International Journal of Safety and Security Engineering
Vol. 13, No. 4, August, 2023, pp. 657-664
Journal homepage: http://iieta.org/journals/ijsse

Ultra-Lightweight Encryption for STL Files in IoT-based 3D Printing

Nilufar Yasmin*, Richa Gupta

Department of ECE, Jaypee Institute of Information Technology (JIIT), Noida 201307, India

Corresponding Author Email: nilufar.yasmin22@gmail.com

https://doi.org/10.18280/ijsse.130407

Received: 8 July 2023
Revised: 5 August 2023
Accepted: 16 August 2023
Available online: 28 September 2023

Keywords:
cryptography, 3-D printing, stereolithography (*STL*) *files, s-box, PRESENT, security, encryption, internet of things* (*IoT*)

ABSTRACT

In the expanding landscape of open hardware and software, the preservation of privacy is paramount for individuals, products, and systems. This study focuses on the security implications pertaining to stereolithography (STL) files in the 3D printing domain, within the scope of the Internet of Things (IoT). As business models increasingly rely on copyrighted content to fuel free services, the application of lightweight encryption becomes crucial in safeguarding STL files utilized in 3D printing operations. In cognizance of the unique needs of the IoT, such as reduced energy consumption, efficient computation, and superior performance metrics, an adaptation of the pioneering Ultra-Lightweight encryption algorithm, modified PRESENT, is proposed. Modifications are made within the substitution box (s-box) of the PRESENT algorithm, yielding a version that consumes less computational time and power. This modified s-box fulfills several evaluation criteria for assessing security parameters, including bijective property, nonlinearity, and strict avalanche criteria, suggesting a substantial resistance to breaches. The application of this customized PRESENT algorithm to secure STL files in IoT-linked 3D printing demonstrates its efficacy in protecting sensitive data, even under the restrictive resources of IoT environments. The findings of this study contribute to the ongoing dialogue on the intersection of security and accessibility in the age of open-source hardware and software.

1. INTRODUCTION

The Internet of Things (IoT) has permeated everyday life, bringing the necessity of security in data exchange to the forefront of considerations [1]. Additive Manufacturing (AM) processes, now increasingly employed for the mass production of end-user items, have been revolutionized by advancements in 3-D printing technology. These developments have democratized the rapid mass production of novel prototypes. End-to-end encryption offers some degree of assurance for design owners, yet the threat of side-channel attacks persists. The question arises: why the emphasis on shielding a specific file format? With the increasing accessibility of 3-D printing, users can reproduce a vast array of items, even those with patent-protected or copyrighted designs. This potential growth in 3-D printing could catalyze significant shifts in supply chain structures, possibly resulting in traditional manufacturers losing substantial control over their intellectual property [2, 3]. Security is a cornerstone of Intellectual Property (IP) rights and forms the bedrock of any business operation. Protecting 3-D IP is paramount in maintaining a competitive edge in the business landscape. In recent times, the theft of 3-D models has emerged as a prevalent issue, with potential to inflict serious business losses. The stakes are further raised when considering 3-D model files associated with military or governmental entities; exposure of such files could jeopardize national security. In light of these considerations, current strategies revolve around the encryption of files within distributed networks [4].

Among the various formats for representing 3-D files, the Stereolithography (STL) format is the most commonly used. STL offers numerous advantages, including compatibility with Computer-aided Design (CAD) software, which allows for the creation of a myriad of personalized products. These designs are encapsulated as 3-D model files, unique digital blueprints within the CAD software [2]. 3-D images, unlike their 2-D counterparts, convey the illusion of depth, rendering objects as volumetric entities occupying space. The intricate spatial arrangement and the substantial data contained within a 3-D model file underscore the need for suitable encryption methods [5]. Traditional block ciphers such as the Data Encryption Standard (DES) and the Advanced Encryption Standard (AES), despite their utility in various applications, fail to meet the requirements for securing complex 3-D model files within resource-constrained IoT environments, spotlighting the importance of security considerations for cloud storage and portable clients [6, 7]. Given these challenges, attention is redirected towards advancements in 3-D printer technology and the opportunities they present. 3-D printer file formats are emerging at the forefront of manufacturing, offering numerous advantages that will be discussed in the following sections.

1.1 3-D printer advantages and future possibilities

- Flexibility in design: 3D printing enables the production of detailed and sophisticated designs that would be difficult or impossible to produce using conventional manufacturing techniques. This makes it possible to create incredibly distinctive items.

- Rapid prototyping: 3D printing makes it possible to prototype quickly and affordably. This shortens the time it takes to introduce new ideas to the market and quickens product development cycles.
- Manufacturing on Demand: 3D printing makes it possible to produce goods as needed, eliminating the need for large-scale production and storage facilities. Cost savings and more effective supply chains may result from this.

