Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Realization of Eavesdropping Attacks on Dynamic Encryption Decryption Algorithm (DEnDecA) for IoT Applications

Jasvir Singh Kalsi[1], Jagpal Singh Ubhi[2], Kota Solomon Raju[3]

[1]Research Scholar, Department of Electronics and Communication Engg., SLIET Longowal
[2]Professor, Department of Electronics and Communication Engg., SLIET Longowal
[3]Senior Principal Scientist, Aerospace Electronics & Systems Division, CSIR-NAL Bangaluru
Email: jasvirkalsi@sliet.ac.in

The development of cryptographic systems that achieve security and computational efficiency in resource constrained environments of Internet of Things (IoT) networks for secure communication is a dire need. For IoT devices, the light weighted cryptographic algorithms are devised by various researchers. The works include designing new approaches as well as downscaling the existing framework to reduce the parameters such as operational time, computational requirements, memory requirements and power backup. Reducing either computation or iteration of memory requirements may compromise the security of these derivative designs. The use of Dynamic Encryption Decryption Algorithm (DEnDecA), an encryption framework that chooses among multiple cryptographic algorithms, on improving overall security has been discussed in this paper. The algorithms uses a Pseudo Random Number (PRN) generator to select encryption method at random. In its dynamic selection process it obscures the encryption scheme to possible attackers, and maintains robust protection even when less secure algorithms such as AES-32 and AES-64 are utilized. An encrypted part of the transmitted data is available only to the recipient that it is addressed to, which makes it resistant to Physical layer attacks such as Eavesdropping, man-in-the-middle and side channel attacks while also providing this capability. DEnDecA has been implemented in a computationally efficient manner, and is therefore suitable for IoT applications where low power and limited processing capabilities are the norm. Experimental evaluations demonstrate the simulation of Eavesdropping attacks on the DEnDecA envelop and shows the operation of the algorithm against the eavesdroppers hence enhancing data confidentiality and integrity, while maintaining a negligible impact on system resources.

Keywords: Advanced Encryption Standard, IoT Security, light weight cryptography, Dynamic Encryption, Physical layer security

1. Introduction

The recent development of wireless communication technologies has brought important changes to the flow of data through networks that contributed with some risks primarily in regards to the security of the shared information. With regard to IoT systems, the protection of data that are transmitted between nodes and hubs from unauthorized access poses an ever growing issue. Among the developed theories attempting to alleviate these problems, physical layer security (PLS) is a relatively recent research direction. PLS's aims at designing schemes to provide data security at the physical layer of the communication systems using the randomness of the wireless channel and noise inherent in the channel to offer security, apart from employing secure cryptographic methods [1]. However, critical as it is to the performance and potentials of IoT systems, the physical layer is not immune to multiple form of attacks, with eavesdropping being one of the most common at the device level of IoT. Classical techniques for encryption of data are based on the use of what is known as symmetric or bilateral key systems, in which data encrypted and data decrypted use the same key.

For IoT devices the use simple case of symmetric encryption is preferred, because there are less requirements to on-board processor at distant IoT nodes. Although these algorithms offer substantial security gains, they may offer at best only partial protection against such snooping attacks, especially in situations where the attackers have access to signal processing and cryptanalysis resources. That raises the fundamental question of how physical layer security methods can be integrated with encryption techniques to protect against interception and decryption of transmitted information. After recent evolution in lightweight AES algorithms for IoT, they have focused on equal level of security along with better performance. One interesting development is the Lightweight AES (LAES), which was specially designed for low-energy microcontrollers such as those employed in IoT gadgets. LAES is as fast as the traditional AES and other lightweight cryptographic techniques such as PRESENT and CLEFIA as has been brought out by performance assessments [2]. These evaluations highlight that LAES offers strong randomness while substantially reducing both processing time and energy consumption, making it particularly suitable for IoT applications.

To protect image data in smart IoT devices, new techniques such as Preserve Lightweight Image Encryption (PLIE) have been developed [3]. As compared to AES implementation, PLIE enhances throughout and decreases encryption time around 50%. This approach effectively partitions and randomizes the load with the added bonus of protecting user identity and two critical performance factors that are essential in IoT applications. These are important especially in countering security threats while at the same time, taking into account the functionality and real use of many IoT devices. Going further than simple software optimization of AES lightweight designs, hardware accelerators have been explored to enhance it. Out of the two matrix remapping techniques, the findings of studies reveal that using an FPGAs or ASICs can improve the performance of lightweight AES algorithms[4].

All these hardware solutions are subjected to tests against different attacks such as side channel attacks and are designed to make encryption secure even in resource constrained environments[5]. Nevertheless, designing the lightweight AES algorithms is still a concern of difficulties. Conserving resource is normally accompanied by compromise on encryption's speed or it incorporates some weaknesses that may be exploited by attackers. When widespread in key industries like healthcare, transportation, and energy, IoT calls for effective cryptosystems even more. As more connected devices deal with sensitive information needs for lightweight cryptographic solutions will probably rise. By contrasting different lightweight cryptographic algorithms for the Internet of Things, it becomes possible to identify their suitable and unsuitable applications. For instance, though LAES achieves high security with the best performance for constrained platforms, there are other algorithms such as SPECK and ASCON that come with entirely different security and complexity characteristics[4][5][6]. Finding the suitable lightweight encryption for the particular application depends not only on certain requirements or peculiarities of the application itself but also on the consequences that stem from inadequate security measures.

