International Journal of Safety and Security Engineering
Vol. 14, No. 3, June, 2024, pp. 1007-1014
Journal homepage: http://iieta.org/journals/ijsse

Secure File Operations: Using Advanced Encryption Standard for Strong Data Protection

Nikhil Chand Nelakuditi*, Nanda Kishore Namburi, Jilani Sayyad, Dinesh Varma Rudraraju, Raja Govindan, Peddada Venkateswara Rao

Computer Science Engineering, Koneru Lakshmaiah Educational Foundation, Vijayawada 522302, India

Corresponding Author Email: klucse2000030672@gmail.com

Copyright: ©2024 The authors. This article is published by IIETA and is licensed under the CC BY 4.0 license (http://creativecommons.org/licenses/by/4.0/).

https://doi.org/10.18280/ijsse.140330

Received: 6 December 2023
Revised: 6 March 2024
Accepted: 15 March 2024
Available online: 24 June 2024

Keywords:
data security, cyber threats, encryption, data protection, confidentiality, integrity, authenticity

ABSTRACT

In our rapidly digitalizing world, the safeguarding of sensitive information stands as a critical concern for organizations across industries. This paper addresses the evolving complexities in data security by advocating the Advanced Encryption Standard (AES) algorithm as a robust defense against the expanding landscape of cyber threats. As financial, healthcare, and educational data transition to digital formats, vulnerabilities in data transmission become more pronounced. To counter these risks and ensure the integrity, confidentiality, and authenticity of essential data, this paper emphasizes the implementation of AES encryption. Thoroughly scrutinizing the widely acknowledged AES algorithm, we highlight its effectiveness and adaptability in securing data. The paper underscores the indispensability of AES in the modern data security milieu, emphasizing its role not only in securing information during transmission and storage but also in decryption, granting authorized users access to protected data. By embracing AES encryption and decryption, organizations can strengthen their defenses against various data-related threats, maintaining the trust and assurance of stakeholders. This work emphasizes the pressing demand for AES encryption and decryption in an era where data security takes precedence, offering valuable insights into their significance and practical application for the protection of crucial digital assets.

1. INTRODUCTION

In the contemporary digital age, data reigns supreme as the lifeblood of organizations, underpinning their operations, decisions, and competitive advantage. The rapid and relentless digitalization of information has ushered in numerous transformative benefits, revolutionizing the way we store, transfer, and manage data. However, this digital transformation has brought with it a new set of challenges, particularly in the realm of data security. Financial transactions, patient health records, academic achievements they all have one thing in common now: they exist in the digital domain. As data assumes a central role in our daily lives, safeguarding its confidentiality, integrity, and accessibility has become an imperative, notably in the domains of finance, healthcare, and education.

The Advanced Encryption Standard (AES) algorithm's critical significance is the central focus of this journal paper's exploration into data security. In a landscape where data is more precious than ever, AES stands as a formidable guardian, offering a robust defense against a spectrum of cyber threats that haunt the digital realm. But our work extends beyond just highlighting the importance of AES in data encryption and decryption.

As we entrust data with increasingly critical responsibilities and amplify the importance of confidentiality, the methods used to protect it must evolve in kind. Our objective is to delve deep into the core of data security with AES as our guiding star, shedding light on both its encryption and decryption functions. The significance of AES in our work cannot be overstated; it emerges as a versatile and trusted solution that secures data across a multitude of sectors, during transmission, and while at rest. Through the seamless implementation of AES for data protection and accessibility, we advocate proactive measures to safeguard sensitive information, ensuring not only its confidentiality and integrity but also the trust of those who rely upon it.

In this journal paper, we transcend the boundaries of mere encryption and delve into the equally vital sphere of decryption. This dual approach ensures that not only is data safeguarded during transfer and storage but that it is made available to authorized users when needed. AES encryption and decryption serve as gatekeepers, facilitating secure and efficient access to vital digital assets, thereby elevating data security to a critical domain of trust and assurance (Figure 1).

We've paved the way for an in-depth examination of the AES algorithm, highlighting its central role in modern data security. In the forthcoming sections, we'll delve into the reasons why AES is essential for securing digital data in finance, healthcare, education, and beyond. Our focus is on demonstrating the importance of AES in the digital era, emphasizing the need to build trust and ensure the secure and efficient utilization of valuable digital assets, both in terms of protection and accessibility.

This journal aims to bridge the gap between theoretical

advancements and practical implications in the realm of data security. Beyond a theoretical exploration of encryption and decryption, we delve into the real-world scenarios where the efficacy of such algorithms is put to the test. By addressing research questions that extend beyond the technicalities of AES, we strive to contribute to a comprehensive understanding of how encryption methods align with the complex demands of today's interconnected world.

Our broader objective encompasses a nuanced analysis of the challenges faced by organizations in safeguarding sensitive information. From financial transactions to healthcare records and educational data, the spectrum of digital assets requiring protection is vast. The journal seeks to provide insights into the dynamic landscape of data security, offering valuable perspectives on not just the 'what' and 'how' of encryption but also the 'why' in various practical contexts.

