Volume: 16 Issue: 04 Pages: 6473-6486 (2025) ISSN: 0975-0290

Performance and Security Analysis of Symmetric Data Encryption Algorithms: AES, 3DES and Blowfish

Bello Alhaji Buhari*
Department of Computer Science, Usmanu Danfodiyo University, Sokoto – Nigeria
Email: buhari.bello@udusok.edu.ng
Hajara Abdulkadir
Department of Computer Science, Federal College of Education Gidan Madi, Sokoto – Nigeria
Email: hajaraabdulkadir@fcegm.edu.ng
Sadiq Aliyu Ahmad
Department of Cybersecurity, Federal University Dutse, Jigawa – Nigeria
Email: sadiqaliyu@fud.edu.ng
Rilwanu Sulaiman, Maniru Malami Umar, Sirajo Shehu, Abba Almu, Abdullahi M. Abubakar
Department of Computer Science, Usmanu Danfodiyo University, Sokoto – Nigeria
Email: : rilwanusulaiman66@gmail.com, manirutambuwal@gmail.com, siiraa2001@gmail.com, almu.abba@udusok.edu.ng, abdullahi.mabubakar@udusok.edu.ng
Jude Oguejiofor Nwoji
Department of Computer Science, Waziri Umaru Federal Polytechnic, Birnin Kebbi - Nigeria
Email: judycharles222@wufpbk.edu.ng

---**ABSTRACT**---

In the competitive business world of today, effective use of information technologies is essential to a world's success. As a result, sharing data and information over the Internet in an information system could be risky. A comparative analysis of these algorithms is required due to the ever-changing nature of security threats and the continuous advancements in computer technology. Therefore, we evaluate the performance and security strength of AES, 3DES and Blowfish algorithms. Security strength, throughput, memory usage, and execution time are the evaluation measures. Data from text files, audio files, videos, and images were used in our research. The security study was conducted using CrypTool 2, and the performance trials were created using the Java programming language. From the experimental findings, Blowfish performed better in terms of throughput and performance for both large and small files than AES and 3DES. AES ranked second in terms of speed and throughput while maintaining a balance between security and performance. 3DES did the worst in throughput and speed. Blowfish and AES were the two algorithms with the highest memory usage. Furthermore, 3DES is a wise option in resource-constrained scenarios. AES is the most secure algorithm because it offers robust protection against numerous attacks.

Keywords - **Symmetric algorithms, performance evaluation, AES, 3DES, Blowfish, security evaluation**

Date of Submission: November 05, 2024 Date of Acceptance: December 26 ,2024

I. INTRODUCTION

Ensuring the security and integrity of data is crucial in today's interconnected world, when sensitive information is sent globally. Given how much time people spend online, network security is now a crucial component of data transfer [1]. The efficient use of information technologies is critical to a business's success in today's cutthroat environment. Rapid growth is also observed in the values of digital information stored on these systems. The majority of these systems are network-based and connect to open networks like the Internet to share data. Thus, using the Internet in an information system may be dangerous for conducting business. Thus, it is imperative that computer and cyber security measures be put in place in order to safeguard digital assets [2].

Through the Internet, people are sending vast amounts of important data that take a lot of time to process, like emails, bank transactions, and online purchases. However, because of their great visibility, they are vulnerable to heavy attacks or become desirable targets for attackers. Symmetric encryption is a contemporary phenomenon that can be used to overcome this. Information is shielded from users for whom it is not intended by using symmetric encryption [3].

Internet service providers heavily rely on cryptographic encryption and decryption technologies to guarantee data security and confidentiality during transmission [4]. The primary attributes that differentiate and identify one encryption algorithm from another are its capacity to protect data, execution speed, memory consumption, and implementation efficiency. The foundation of contemporary information security are cryptographic algorithms, which offer a way to shield data from unwanted access and preserve its privacy.

