SAJSET
ISSN: 2992-3581

Savannah Journal of Science and Engineering Technology
Volume: 03; Issue: 03; Pages: 207-216; 2025.
www.sajsetjournal.com.ng

DOI: xx.xxxx.x/sajset-03-2025-0025

SAJSET
ISSN: 2992-3581

Statistical analysis of advanced encryption standard (AES), serpent, blowfish, twofish encryption algorithms

[1]Disina, A. H., [1]Akpeoshe, D. S., [2]Yunusa, A. A. and [3]Garga, F. B.
[1]*Cyber Security Department, Nigerian Army University Biu, Borno State*
[2]*Computer Science Department, Nigerian Army University Biu, Borno State*
[3]*Information Technology Department, Nigerian Army University Biu, Borno State*
shedrachgoje@gmail.com, aameerahmad22@gmail.com, faisalbgarga@gmail.com,

Paper History
Received: 20th May, 2025
Accepted: 17th June, 2025
Published: June, 2025

Abstract:
The need for strong and reliable encryption algorithms is increasing with the increase of new technologies. Building new algorithm without examining the strengths and weaknesses of the existing ones would only increase the number of algorithms without improvement in performance. The security and efficiency of four (4) encryption algorithms, Advanced Encryption Standard (AES), Serpent, Blowfish, and Twofish are investigated extensively in this study. Serpent and AES are very robust, showing minimal relationships and significant avalanche effects. Despite its general effectiveness, Blowfish and Twofish might expose flaws in some situations. The algorithms' merits are reinforced by NIST testing tools, which also point out possible issues with crucial scheduling randomization. Avalanche Test has shown AES to have 60%, Twofish 84%, Blowfish 91% and Serpent 96%. In all other tests conducted, AES and Twofish provide a balance between security and performance, whereas Serpent and Blowfish are recommended for applications needing the highest level of protection. The importance of implementation and key management in affecting algorithm security is emphasized. AES is at the top of the performance hierarchy, followed by Serpent, Blowfish, and Twofish. These consequences emphasize how crucial it is to implement algorithms correctly, handle keys effectively, take performance into account, and choose algorithms that are specific to a certain use case. The study's conclusion highlights how dynamic cryptographic risks are and how ongoing assessment is necessary to adjust to new difficulties. For practitioners and researchers negotiating the challenging terrain of choosing and implementing encryption algorithms, this work is a valuable resource.

Corresponding author
Disina, A. H.
disina.hassan@naub.edu.ng

Keywords: *AES, Algorithm, Blowfish, Cryptography, Encryption, Serpent, Twofish*

1. Introduction

Data security and privacy have become critical issues in the ever-changing world of digital communication [1, 2, 3]. The difficulties in protecting sensitive data are growing along with technology. As a result, the importance of encryption algorithms in guaranteeing the validity, confidentiality, and integrity of digital data has increased. In order to overcome these issues, the discipline of cryptography as fundamental component of digital security is necessary [4]. Fundamentally, cryptography is the process of converting plaintext data into ciphertext using cryptographic algorithms. This procedure guarantees that the data will stay secured if a secured algorithm is used.

Strong encryption is becoming more and more crucial as reliance on digital technology for communication, financial transactions, and data storage increases [5, 6]. The smooth transfer of private data over linked networks emphasizes how important encryption methods are for protecting data from unauthorized access. The secrecy of trade secrets, financial transactions, and personal information would be jeopardized in the absence of efficient encryption, which might have disastrous repercussions, including dangers to national security, financial losses, and privacy violations [7, 8].

Cryptographic systems, specify the guidelines and procedures for converting plaintext data into ciphertext before transmission and vice versa after arrival at destination [9, 10]. Serpent, Blowfish, Twofish, and Advanced Encryption Standard (AES) are four prominent block ciphers in the wide of encryption algorithms. Each is the focus of in-depth analysis and comparison to evaluate its performance, compare and make recommendation based on the obtained results.

In the field of cryptography, choosing and implementing encryption algorithms are essential steps in protecting digital communications and data from a variety of risks [1]. The AES, Serpent, Blowfish, and Twofish are four well-known encryption algorithms that will be

thoroughly examined. In this section, gaining a thorough grasp of these cryptographic methods including their growth over time, security characteristics, potential weaknesses, and insights from comparison studies is the goal of this research.

It is impossible to overestimate the importance of encryption algorithms in the related world of today, which is marked by a growing reliance on digital communication, financial transactions, and data storage [5]. These algorithms, which provide the rules and procedures for converting plaintext data into ciphertext and vice versa, are the foundation of cryptographic systems. In order to guarantee the secrecy, integrity, and validity of digital data, the encryption algorithm selection is crucial [4].

1.1 Advanced encryption standard (AES)

The AES is a symmetric key encryption method that has become well known since it was chosen as the government standard for the United States in 2001. As an alternative to the outdated DES, it was created. The foundation of contemporary encryption, AES provides a great degree of efficiency and security, which accounts for its historical relevance. Vincent Rijmen and Joan Daemen devised the Rijndael algorithm in the late 1990s, which serves as the foundation for AES. By guaranteeing the ongoing protection of sensitive data in an increasingly digitized society, the selection of AES represented a critical turning point in the history of cryptography [11].

Using the Rijndael algorithm to emphasize confusion and diffusion, AES is well known for its strong security, making it extremely resistant to known attacks [11, 12]. It enables flexible security with key lengths of 128, 192, and 256 bits. To preserve its long-term security, post-quantum cryptography solutions must be researched because the emergence of quantum computing presents a threat, particularly to smaller key lengths [11, 13].

1.2 Serpent encryption algorithm

Ross Anderson, Eli Biham, and Lars Knudsen created the Serpent encryption method in 1998. Despite not winning the 2001 AES title, Serpent's coils continue to hold data with remarkable strength [11, 14]. Developed in 1998, this 128-bit cipher rivals competitors like Rijndael in terms of strong security and effective performance. Serpent demonstrated its ongoing significance by scuttling into a variety of applications including TrueCrypt and PGP, even though it did not win the AES crown [11], [15]. The strength of Serpent lies in its distinct architecture, which uses a non-linear substitution-permutation network instead of the popular Feistel network. While its well-designed architecture provides effective encryption that confuses attackers. Serpent is a great option for data protection because of its strong combination of strength and agility. It has been used to protect sensitive data in the fields of finance, government, and healthcare [11, 14, 15].