As we progress through the subsequent sections, the narrative will unfold, shedding light on specific objectives, methodologies employed, and the implications of our findings. This journal is more than a technical exploration; it is a holistic endeavour to decipher the complex tapestry of data security in the contemporary digital age.

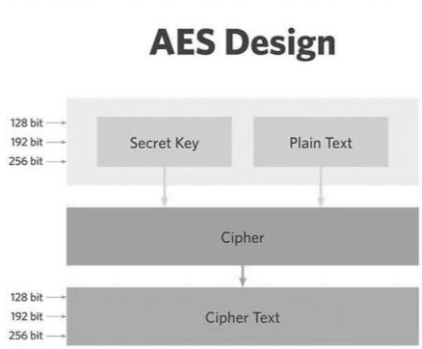

Figure 1. Conceptual diagram of AES encryption

2. LITERATURE WORK

In essence, our comprehensive literature review meticulously navigates through both foundational and contemporary works, offering a panoramic perspective on the dynamic landscape of data security. Each meticulously chosen citation serves as a building block, collectively reinforcing the overarching theme of strengthening the fortifications around data security, with a distinctive focus on the pivotal role played by the Advanced Encryption Standard (AES) algorithm (Table 1).

Yu et al. [1] demonstrate the effectiveness of this protection in preventing scan-based attacks, thus averting the leakage of critical data. Additionally, they proposed protection extends its security measures to disable key Trojan attacks, bolstering the overall robustness of AES implementations.

The experimental results included by Liu [2], illustrate that the suggested chaotic algorithm yields an average P-value uniformity of 0.714. This finding suggests that the data security system which is based on the chaotic algorithm provides a high degree of security.

Wardhani et al. [3] explicates the fundamental objective of the described implementation, focusing on the achievement of secure communication through a 128-bit encryption key. Initially, data undergoes encryption before transmission, and the resulting encrypted voice data is then sent to another node using a communication module. When the communication module receives the encrypted voice data, it triggers the decryption process to retrieve the original plaintext.

Differential fault analysis (DFA) remains a critical area of research, with an ongoing exploration of various techniques and dimensions for compromising AES implementations [4]. The findings and methodologies underscore the importance of continually enhancing the security of AES encryption in the face of emerging attack vectors.

The significance of security in integrated circuits within the contemporary global supply chain is undeniable. Chhabra and Lata [5] emphasizes the value of AES as a security measure and underscores the potential of hardware obfuscation techniques in strengthening the AES implementations. The findings and analyses presented represent a valuable contribution to the continuous endeavors to bolster IC security and safeguard critical data from possible risks and vulnerabilities.

Yang et al. [6] present an innovative method that involves integrating chaos theory into the AES encryption process. This method leverages two separate chaos systems to produce two unique sequences. One of these sequences is utilized as the encryption key, while the other is responsible for governing the row-shifting operations. This creative fusion of chaos theory with AES leads to the creation of a novel encryption algorithm.

In the realm of database security, the protection of sensitive information and data integrity is of utmost importance. Zaw et al. [7] explore the use of multiple security techniques, including AES encryption, elliptic curve encryption, and digital signatures, to enhance the security of databases and proposed a comprehensive approach to safeguarding sensitive information in database systems.

Wu et al. [8] describes the swift advancements and global reach of semiconductor design and manufacturing have heightened the susceptibility of integrated circuits (ICs) to potential tampering, known as hardware Trojans. Notably, they point out that the Advanced Encryption Standard (AES) core, extensively used in security-critical applications, has emerged as a prime candidate for such hardware Trojan attacks.

In the realm of database security, the protection of sensitive information and data integrity is of utmost importance. Zaw et al. [7] explore the use of multiple security techniques, including AES encryption, elliptic curve encryption, and digital signatures, to enhance the security of databases and proposed a comprehensive approach to safeguarding sensitive information in database systems.

The description of the importance of selecting suitable encryption methods for data security in the (WSN's) wireless sensor networks that is provided by Panda [9]. The implementation of AES represents a significant step in reinforcing the security of WSNs and ensuring the confidentiality of the data transmitted within the network.

Liu et al. [10] underscore the significance of bolstering AES algorithm security by addressing possible vulnerabilities in its design, notably within the S-box and key schedule. The implementation strategy outlined here, customized for Java and PDA applications, represents a forward-looking approach to adapting to evolving security concerns and laying the foundation for secure digital-age communication.

Dilip and Sebastian [11] highlights the promising prospects of employing GPUs for accelerating AES encryption. By leveraging the high-performance computing capabilities of

GPUs, the AES algorithm can be significantly enhanced in terms of encryption speed, addressing the need for fast and efficient encryption in various contemporary applications.

Thiyagarajan and Kamalakannan [12] acknowledges the vital necessity of securing data in the age of the cloud computing and it underscores the significance of the AES algorithm. This importance is further underpinned by the incorporation of inventive data reading protocols and compression methods, all working collectively to attain this essential goal.