There are numerous cryptographic approaches available to secure information systems. AES (Advanced Encryption Standard), 3DES (Triple Data Encryption Standard), and Blowfish are the most popular symmetric encryption algorithms among the many that are available. The constant improvements in computer technology and the changing nature of security risks necessitate a comparative review of these algorithms. Attackers may be able to crack encryption algorithms using more advanced or brute-force methods as processing power rises. As a result, in order to guarantee that encryption algorithms are appropriate for a variety of applications and security needs, it is imperative that their strengths and weaknesses be evaluated.

AES has the lowest execution time, according to some research [1, 5-6], but the majority of research [7, 4, 8-9, 3, 10-12] concluded that Blowfish has better execution time. Memory utilization is only taken into account by a small number of studies [7, 6, 10-14,] as one of the performance indicators when comparing symmetric algorithms, despite the fact that it is a crucial component in assessing an algorithm's performance. Together with performance examination of a few cryptographic methods, Mota in [11] also carried out security analysis. On the other hand, the security comparison is based solely on theoretical claims.
Therefore, this research evaluates both the performance and security strength of AES, 3DES and Blowfish symmetric algorithms. Security strength, throughput, memory usage, and execution time are the evaluation measures. Data from text files, audio files, videos, and images were used as specimens in our research. An experiment is conducted, where the security analysis was performed using CrypTool 2 instead of just theoretical backgrounds, and the performance trials were created using the Java programming language.

This is how the remainder of the paper is structured: The related work is shown in Section II. Section III presents the methodology. The results and discussion of this study are presented in Sections IV and V. In Section VI, a conclusion is presented.

II. RELATED WORKS

The development and in-depth analysis of popular cryptographic algorithms, such as DES, 3DES, AES, RSA, and blowfish, was done by Radhi and Ogla in [7]. The results show that blowfish uses the least amount of memory and takes the least amount of time to decrypt files of different sizes (25K, 50K, 1M, 2M, 3M, and 4M). In comparison to previous cryptographic algorithms, this results in a speed increase of about three times.

Using a variety of file formats, Assa-Agyei and Olajide in [1] examined the encryption, decryption times, and throughput (speed) of the three most used block cipher algorithms: Twofish, Blowfish, and AES. They used Python to implement the various symmetric encryptions in their experiment. In their experiment, they used a benchmark key size of 128 bits to obtain reliable values for assessing the effectiveness of the cryptographic algorithms that were being evaluated. The experiment was also conducted three times, and the average execution time was noted each time. Overall findings demonstrated that the AES method is better suited for safe data transmission.

The encryption speeds of five distinct cryptographic symmetric block-cipher algorithms, DES, Triple DES, Blowfish, Twofish, and Threefis, are compared by Alabdulrazzaq and Alenezi in [4]. With different text file sizes, the simulation is run in Python, and the results demonstrate that Blowfish performs better than the other examined methods. They haven’t taken into account elements like the kind and size of the file, the machine on which the cryptographic technique would run, and the required degree of security.

A comparative analysis of the encryption process and throughput for the AES, DES, 3DES, and Blowfish algorithms in Internet of Things devices was carried out by Kureshi and Mishra in [8]. A system for measuring temperature and humidity, based on the Raspberry Pi 3B+, has been created so that these methods may be experimentally compared. The sensor data was sent to the server via a variety of encryption techniques with variable key lengths. The strengths and drawbacks of each encryption algorithm in their IoT application have been assessed through a comparative examination of the encryption process time and throughput. According to the experimental results, Blowfish has the maximum throughput while DES and 3DES have the lowest encryption time requirements.