Many researchers have worked on downscaling of AES-128 algorithm to reduce the datapath of the key state for light-weight option of encryption IoT constrained devices. The proposals of the researchers concentrated in designing AES-64 and AES-32 variants in which, the computational requirements are significantly decreased. The reduction in datapath as well as key state array also minimizes the on-board memory requirements. The challenge faced in the downscaling is that there is a trade-off with the security measures. The data flow at physical layer is available freely especially in case of wireless transmission. There are many possible intrusions and data breach probability is high. The data can be easily intercepted by the attackers at physical layer but unless the data is secured with a reliable security algorithm, it possess no harm. The secure encryption technique acts as a backbone to the physical layer. In case of downscaling the AES-128 algorithm, the data becomes quire vulnerable to the eavesdropping attacks. This possess a challenging situation for IoT applications.

Eavesdropping attacks are among the most common and harmful threats at the physical layer. These attacks exploit the fact that electromagnetic waves, which carry information, are broadcast through the air, making them inherently vulnerable to interception. While conventional encryption algorithms can secure data at higher layers of the communication stack, they may not fully address the vulnerabilities present at the physical layer.

The eavesdroppers do not need to tap into the source or destination of the signal transmitted in the wireless communication; they only have to be near to the signal source. In case of IoT, the devices are power constrained devices with less computation power and marginally better memory. Thus, protection of the physical layer is a critical in preventing other entities from intercepting and signals and alter the communicated data. Several approaches have been investigated to mitigate different physical layer eavesdropping: artificial noise, beamforming, and cooperative communication [7]. These approaches are intended to increase the level of protection for wireless systems by complicating the understanding of signals intercepted by an adversary. The advancements in lightweight AES algorithms are a positive sign that have shown, successful ways of addressing problems of IoT applications are in process. This way, the researchers are working on improving the security measures while, at the same time, improving the communication efficiency in different IoT networks. There are still some aspects that need to be figured out before the simultaneous use of encryption algorithms and physical layer security to counteract eavesdropping is most effectively achieved. A major challenge is the development of effective encryption algorithms in B3G mobile systems that have to be secure against the attacks at the physical layer, and at the same time they have to consume moderate computational and power resources. For as this field grows, more research will be important so that lightweight cryptographic innovations can address newly emerging security threats while satisfying the stringent challenges associated with current IoT deployments. The future applied research and development question will be to find the optimal balance between the high security level and the impossible or at least very difficult if not impossible to implement on devices with limited resources. However, there is still much that needs to be learned about the relationship between encryption algorithms and the characteristics of wireless channels, including fading and interference[8].

In this research, the proposed DEnDecA in [9] is implemented on downscaled AES derivatives as discussed below. This research aims to investigate the susceptibility of downscaled AES-128 encryption algorithm to eavesdropping at the physical layer norm and recommend ways of improving the effectiveness of the schemes in protecting wireless communication systems. Using literature review and eavesdropping simulations, this research intends to present a clear revelation on the difficulties realized when implementing physical layer security in symmetric encryption. In addition, it aims to find the best strategy to strengthen the data breach in downscaled AES derivatives [9]. The research tries to find the solution by dynamically changing the algorithms used in encrypting at physical layer hence offering a better chance in fending off attacks at this physical layer. The result of this study adds to the literature in physical layer security development to enhance the IoT communication system’s reliability and security.

2. Dynamic Encryption Decryption Algorithm (DEnDecA)

The increase in the need for secure communication in IoT requires effective encryption protocols, although the IoT devices have limited resources. These increases, along with limits on processing power, memory, and energy storage, require the use of cryptographic protection schemes for data security against a growing variety of threats [10]. Out of all the encryption models, the most commonly used area the Advanced Encryption Standard or AES, AES-128, AES-192, AES-256 of them. All three variants of AES operate on a 128-bit block size, with the primary distinction being the length of the encryption key: 128 bits, 192 bits and 256 bits each [11]. An architectural overview of AES is as shown in figure 2.1.