A sophisticated approach to mitigate side-channel attacks, specifically collision attacks, in the context of edge computing was introduced by Ding et al. [13]. Their approach, which capitalizes on the correlation between distance metrics and the reduction of plaintext candidate space, leads to a substantial boost in the efficiency of their proposed technique. These results carry significance for enhancing the security of edge devices in the ever-evolving landscape of security threats.

Dilip and Sebastian [11] highlights the promising prospects of employing GPUs for accelerating AES encryption. By leveraging the high-performance computing capabilities of GPUs, the AES algorithm can be significantly enhanced in terms of encryption speed, addressing the need for fast and efficient encryption in various contemporary applications.

Thiyagarajan and Kamalakannan [12] acknowledges the vital necessity of securing data in the age of the cloud computing and it underscores the significance of the AES algorithm. This importance is further underpinned by the incorporation of inventive data reading protocols and compression methods, all working collectively to attain this essential goal.

Using the Synopsys library and considering various supply voltages, Tsai et al. [14] conducted simulations. Their experiments' results are outstanding, showing a significant 62.0% reduction in dynamic power and an astonishing 88.5% decline in leakage power compared to traditional AES data encryption methods. Additionally, their security research highlights how the LPADA key update process enables mutual authentication between end nodes and application servers, enhancing the system's resistance to replay attacks and preventing hostile actors' attempts to eavesdrop on legitimate users.

Mitchell's study [15] underscores that 2-key triple DES, while not entirely compromised, offers only a minimal margin of security. Hence, there is a pressing need to explore alternatives for its replacement, with a preference for the 3-key version or a contemporary cipher like AES.

Each referenced work plays a vital role in strengthening the narrative, with the AES algorithm at its core. This varied collection of literature not only lays the foundation but also captures the fundamental essence of our exploration, smoothly connecting theoretical advancements with practical implications in the continuously evolving landscape of data security. The inclusion of each cited work is strategically placed to enhance the overall narrative, with particular emphasis on the pivotal role played by the AES algorithm. This diverse compilation of literature not only establishes the groundwork but also encapsulates the fundamental essence of our exploration. It adeptly intertwines theoretical progress with practical implications, providing a nuanced perspective in the ever-changing domain of data security.

Table 1. Comprehensive AES study review

S. No.	Title	Methodology	Findings
1	"AES Design Improvements Towards Information Security Considering Scan Attack" [1].	Scan attack analysis	Protection against scan attacks is effective
2	"Information Encryption Security System Based on Chaos Algorithm" [2].	Data analysis	Chaotic algorithm provides high information security
3	"Fast implementation of AES on Cortex-M3 for security information devices" [3].	Implementation and simulation	Enables secure communication with 128-bit encryption
4	"Improved Differential Fault Analysis on AES Key Schedule" [4].	Analysis and attack enhancements	Highlights the need to enhance AES security
5	"Enhancing Data Security using Obfuscated 128-bit AES Algorithm - An Active Hardware Obfuscation Approach at RTL Level" [5].	Hardware obfuscation	Enhances data security with minimal impact
6	"An Improved AES Algorithm Based on Chaos" [6].	Chaos theory integration	Develops a new encryption algorithm
7	"Database Security with AES Encryption, Elliptic Curve Encryption and Signature" [7].	Multiple security techniques	Enhances security in databases
8	"AES design improvement towards information safety" [8].	AES design analysis	Contributes to improving AES information safety
9	"Data security in wireless sensor networks via AES algorithm" [9].	AES implementation	Enhances data security in wireless sensor networks
10	"AES Algorithm Implemented for PDA Secure Communication with Java" [10].	Implementation and simulation	Achieves secure communication using AES with Java
11	"AES Encryption Algorithm Based on the High-Performance Computing of GPU" [11].	GPU-based AES acceleration	Significantly improves AES encryption speed
12	"Data integrity and security in a cloud environment using AES algorithm" [12].	Cloud security analysis	AES ensures data integrity and security in the cloud
13	"Adaptive Chosen-Plaintext Collision Attack on Masked AES in Edge Computing" [13].	Side-channel attack analysis	Introduces low-power AES encryption for edge computing
14	"Low-Power AES Data Encryption Architecture for a LoRaWAN" [14].	LPADA implementation	Achieves significant power reduction compared to traditional AES
15	"On the Security of 2-key Triple DES" [15].	Cryptanalysis of 2-key triple DES	Urges replacement of 2-key triple DES with more secure alternatives

3. METHODOLOGY

In the realm of data security, several encryption methods have been developed to safeguard sensitive information. These methods include the Rivest–Shamir–Adleman (RSA) algorithm, (DES) the Data Encryption Standard, Blowfish, (3DES) the Triple Data Encryption Standard, and many more. Each method employs distinct cryptographic techniques to protect data. The choice of encryption method often depends on the specific security requirements of an application. While these encryption methods are valuable, the Advanced Encryption Standard (AES) stands out as the preferred choice for various reasons. Notably, AES is renowned for its exceptional combination of security and efficiency, making it a suitable choice for a wide range of applications. Its adoption as a federal standard underscores its reliability and robustness. In this methodology, we will elucidate why AES surpasses its counterparts, covering key aspects such as encryption strength, computational speed, and versatility. The AES algorithm, adopted as a federal standard in the United States, is a symmetric key encryption algorithm. It supersedes (DES) the Data Encryption Standard and comprises various key lengths, with the most common ones being AES-128, AES-192, and AES-256 (Table 2).