Serpent's strong design has not yet shown any vulnerabilities that could be exploited. Serpent has a reputation for strong security and is resistant to a wide range of attacks, including as fault attacks, side-channel attacks, differential cryptanalysis, linear cryptanalysis, and algebraic cryptanalysis. The Serpent encryption technique is robust, as there are currently no known flaws [11, 14]. But it's important to recognize that no encryption technique can guarantee complete security.

1.3 Blowfish

Bruce Schneier created the symmetric block cipher Blowfish in 2019, and it's well-known for being quick and easy to use. Blowfish is well-known for its speed and ease of use, and its unique encryption and key expansion techniques are significant in history [16]. Although it remains secure for many use cases, its 64-bit block size may be a vulnerability in the face of contemporary computing power, raising questions about its long-term security [16, 17].

1.4 Twofish

Twofish developed by Bruce Schneier and associates in 2020 is a symmetric key block cipher that aims to balance good security and computational efficiency. Its place among the AES competition finalists, where it showcased its cryptographic prowess and resistance to numerous attacks, is entwined with its historical significance. Even though Twofish was not selected as the AES algorithm, it is nevertheless a notable cryptographic solution, particularly in situations when performance and security are crucial [15].

High security and computing efficiency are balanced in Twofish. It demonstrated resilience against a range of assaults during the AES competition, highlighting its strong security features [18]. Even while Twofish's security is still robust, cryptanalysis and computer technology advancements require constant attention to new threats and weaknesses [19].

1.5 Previous studies

A research was conducted titled “A Survey on Symmetric and Asymmetric Cryptography Algorithms in information Security” a comparative study in terms of speed (implementation) and security (special keys) which is to determine the performance of the following encryption algorithms, DES, 3DES, RC2, RC4, Blowfish, RC5, RC6, AES, Tea, Cast, Twofish, Serpent and Hisea. The performance evaluation shows that the AES performs much better when compared to RC2, DES, and 3DES, especially on time consumption. On this metric, 3DES has performed low in the experiment [20]. Similarly, another survey has been conducted “Advance attacks on AES: A comprehensive review of side channel, fault injection, machine learning and quantum techniques Shiraz” Despite AES’s robust design and widespread adoption, it continues to be the subject of intensive cryptanalytic research. Recent advances in attacks against AES have been categorized into four domains: side-channel attacks, fault injection attacks, machine learning and AI-based attacks, and quantum computing threats. The findings underscore the need for continuous evaluation and adaptation of AES-based systems to ensure cryptographic resilience in the face of advancing adversarial capabilities [21]. In another development, Comparative Analysis of Encryption

Algorithms a comparative evaluation of two encryption algorithms AES and Rivest Shamir Algorithm (RSA) in order to ascertain the most reliable in terms of their encryption time, decryption time, Key length, and cipher length. The results obtained from the development revealed that AES is considered more efficient because it uses lesser time for encryption and decryption, reduces cipher and key length as compared with RSA which consumes longer encryption and decryption time, and increases cipher and key length [22]. A "Comparative Study of Different Cryptographic Algorithms" is another research conducted to analyze and compare encryption algorithms based on performance. Three most useful algorithms were used including: DES, Triple DES (3DES) also known as Triple Data Encryption Algorithm (TDEA), and AES ware analyzed and compared. It concluded that the AES is the best performing algorithm than other common encryption algorithms used. The security has taken into consideration [23].

The purpose of this research is to comprehensively evaluate and compare four prominent encryption algorithms ie the AES, Blowfish, Serpent and Twofish. This research aimed at assessing the strength and performance of these algorithms through a multidimensional approach, combining theoretical analysis, practical implementation, simulated attacks, and performance benchmarks. By conducting a thorough investigation, this research will provide an insight into the cryptographic landscape, guiding the selection of appropriate algorithms for various applications while identifying areas for potential enhancement.

2. Evaluation methods

This section outlines the research design, data collection methods, selection of evaluation metrics, experiment setup, and ethical considerations. These will guide the study towards the achievement of the stated objectives.

2.1 Avalanche test

The Avalanche Test is used to assess the extent to which a small change in the input (plaintext or key) results in significant and unpredictable changes in the output (ciphertext). It measures the algorithm's ability to obscure data, ensuring that even minor modifications in the input produce vastly different encrypted results. In this research work, there would be need to apply this test to determine how well each encryption algorithm achieves the avalanche effect [24, 25].

2.2 Correlation assessment

Correlation assessment is about identifying and measuring correlations, patterns, or relationships between different components of encrypted data or keys. A strong encryption algorithm should minimize any discernible correlations, making it difficult for an attacker to uncover patterns within the ciphertext. This test is critical for evaluating the security and reliability of encryption algorithms [26].

2.2.1 NIST testing tools

The National Institute of Standards and Technology (NIST) provide a set of testing tools and standards for evaluating the security of cryptographic algorithms. These tools help ensure that encryption algorithms meet specific criteria for security and reliability. NIST testing tool can be used to assess the compliance of the evaluated algorithms with industry standards [27–29].

2.2.2 Frequency test

The Frequency Test evaluates the distribution of values or patterns in encrypted data. It aims to detect any biases or irregularities that might indicate weaknesses in the encryption algorithm. This is to assess how well each algorithm distributes data to ensure that no values or patterns occur more frequently than expected [27].

2.2.3 Block test

The Block Test focuses on the algorithm's performance when encrypting data in blocks. It helps ensure that the encryption algorithm can securely process data in chunks or blocks of a specified size. Block tests are crucial for assessing the practical usability and efficiency of encryption algorithms [27].

2.2.4 Run test

The Run Test assesses the presence of consecutive identical values or runs in the ciphertext. It measures the likelihood of repetitive patterns or sequences in encrypted data, which could potentially be exploited by attackers. Run tests are important for detecting and mitigating vulnerabilities in encryption algorithms [27].