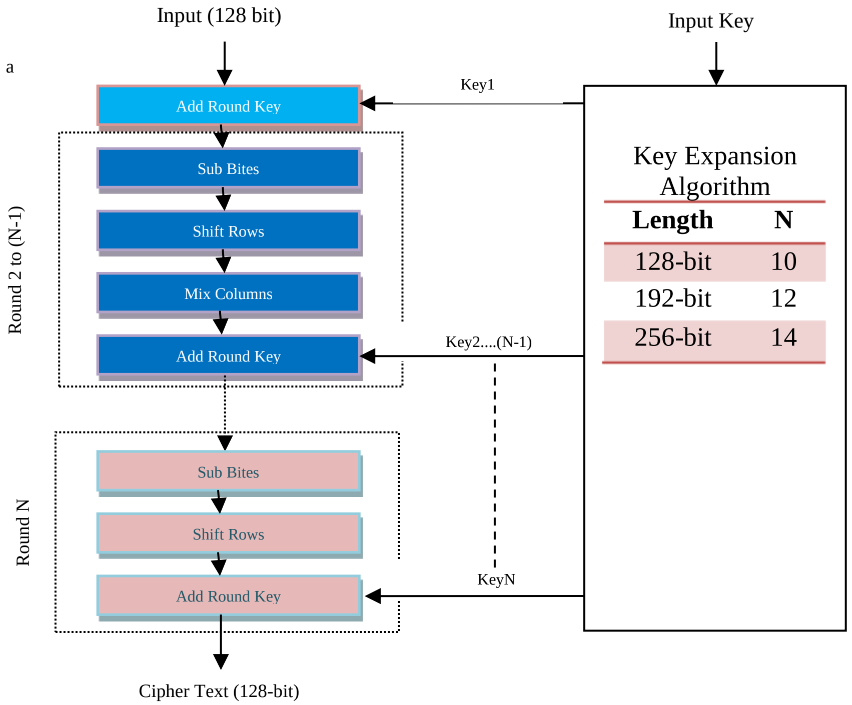

Figure 2.1: AES Architecture

While the AES algorithm has been extensively studied in software implementations, its hardware realization is of equal importance, particularly in the context of low-power, resource-constrained IoT devices. These devices often lack sufficient computational resources and energy, making hardware implementations of cryptographic algorithms essential. Researchers have proposed various hardware architectures for AES, each aiming to either minimize area usage or maximize performance. These optimizations offer IoT system designers a broad range of solutions to choose from, depending on the specific requirements and constraints of their applications [4]. Hardware-based implementations generally outperform software-based solutions in terms of both processing speed and energy efficiency, which is particularly crucial for IoT devices that must meet stringent power consumption and performance criteria. AES is implemented in a 2-dimensional array of bytes. These states are arranges as shown in figure 2.2.

S00	S10	S20	S30
S01	S11	S21	S31
S02	S12	S22	S32
S03	S13	S23	S33

Figure 2.2: Two dimensional ‘State’ array

The operation of AES is sub-divided in distinct byte-oriented iterative transformations called rounds. In the process of encryption, these rounds further are composed of a sequence-based byte-transformations such as substitution of byte from S-box (Sub-byte), shifting of each row horizontally in a selective pattern (Shift-Rows), column state mixing of data (Mix-Column) and XOR operation or modulo 2 addition of the state with the round Key (Add Round-Key) and the plain text is transformed into secure cipher text. The decryption process is exactly identical but reciprocal of encryption to decrypt plain text from cipher[12][13].

In the context of IoT, these hardware solutions are critical, as they not only ensure data confidentiality but also address the need for authenticated encryption, which ensures the integrity and authenticity of the transmitted data [14]. AES is recognized for its versatility and reliability in securing data transmission. The algorithm follows a Feistel structure, with a 128-bit plaintext block being processed through multiple rounds of encryption. The number of rounds varies depending on the key size: 10 rounds for a 128-bit key, 12 rounds for a 192-bit key, and 14 rounds for a 256-bit key. This iterative process converts the plaintext into ciphertext, ensuring high levels of security across different key sizes. In our research, we have downscaled AES algorithm breaching the process of Fiestel Structure followed by AES algorithm. The Key size and number of rounds are reduced to design AES-32 and AES-64 algorithms as proposed in [9]. Then the DEnDecA envelop is applied on the downscaled algorithms.

The Dynamic Encryption Decryption Algorithm (DEnDecA) architecture is distinguished by its capability to dynamically switch between encryption algorithms without requiring any input from the user. In this approach, the details of the selected encryption algorithm are transmitted alongside the data as part of an 8-bit header byte. Upon receipt of this data, the plaintext is encrypted using the chosen algorithm, resulting in ciphertext. The dynamic nature of the algorithm prevents unauthorized decryption by intruders, as they are unaware of which encryption method was employed. A Pseudo-Random Number (PRN) generator is employed to randomly select an algorithm for encryption. This number is consistently used at both the transmission and reception terminals, ensuring that both ends are synchronized in their choice of algorithm [9].