Table 2. Variants of AES encryption

Variant	No. of Rounds	Key Size
AES-128	10 Rounds	128 bits
AES-192	12 Rounds	192 bits
AES-256	14 Rounds	256 bits

The widely embraced symmetric encryption algorithm, the Advanced Encryption Standard (AES), is renowned for its strong security (Figure 2).

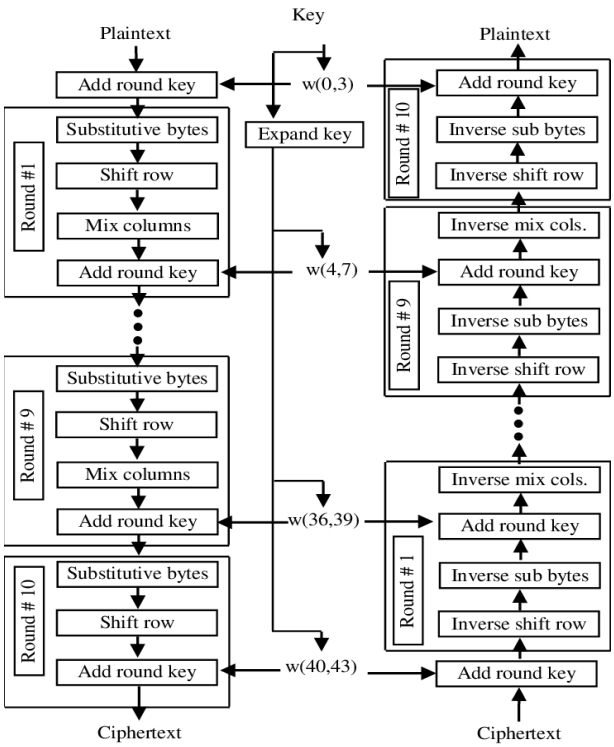

Figure 2. Block diagram for AES encryption and decryption

AES functions via a sequence of well-defined phases, classifying them into key expansion, substitution, permutation, and transformation stages.

3.1 SubBytes

In this step, AES employs a substitution table known as the S-box. The S-box replaces each byte of data with a corresponding value from the table. The S-box is designed to introduce confusion into the data, making it challenging for attackers to discern any patterns or relationships.

Given a byte **B[i][j]** in the State array, the substituted value is **S[i][j] = SBox[B[i][j]]**, where **SBox** is the predefined substitution box (Figure 3).

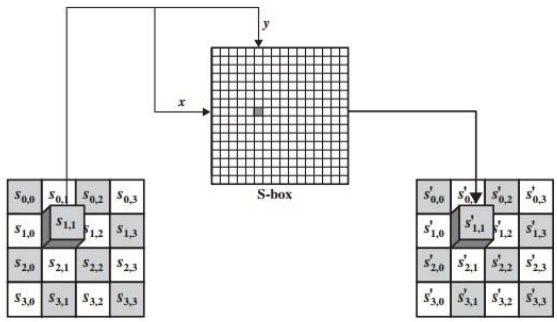

Figure 3. Substitution bytes transformation

The Inverse Sub Bytes operation involves replacing each byte in the State array with a corresponding byte from an inverse substitution box. Hence, the inverse substituted values are $\mathbf{S'[i][j] = SBox^{-1}\ [S[i][j]]}$ where **S'[i][j]** is the resulting state array, **S[i][j]** is the original state array, $\mathbf{SBox^{-1}}$ is the inverse substitution box.

3.2 ShiftRows

The ShiftRows step shuffles the data within the rows of the data block. In this process, bytes in each row are shifted left by different offsets. This permutation operation ensures that no simple linear relationships exist between the input and output, adding another layer of complexity to the encryption.

For the State array **S[i][j]**, the shifted value is **S[i][j] = S[i][(j + i) mod 4]** (Figure 4).

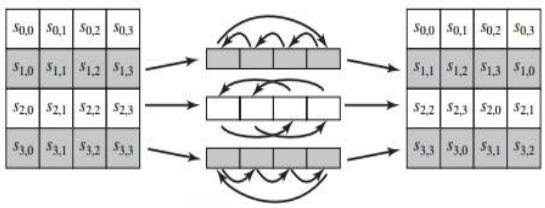

Figure 4. Shift row transformation

The Inverse Shift Rows operation is the reverse of the Shift Rows operation. It involves shifting the bytes of each row in the State array to the right. Hence, the inverse shifted values is **S'[i][j] = S[i][(j - i) mod 4]** where, **S'[i][j]** is the resulting state array, **S[i][j]** is the original State array, the **(j - i) mod 4** expression represents the backward shift.