AES, 3DES, Blowfish, and Twofish are also the four symmetric algorithms that were the subject of a performance comparison by Dibas and Sabri in [5]. For both the encryption and decryption operations, they assessed the ciphertext size, memory usage, and execution time. They used C# to create a.NET application, which allowed them to compare the results against various file sizes. Each of the targeted algorithms was encrypted and decrypted five times against a range of file sizes, including 1 KB, 100 KB, 1 MB, 10 MB, and 100 MB. They concluded from their findings that Twofish has the longest execution time and AES has the shortest execution time for both encryption and decryption procedures. AES and 3DES use less memory during the encryption process than blowfish and twofish, albeit their memory usage is fairly similar. AES, however, used less RAM for decryption. The largest ciphertext sizes are, finally, shared by Blowfish and Twofish.

Five encryption methods for mobile devices were the subject of a performance evaluation research by Rouaf and Yousif in [15]. The algorithms that are being examined are REA, TEA, RSA, DES, and AES. They conduct two investigations. The first experiment measures the amount of time it takes for three mobile devices to encrypt ten files using five different encryption techniques. Using the same five encryption techniques, the second experiment manages one mobile device's battery consumption for the same contents. The experiment's findings showed that REA is the slowest algorithm and AES is the fastest. With less than 1 mAh needed to encrypt 20 files totaling 1500 KB, AES is the least power-hungry algorithm.

Using simulations to calculate time and memory use metrics, Commey et al. In [9] assessed the performance of AES, 3DES, Blowfish, and RSA on records in a particular dataset. Simulations have shown that Blowfish outperforms AES, which outperforms 3DES in terms of processing time, with RSA being the slowest procedure. When it came to memory use, the symmetric encryption methods (AES, 3DES, and Blowfish) and the asymmetric encryption algorithm (RSA) utilized around twice as much as each other.

The inner workings of common encryption algorithms are explained and a comprehensive overview of them is presented by Alenezi et al. In [16]. We also take ten different symmetric encryption techniques and test their performance using a Java simulation. We compare the following algorithms: DES, DESede, XTEA, IDEA, BlowFish, RC2, RC4, RC6, DES, BlowFish, and SEED. Several plaintext file sizes, including 1GB, 500MB, 100MB, 10MB, and 1MB, were used to simulate these techniques. They noticed from their findings that the best results in terms of encryption time and throughput were obtained by RC4, RC6, and AES.

Using a Java cryptography package, Advani and Gonsai in [17] examined several file types—such as image, audio, and video—with different asymmetric algorithms. Comparing files of different sizes is done with AES, DES, DESede, Blowfish, Twofish, and so on. We have applied a number of these padding strategies, including PKCS5Padding, CBS, CBC, and others. After 48 different evaluations on a variety of file formats, it was ultimately determined that, in terms of encryption and decryption times, the symmetric algorithms used by AES and Blowfish perform better.

A performance evaluation of the symmetric data encryption algorithms Blowfish and Advanced Encryption Standard (AES) was carried out by Buhari et al. In [3]. A total of four distinct data kinds are evaluated: textual files, audio files, video files, and image files. The metrics used for performance evaluation are throughput and encryption time. The prototype is written in Java and compiled using the JDK 7.1 development kit using the Netbeans IDE7.1.2 and default settings. The evaluation's findings showed that blowfish outperforms AES in terms of efficiency. However, for Blowfish, when data size increases, the encryption time occasionally gets shorter. This is explained by the fact that Blowfish employs key sizes of 126, 192, or 256.

In their practical implementation using Java, Vyakaranal and Kengond in [6] examined several symmetric key cryptographic algorithms, including DES, 3DES, AES, and Blowfish, taking into account factors like encryption time, decryption time, entropy, memory usage, throughput, avalanche effect, and energy consumption. Proposed work that takes into account tradeoff performance in terms of cost of different parameters has highlighted the practical implementation of algorithms instead of only theoretical ideas. The avalanche impact of algorithms and battery usage have been examined. It demonstrates how well AES performs overall among the algorithms under consideration in the performance analysis.