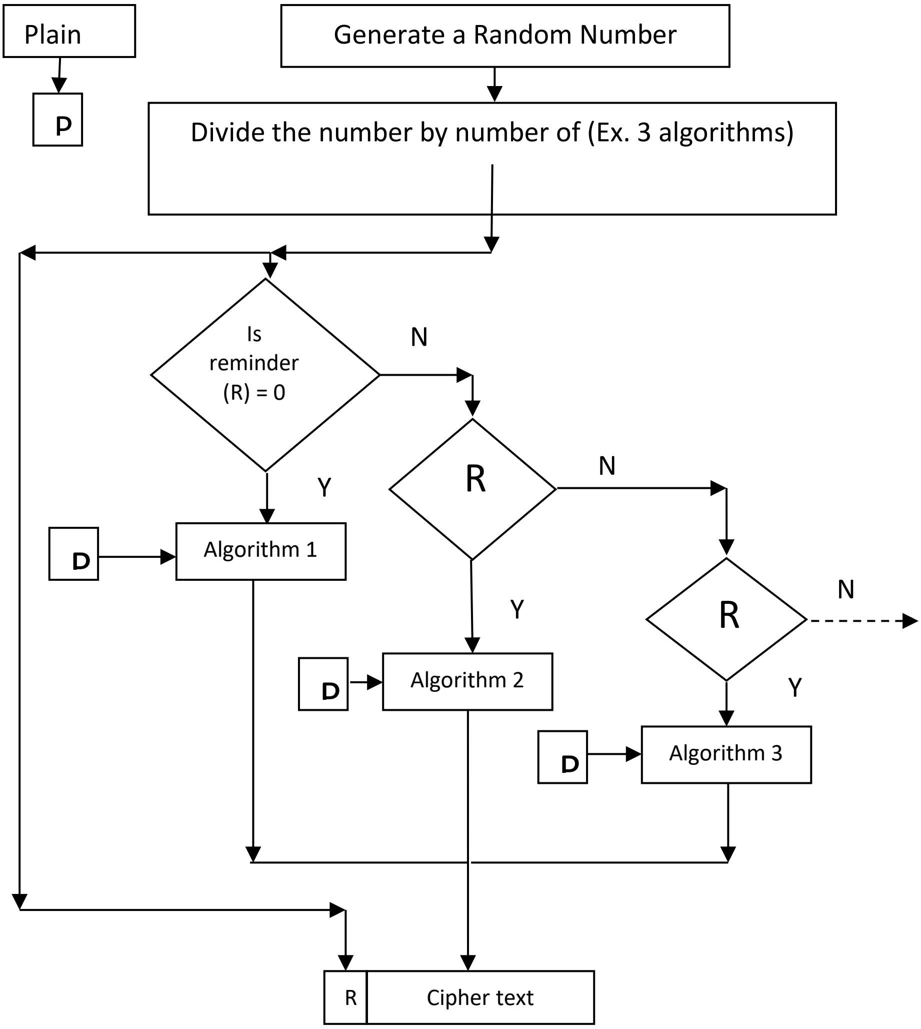

Figure 2.2: Architecture of DEnDecA

Before the encryption process, the PRN generates a random number, which is then divided by the total number of available encryption algorithms in the DEnDecA system. The remainder of this division indicates the specific algorithm to be used. For instance, if three algorithms—such as AES-32, AES-64, and AES-128—are programmed into the system, the PRN might generate the number ‘47’. When divided by the total number of algorithms (in this case, ‘3’), the remainder is ‘2’, indicating that AES-64 should be selected for the encryption process. The corresponding algorithm number is then transmitted as an 8-bit header along with the ciphertext to the receiving terminal. This header, though necessary, imposes minimal overhead, especially when encrypting large volumes of data. At the receiving end, the first 8 bits of the incoming data are extracted to identify the algorithm used for encryption. This number is then employed to select the appropriate decryption algorithm, ensuring that the ciphertext can be correctly decrypted and converted back to plaintext[9][13]. For instance, encrypting 1MB of data results in only an 8-bit increase in data size. The concept of DEnDecA is shown in Figure 2.2

This process enhances the security of the communication in multiple ways. First, by automatically selecting the encryption algorithm, the system ensures that the specific algorithm used to encrypt the data is unknown to the attacker, even at the time of transmission. Although AES-32 and AES-64 provide lower levels of security than AES-128, the security strength lies in the randomization of the algorithm selection itself, rather than in the inherent strength of any single algorithm. The unpredictability of the encryption method prevents an attacker from targeting specific vulnerabilities associated with one encryption algorithm. Second, the dynamic nature of the algorithm selection significantly reduces the likelihood of man-in-the-middle and micro-probing attacks. Because the algorithm changes with each data chunk, even if an attacker successfully breaches the encryption once, they would only have access to a single segment of the data stream. When the next data chunk is available, the algorithm shifts again which makes it very hard for the attacker to stay connected. This dynamic behavior provides more security than the static forms of encryption like the AES 128. In the case of IoT which tends to work under constrained bandwidth, the AES-32 and AES-64 are properly suitable for encrypting data. These algorithms offer a good proportion of protection and usage consumption as they use less CPU than AES-128 and still enough to protect the data. The DEnDecA system is easy to deploy and introduces no overhead and, therefore, is suitable for IoT devices with a low processing ability. Yet, the implementation of the algorithm guarantees the confidentiality and data’s informational safety, even with restricted computational power.