3.3 MixColumns

In the MixColumns stage, AES applies a mathematical transformation to the columns of the data block. This transformation mixes the bytes within each column, further increasing the encryption's diffusion properties.

If C is the column matrix representing the State column and M is the fixed matrix used for multiplication, the result is **C' = M * C**, where **C'** is the resulting column.

The Inverse MixColumns operation involves multiplying each column of the State array with an inverse matrix. So, the inverse mixed column values are **C'[i] = C[i]*M^(-1)** where **C'** is the resulting column, **C[i]** is the original column, **M ^ (-1)** is the inverse MixColumns matrix (Figure 5).

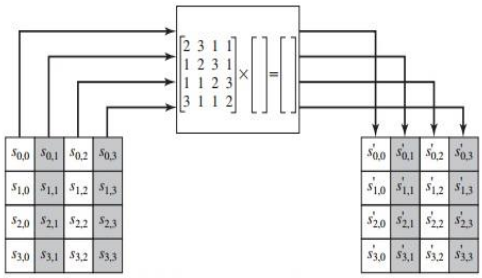

Figure 5. Mix column transformation

3.4 Add round key

At the end of each round, a round key generated during the key expansion phase is XORed with the data. This step adds a unique portion of the round key to the data, ensuring that each round of encryption is distinct, and the key material is integrated.

In this step, each byte of the State array is XORed with a byte from the round key. For a byte **S[i][j]** in the State array and the corresponding round key byte **K[i][j],** S**[i][j] = S[i][j] ⊕ K[i][j]** (Figure 6).

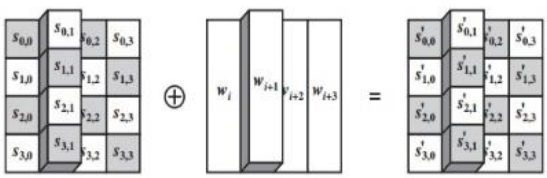

Figure 6. Add round key transformation

Overall, AES stands out as the top choice for data encryption in the field of data security due to its exceptional balance of security, efficiency, and adaptability. Its range of key lengths allows it to cater to various security needs in different applications. An in-depth examination of the AES algorithm reveals the complexity that underlies its robust security features. This thorough grasp of AES paves the way for the upcoming sections of this research, where we explore its practical application and assessment to enhance data security and mitigate potential risks.

4. ALGORITHMIC FRAMEWORK

Before delving into the practical implementation of the (AES) Advanced Encryption Standard, it is essential to grasp the underlying algorithms that make AES a robust encryption method. This section provides a comprehensive overview of the AES algorithm, detailing its encryption and decryption processes. Understanding these cryptographic algorithms is paramount to appreciating how AES ensures data security. The algorithms presented here set the stage for subsequent implementation and practical application of AES.

Algorithm: AES Encryption and Decryption

Input:
- User-selected file
- Encryption or decryption mode
- User-provided key

Output:
- Encrypted or decrypted file

Step 1: User Interaction
- The user selects a file for encryption or decryption.
- The user specifies the mode (encryption or decryption).
- The user provides a key.

Step 2: AES Encryption
- If encryption mode is selected:
- The application initiates encryption with the provided key.
- The result is an encrypted file.

Step 3: AES Decryption
- If decryption mode is selected:
- The application initiates decryption with the provided key.
- The outcome is the restored file in its original form.

Step 4: Save File Dialog
- Users are offered the option to save the resulting file.
- A customizable "Save File" dialog allows users to specify the destination and file name for the encrypted or decrypted file.

Now we outline the algorithm for creating a user-friendly desktop application designed to interact seamlessly with the AES encryption and decryption processes. The features offered by the application are.

1. **User Input:** The application features a file input field that enables users to select the file they wish to encrypt or decrypt. The file path is captured, and users are prompted to choose between encryption and decryption modes.

2. **Encryption with AES:** For encryption, the chosen file is processed through the AES encryption algorithm. A user-provided encryption key is utilized in this process. The encrypted file is then saved in the specified destination, ensuring that sensitive data is securely protected.

AES.Encrypt(inputFile, outputFile, encryptionKey);

3. **Decryption with AES:** On selecting decryption mode, the application applies the AES decryption algorithm to the chosen file, utilizing the provided decryption key. The decrypted file is **saved** to the designated location, making the original data accessible again.