To lower power consumption in Wireless Sensor Networks (WSNs), Al Sibahee et al. In [18] evaluates four of the most used encryption algorithms: RSA for an asymmetric cipher and RC4, DES, and AES for a symmetric cipher. For those encryption techniques, a comparison has been done for various parameters, including data block sizes, key sizes, and encryption/decryption speeds. The simulation data are provided to show how effective each algorithm is in using up electricity.

According to a few chosen important criteria, Semwal and Sharma in [10] compared several cryptographic encryption algorithms based on their main features and then talked about how much each algorithm cost in terms of performance. DES, 3DES, IDEA, CAST128 AES, Blowfish, RSA, ABE, and ECC are a few of the algorithms selected for the task. In terms of memory requirements, Blowfish is the best option (RSA has a high demand). This makes Blowfish suitable for small applications, particularly embedded ones and devices with limited memory. When it comes to encryption and decryption times, blowfish has the shortest encryption and decryption times whereas RSA takes the longest. When message integrity and privacy are of utmost importance, AES may be the recommended option.

By taking into account theories and researches, Yassein et al. In [19] provided a thorough analysis of symmetric key and asymmetric key encryption algorithms that improved data security in cloud computing systems. They talk about symmetric encryption methods like AES, DES, 3DES, and Blowfish, and asymmetric encryption algorithms like RSA, DSA, Diffie-Hellman, and Elliptic Curve. They discovered that the difference parameter had an impact on the various algorithms' efficiency. The need for efficient, reliable, and highly secure algorithms that are compatible with the vast amount of data stored in cloud computing has arisen due to the current situation of growing demand for cloud applications. When it comes to cloud apps, security and speed are the most crucial factors.

Using standard encryption techniques, Mota et al. In [11] conducted a comparison. Data encryption and decryption times, security efficiency, memory utilization, power consumption, jitter, and latency will all be taken into account while making this comparison. AES, DES, 3DES, Blowfish, and other symmetric algorithms are compared. Elgamal and the RSA ECC asymmetric algorithms are contrasted. Blowfish and AES are the top two symmetric algorithms, respectively. Save for the time required for the signature verification key, ECC outperforms RSA in most aspects when it comes to asymmetric algorithms. But theoretical foundations support the security comparison.

Based on encryption and decryption times, throughput, key sizes, avalanche effects, memory, correlation analysis, and entropy, Mushtaq et al. in [12] assessed and contrasted the performance of different encryption algorithms. Based on the findings of many researchers, they elucidated the performance analysis and addressed the security considerations in the construction of the encryption algorithm according to the assessment parameters. The performance evaluation indicates that, given the available resources, the outcomes of Blowfish, AES, and HiSea offer more security. When memory and encryption/decryption time are critical factors, Blowfish is the best choice because it is a software solution that operates efficiently. But the avalanche effect, which performs exceptionally well, can be used to examine AES, while HiSea performs well when it comes to entropy and correlation analysis. They therefore come to the conclusion that applications where integrity and secrecy are of the utmost importance can make use of the AES and HiSea.

The memory building rate, various key sizes, CPU use time period, and encryption speed of the four methods were examined by Awotunde et al. in [14] in order to ascertain the computational resource consumption and execution time of each algorithm. The results demonstrate that, in most circumstances, the cryptographic algorithm's key length and resource consumption are proportionate, as demonstrated by the key lengths of the Blowfish, AES, 3DES, and DES algorithms, respectively.

Four factors are taken into consideration in Okolie and Adetoba's in [13] comparative examination of four symmetric key encryption algorithms: AES, DES, 3DES, and Blowfish: encryption time, decryption time, memory use, and amount of output bytes. The effectiveness of each of these methods was examined through the usage of Data Security Model Analyser to analyze experimental data. According to the experimental results, DES requires the least amount of time to decode data, while AES requires the least amount of time to encrypt it. While there is a slight difference in the encryption and decryption times between DES and 3-DES, 3-DES has the smallest output byte size. It has been determined that AES uses the least amount of memory size when encrypting data.