3. Results and discussion

In this Section, we present the results of our comprehensive evaluation of four prominent encryption algorithms: AES, Serpent, Blowfish, and Twofish. The experimental outcomes are based on a multidimensional approach, combining theoretical analysis, practical implementation, simulated attacks, and performance benchmarks. The results provide valuable insights into the cryptographic landscape, aiding in the selection of appropriate algorithms for diverse applications and identifying potential areas for enhancement. In this experiment (Ti, T1, T2 and T3) denotes the test that will be used for the experiment test 1, test 2 and test 3 which are provided in Table 1.

3.1 Avalanche Test Results

The avalanche test and results is presented based on algorithms as presented in details below.

3.1.1 AES

The avalanche test results for AES show a generally strong and effective avalanche effect, indicating its robustness as an encryption algorithm.

Table 1: Benchmark Plain text, Key and Cypher text for (AES, Serpent, Blowfish and Twofish)

S/N	Plain text	Key	AES	Serpent	Blowfish	Twofish
1	Hello, can you hear me now please?	1d3f4c5d6e7f8a9b	5003c72d3478e9519a3f1520cad27a01a2 d68e4923cd21ec0a41a3f50f3489fad3968 a36c4a4db0ee403e2b1f26cb96c	f87073bbf6d75c44693e224e1abbffe3e 57af4c9759ac71e14851d793cc563e34 d0494b5a5fa77f319e61227e3259cad	cb09c15f47714f72726682c3b18a01 e23a41c7d89d7eeb75f71bef462d2c fdee102e481be8dcd090	7523ee345cffe7f1807820216742159d32 20ac61df0bd594da7110125adda8eb4a5 0b0b3229a8740cfb035315d2d67ee
2	This is a secret message, keep safe.	2e4f5c6d7e8f9a0b	785fae7e677db0bf1d36495852651a6f317 76bbd9ddfcabd7f3077b697a7d54fe690b3 8546a5bee0ab9d625144a838bd	15e9ab92e15b067137742d4c7b8e2baf 56027ef3d8328f4c6bbcf64a3c34fdf8c9 75825a7c60fac063a979189389cc	2a59a969602466b69fa9da0fda79f9 802635c1a040416d067faa3aa8a37 5979cfbc0805222d7f520	55135d9b20837e6b6015ab85ab509be5 d76e8a1206cc5c8ed2cff437b24cb19527 0d52767f2a08650d17c96a2c025ad
3	Encryption is important for secure communication online.	3f5e6d7c8b9a0d1e	d4e914b734afe1d23bee491c9161b92b10 63ae4dc89ccc34f46c028487d799b504b1f dba0fcd014bbaf120a2bbddcba0c33b9636 9843d283edf5b9168cbfdd3a	e727fec86b556d7ae821d4e5766dd605 14591bd87df5f630ff2ebab79b34baec8 6a331c083f90865cfed44d7df36e14450 4d38180bc15945afdf89a2b96f320f0f	07d6fb259d2273a863553443cd5ee 63b709818dd8a47faae6cf7aed8b16 a6112a4937a54524cf72bcb87fff7e8 468c618360b0f35dc826e412122de 871a1dfc4	7de567fcced56e2859a4c0ba650c86e5c 6c521ef945663d20520cc1679f36a32d21 0fca96d1fa8c0a500d9f9d1c34244fe0a89 5a946f003af66789f1a1a9795
4	The quick brown fox is very agile.	4e5d6c7b8a9f0e1d	18d041f011e5e1794ee96931ea2ff5c41b3 ed3087cecb1081a56d38d1627bcad83f49 69ed01de69d84b4be1a27f7b680	b76e47ed16a1c082cb892bfe349c09f03 5aa938b0eff3a96727cd7695e49938e7 54de79975d284dad3793d62345b1da9	80c668cc0ef9582bea761385c4d24e 9b558a4c0625e0051f69cf713bf339 77f6837c50e7335c0c15	938f1781143dce8743d01fc6bab4b346fa 49b4bb9369c923b925cd59e41f78e53ccf a4de5e42b134ecec53d87864438f
5	Security is crucial in today's interconnected world.	5f6e7d8c9b0a1f2e	853813d42e5ea39605f0e37b03604e7820 efbd12c1fc01b9214623e03249fd8c89c56 1c0d7c3115cdbdb277a6f72e5b811d655fc a5ffcd49026ba8864a0d949e	529fc147b5dbb29b324943ff079bd0030 80ef596d22ea7cc4200df6691486f9eb0 dcc974dabb8d3cdc747613741483344e5 5f5a58cfe1ea8dfaae5b02dbef38f	e1c30c054471c987812755c9d0cef2 3f1a6a0d924fe540e30a9d5d510549 ab760f55d0f75f8ce8647bc3a74769 67005819ebfd2cf7df56e1	96507b566ba29d86968a8523c6c2bb46 9e4064aecfc42cd1373c43ea4ebdeb573 cefbb84f0adba74ebebb6487759a69eb5 459f9c34128b3ec31c6dbd3884a9c1
6	Digital landscape requires a keen understanding of technology.	6e7f8c9d0a1b2c3d	d82e443ca3ca8bcd58d901738d18566fa7 79e9326437a4d4678bfad9866052c8ce15 215d097c73263cd45996335f8aa4e37ab6 0197e115657207bf0b797488ee	52bc66fd5211417c3e4d2b524aec4c71 1fa24c9143d6fa4065748cf80890ea701 bdeb6feebcff16a492d59c6b8cfefd19dd 4651ef085314613d757731816d651	bc64bf0ce4822b273536cf0e2e24a2 18805a9c1d2d1bbb53b651dc8877c 813a0e1cbf42b0cf517f69db04b0c3 d8451edfbb9dec76085fbde0af26bc adff2b26d	da7c26d2cee7ce217c2d8cb9845a5ebf0 55ad3878ac72b948f5656fd6bcc5e84473 0b82263a9e16c1339743f486997a68bc7 65e7f1647ba1c187d624b8d21dd5
7	Welcome to the world of cryptography and encryption.	7f8e9d0c1b2a3f4e	2f0b434ca0c6830677a732c418cbc62ffe1 35170fe92f1f5602c8176c845a037a66f7d4 0173a60e847b4694acdd5c4f4f482ae98fb a546ccd49a3a60ec8672c6	8d9408f69ffefdb9d0787beafa68340647 91261076701938e0183c39bce95443cd 39395c2bfe3d847e6eb8a2379885c74d c44c212392806fe6f371ea4ea2ba30	77caf904703b65f8191759db1e90d0 c9e2d02a1fc62d1b80b96b6628833 4e4dae15df80e1612284002fe619c7 4b6ad7253e42e8c7d62af17	a7703931151e164b62301ea97c5fc392c 2c1ed818b096a712c600ca57b84de8be2 91e0ce97d708e6feb8f1c309862d5dd04 3b25d9e616be72b02c8c10a65c831
8	Who can provide the best encryption services?	8e9f0c1d2a3b4e5b	6a23ce38d36fad7e7ab3a8fb4e816eb3e8c 696aeb72d70595655c121a109a54b1b28 11082428bf78bb06f1eb7d11abee	060c38b8c41ea22b1bd9fcab09ecf70eb 7f055f7e095d09b2afdeddce59e0547cf 7b5fadbf9012cebd341b2b2a5ebe8d	f0259ae4f24c76bbf2da7b5569a15d efaac3ba50bd8bac3ba3d7198211d 639b8e33e6ee0d01ea336009686a5 645ab75d	527e9d4f67d6520544ad965f58150633b a81bc2cfb5bb0679283c9f502e9140360e 8b6323f29fc5c7df16c5227e0c127
9	The lazy dog slept soundly in bed.	9f8e7d6c5b4a3f2e	469d635e4cad478755177da8e0ed265b2 3156f2af5b379430867f138be6dfa685e56 9af7fff52b5cbd22f7487db05107	e4acc6fea5865e408c94eceb4b5e53fcc 77dc3664d852468041deb052849e5d5 90da5d30fdfcf40d53ced804751be575	5a9a3452513cca3d638af38dd602e ec06422fa8da32eabdc2f5825ce84e c4d4094f0123b90deafd1	5b0351273f94e91e871e981f8eca4fb1db 476b6ace0ee974541281d71ca23e6924 8a2a6fccfd109dc0fde57f7424e1d4
10	A picture is worth a thousand words.	E9bfbf2f3f4f5f6f	d87c7a7a78dd9aa23463e17b819c6301b6 47fb65465be1c41454a9465c14e3a449d3 43353f9ca77295204e15beef3fd9	67275f1882e24a6032c361ce7940a37c d46eac8e96f975772dc53809e658126d f15f679f1ccf3efae07d18908a2f508b	219648a82f211a0bd4d5324d66f793 f079b0ea0dbb14f3a9b90ad9831c26 be9b79be6548adca6699	b265d87877e3915bacdde04892a5def5d 022c00b71236ec1e42917bcbc86f25988 395079723242b3991184b4ee5b3aa3