AES.Decrypt(inputFile, outputFile, decryptionKey);

4. **File Selection Functionality:** The file selection mechanism using OpenFileDialog tool. Allow users to browse and select the file they want to encrypt or decrypt.

```
OpenFileDialog openFileDialog = new OpenFileDialog();

if (openFileDialog.ShowDialog() == DialogResult.OK)
{
string selectedFilePath = openFileDialog.FileName;

filePathTextBox.Text = selectedFilePath;
}
```

5. **Customization with Save File Dialog:** To enhance user experience, a "Save File" dialog is incorporated into the application. This feature empowers users to choose the location and filename when saving their encrypted or decrypted files.

```
SaveFileDialog saveFileDialog = new SaveFileDialog();

if (saveFileDialog.ShowDialog() == DialogResult.OK)
{
string filePath = saveFileDialog.FileName;

Console.WriteLine("File saved to: " + filePath);
}
```

This algorithm serves as the foundation for a user-friendly desktop application that empowers users to perform AES encryption and decryption with ease.

5. IMPLEMENTATION

In the preceding sections, we have laid the foundation for our study on the Advanced Encryption Standard (AES). We have explored its significance, delved into the intricacies of its algorithm, and discussed the method of implementation. Now, it's time to put our knowledge into practice and implement AES encryption and decryption in a real-world scenario.

Let's explore the steps taken to develop this application.

5.1 Application setup

Begin by launching Visual Studio and creating a new Windows Form App (.Net Framework) project (Figure 7).

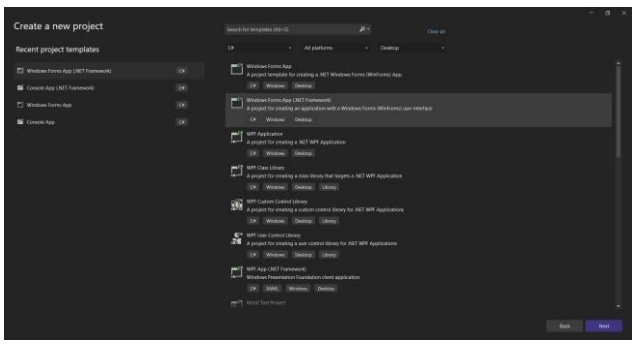

Figure 7. Creating a new project

5.2 Form configuration

Configure the project by adding the necessary tools to the form (Figure 8).

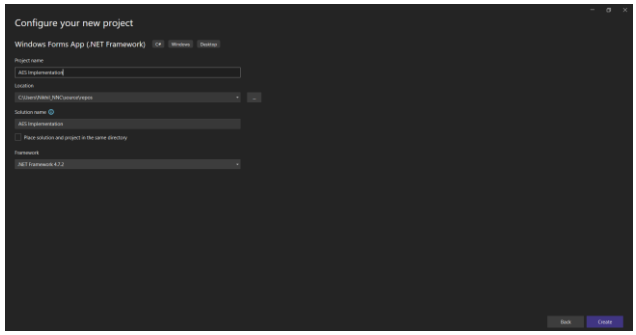

Figure 8. Configuring project

5.3 Start application

After the form is set up, start the application. The interface is designed to facilitate user interactions (Figure 9).

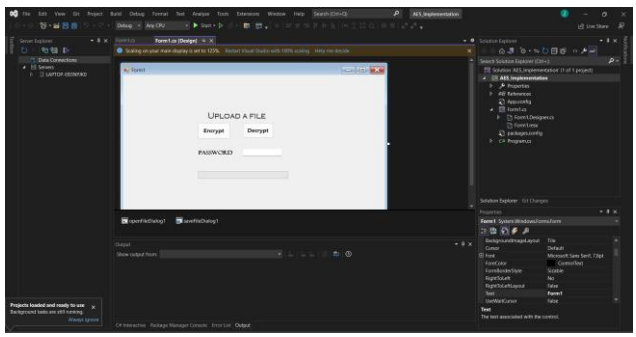

Figure 9. Run the application

5.4 File encryption

When selecting the "Encrypt" option, users can upload a file. In this example, a text file is used for clarity. After selecting the file and entering a password, click "OK." (Figure 10).

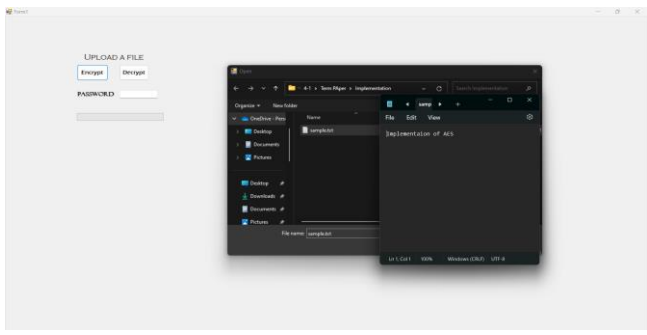

Figure 10. Upload file for encryption

5.5 Destination selection

The application prompts users to choose where to store the encrypted file. Users also have the flexibility to choose the file's name (Figure 11).