3. Results and Discussion

Due to the growth of wireless networks, eavesdropping attacks are considered major threats to the privacy and confidentiality of communications. These attacks exploit how wireless channels are designed that data can be accessed by unauthorized people. In wireless communication data is transmitted through radio signals and therefore anyone maybe be in a position to intercept them as the signal has to pass through the air [15]. Wireless networks have becoming a requirement to any modern network but this also comes with the risk of security threats. Eavesdropping is a primary threat for communication systems because the signals propagated by the communicating parties can be intercepted by unauthorized third parties, which endanger the confidentiality of transmitted information. Wireless network on the other hand are easily accessible since they emit signals in the environment and therefore very easy to intercept[16].

At the Physical Data Layer where data is transmitted over Radio Frequencies, there is vulnerability to tapping. This layer is used to guarantee that communication of information will be rightful and secure, however it presents a number of prospects for the intruder or other unauthorized personnel to penetrate. For instance, there is a high risk of private data leakage undermining the general data availability, security, and confidentiality [17].

Eavesdropping is a kind of interception; the attacker listens and perhaps decode the exchanged information without disrupting the actual communication. These attacks are most effective in the wireless environments because of the Transmission mediums used by the signals. The chief concern in the physical layer is the tap on the radio signals conveying sensitive information [18]. Some techniques for instance, frequency hopping or even signal jamming can easily be exploited by the attackers. At times they might record the frequencies being used in the radio spectrum for communication. It means that such signals enable the eavesdropper to obtain a password or encryption key, for instance.

As to eavesdropping attacks in the physical layer, the main type cunning the architecture of the used network and the skills of the intruder [19]. The kinds of eavesdropping attacks that follows are often experienced; Passive Eavesdropping is the simplest form of the attack where the attacker floats around and captures the content of the communications link without attempting to communicate with the network. The passive dumping does not modify the operation of the system in any way; therefore, it is hard to notice. Another type is the Active Eavesdropping where the attacker listens to the communication and in some cases tries to alter the message being passed out or even inject other wrong information into the passing out message. This could lead them to reveal some unwanted information or even bring in question the authenticity of the communication signal. On of a very prominent attack is Man-in-the-Middle Attack. While technically is more than simply an eavesdropping attack the Man-in-the Middle attack describes a scenario in which an unauthorized third party inserts themselves into the communication between two parties and can possibly modify the contents of the transmitted information. The attacker can then listen to the entire dialogue and may also supply fake data to the communication[17][19][20].

As our algorithms are to be tested in real time environment on physical layer, the testing of designed algorithm was performed by subjecting the data encryption to Eavesdropping attacks. There are several parameters that can be done out of which, our research concentrated on Power Analysis, Throughput and Time Analysis of the noisy signal that contains the encrypted data, Eavesdropping signal and Legitimate signal. The analysis was performed on MATLAB where an arbitrary eavesdropping environment is created to measure the actual values of performance parameters. The analysis was done with the comparison of these parameters with respect to channel quality starting from 3dB noisy channel to 30dB SNR channel. The readings are recorded at a resolution of 1dB.

The results of various parameters are as follows:

Power Analysis:

The Power analysis of the eavesdropping attack shows the power of Encrypted Signal Power, Noisy signal Power and Eavesdropping Signal Power against the Signal to Noise Ration of the Channel. In ideal case, the encrypted signal power must remain constant and as SNR increase from lower values to 30 dB, the channel becomes ideal. The Noisy signal power and eavesdropping signal power values should get lowered as the channel approached 30dB SNR. The results of the analysis is summarized in table 3.1. The results at a resolution of 1dB SNR are shown in figure 3.1. The figure shows that in noisy channel, the power of eavesdropping signal is high which in turn increases the noisy signal power. As the implementation moves towards better SNR channel at 30dB, the power of encrypted signal remains constant but the eavesdropping signal power decreases which in turn decreases the noise power. This analysis shows that in ideal conditions of Eavesdropping attack, DEnDecA environment performs satisfactory. The down-scaling of AES algorithm has no effect on the power as the signal power remains constant in attack environment.

Table 3.1: Power Analysis of Enc. Signal Power, Noisy Signal Power and Eavesdropper Signal Power

SNR (dB)	Encrypted Signal Power (dB)	Noisy Signal Power (dB)	Eavesdropper Signal Power (dB)
0	42.52	45.41	49.01
5	42.52	43.43	45.55
10	42.52	42.97	43.76
15	42.52	42.63	42.91
20	42.52	42.58	42.69
25	42.52	42.54	42.53
30	42.52	42.51	42.50

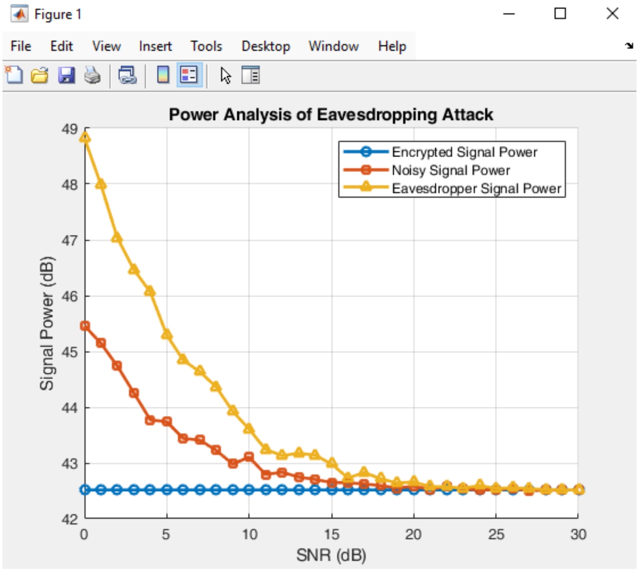

Figure 3.1: Analysis of Encrypted Signal Power, Noisy Signal Power and Eavesdropping Signal Power