The performance of the widely used Advanced Encryption Standard (AES) and Blowfish algorithms is compared and evaluated by Raigoza and Jituri in [20]. For various kinds of data string values, the execution time is measured. According to their tests, there is a 200–300 millisecond speed gap between the AES and Blowfish algorithms. Furthermore, no significant differences were found between the algorithms evaluated in their studies where the data size was adjusted, leading to an approximate length of encrypted data for both the AES and Blowfish algorithms.

Panda in [21] evaluates asymmetric (RSA) and symmetric (AES, DES, Blowfish) cryptographic methods using a variety of file types, including text, picture, and binary files. Evaluation metrics including encryption time, decryption time, and throughput have been compared for these encryption algorithms. According to simulation data, AES outperforms competing algorithms in terms of throughput and encryption-decryption time.

III. METHODOLOGY

This section describes the techniques and simulation choices made to evaluate the performance and security of the selected algorithms.

A. Evaluation Metrics

The study uses a set of assessment measures to evaluate several aspects of their effectiveness and usefulness. The specified metrics are intended to offer a thorough grasp of the many contexts in which these algorithms operate. These include security strength, throughput, memory utilization, and execution time. The duration of an encryption algorithm's execution is measured in milliseconds; memory usage is used to evaluate how much memory each algorithm uses during encryption procedures; throughput is the amount of encrypted plaintext divided by the encryption time in milliseconds; and security strength is the degree of resilience or resistance a cryptographic algorithm demonstrates against different kinds of attacks. In this experiment, a brute force cryptanalysis assault was employed.

B. Experimental Environment

All test experiments were conducted using the Intel(R) Core (TM) i5-3337U CPU @ 1.80 GHz, 4.00 GB (3.87 GB useable), 64-bit operating system, x64-based processor, and Windows 11 Pro as hardware and software.

C. Experimental Setup

The Java programming language has been used to create the experiments. To track the execution time in milliseconds and the memory consumption in kilobytes, two customized programs are developed: EncryptionSpaceMonitor.java to track memory usage and EncryptionTimer.java to calculate encryption time. Six distinct data file types and sizes are used during program execution.

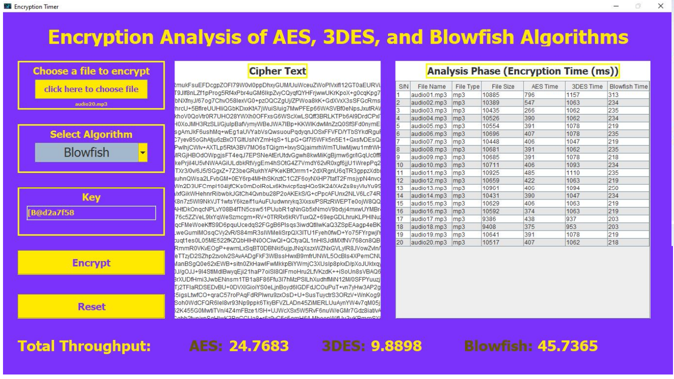
i. Performance Evaluation

With the default settings in the JDK21 development kit for Java, the IntelliJ IDE 2021.2.1 was used to compile the experiment application. Several iterations of the experiment were conducted to ensure that the outcomes are reliable and suitable for comparing the various algorithms.

The specimens for our studies included image, audio, video, and text file data. The Java Interface Development Environment was used to create the prototype, which is made up of interfaces. Using the computer keyboard to upload the original data (plaintext) file, the monitor to display the output (ciphertext) along with the encryption key, the computer processing unit to process tasks, and the encryption time for both algorithms and the interface enables the user to interact with the application through a user-friendly designed graphical user interface.