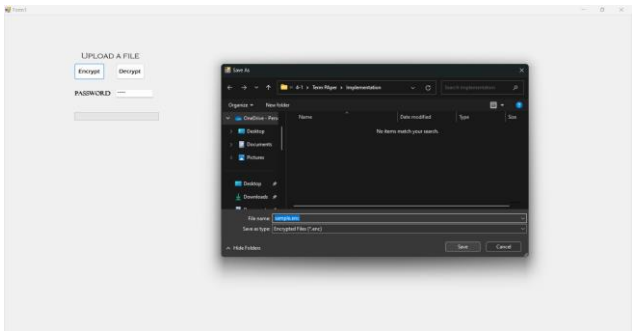

Figure 11. Select the destination path

5.6 Successful encryption

The file is successfully encrypted, ensuring data security (Figure 12 and Figure 13).

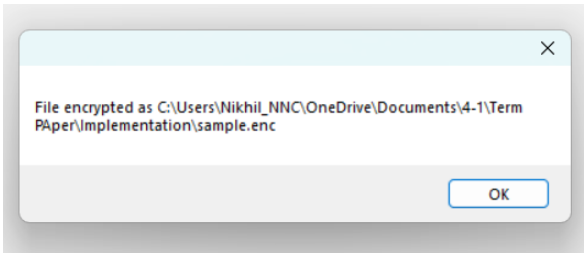

Figure 12. Message box with successful encryption message

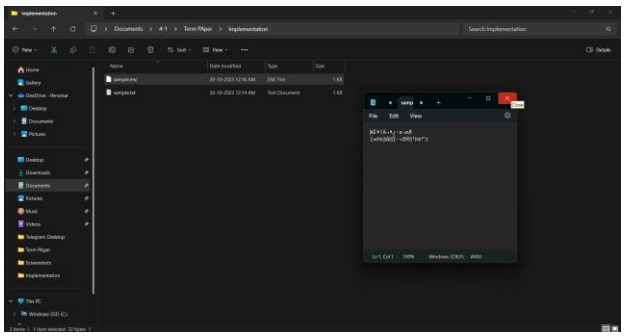

Figure 13. The encrypted text

5.7 Decryption

The same process applies to decryption. After selecting the "Decrypt" option, choose the file to decrypt (Figure 14).

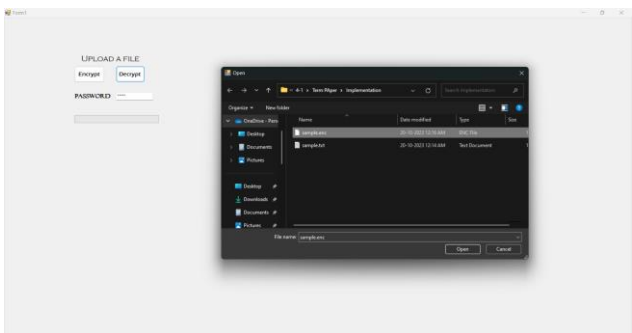

Figure 14. Upload file for decryption

5.8 Password entry

Enter the decryption password, and the application will decrypt the file. Users can choose the location for storing the decrypted file (Figure 15 and Figure 16).

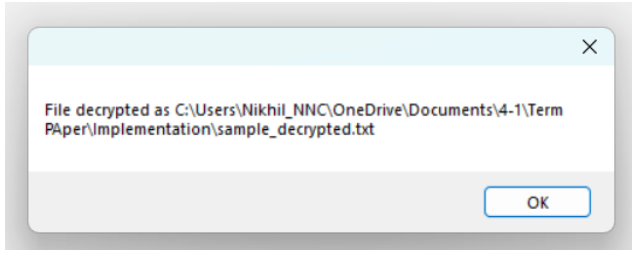

Figure 15. Message box with successful decryption message

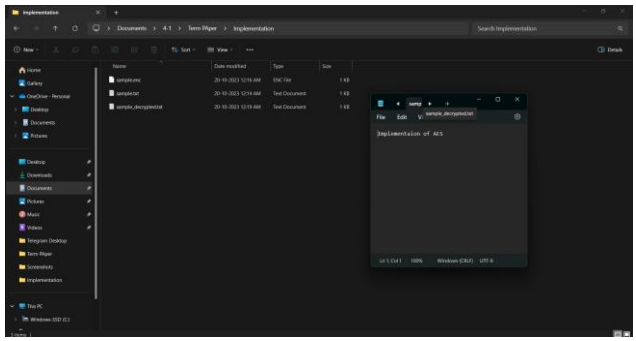

Figure 16. The decrypted text

The implementation of AES encryption and decryption serves a paramount purpose. In an era marked by growing concerns over data security, safeguarding sensitive information is of utmost importance. By developing and utilizing this application, we equip users with a practical tool to protect their data with strong encryption. Whether it's personal files, confidential documents, or critical information, the significance of AES encryption cannot be overstated. The robust encryption capabilities of AES, coupled with the ease of our application, allow users to protect their data with confidence.

6. CONCLUSION

This study illuminates the crucial role of encryption techniques, with the Advanced Encryption Standard (AES) standing out as a robust solution, balancing security, efficiency, and adaptability. Despite resource constraints impacting our implementation, the work underscores the vital importance of encryption in the digital era. AES, with its varied key lengths, remains a preferred choice across applications, emphasizing its significance in contemporary data security.