Other advantages associated include customization & personalization, manufacturing complex geometries, space exploration and remote manufacturing, medical innovations, Cost-effectiveness, and many more. Apart from the various advantages of a 3-D printer, it has diversified future possibilities like:

- Bioprinting: The ability to print functional human tissues and even organs have the potential to revolutionise medicine by removing the need for animal testing in drug development and organ transplantation.
- Buildings and infrastructure might theoretically be built using large-scale 3D printers, which would speed up the process and cut costs while allowing for creative architectural concepts.

Other important future approaches includes electronics printing, space exploration, sustainable manufacturing, multi-material printing, global access like communities could manufacture necessary commodities locally if 3D printing brought manufacturing capabilities to isolated and underdeveloped regions [8].The possibilities for 3D printing are set to increase as technology develops and researchers look into new materials and methods, revolutionising various industries and how we produce, design, and create.

1.2 How do 3-D printers work?

In 3-D printing whole object is divided into thousands of tiny little slices, then manufacturing is carried from the bottom to top, layer-by-layer. Every layer stick to its previous layer beneath it and combining the whole structure the final object is formed. The 3-D printing workflow can be understood with the help of the diagram shown below as Figure 1 [9].

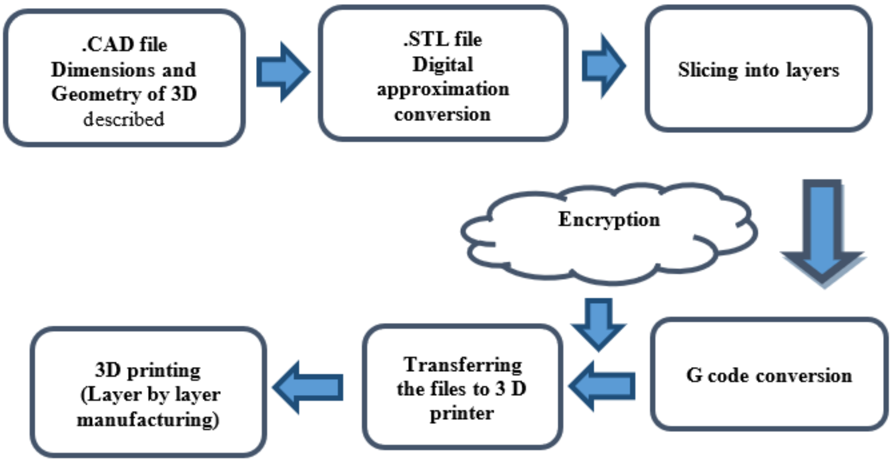
.CAD file
Dimensions and Geometry of 3D described
.STL file
Digital approximation conversion
Slicing into layers
Encryption
G code conversion
Transferring the files to 3 D printer
3D printing (Layer by layer manufacturing)

Figure 1. 3-D printing workflow and scope of encryption [9]

3-D printing workflow consists of the following steps:

i. Computer-aided design (CAD) software is used to design the 3-D structure of a product to be manufactured.
ii. Triangulation method is used to convert CAD files into an STL file format.
iii. A dedicated software is used for slicing the 3-D model into layers of specified thickness.
iv. Next G-code commands; programming language for Computer Numerical Control (CNC), are used to make the device work in a particular manner.
v. At the time of file transfer; if it happens in a distributed network, then encryption must be done on the transmitter side to provide necessary security to the file. It must be followed by decryption at the receiving end.
vi. Manufacturing occurs layer-by-layer, followed by post-processing operations.

The unique advantages of 3D printing, along with its prospective advancements, provide a compelling research trajectory for emergent scholars. Given the contemporary landscape, it is incumbent upon all to equip themselves for the Internet of Things (IoT) environment. In the context of IoT communications, the necessity for encryption becomes paramount, particularly when transmitting sensitive information. This includes, but is not limited to, governmental, medical, private, or personal data. To safeguard against third-party attacks, it is crucial to implement protections that prevent unauthorized access, modifications and restrict the number of copies. Such measures not only serve to maintain the integrity of the files in question but also uphold intellectual property rights. This is relevant even in scenarios involving distributed networks, where the need for data security is amplified. In this context, the foundational role of encryption in preserving data integrity and intellectual property rights is underscored, irrespective of the network configuration.

2. BACKGROUND

This section discusses the need of encryption for STL files in a distributed network. Also, it discusses about the related work in this field including summery of few research papers.

2.1 3D Printing in a distributed and the essential role of encryption

Instead of replacing the complete product on account of losing a small part of it, manufacturing the lost/damaged part quickly is not a future thing anymore. Thanks to Additive manufacturing practices we can manufacture any customized product at our own ease. IoT has revolutionized our lives from head to toe. Creating things from home or the office with a single command is an amazing technical feat. Hence use of 3-D printers in a distributed network is the thing for today. It helps us exploit IoT in its full swing. With luxury comes complexities. As we have to be more alert about whom to trust or whether the network, we use for printing is secure or not? Along with security, there are several issues related to Intellectual property rights [3]. When a content originator distributes a file to a recipient, various concerns emerge, including security vulnerabilities, readability concerns, susceptibility to unauthorized duplication, and the potential for exploiting the file to generate an unrestricted quantity of product copies. Hence classic encryption is futile here [1]. In the world of 3D printing, encryption is essential for a number of reasons.