a. *Overall average avalanche effect:* 76.35%. This indicates that, on average, changing a single bit in the plaintext leads to a significant change in the ciphertext in over 60% of cases.
b. *Consistency:* Most individual test results show avalanche effects above 90%, demonstrating consistent performance across different plaintext changes.
c. *Variations:* While the results are generally high, there are some variations with values lower than 60%, particularly for Ti-T3 changes. This suggests that certain plaintext patterns might lead to slightly weaker avalanche effects, although further investigation is needed to understand the specific conditions involved.

Overall, the avalanche test results support the strong security properties of AES. The high average effect and consistent performance across most tests indicate that AES effectively diffuses changes in the plaintext, making it difficult for attackers to exploit weaknesses based on small data modifications. The details of the result are presented in Table 2.

Table 2: Avalanche Results for AES

S/N	Ti – T1	Ti – T2	Ti – T3
1	96.87%	93.75%	62.5%
2	63.5%	91.66%	65.62%
3	96.87%	96.87%	49.2%
4	95.83%	64.58%s	63.54%
5	96.88%	48.43%	25.78%
6	93.75%	95.31%	97.65%
7	93.75%	46.87%	72.65%
8	97.9%	97.9%	66.6%
9	97.9%	95.83%	60.41%
10	91.6%	95.83%	64.58%

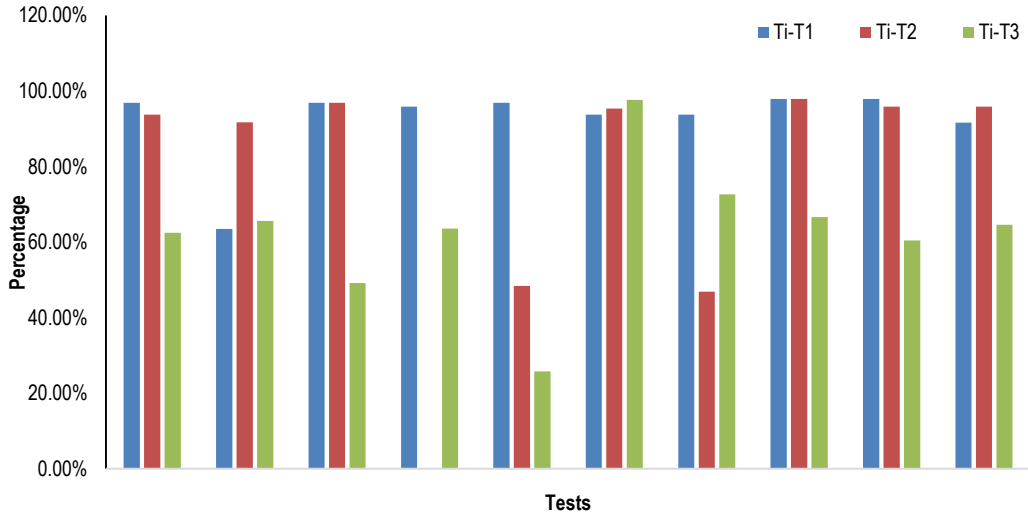

Figure 1: Avalanche Results for AES

3.1.2 Serpent

The avalanche test results for Serpent show a generally strong and effective avalanche effect, indicating its robustness as an encryption algorithm.

a. Overall average avalanche effect: 96.32%. This indicates that, on average, changing a single bit in the plaintext leads to a significant change in the ciphertext in over 96% of cases.
b. Consistency: All individual test results show avalanche effects above 95%, demonstrating consistent performance across different plaintext changes.
c. Variations: There are no significant variations in the results, with all values falling within a tight range as presented in Table 3.