Two boxes labelled "analysis phase" and "cipher text" are present in the interface. When any of the algorithms are chosen in a dropdown field and the Encrypt button is clicked, the original data is uploaded to the box using the Select file button, and the encrypted data is stored in the cipher text. The buttons labelled "click to encrypt using AES algorithm" and "click to encrypt using Blowfish algorithm" are used to encrypt the original material and, respectively, transform plaintext into cipher text. The encrypted datakey and the reset data button, which is used to refresh the data boxes, are displayed in a column beneath the cipher text. The experiment's outcome is shown in a table on the opposite side called the "Result table."

By pressing the Encrypt button, the file name, size, encryption time, and memory usage for the algorithms under study are displayed automatically. The throughput of the encryption time will be computed later. This can be shown in Fig. 1 and Fig. 2, for speed and throughput analysis and memory utilization analysis respectively.



Encryption Analysis of AES, 3DES, and Blowfish Algorithms

audio20.mp3

Select Algorithm: Blowfish

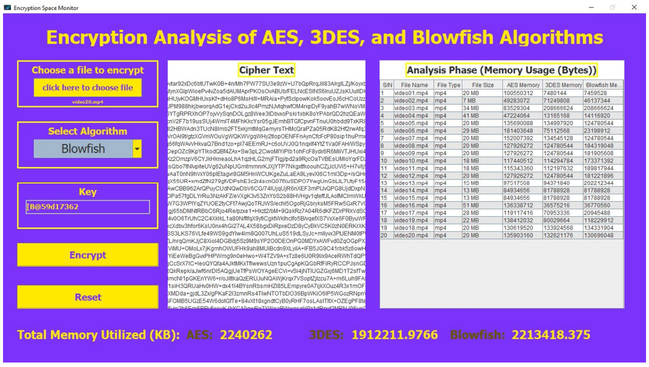
Key: [B@d2a7f58

Analysis Phase (Encryption Time (ms))

S/N	File Name	File Type	File Size	AES Time	3DES Time	Blowfish Time
1	audio01.mp3	mp3	10885	796	1157	313
2	audio02.mp3	mp3	10389	547	1063	234
3	audio03.mp3	mp3	10435	266	1062	235
4	audio04.mp3	mp3	10526	390	1062	234
5	audio05.mp3	mp3	10554	391	1078	219
6	audio06.mp3	mp3	10696	407	1078	235
7	audio07.mp3	mp3	10448	406	1047	219
8	audio08.mp3	mp3	10681	391	1062	235
9	audio09.mp3	mp3	10685	391	1078	218
10	audio10.mp3	mp3	10711	406	1093	234
11	audio11.mp3	mp3	10925	485	1110	235
12	audio12.mp3	mp3	10659	422	1063	219
13	audio13.mp3	mp3	10961	406	1094	250
14	audio14.mp3	mp3	10431	390	1047	234
15	audio15.mp3	mp3	10629	406	1063	219
16	audio16.mp3	mp3	10592	374	1063	219
17	audio17.mp3	mp3	9386	438	937	203
18	audio18.mp3	mp3	9408	375	953	203
19	audio19.mp3	mp3	10641	391	1078	219
20	audio20.mp3	mp3	10517	407	1062	218

Total Throughput: AES: 24.7683 3DES: 9.8898 Blowfish: 45.7365

Fig. 1: The main interface of the prototype (for speed and throughput analysis)



Encryption Analysis of AES, 3DES, and Blowfish Algorithms

video20.mp4

Select Algorithm: Blowfish

Key: [B@59d17362

Analysis Phase (Memory Usage (Bytes))