While the study acknowledges limitations in its implementation, it reaffirms the pivotal role of encryption, urging further research and development to enhance methods like AES in addressing evolving data security challenges. As the digital landscape advances, the study emphasizes the ongoing need for robust encryption methods.

In future works, we are poised to integrate the implemented solution from this paper into real-world applications. This strategic move is designed to bolster the security of real-world data, aligning with the ever-evolving demands of our interconnected world. The seamless transition from theory to application underscores the study's practical relevance, emphasizing its potential to make a substantial impact on addressing contemporary data security needs.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to G. Raja and P. Venkateswara Rao for their guidance during this project.

REFERENCES

[1] Yu, L.T., Zhang, D.R., Wu, L., Xie, S.G., Su, D.L., Wang, X.X. (2018). AES design improvements towards information security considering scan attack. In 2018 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/ 12th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE), New York, NY, USA, pp. 322-326. https://doi.org/10.1109/TrustCom/BigDataSE.2018.00056

[2] Liu, X.Y. (2022). Information encryption security system based on chaos algorithm. In 2022 7th International Conference on Cyber Security and Information Engineering (ICCSIE), Brisbane, Australia, pp. 128-131. https://doi.org/10.1109/ICCSIE56462.2022.00033

[3] Wardhani, R.W., Ogi, D., Syahral, M., Septono, P.D. (2017). Fast implementation of AES on Cortex-M3 for security information devices. In 2017 15th International Conference on Quality in Research (QiR): International Symposium on Electrical and Computer Engineering, Nusa Dua, Bali, Indonesia, pp. 241-244. https://doi.org/10.1109/QIR.2017.8168489

[4] Kim, C.H. (2012). Improved differential fault analysis on AES key schedule. IEEE Transactions on Information Forensics and Security, 7(1): 41-50. https://doi.org/10.1109/TIFS.2011.2161289

[5] Chhabra, S., Lata, K. (2018). Enhancing data security using obfuscated 128-bit AES algorithm - An active hardware obfuscation approach at RTL level. In 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Bangalore, India, pp. 401-406. https://doi.org/10.1109/ICACCI.2018.8554562

[6] Yang, Z.H., Li, A.H., Yu, L.L., Kang, S.J., Han, M.J., Ding, Q. (2015). An improved AES algorithm based on chaos theory in wireless communication networks. In 2015 Third International Conference on Robot, Vision and Signal Processing (RVSP), Kaohsiung, Taiwan, pp. 159-162. https://doi.org/10.1109/RVSP.2015.45

[7] Zaw, T.M., Thant, M., Bezzateev, S.V. (2019). Database Security with AES Encryption, Elliptic Curve Encryption and Signature. In 2019 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF), St. Petersburg, Russia, pp. 1-6. https://doi.org/10.1109/WECONF.2019.8840125

[8] Wu, L., Wang, X.X., Zhao, X.Y., Cheng, Y.Q., Su, D.L., Chen, A.X., Shi, Q.H., Tehranipoor, M. (2016). AES design improvement towards information safety. In 2016 IEEE International Symposium on Circuits and Systems (ISCAS), Montreal, QC, Canada, pp. 1706-1709. https://doi.org/10.1109/ISCAS.2016.7538896

[9] Panda, M. (2015). Data security in wireless sensor networks via AES algorithm. In 2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO), Coimbatore, India, pp. 1-5. https://doi.org/10.1109/ISCO.2015.7282377

[10] Liu, N.S., Guo, D.H., Huang, J.X. (2007). AES algorithm implemented for PDA secure communication with Java. In 2007 International Workshop on Anti-Counterfeiting, Security and Identification (ASID), Xizmen, China, pp. 217-222. https://doi.org/10.1109/IWASID.2007.373730

[11] Dilip, V., Sebastian, S. (2017). Date security in smartphones using bit-locker technology. International Journal of Advanced Research (IJAR). http://dx.doi.org/10.21474/IJAR01/3698

[12] Thiyagarajan, B., Kamalakannan, R. (2014). Data integrity and security in cloud environment using AES algorithm. In International Conference on Information Communication and Embedded Systems (ICICES2014), Chennai, India, pp. 1-5. https://doi.org/10.1109/ICICES.2014.7033923

[13] Ding, Y., Shi, Y., Wang, A., Zheng, X., Wang, Z., Zhang, G. (2019). Adaptive chosen-plaintext collision attack on masked AES in edge computing. IEEE Access, 7: 63217-63229. https://doi.org/10.1109/ACCESS.2019.2916553

[14] Tsai, K.L., Leu, F.Y., You, I., Chang, S.W., Hu, S.J., Park, H. (2019). Low-power AES data encryption architecture for a LoRaWAN. IEEE Access, 7: 146348-146357. https://doi.org/10.1109/ACCESS.2019.2941972

[15] Mitchell, C.J. (2016). On the security of 2-key triple DES. IEEE Transactions on Information Theory, 62(11): 6260-6267. https://doi.org/10.1109/TIT.2016.2611003