Time Analysis

Time analysis plays a vital role in evaluating the performance of an algorithm. It is basically time taken by an algorithm to encrypt one state of data. This data could be of any size for example for AES-128 algorithm, the block data size is of 128 bit. Ideally, the Eavesdropping signal must take more time than the encrypted signal for better performance.

Table 3.2: Time analysis of Encryption and Decryption Time of signal and Decryption Time taken by Eavesdropper

SNR (dB)	Encryption Time of signal (ms)	Decryption Time of Signal (ms)	Decryption Time Eavesdropper (ms)
0	0.07	0.06	0.59
5	0.06	0.07	0.61
10	0.07	0.06	0.59
15	0.08	0.08	0.58
20	0.06	0.07	0.59
25	0.07	0.07	0.61
30	0.07	0.07	0.59

The results are shown in figure 3.2 in the implementation, it is seen that the time taken to intercept and decode the signal is much higher than that of the cryptographic algorithms in ideal conditions. The time taken do decrypt the cipher aligns with that of encryption time but the attacker requires approximately six times the time to decrypt the information. In our case, as the algorithm hops at every state array, the information is further safe as if the intruder may be able to decrypt the data, by the time, the algorithm is switched and the data is again secured. The results are summarized in table 3.2.

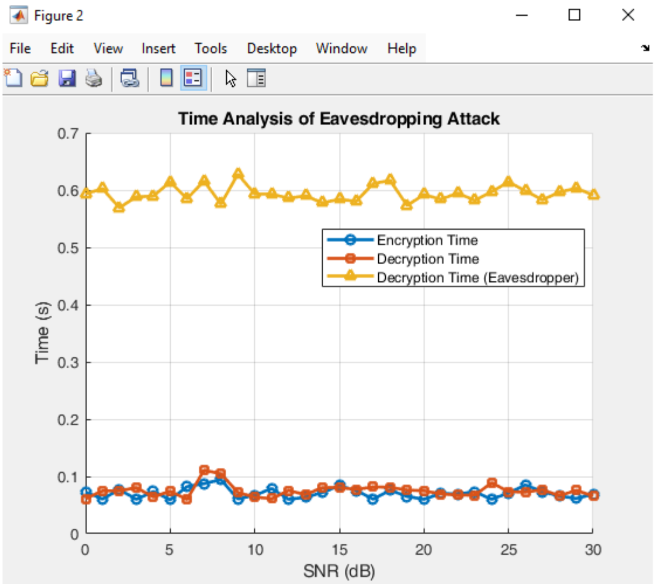

Figure 3.2: Analysis of Encryption Time, Decryption Time and Decryption Time of Eavesdropping signal.

Throughput Analysis (Data Rate)

Throughput is the amount of data successfully delivered over the network within a given time frame, typically measured in bits per second (bps). For eavesdropping countermeasures, it is essential to ensure that any introduced interference or security measure does not significantly degrade the communication speed for legitimate users. This is a ratio of total data transferred to total transmission time taken. Ideally, throughput should be high and must increase with the channel approaching 30dB. The results are summarized in table 3.3.

Table 3.3.

SNR (dB)	Throughput of Signal (bps)	Throughput Eavesdropper (bps)
0	1.00	0.9
5	2.06	1.9
10	3.46	2.9
15	5.03	3.9
20	6.66	4.9
25	8.31	5.9
30	9.97	6.9

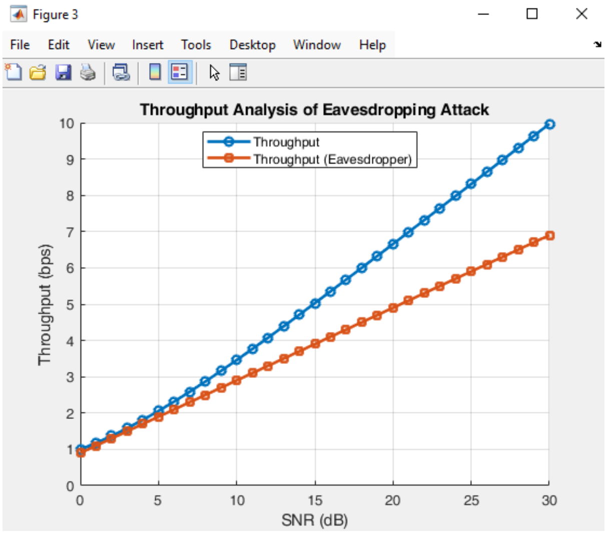

Figure 3.3: Analysis of Throughput of Encrypted signal and Eavesdropping signal.