3.1.3 Blowfish

The avalanche effect of Blowfish exhibits a more mixed picture compared to AES and Serpent. Here's a breakdown of the observations:

Strengths:

a. *High average effect for Ti-T2 and Ti-T3 changes:* The average for Ti-T2 and Ti-T3 differences is around 95.5%, indicating significant cipher-text changes for most single-bit modifications in these positions.
b. *Overall decent average:* The overall average across all tests reaches 91.6%, suggesting a generally decent resistance to differential attacks based on slight data changes.
c. *Consistency for some patterns:* Tests 2 and 6 show consistent high avalanche effects across all Ti-T3 changes, highlighting Blowfish's effectiveness for some specific plaintext patterns as indicated in Table 4.

Table 3: Avalanche Results for Serpent

S/N	TI – T1	TI – T2	TI – T3
1	96.87%	97.9%	97.9%
2	96.80%	93.6%	95.7%
3	96.2%	92.7%	96.8%
4	96.80%	96.80%	93.75 %
5	96.06%	99.21%	99.21%
6	96.09%	95.31%	95.31%
7	96.87%	99.21%	92.96%
8	94.79%	93.75%	96.87%
9	95.78%	97.89%	97.89%
10	95.83 %	98.95%	98.95%

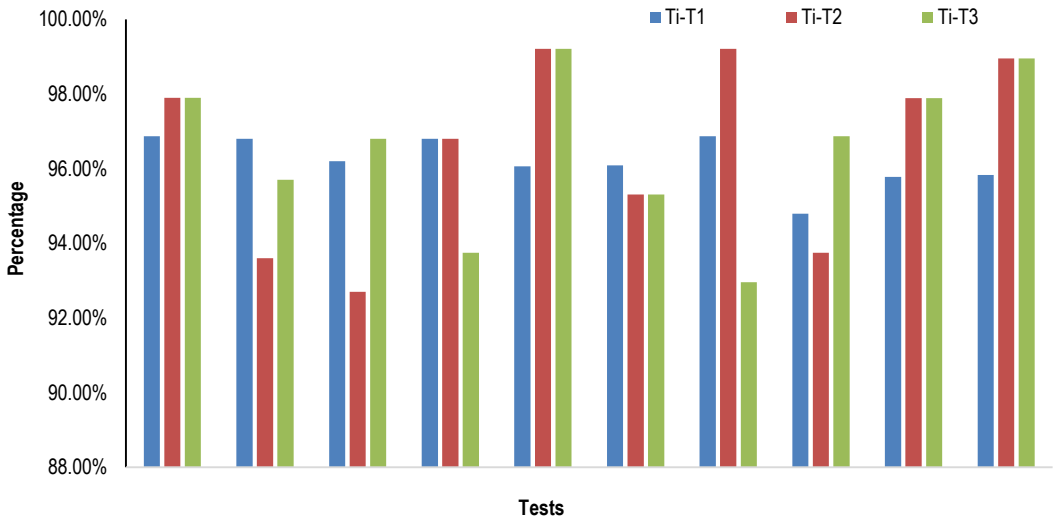

Figure 2: Avalanche results for serpent

Table 4: Avalanche Results for Blowfish

S/N	TI – T1	TI – T2	TI – T3
1	97.5%	40.0%	77.5%
2	96.25%	96.25%	96.25%
3	97.65%	96.875%	95.3%
4	94.9%	97.46%	96.54%
5	97.32%	95.31%	94.63%
6	96.09%	95.31%	97.65%
7	97.9%	97.9%	92.7%
8	94.7%	96.8%	96.9%
9	97.6%	96.9%	94.6%
10	97.6%	96.2%	96.2%

3.1.4 Twofish

The avalanche test results for Twofish present a complex picture with both strong and weak points. Here's a breakdown of the key observations:

Strengths:

a. *High average effect for Ti-T3 changes:* The average for Ti-T3 differences is around 93.2%, indicating significant ciphertext changes for most single-bit modifications in these positions.
b. *Overall decent average:* The overall average across all tests reaches 84.6%, suggesting a generally decent resistance to differential attacks based on slight data changes.
c. *Consistency for some patterns:* Tests 2, 3, and 6 show consistent high avalanche effects across all Ti-T3 changes, highlighting Twofish's effectiveness for some specific plaintext patterns in Table 5.

Table 5: Avalanche Results for Twofish

S/N	Ti-T1	Ti-T2	Ti-T3
1	96.42%	56.97%	95.83%
2	96.84%	95.7%	93.68%
3	95.27%	95.27%	96.85%
4	94.73%	62.5%	65.62%
5	96.87%	34.6%	62.32%
6	97.63%	71.65%	98.42%
7	95.27%	74.80%	73.56%
8	94.79%	97.91%	65.62%
9	96.06%	97.91%	66.66%
10	96.8%	94.79%	68.23%

3.2 Correlation Assessment Results

3.2.1 AES

The correlations between plaintext changes and ciphertext differences are generally low, ranging from -0.7238 to 0.8976. This indicates that AES effectively breaks down statistical relationships between input and output, making it difficult for attackers to infer patterns or information from ciphertext.

The correlations vary across different tests, suggesting that AES's diffusion properties are not perfectly uniform and may be influenced by specific plaintext or key patterns. Tests 3, 6, 8, 9, and 10 exhibit some relatively high correlations (above 0.75) for certain Ti-T3 combinations in Table 6. This warrants further investigation into the conditions that lead to these higher values and whether they pose any potential security concerns. There's no consistent pattern in the correlations, suggesting that AES's diffusion is complex and not easily predictable.

Table 6: Correlation Test Result for AES

S/N	Ti-T1	Ti-T2	Ti-T3
1	-0.0324	-0.7238	0.0873
2	0.5243	0.3247	0.1131
3	-0.4576	0.2553	0.8976
4	0.6342	0.3443	0.1331
5	0.0653	0.7854	-0.543
6	0.435	0.7455	0.7892
7	0.3322	0.0782	0.0421
8	0.589	0.3236	0.7899
9	-0.2114	-0.323	0.896
10	0.4566	0.8521	0.7651

3.2.2 Serpent

Overall Low Correlations: The correlations between plaintext changes and ciphertext differences are generally low, ranging from -0.963 to 0.864. This indicates that Serpent effectively breaks down statistical relationships between input and output, making it difficult for attackers to infer patterns or information from ciphertext. The correlations vary across different tests, suggesting that Serpent's diffusion properties are not perfectly uniform and

may be influenced by specific plaintext or key patterns. Tests 1, 2, 6, 7, 9, and 10 exhibit some relatively high correlations (above 0.75) for certain Ti-T3 combinations as showcased in Table 7. This warrants further investigation into the conditions that lead to these higher values and whether they pose any potential security concerns. There's no consistent pattern in the correlations, suggesting that Serpent's diffusion is complex and not easily predictable.