IP Protection: Encryption protects 3D model design files by preventing unauthorized access and reproduction and so safeguarding the inventors' and manufacturers' intellectual property rights.

Secure transfer: Encrypting data reduces the possibility of interception or tampering while ensuring that design files stay private and unchanged throughout the transfer between devices and printers. Encryption prevents the unauthorised copying of valuable or sensitive items, which lowers the possibility of fake goods reaching the market [10].

Apart from these, other important reasons include data privacy, regulatory compliance, sticking to anti-piracy measures, secure collaborations, and future-proofing. Integration with the Internet of Things: As IoT-connected 3D printers become more prevalent, encryption is essential to protect data transfers and communications between printers and devices [11]. In the end, encryption in 3D printing aids in safeguarding priceless designs, preserving the integrity of printed things, and promoting confidence among designers, producers, and clients.

2.2 Related work

This section explores the contributions of various researchers in the field, focusing particularly on the critical issue of security in relation to 3-D printers.

In 2021, Silva et al. unveiled an effective encryption algorithm designed to enhance the security of 3D printed models, operating within the frequency domain of the discrete cosine transform [12]. The primary objective of this algorithm is to thwart unauthorized copying and access during storage and transmission. The algorithm targets the encryption of the Direct Current (DC) coefficients of facet matrices within the frequency domain of the discrete cosine transform. An encrypted model for 3D printing is the output of this procedure. In this review, it is noted that the discussed composite formations utilizing graphene exhibit notable enhancements compared to unreinforced polymer matrices, in addition to eliciting an improved cellular response.

In 2020, Ji Xu et al. proposed a novel method of encrypting 3D images utilizing a unique chaotic system [13]. An analysis of the system's properties facilitated the development of an encryption scheme for 3D images. The procedure involves altering the chaotic system's initial condition using a hash value and scrambling image coordinates using chaotic sequences. A security analysis confirmed the method's robustness against attacks, thereby underscoring its potential for secure 3D image encryption. The experimental results demonstrated that the proposed research exhibited desirable outcomes in terms of capacity, resilience against brute force attacks, key space, and entropy evaluations.

In 2019, the focus of Shui-li et al. was centered upon the protection of big data privacy during storage [14]. They proposed a customizable encryption method, capable of satisfying individual privacy preferences. This versatile method accommodates a variety of file formats, including audio, video, images, and text. Additionally, it caters to diverse privacy requirements by offering tailored encryption schemes. This method has demonstrated its efficiency in terms of entropy and computation time when contrasted with alternative methods.

In 2020, an innovative encryption algorithm for 3D image files was presented by Xu et al. [13] leveraging a novel discrete chaotic system [15]. The chaotic characteristics of this system were scrutinized through phase diagrams, Lyapunov exponents, and bifurcation diagrams. These analyses informed the development of a secure encryption method for 3D image files.

Transitioning from the realm of 3D printing, the purview is broadened to embrace the expansive domain of the Internet of Things (IoT). This shift brings to light the intersection of these two technologies, unveiling a landscape rich with potential synergies and intricate interconnections. The IoT has recently revolutionized printing by enabling remote management and proactive maintenance; however, this has also elicited security concerns. In 2021, research demonstrated how 3D printers with IoT capabilities could be exploited, leading to unauthorized printing or sabotage. The MyQbot IoT botnet further underscored the imperative for robust security in 3D printers by exploiting lax IoT device security. The significant breach of IoT-connected 3D printers at major automobile manufacturers in 2020 highlighted the risks of unauthorized access, despite the improved supply chain efficiency afforded by this connection. As quality control driven by IoT gains momentum, the importance of maintaining firmware integrity has been emphasized, as instances have been recorded where printer performance was compromised by insecure updates [16].

3. PROPOSED METHOD

This document describes a system, that facilitates the secure distribution of electronic files on a computing network with the help of encryption techniques using Lightweight cryptography. This technique effectively blocks unauthorized access to files shared by content providers, ensuring their confidentiality. It incorporates lightweight cryptography to enhance security, especially in pervasive computing environments. applications, which are characterized by resource-constrained devices. To countermeasure major security threats, IoT needs to apply encryption to sensor devices in the environment. Symmetric and asymmetric key-based cryptography stand out among cryptographic techniques. The same secret key is employed in symmetric-key cryptography for both encryption and decryption. This method is frequently used in many protocols to guarantee data integrity, privacy, and secure authentication. It's comparable to having just one key to use to lock and unlock a safe.