The results are shown in figure 3.3. The results show that as the SNR of channel increases, the throughput of the encrypted signal is increasing. The throughput of the Eavesdropper is also increasing as the channel becomes lesser noisy. This is due to the fact that the a lesser noisy channel also benefits the Eavesdropping attacks such as Man-in-the Middle attack but the throughput of encrypted signal is much higher than that of eavesdropping signal.

4. Conclusion and Future Scope

The above implementation showed that the use of DEnDecA envelop over the designed AES-64 and AES-32 algorithms preserved the Security of data. This in turn provides a better and light-weighted option for constrained IoT devices and nodes. There is a very small computational addition in the implementation but the downscaling of the derivatives of AES saves much of the computation as compared to the designed encryption envelop. Hence, the DEnDecA algorithm serves as a robust envelop in designing light weight encryption techniques for IoT devices and Systems.

The software design and hardware implementation of various cryptographic techniques offer significant potential for securing IoT communication, various challenges remain to be addressed. The need for security protocols that can adapt to the different resource capabilities of IoT devices and powerful edge or cloud platforms is crucial. The oversight of transport layer security features in IoT implementations poses notable security risks. The complexity introduced by cryptographic solutions in software further complicates matters and can create vulnerabilities. Thus, ongoing research into lightweight cryptographic protocols represents a promising path forward, aiming to provide efficient security solutions that are suitable for resource-constrained IoT devices while ensuring robust protection.

References

1. El-Latif A.A.A., Abd-El-Atty B., Venegas-Andraca S.E., Elwahsh H., Piran M.J., Bashir A.K., Song O.Y. and Mazurczyk W. (2020), “Providing End-to-End Security Using Quantum Walks in IoT Networks”, IEEE Access 8, 92687–92696. https://doi.org/10.1109/ACCESS.2020.2992820
2. NIST Special Publication 800-131A, “Transitions: Recommendations for Transitioning the Use of Cryptographic Algorithms and Key Lengths”, Revision 1, National Institute of Standards and Technologies, USA, (2015). https://doi.org/10.6028/NIST.SP.800-131A
3. Tsai M. M. Y. and Cho H. H., “A High Security Symmetric Key Generation by Using Genetic Algorithm Based on a Novel Similarity”, Mobile Networks and Applications Journal, Springer, (2021). https://doi.org/10.1007/s11036-021-01753-1
4. Yue H. and Zheng X., “Research on Encrypting Accounting Data Using DES Algorithm under the Background of Microprocessor System”, Microprocessors and Microsystems Journal, (2021). https://doi.org/10.1016/j.micpro.2021.104061.
5. J. S. Kalsi and J. Singh Ubhi, "A study of Conventional Protocols applicable to the emerging IoT Systems and Devices," 2019 International Conference on Automation, Computational and Technology Management (ICACTM), London, UK, 2019, pp. 395-399, https://doi.org/10.1109/ICACTM.2019.8776826.
6. Suchdeo M. and Gandhewar N., “Elevating IoT Security: Integrating LSTM with Symmetric Key Protocols in Distributed Environments”, Int J Intell Syst Appl Eng, 12(19) , pp. 252–275, Mar. 2024.
7. S. Kumar and R. Kaur, "Symmetric and Asymmetric Cryptography using RSA Algorithm," International Journal of Science and Research, vol. 12, no. 9, pp. 919–923, Sep. 2023. https://www.doi.org/10.21275/SR23906202504
8. Kaur and G. Singh, "Encryption Algorithms based on Security in IoT (Internet of Things)," 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), Solan, India, 2021, pp. 482-486, https://www.doi.org/10.1109/ISPCC53510.2021.9609495.
9. J. S. Kalsi, J. S. Ubhi, K. S. Raju, "Light-Weighted Dynamic Encryption Decryption Algorithm (DEnDecA) for Internet of Things", International Journal of Sensors, Wireless Communications and Control; Volume 15(1), 2025. https://doi.org/10.2174/0122103279320313240805055515
10. G. Sittampalam and N. Ratnarajah, "Enhanced Symmetric Cryptography for IoT using Novel

Random Secret Key Approach," 2020 2nd International Conference on Advancements in Computing (ICAC), Malabe, Sri Lanka, 2020, pp. 398-403, https://www.doi.org/10.1109/ICAC51239.2020.9357316