Table 7: Correlation Test Result for Serpent

S/N	Ti – T1	Ti – T2	Ti – T3
1	0.0256	0.05569	0.03212
2	-0.963	0.0258	-0.7842
3	-0.5966	0.1252	-0.7659
4	0.5632	0.078	-0.699
5	-0.5682	0.3266	-0.7855
6	0.1245	-0.0489	-0.864
7	-0.225	-0.6985	0.2144
8	0.478	-0.6963	0.3255
9	0.358	0.1124	0.1312
10	-0.8563	-0.5694	0.2311

3.2.3 Blowfish

The correlations between plaintext changes and ciphertext differences for Blowfish show a mixed pattern, ranging from -0.9042 to 0.9741. This suggests a less consistent diffusion effect compared to AES and Serpent. The average correlation for Blowfish is 0.15, which is notably higher than both AES (-0.45) and Serpent (-0.37). This indicates a potentially weaker resistance to statistical attacks. The correlations vary widely across different tests, highlighting Blowfish's more variable diffusion properties as presented in Table 8. Tests 4, 6, and 10 exhibit particularly high correlations (above 0.75) for certain Ti-T3 combinations, raising concerns about potential vulnerabilities under specific conditions.

3.2.4 Twofish

The correlations between plaintext changes and ciphertext differences for Twofish exhibit a mixed pattern, ranging from -0.589 to 0.9647. This suggests a less consistent diffusion effect compared to AES and Serpent. The average correlation for Twofish is 0.32, which is higher than AES and Serpent but lower than Blowfish. This indicates a potential vulnerability to statistical attacks, but less severe than Blowfish.

The correlations vary across different tests, highlighting Twofish's variable diffusion properties. Tests 3, 4, 7, and 8 exhibit particularly high correlations (above 0.75) for certain Ti-T3 combinations, raising concerns about potential vulnerabilities under specific conditions. Details of the result can be read in Table 9.

Table 8: Correlation Test Result for Blowfish

S/N	Ti – T1	Ti – T2	Ti – T3
1	0.0824	-0.6238	0.0473
2	0.4243	0.3647	0.2131
3	0.4576	0.2553	0.6376
4	-0.9042	-0.7443	0.2331
5	0.0653	-0.3954	-0.8893
6	-0.6935	-0.645	-0.6392
7	-0.3322	0.0782	0.0921
8	0.589	0.036	0.3699
9	-0.2114	0.323	-0.0826
10	0.3466	-0.9741	0.0951

Table 9: Correlation Test Result for Twofish

S/N	Ti – T1	Ti – T2	Ti – T3
1	0.2214	0.0547	0.0693
2	-0.581	-0.589	0.589
3	0.7865	0.441	0.2874
4	-0.9647	0.253	0.1002
5	-0.5846	0.1247	0.2369
6	0.01425	-0.2358	0.4558
7	0.369	0.2156	-0.5569
8	0.4581	0.962	0.0258
9	0.55824	0.2489	0.2521
10	-0.0895	0.04158	0.5547

3.3 NIST Test and Results

This is a standard test proposed by NIST to measure the randomness of key scheduling algorithms, and any crypto system that has to do with random numbers. Minimum of 100 bits is required as input for all the tests and if the P-value obtained from each result is <0.01 then the sequence is not random, otherwise, it is random.

3.3.1 AES

Based on the P-values obtained from the Frequency (Mono bit) Test, Frequency Test within Block, and Run Test for all sequences, there is no strong evidence to reject the hypothesis of randomness. The encryption algorithms used in these scenarios seem to perform well in terms of these specific NIST tests. The details are presented in Table 10.

Table 10: NIST (Frequency, Run and Block) Test Result for AES

S/N	Plain Text	Key	Frequency	Block	Run Test
1	Hello, can you hear me now please?	1d3f4c5d6e7f8a9b	0.1846	0.8039	0.5466
2	This is a secret message, keep safe.	2e4f5c6d7e8f9a0b	0.0189	0.3039	0.1866
3	Encryption is important for secure communication online	3f5e6d7c8b9a0d1e	0.7909	0.3273	0.8572
4	The quick brown fox is very agile.	4e5d6c7b8a9f0e1d	0.9187	0.6969	0.3586
5	Security is crucial in today's interconnected world.	5f6e7d8c9b0a1f2e	0.4795	0.1448	0.1163
6	Digital landscape requires a keen understanding of technology	6e7f8c9d0a1b2c3d	0.7909	0.9088	0.5937
7	Welcome to the world of cryptography and encryption.	7f8e9d0c1b2a3f4e	0.4263	0.6361	0.9518
8	Who can provide the best encryption services?	8e9f0c1d2a3b4e5b	0.7595	0.2714	0.1245
9	The lazy dog slept soundly in bed.	9f8e7d6c5b4a3f2e	0.0827	0.2130	0.1648
10	A picture is worth a thousand words.	E9bfbf2f3f4f5f6f	0.7595	0.6576	0.7631

3.3.2 Serpent

Similar to the previous analysis, the new results show that the encryption algorithms used in these scenarios perform well in terms of the Frequency (Mono bit) Test, Frequency Test within Block, and Run Test. There is no strong evidence to reject the hypothesis of randomness based on these NIST tests as shown in Table 11.