Asymmetric key cryptography, in contrast, uses two keys: a private key for decryption and a public key for encryption. The validity of messages can be confirmed using this technique, and data integrity can be upheld. The statement emphasizes the widespread usage of symmetric-key cryptography and how adaptable it is in protecting private data and fostering secure communication. Block ciphers are better in robustness and security performances. Lightweight cryptography is a low-computation encryption method [17]. Keeping in mind the power constraints and other resource constraints in IoT devices an ultra-lightweight symmetric block cipher PRESENT has been chosen. PRESENT is the pioneer algorithm in the field of lightweight cryptography and is one of the very few lightweight algorithms which are ISO certified [18]. It has been modified to attain better results in terms of various parameters. In this context, a non-linear layer employs a 4-bit S-box, which is repeated 16 times simultaneously in each round. The PRESENT cipher follows the structure of an SP network (Substitution-Permutation network) and involves a total of 31 rounds. There are 31 rounds total, and each one contains a series of events. These actions include using an XOR operation to include a round key Ki, where i is a number between 1 and 32. K32, the final key, is used for post-whitening operations. After that, a non-linear substitution layer is applied, followed by a linear bitwise permutation. Here algorithm uses a single 4-bit to 4-bit S-box S: F24 → F24. It is much more compact as compared to an 8-bit S-box [19]. The optimal S-box generation is designed by modifying the existing PRESENT block cipher using a novel technique.

Initially, a static S-box is generated. In static S-box generation, Gaussian random sequences are created using Box Muller transform and the pseudo number sequences are generated using the Central Limit Theorem. Then floating points are removed from both sequences which attain the compound integer sequences using both Gaussian random sequence and pseudo number sequences. Subsequently, obtained compound integer sequences are converted into 64 distinct integers. Finally, 64 distinct integers are transformed into a 4× 4 matrix and obtain the proposed static S-box. Further, this generated S-box is utilized as a seed to the dynamic S-box generation for obtaining optimal S-box. Here, a gradient-based optimization algorithm is utilized for obtaining optimal S-box generation to secure the data.

Firstly, random data are collected from various IoT devices/sensors which act as initial input to the system, for Encryption. The input data is secured by the proposed optimal S-box generation process. Initially, input data creates a sequence by using the Gaussian distribution algorithm named as Box-Muller transform. Further, generating the pseudo-number sequences using the "Central Limit Algorithm" is carried out. This algorithm states that, when a large number of samples are attained from the autonomous variables, then the arithmetic mean of their distribution will be the normal distribution [20]. The normal distribution is generally known as a bell-shaped distribution. The generated good performance static form of the S-box is utilized for the generation of dynamic S-box. Dynamic S-Box Optimization is carried out with a Gradient-Based Optimization Approach. The optimization technique is utilized to generate an optimal S-box from a static form of an S-box. The main importance of this process is to choose the optimal S-box [21, 22]. The Schematic for the encryption of data using optimal S-box is shown in Figure 2.

The commonly known approaches to creating S-boxes involve using chaotic maps, power polynomials, DNA sequences, TDERC sequences, Galois Field operations, machine learning techniques, inversion mapping, and pseudorandom number generators [23]. Further investigation in this area has identified vulnerabilities in the behavior of chaotic systems. These weaknesses include uneven distribution of data, interruptions in chaotic sequences, a limited amount of randomness, the impact of finite precision, and challenges related to computational complexity [24, 25].

The method proposed here is a hybrid model combining static s-box generation and dynamic s-box generation. The benefit of employing the Box-Muller transform to alter S-boxes in cryptographic algorithms is that it introduces controlled randomness, which improves the security (non-linear feature) of the S-box. This could help develop stronger, more resilient cryptographic systems that are better able to safeguard sensitive data. Using a Gradient-Based Optimization Approach to dynamically optimize S-boxes has the benefit of making the cryptographic system flexible and well-tuned. It's similar to having an algorithm that can modify itself to block many attack tactics, thereby enhancing system security. This can contribute to creating more robust and resilient cryptographic systems that are better equipped to protect sensitive information.

For the purpose of simulation and analyzing results, the following procedures have been undertaken:

- **Use of Modified PRESENT**: The modified PRESENT encryption method has been employed to encrypt STL files used in 3-D printing. Various advanced parameters have been calculated and then compared to assess their effectiveness.
- **Conversion of STL Format**: Initially, the 3-D file in STL ASCII format is transformed into the STL Binary format. Using a binary format reduces storage space usage and minimizes RAM consumption when loading into memory.

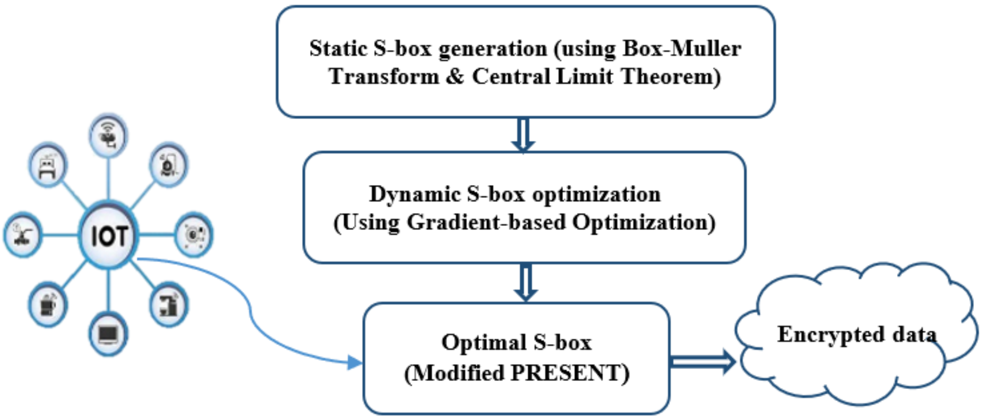
Static S-box generation (using Box-Muller Transform & Central Limit Theorem)
Dynamic S-box optimization (Using Gradient-based Optimization)
Optimal S-box (Modified PRESENT)
Encrypted data
IOT