11. “Specifications for the ADVANCED ENCRYPTION STANDARD (AES)”, Federal Information Processing Standards Publication 197, (2001).
12. Nechvatal J, Barker E., Bassham L., Burr W., Dworkin M., Foti J., and Roback E., "Report on the Development of the Advanced Encryption Standard (AES)", National Institute of Standards and Technology, US Department of Commerce, 2000. https://doi.org/10.6028/jres.106.023
13. J. S. Kalsi, J. S. Ubhi, K. S. Raju, "Design Advancements in Light-weighted Symmetric Encryption for IoT applications on FPGA: Focusing on AES and DES Derivatives," International Journal of Engineering Trends and Technology, vol. 72(8), pp. 292-311, 2024. https://www.doi.org/10.14445/22315381/IJETT-V72I8P128
14. Federal Information Processing Standards Publication 197, "Announcing the Advanced Encryption Standard (AES)", November 26, 2001
15. Crocetti L., Baldanzi L., Bertolucci M., Sarti L., Carnevale B. and Fanucci L., “A simulated approach to evaluate side-channel attack countermeasures for the Advanced Encryption Standard”, 68 (2019), pp.80-86. https://doi.org/10.1016/j.vlsi.2019.06.005
16. Zhou C., Zhang W., Ding T. and Xiang Z., “Improving the MILP-based security evaluation algorithms against differential cryptanalysis using divide-and-conquer approach”, IACR Crypto-analysis ePrint, 19(2019).
17. Chen L., Wang G. and Zhang G., “Milp-based related key rectangle attack and its application to GIFT, Khudra, MIBS”, The Computer Journal, 62(12), (2019), pp.1805–1821. https://doi.org/10.1093/comjnl/bxz076
18. Cao M. and Zhang W., “Related-key differential crypt analysis of the reduced-round block cipher GIFT”, IEEE Access, 7 (2019), pp.175769–175778. https://doi.org/10.1109/ACCESS.2019.2957581
19. Wang H., Shi Q., Nahiyan A., Forte D. and Tehranipoor M. M., “A Physical Design Flow against Front-side Probing Attacks by Internal Shielding”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 39(10)(2020). https://doi.org/10.1109/TCAD.2019.2952133
20. Wang H., Shi Q., Forte D., Tehranipoor M. M., “Probing Attacks on Integrated Circuits: Challenges and Research Opportunities”, IEEE Design & Test, 34(5)(2017). https://doi.org/10.1109/MDAT.2017.2729398
21. Wang H., Shi Q., Forte D., Tehranipoor M. M., “Probing Assessment Framework and Evaluation of Antiprobing Solutions”, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 27(6)(2019), pp.1239-1252. https://doi.org/10.1109/TVLSI.2019.2901449
22. Weiner M., Wieser W., Lupon E., Sigl G. and Manich S., “A Calibratable Detector for Invasive Attacks”, IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 27(5)(2019), pp. 1067-1079. https://doi.org/10.1109/TVLSI.2019.2892408
23. Wang H., Shi Q., Nahiyan A., Forte D. and Tehranipoor M. M., “A Physical Design Flow against Front-side Probing Attacks by Internal Shielding”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 39(10)(2020). https://doi.org/10.1109/TCAD.2019.2952133.
24. Madhavapandian S. and MaruthuPandi P., “FPGA implementation of highly scalable AES algorithm using modified mix column with gate replacement technique for security application in TCP/IP”, Microprocessors and Microsystems Journal, Elsevier, 73 (2019). https://doi.org/10.1016/j.micpro.2019.102972.
25. Noor S. M. and John E. B., “Resource Shared Galois Field Computation for Energy Efficient AES/CRC in IoT Applications”, IEEE Transactions on Sustainable Computing, 4(4)(2019). https://doi.org/10.1109/TSUSC.2019.2943878.

26. Verbauwhede I., Hoornaert F., Vandewalle J. and Man H. J., “Security and Performance Optimization of a New DES Data Encryption Chip”, IEEE Journal of Solid-State Circuits, 23(3)(1988), pp.647-656. https://doi.org/10.1109/4.302
27. Smld M. E. and Branstad D. K., “The Data Encryption Standard: Past and Future”, Proceedings of the IEEE, 76(5)(1988), pp.550-559. https://doi.org/10.1109/5.4441
28. Coppersmith D., “The Data Encryption Standard (DES) and its strength against attacks”, IBM Journal of Research&Development, 38(1994). https://doi.org/10.1147/rd.383.0243
29. Gong G. and Golomb S. W., “Transform Domain Analysis of DES”, IEEE Transactions on Information Theory, 45(6)(1999). https://doi.org/10.1109/18.782138. Pammu A. A., Weng-Geng H., Lwin N. K. Z., Chong K. S. and Gwee B. H. “A High Throughput and Secure Authentication-Encryption AES-CCM Algorithm on Asynchronous Multicore Processor”, IEEE Transactions on Information Forensics and Security, 14(4)(2019), pp.1023-1036. https://doi.org/10.1109/TIFS.2018.2869344
30. Masoumi M., “A highly efficient and secure hardware implementation of the advanced encryption standard”, Journal of Information Security and Applications, 48(2019). https://doi.org/10.1016/j.jisa.2019.102371
31. Lumbiarres-Lopez R., Lopez-Garcia M., Canto-Navarro E. “Hardware architecture implemented on FPGA for protecting cryptographic keys against side-channel attacks”, IEEE Transactions on Dependable Secure Computing, (2016).
32. Moore J. H. and Simmons G. J., “Cycle Structure of the DES for Keys Having Palindromic (or Antipalindromic) Sequences of Round Keys”, IEEE Transactions On Software Engineering, 13(2)(1987), pp.262-273. https://doi.org/10.1109/TSE.1987.233150