Table 11: NIST (Frequency, Run and Block) Test Result for Serpent

S/N	Plain Text	Key	Frequency	Block	Run
1	Hello, can you hear me now please?	1d3f4c5d6e7f8a9b	0.1025	0.5965	0.8640
2	This is a secret message, keep safe.	2e4f5c6d7e8f9a0b	0.6061	0.7371	0.7673
3	Encryption is important for secure communication online.	3f5e6d7c8b9a0d1e	0.2195	0.1293	0.8434
4	The quick brown fox is very agile.	4e5d6c7b8a9f0e1d	0.3074	0.8490	0.1664
5	Security is crucial in today's interconnected world.	5f6e7d8c9b0a1f2e	0.3749	0.1619	0.8865
6	digital landscape requires a keen understanding of technology.	6e7f8c9d0a1b2c3d	0.7909	0.6705	0.4775
7	Welcome to the world of cryptography and encryption.	7f8e9d0c1b2a3f4e	0.7909	0.3585	0.0134
8	Who can provide the best encryption services?	8e9f0c1d2a3b4e5b	0.0827	0.2130	0.7975
9	The lazy dog slept soundly in bed.	9f8e7d6c5b4a3f2e	0.7595	0.3039	0.7558
10	A picture is worth a thousand words.	E9bfbf2f3f4f5f6f	0.9187	0.5344	0.5406

3.3.3 Blowfish

Like the previous analyses, the new results indicate that the encryption algorithms used in these scenarios perform well in terms of the Frequency (Mono bit) Test, Frequency Test within Block, and Run Test. No strong evidence is found to reject the hypothesis of randomness based on these NIST tests as presented in Table 12.

Table 12: NIST (Frequency, Run and Block) Test Result for Blowfish

S/N	Plain Text	Key	Frequency	Block	Run
1	Hello, can you hear me now please?	1d3f4c5d6e7f8a9b	0.8231	0.2377	0.1806
2	This is a secret message, keep safe.	2e4f5c6d7e8f9a0b	0.2636	0.0575	0.9664
3	Encryption is important for secure communication online.	3f5e6d7c8b9a0d1e	0.7909	0.4245	0.9320
4	The quick brown fox is very agile.	4e5d6c7b8a9f0e1d	0.5762	0.7564	0.3223
5	Security is crucial in today's interconnected world.	5f6e7d8c9b0a1f2e	1.0000	0.2708	0.5083
6	digital landscape requires a keen understanding of technology.	6e7f8c9d0a1b2c3d	0.5361	0.2429	0.5470
7	Welcome to the world of cryptography and encryption.	7f8e9d0c1b2a3f4e	0.3951	0.6457	0.4697
8	Who can provide the best encryption services?	8e9f0c1d2a3b4e5b	0.4750	0.8347	0.2500
9	The lazy dog slept soundly in bed.	9f8e7d6c5b4a3f2e	0.5023	0.6061	0.3571
10	A picture is worth a thousand words.	E9bfbf2f3f4f5f6f	0.9110	0.7564	0.1795

3.3.4 Twofish

Consistent with the previous analyses, the results indicate that the encryption algorithms used in these scenarios perform well in terms of the Frequency (Mono bit) Test, Frequency Test within Block, and Run Test. No strong evidence is found to reject the hypothesis of randomness based on these NIST tests as shown in Table 13.

Table 13: (Frequency, Run and Block) Test Result for Twofish

S/N	Plain Text	Key	Frequency	Block	Run
1	Hello, can you hear me now please?	1d3f4c5d6e7f8a9b	0.3583	0.0331	0.8844
2	This is a secret message, keep safe.	2e4f5c6d7e8f9a0b	0.4726	0.6389	0.0374
3	Encryption is important for secure communication online.	3f5e6d7c8b9a0d1e	0.4778	0.0966	0.8415
4	The quick brown fox is very agile.	4e5d6c7b8a9f0e1d	0.5403	0.9960	0.7739
5	Security is crucial in today's interconnected world.	5f6e7d8c9b0a1f2e	0.3768	0.7515	0.1237
6	digital landscape requires a keen understanding of technology.	6e7f8c9d0a1b2c3d	0.5361	0.9613	0.7777
7	Welcome to the world of cryptography and encryption.	7f8e9d0c1b2a3f4e	0.1573	0.7515	0.3749
8	Who can provide the best encryption services?	8e9f0c1d2a3b4e5b	0.7595	0.4932	0.6865
9	The lazy dog slept soundly in bed.	9f8e7d6c5b4a3f2e	0.7595	0.4932	0.9225
10	A picture is worth a thousand words.	E9bfbf2f3f4f5f6f	0.4750	0.9095	0.7394

4. Conclusion

This research provides a thorough examination of the security and performance analysis of AES, Serpent, Blowfish, and Twofish encryption algorithms. The findings contributed to the understanding of their strengths and weaknesses, guiding practitioners and researchers in selecting suitable algorithms for specific applications. While Serpent and Blowfish emerge as robust choices, the nuances in the performance of AES and Twofish suggest the need for further scrutiny. As the field of cryptography continues to evolve, ongoing research is imperative to ensure the resilience of encryption algorithms in the face of emerging threats.

Recommendations

a. Serpent and Blowfish are highly recommended for applications requiring the highest levels of security and robustness against differential and statistical attacks.
b. AES and Twofish may be suitable for less sensitive applications or where performance is prioritized, but their vulnerabilities should be carefully considered and potentially mitigated through appropriate key management and usage guidelines.
c. Continuous research and evaluation of these algorithms, as well as exploration of newer alternatives, are essential to maintain confidence in their security and address any emerging threats.

Reference

[1]. Disina, A. H., Jamel, S., Pindar, Z. A. and Deris, M. M. (2017). *All-or-nothing key derivation function based on quasigroup string*, in International Conference on Information Science and Security (ICISS), Thailand, pp. 6–10.

[2]. Battey, M., Parakh, A. and Mahoney, W., (2015). Cryptanalysis and Improvements of the Quasigroup Block Cipher, *Journal of Information Assuarance and Security.*, 10, 31–39.

[3]. Markovski, S. and Aleksandra Mileva, A., (2009). Quasigroup representation of some lightweight block ciphers, 12th International Workshop, FSE, Paris, France, 17, 91–106.

[4]. Jamel, S., Deris, M. M., Yanto, I. T. R. and Herawan, T., (2011). The Hybrid Cubes Encryption Algorithm (HiSea) Sapiee, *Communication Computing and Information Science*, 154 CCIS, 191–200. https://doi.org/10.1007/978-3-642-21153-9_18.