Figure 2. Schematic diagram of encryption of data using proposed optimal S-box

- **Conversion Process**: To convert an STL ASCII file to STL Binary, software tools such as Autodesk 123-D Design and Meshmixer are utilized. Autodesk 123-D Design is a free CAD designer software provided by Autodesk for Windows. Meshmixer, on the other hand, is used to generate the binary STL file.
- **Performance Metrics**: Computational time and power consumption are computed and then compared with the original version of the PRESENT cipher. This assessment helps to understand the efficiency and resource utilization of the modified method.
- **File Reconstruction:** The reconstructed files are used for further validation and verification purposes. This step ensures that the encryption and conversion processes have been accurately executed, and the original data can be successfully retrieved.

4. RESULT & DISCUSSION

Various standard STL files are provided as input. Here, a standard STL file for a cube (3-D structure of cube), a version of the cube with engraving on it (xyz_Calibration_cube.STL), and a standard file for Eiffel.STL is considered for analysis. After encryption, computation Time and Power consumed are calculated using a comparison code in MATLAB. Results for the same are presented in the Tables 1-3.

Table 1. Power and time comparison of existing and modified algorithm (for file1)

Cipher	Computation Time (Sec)	Power (µW)
Modified PRESENT	653.5436	1602.539
PRESENT	721.894	1698.885

Table 2. Power and time comparison of existing and modified algorithm (for file2)

Cipher	Computation Time (Sec)	Power (µW)
Modified PRESENT	631.7973	1588.938
PRESENT (SPN with a bit-based permutation layer) 2007	717.579	1695.557

Table 3. Power and time comparison of existing and modified algorithm (for file3)

Cipher	Computation Time (Sec)	Power (µW)
Modified PRESENT	53454.5436	11763.6791
PRESENT	56675.7693	11790.545

Input Files:

- File 1: Standard STL file of a cube.
- File 2: STL file of a cube with engraving.
- File 3: Standard STL file of the Eiffel Tower.

Simulation Results:

- For each of the input files, Modified PRESENT has been tested against the original PRESENT.
- Across all scenarios (File 1, File 2, and File 3), simulation results consistently indicate that Modified PRESENT outperforms the original PRESENT.

Comparison Metrics:

- The performance of Modified PRESENT is measured against the original PRESENT in terms of factors such as computation time and power consumption.

Source of STL Files:

- The input STL files used for testing are obtained from "Thingiverse.com," a platform that offers various standard STL files for 3D printing and modeling [26].

In summary, the findings from the simulations demonstrate that across different types of STL files (cube, engraved cube, Eiffel Tower), the Modified PRESENT encryption method consistently exhibits better performance compared to the original PRESENT. This suggests that the modifications applied to the PRESENT cipher have resulted in improved efficiency and effectiveness, making it a more suitable choice for securing various types of standard STL files. Table 1 results show that the modified version of the PRESENT cipher demonstrates a shorter computation time (653.5436 Sec ie. in Seconds) and lower power consumption (1602.539 **µW** ie. in Micro-Watt) compared to the original PRESENT cipher, which required a longer computation time (721.894 Sec) and higher power consumption (1698.885 **µW**). This suggests that the modification has led to improved efficiency in terms of both computation time and power usage. Similarly results as per Table 2, and 3 show that irrespective of input STL file size, computation time required and power consumed are always lesser in modified PRESENT as compared to existing PRESENT. In essence, the results substantiate the advantage of the proposed modified PRESENT encryption method, making it a strong candidate for securing data in environments where efficiency and resource optimization are critical, such as the realm of IoT devices.

4.1 Performance analysis and validation of the algorithm

In this section, the performance of the proposed methodology is analyzed and compared with existing approaches in terms of computation time and power consumed. These are considered to be state-of-the-art parameters i.e., the standard parameters for analyzing the performance of various algorithms. The effectiveness of the suggested system is assessed using MATLAB's 2021 version on a Dell Vostro Laptop, which is equipped with an Intel(R) Core (TM) i5-8265U CPU running at 1.60GHz (with a maximum speed of 1.80GHz), and operates on an 8 GB RAM and a 1 TB HDD storage. MATLAB is a widely used programming environment that's particularly popular for simulations in various fields, including engineering, physics, and computer science. The performance matrices and their evaluations are already described in the section under results.

- ***Power consumption:*** The power consumption is calculated for software/hardware-implemented [27].
- ***Time:*** The time required for the computation of the encryption algorithm is calculated using MATLAB 2021 version.

Modified PRESENT can compute in lesser time with comparative power consumption, making it a contender for IoT devices. The observed decrease in both computation time and power consumption when using the modified PRESENT implies that this encryption approach is more efficient in terms of processing speed and energy utilization. These factors are of paramount importance, especially in resource-constrained scenarios like the IoT, where devices often operate with limited computational capabilities and power availability.

4.2 S-Box evaluation of proposed modified PRESENT

Here the input files are encrypted using the proposed algorithm that has the modified s-Box. The S-Box is a key element in many encryption schemes, greatly enhancing their robustness and security. Due to its confusion, non-linearity, avalanche effect, diffusion, and complexity qualities, contemporary cryptographic systems are capable of safeguarding sensitive data from intrusion and attacks. As the performance of an S-box is analyzed by evaluating its non-linearity, bijectivity, equiprobable input or output XOR distribution, and strict avalanche criterion, details of which are undersigned.

1) Non-linearity

Non-linearity shows the amount of confusion created by the S-box. Non-linearity ensures that the S-box doesn't behave in a linear or predictable manner. Higher the non-linearity of the S-box; the stronger against differential and linear attacks it is [28]. The non-linearity measure can be expressed as,

$$N_x = 2^{p-1}\left(1 - 2^{-p}\, max(y_p)\right) \tag{1}$$

here, N_x represents the non-linearity, y_p represents the output bits. The non-linearity value for proposed modified PRESENT is 0.541 whereas for the existing PRESENT, it is 0.507.

2) Bijectivity

It says that obtained output vectors must appear only once [24]. This property is essential to ensure reversibility during encryption and decryption. It prevents information loss and helps maintain the integrity of the data being processed. Bijectivity also ensures that there are no collisions, where two different inputs produce the same output. The sum of every component in the function is f_p, equivalent to 2^{p-1} what is described as

$$w\left(\sum_{p=1}^{n} b_p f_p\right) = 2^{p-1} \tag{2}$$

Here in Eq. (2); $b_p = \{b_1, b_2, b_3 \ldots\ldots b_p\} \in (0,1)$. The proposed S-box generates 16 distinct output values: [7, 11, 10, 5, 3, 4, 0, 8, 12, 2, 14, 6, 13, 1, 9, 15]. Here Bijectivity criteria are fulfilled as each value has appeared once only.

3) Equiprobable input or output XOR distribution

Differential probability serves as a tool for assessing the effectiveness of the encryption function [29]. This criterion focuses on the statistical distribution of input-output pairs and their XOR relationships. Ideally, an S-box should ensure that the XOR distribution of input-output pairs is balanced, meaning that an equal number of ones and zeros occur in the XOR result. This property helps in preventing various attacks, including differential cryptanalysis, by making it harder to find patterns in the XOR differences. The equiprobable output values obtained here varies are from 0 to 15. Hence the maximum value is 15. Differential uniformity is represented by Eq. (3).

$$D_y = Max\left(\frac{x_p}{2^p}\right) \tag{3}$$

Here in Eq. (3), D_y represents the differential probability, x_p represents the input bits. The modified S-box has a differential uniformity of 15. Hence it can be considered a fairly resistant S-box against differential attacks.

4) Strict avalanche criterion (SAC)

As per SAC Making small changes in input will create a large amount of change in output [30]. The strict avalanche criterion (SAC) measures the sensitivity of an S-box's output to changes in its input. SAC is crucial for diffusion, which is the process of spreading the influence of a single input bit change throughout the entire output. Here the matrix values obtained or the modified S-box is presented in Table 4.

Table 4. Matrix elements satisfying SAC of modified PRESENT

0.5010	0.5047	0.4897	0.4989
0.5082	0.5155	0.5559	0.4878
0.4898	0.4383	0.4696	0.4975
0.4973	0.4388	0.4639	0.4650

The matrix values mentioned in Table 4 prove that the developed Modified PRESENT fulfills the SAC by obtaining an average value of 0.488. Whereas the existing PRESENT algorithm-based generated matrix has an SAC value of approximately 0.35.

5) Output bits independence criterion

This concept implies that changing a single input bit results in the independent alteration of the corresponding output bits. Here, all the obtained bits are independent of each other [23]. For every varying input bit, the output bit also changed accordingly. Moreover, the proposed modified PRESENT satisfies the independence of bits property by obtained values of [0.50, 0.49, 0.504, 0.49, 0.509, 0.51, 0.53, 0.497, 0.48, 0.43, 0.44, 0.49, 0.493, 0.439, 0.46, 0.47]. Here, each and all attained values are independent of each other, whereas in the existing cipher some of the values are obtained repeatedly.

Statistical parameters used here includes, Power, Time and, S-box evaluation criteria values. All are mentioned in Tables 1, 2, 3 and 4.

5. CONCLUSION

In this paper, an effort has been made to encrypt STL files for 3-D printers using the modified PRESENT. The modification involves adapting the S-box to meet security criteria while maintaining performance. This encryption approach is particularly suited for binary input data, such as standard STL files used in 3-D printing. The process involves converting the STL files into binary format, encrypting the data, and then decrypting it to reconstruct the STL file. The results show significant improvements in terms of time and power consumption compared to the original PRESENT version, making it a promising solution for secure communication in additive manufacturing within distributed networks, contributing to enhanced reliability in the IoT landscape.