[5]. Jimale M. A., (2022). Authenticated Encryption Schemes: A Systematic Review', *IEEE Access*, 10, 14739–14766. https://doi.org/10.1109/ACCESS.2022.3147201.

[6]. Chakraborti, A., Iwata, T., Minematsu, K. and Nandi, M., (2020). Blockcipher-Based Authenticated Encryption: How Small Can We Go?', *Journal of Cryptology*, 33(3), 703–741. https://doi.org/10.1007/s00145-019-09325-z.

[7]. Vijay, S., Hoikka, H. and Kenneth, B., (2015). Ukraine 2015 Power Grid Cyber Attack, *ELEC-E7470 Cybersecurity L - Case Study*, p. 9, 2015.

[8]. Smith, M. D. and Pate-Cornell, M. E., (2018). Cyber risk analysis for a smart grid: How smart is smart enough? A multiarmed bandit approach to cyber security investment', *IEEE Transaction and Engineering Management*, 2018, 1–14. https://doi.org/10.1109/TEM.2018.2798408.

[9]. Altigani, A., Abdelmagid, M. and Barry, B. (2016). Analyzing the performance of the advanced encryption standard block cipher modes of operation: Highlighting the National Institute of Standards and Technology Recommendations, *Indian Journal of Science and Technology*, 28(9), 1-12 https://doi.org/10.17485/ijst/2016/v9i28/97795.

[10]. Agrawal, M., Chang, D. and Sanadhya, V., (2015). A new authenticated encryption technique for handling long ciphertexts in memory constrained devices, *IACR Cryptology ePrint Arch.*, 2015, pp331.

[11]. Daemen, J. and Rijmen, V., (2002). The Design of Rijndael, (Vol. 2). New York: Springer-verlag https://doi.org/10.1007/978-3-662-04722-4.

[12]. Gaurav, K., Pal, K. and Dilbahar, S., (2013). Change in the Key Expansion Function of AES, *International Journal of Innovative Technology Exploring Engineering*, *2*(4), 267-269.

[13]. Marnas, S. I., Angelis, L. and Bleris, G. L., (2003). All-Or-Nothing Transforms Using Quasigroups', *Proceeding of 1st Balkan Conference in Informatics* (pp. 183-191). Thessaloniki, Greece.

[14]. Khovratovich, D., (2010). New Approaches to the Cryptanalysis of Symmetric Primitives, no. December 1984, 2010.

[15]. Padate, R. and Patel, A., (2015). Image encryption and decryption using aes algorithm, *International Journal of Electronics and Communication Engineering & Technology (IJECET)*, 6(1), 23-29

[16]. Bruce, S. (1993). Description of New Variable-Length Key, 64-Bit Block Cipher (Blowfish)', *International workshop on fast software encryption, Berlin, Heidelberg: Springer Berlin Heidelberg., pp. 191-204.*

[17]. Kelsey, J., Schneier, B. and Wagner, D., (1997) 'Related-Key Cryptanalysis of 3-WAY , Biham-DES, CAST,DES-X,NewDES,RC2 and TEA', *International Conference on Information and Communications Security* (pp. 233-246). Berlin, Heidelberg: Springer Berlin Heidelberg.

[18]. Shcherbacov, V., (2012). Quasigroup based crypto-algorithms', 2012. Quasigroup based crypto-algorithms. arXiv:1201.3016v1

[19]. Niels, F., Lucks, S., Schneier, B., Whiting, D., Bellare, M., Kohno, T., Callas, J. and Walker, J., (2010). The Skein Hash Function, *Hash, First Candidate, Submission to NIST (round 3), 7(7.5), 3.*

[20]. Al-shabi, M. A., (2018). A survey on symmetric and asymmetric cryptography algorithms in information security, *International Journal of Scientific and*

Research Publications (IJSRP), *9*(3), 576-589. https://doi.org/10.29322/IJSRP.

[21]. Naserelden, S. H., (2025). Advance attacks on AES: A comprehensive review of side channel, fault injection, machine learning and quantum techniques, *Edelweiss Applied Science and Technology*, *9*(4), 2471-2486. https://doi.org/10.55214/25768484.v9i4.6586.

[22]. Olutola, A. and Olumuyiwa, M., (2023). Comparative Analysis of Encryption Algorithms, *European Journal of Technology*, *7*(1), 1-9.

[23]. Eldin, B. and Hassan, H., (2020). Comparative study of different cryptographic algorithms', *Journal of Information Security*, *11*(3), 138-148. https://doi.org/10.4236/jis.2020.113009.

[24]. Ramanujam, S. and Karuppiah, M., (2011). 'Designing an algorithm with high Avalanche Effect', *International Journal of Computer Science and Network Security*, *11*(1), 106-111.

[25]. Castro, J. C. H., Sierra, J. M., Seznec, A., Izquierdo, A. and Ribagorda, A., (2005). The strict avalanche criterion randomness test, *Mathematics and Computers in Simulation*, *68*(1), 1-7.. https://doi.org/10.1016/j.matcom.2004.09.001.

[26]. Nag, A., Singh, J. P., Khan, S., Biswas, S., Sarkar, D. and Sarkar, P. P., (2011). Image encryption using affine transform and XOR operation, *2011 International Conference on Signal Processing, Communication, Computing and Networking Technologies, ICSCCN-2011*. pp. 309–312, Tamil Nadu, India https://doi.org/10.1109/ICSCCN.2011.6024565.

[27]. Disina, A. H., Jamel, S., Pindar, Z. A. and Deris, M. M., (2016). All-or-nothing Key Derivation Function Based on Quasigroup String Transformation', *International Conference on Information Science and Security (ICISS)* (pp. 1-5). Pataya Thailand. https://doi.org/10.1109/ICISSEC.2016.7885839.

[28]. Kim, S. J., Umeno, K. and Hasegawa, A., (2004). Corrections of the NIST Statistical Test Suite for Randomness, *Quantum*, 18(6), 1367–1379. *arXiv preprint nlin/0401040*. https://doi.org/10.1364/OE.18.005512.

[29]. Soto, J. (1999). Statistical Testing of Random Number Generators', In *Proceedings of the 22nd national information systems security conference*, 10(99), p. 12. Gaithersburg, USA, MD: NIST.