5.1 Key findings and future work

The key findings of the work are as below:

- Modification of the PRESENT block cipher's S-box to meet security criteria and enhance performance.
- Successful encryption and controlled decryption of standard STL files after conversion to binary format.
- Improved performance in time and power consumption compared to the original PRESENT version.

- Applicability of lightweight ciphers for secure communication in additive manufacturing within distributed networks and IoT environments.

Hence, encryption in the Additive manufacturing field in a distributed network in a secure way can be done with the help of such lightweight ciphers. This gives us a lead toward reliable communication in the IoT environment.

As future work, the research can delve into a comprehensive cryptanalysis of the modified PRESENT cipher, thoroughly assessing its resistance against various cryptographic attacks. Furthermore, expanding the experimentation to encompass different platforms and network settings, including unsecured environments, would provide a more holistic understanding of the cipher's versatility and security. In the evolving landscape of encryption, exploring the synergy of artificial intelligence and machine learning could offer innovative ways to bolster the cipher's effectiveness by enabling adaptive responses to emerging threats and optimizing performance parameters.

5.2 Significance of work and its limitations

The implications of this work extend beyond the immediate application of encrypting STL files for 3-D printers. In essence, this work not only offers a practical solution for securing STL files but also contributes to the broader realms of data security, IoT communication, and additive manufacturing. Its implications hold potential benefits for industries, consumers, and society at large, fostering safer and more resilient technological landscapes.

While this work introduces a modified PRESENT block cipher for 3-D printing STL file encryption, there are a number of limitations that should be taken into account while analyzing the findings. The encryption method's emphasis on binary data may limit the scope of its use beyond particular file types. The observed improvements in performance in controlled situations might not always apply to complex industrial scenarios. It is still unknown whether the approach can handle larger or more complicated files. Additionally, it is crucial to consider a thorough security assessment, a comparison with alternative encryption techniques, and the approach's long-term sustainability. Acknowledging these limitations provides a realistic context for the findings and highlights areas for future research and refinement.

REFERENCES

[1] Gatlin, J., Belikovetsky, S., Elovici, Y., Skjellum, A., Lubell, J., Witherell, P., Yampolskiy, M. (2021). Encryption is futile: Reconstructing 3d-printed models using the power side-channel. In Proceedings of the 24th International Symposium on Research in Attacks, Intrusions and Defenses, pp. 135-147. https://doi.org/10.1145/3471621.3471850

[2] Ngo, T.D., Kashani, A., Imbalzano, G., Nguyen, K.T., Hui, D. (2018). Additive manufacturing (3D printing): A review of materials, methods, applications and challenges. Composites Part B: Engineering, 143: 172-196. https://doi.org/10.1016/j.compositesb.2018.02.012

[3] Clarke, C. (2017). Create it real creates encryption platform for 3-D printers to prevent intellectual property right. 3-D Printing Industry.

[4] Milazzo, A.M., et al. (2018). U.S. Patent No. 10,063,529. Washington, DC: U.S. Patent and Trademark Office.

[5] Hua, Z., Zhou, Y. (2016). Image encryption using 2D Logistic-adjusted-Sine map. Information Sciences, 339: 237-253. https://doi.org/10.1016/j.ins.2016.01.017

[6] Alapati, Y.K., Ravichandran, S. (2020). Secure data transfer in manet with key calculator and key distributer using cryptography methods. International Journal of Safety and Security Engineering, 10(4): 567-572. https://doi.org/10.18280/ijsse.100417

[7] Yadav, A.K., Ritika, M.G., Garg, M. (2021). Cryptographic solution for security problem in cloud computing storage during global pandemics. International Journal of Safety and Security Engineering, 11(2): 193-199. https://doi.org/10.18280/ijsse.110208

[8] Bozkurt, Y., Karayel, E. (2021). 3D printing technology; methods, biomedical applications, future opportunities and trends. Journal of Materials Research and Technology, 14: 1430-1450. https://doi.org/10.1016/j.jmrt.2021.07.050

[9] Silva, M., Pinho, I.S., Covas, J.A., Alves, N.M., Paiva, M.C. (2021). 3D printing of graphene-based polymeric nanocomposites for biomedical applications. Functional Composite Materials, 2(1): 1-21. https://doi.org/10.1186/s42252-021-00020-6

[10] Balogh, S., Gallo, O., Ploszek, R., Špaček, P., Zajac, P. (2021). IoT security challenges: Cloud and blockchain, postquantum cryptography, and evolutionary techniques. Electronics, 10(21). https://doi.org/2647.10.3390/electronics10212647

[11] Şatir, E., Kendirli, O. (2022). A symmetric DNA encryption process with a biotechnical hardware. Journal of King Saud University-Science, 34(3): 101838. https://doi.org/10.1016/j.jksus.2022.101838

[12] Pham, N.G., Moon, K.S., Lee, S.H., Kwon, K.R. (2018). An effective encryption algorithm for 3D printing model based on discrete cosine transform. Journal of Korea Multimedia Society, 21(1): 61-68. https://doi.org/10.9717/kmms.2018.21.1.061

[13] Xu, J., Zhao, C., Mou, J. (2020). A 3D image encryption algorithm based on the chaotic system and the image segmentation. IEEE Access, 8: 145995-146005. https://doi.org/10.1109/ACCESS.2020.3005925

[14] Li, S., Li, M., Xu, H., Zhou, X. (2019). Searchable encryption scheme for personalized privacy in IoT-based big data. Sensors, 19(5): 1059. https://doi.org/10.3390/s19051059

[15] Pham, G.N., Lee, S.H., Kwon, O.H., Kwon, K.R. (2018). Two-dimensional (2D) slices encryption-based security solution for three-dimensional (3D) printing industry. Electronics, 7(5): 64. https://doi.org/10.3390/electronics7050064

[16] Rahmani, H., Shetty, D., Wagih, M., et al. (2023). Next-generation IoT devices: Sustainable eco-friendly manufacturing, energy harvesting, and wireless connectivity. IEEE Journal of Microwaves, 3(1): 237-255. https://doi.org/10.1109/JMW.2022.3228683

[17] Thangamani, N., Murugappan, M. (2019). A lightweight cryptography technique with random pattern generation. Wireless Personal Communications, 104: 1409-1432. https://doi.org/10.1007/s11277-018-6092-8

[18] Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J., Seurin, Y., Vikkelsoe, C. (2007). PRESENT: An ultra-lightweight block cipher. In Cryptographic Hardware and Embedded Systems-CHES 2007: 9th International Workshop, Vienna, Austria, pp.

450-466. https://doi.org/10.1007/978-3-540-74735-2_31

[19] Yasmin, N., Gupta, R. (2023). Modified lightweight GIFT cipher for security enhancement in resource-constrained IoT devices. International Journal of Information Technology, 1-13. https://doi.org/10.1007/s41870-023-01439-9

[20] Zhang, Y., Jin, Z. (2020). Group teaching optimization algorithm: A novel metaheuristic method for solving global optimization problems. Expert Systems with Applications, 148: 113246. https://doi.org/10.1016/j.eswa

[21] Zhu, H., Tong, X., Wang, Z., Ma, J. (2020). A novel method of dynamic S-box design based on combined chaotic map and fitness function. Multimedia Tools and Applications, 79: 12329-12347. https://doi.org/10.1007/s11042-019-08478-0

[22] Khan, M.F., Ahmed, A., Saleem, K. (2019). A novel cryptographic substitution box design using Gaussian distribution. IEEE Access, 7: 15999-16007. https://doi.org/ 10.1109/ACCESS.2019.2893176

[23] Beg, S., Ahmad, N., Anjum, A., Ahmad, M., Khan, A., Baig, F., Khan, A. (2020). S-box design based on optimize LFT parameter selection: A practical approach in recommendation system domain. Multimedia Tools and Applications, 79: 11667-11684. https://doi.org/10.1007/s11042-01

[24] Wang, Y., Wong, K.W., Li, C.B., Li, Y. (2012). A novel method to design S-box based on chaotic map and genetic algorithm. Physics Letters A, 376(6-7): 827-833. https://doi.org/10.1016/j.physleta.2012.01.009

[25] Ning, L., Ali, Y., Ke, H., Nazir, S., Huanli, Z. (2020). A hybrid MCDM approach of selecting lightweight cryptographic cipher based on ISO and NIST lightweight cryptography security requirements for internet of health things. IEEE Access, 8: 220165-220187. https://doi.org/10.1109/ACCESS.2020.3041327

[26] Thingiverse. https://www.thingiverse.com/, accessed on Feb 4, 2023.

[27] Yasmin, N., Gupta, R. (2022). Performance analysis of lightweight algorithm GIFT-COFB for 3-D printer security. In 2022 8th International Conference on Signal Processing and Communication (ICSC), pp. 475-478. https://doi.org/10.1109/ICSC56524.2022.10009201.

[28] Mohamed, K., Pauzi, M.N.M., Ali, F.H.H.M., Ariffin, S., Zulkipli, N.H.N. (2014). Study of S-box properties in block cipher. In 2014 International Conference on Computer, Communications, and Control Technology (I4CT), pp. 362-366. https://doi.org/10.1109/I4CT.2014.6914206

[29] Zhu, D., Tong, X., Zhang, M., Wang, Z. (2020). A new S-box generation method and advanced design based on combined chaotic system. Symmetry, 12(12): 2087. https://doi.org/10.3390/sym12122087

[30] Shanthi Rekha, S., Saravanan, P. (2017). Low cost circuit level implementation of PRESENT-80 S-BOX. In VLSI Design and Test: 21st International Symposium, VDAT 2017, Roorkee, India, pp. 354-362. https://doi.org/10.1007/978-981-10-7470-7_35