S/N	File Name	File Type	File Size	AES Memory	3DES Memory	Blowfish Me...
1	video01.mp4	mp4	20 MB	100550312	7480144	7459528
2	video02.mp4	mp4	7 MB	49283072	71249808	46137344
3	video03.mp4	mp4	34 MB	83529304	208666624	208666624
4	video04.mp4	mp4	41 MB	47224064	13165168	14116920
5	video05.mp4	mp4	20 MB	135690088	134997920	124780544
6	video06.mp4	mp4	29 MB	181403648	75112568	23198912
7	video07.mp4	mp4	20 MB	152007392	134545128	124780544
8	video08.mp4	mp4	20 MB	127926272	124780544	184319048
9	video09.mp4	mp4	20 MB	127926272	124780544	181905608
10	video10.mp4	mp4	18 MB	117440512	114294784	173371392
11	video11.mp4	mp4	18 MB	115343360	112197632	189817944
12	video12.mp4	mp4	20 MB	127926272	124780544	181221896
13	video13.mp4	mp4	15 MB	97517568	94371840	202212344
14	video14.mp4	mp4	13 MB	84934656	81788928	81788928
15	video15.mp4	mp4	13 MB	84934656	81788928	81788928
16	video16.mp4	mp4	51 MB	136338712	36575216	36770560
17	video17.mp4	mp4	28 MB	119117416	70953336	20945488
18	video19.mp4	mp4	22 MB	138412032	80029664	118229912
19	video18.mp4	mp4	20 MB	130619520	133924568	134331904
20	video20.mp4	mp4	20 MB	135903160	132621176	130696048

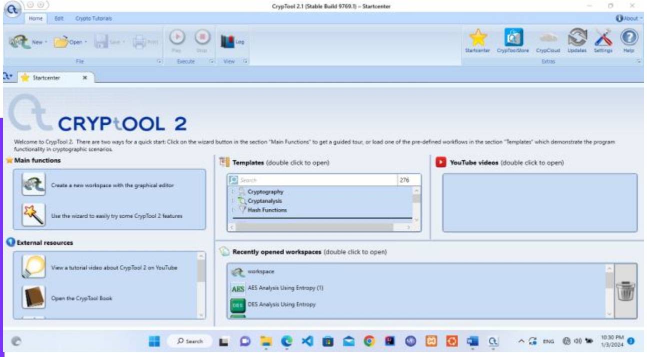
Total Memory Utilized (KB): AES: 2240262 3DES: 1912211.9766 Blowfish: 2213418.375

Fig. 2: The main interface of the prototype (for memory utilization analysis)

ii. Security Strength Evaluation

For security strength analysis, CrypTool 2 is utilized. It is an open-source initiative that offers used for the analysis of the security strength of the implemented cryptographic algorithms. executes more than 400 algorithms that span a broad spectrum of cryptographic methods, from public key cryptography to classical ciphers.

It covers a wide range of cryptography topics, from basic concepts to advanced algorithms, and supports the analysis of cryptographic algorithms, allowing users to simulate and study their behavior. Its cryptanalysis tools allow users to evaluate the security strength of various cryptographic primitives and protocols. Finally, it visualizes cryptographic algorithms and processes, making them easier to understand. This can be shown in Fig. 3.



Welcome to CrypTool 2. There are two ways for a quick start: Click on the wizard button in the section "Main Functions" to get a guided tour, or load one of the pre-defined workflows in the section "Templates" which demonstrate the program functionality in cryptographic scenarios.

Create a new workspace with the graphical editor

Use the wizard to easily try some CrypTool 2 features

View a tutorial video about CrypTool 2 on YouTube

Open the CrypTool Book

workspace

AES Analysis Using Entropy (1)

DES Analysis Using Entropy

Fig. 3: The home interface of CrypTool 2

D. Experimental Workload

The study included six example data files, each of which came in a variety of data types: text, images, audio, video, and PDF files. The sample data files were tested with data sizes ranging from 1MB to 30MB.

IV. RESULTS

A. Execution Time and Throughput

An encryption algorithm's throughput is calculated using encryption time. The performance of each encryption technique is demonstrated in this subsection by computing encryption time and throughput.

iii. The Effect of Text Files for Cryptography Algorithm on Encryption Time and Throughput

Table 1: Experimental result of encryption time and throughput for audio files

S/N	File Name	File Type	File Size	Execution Time (milliseconds)		
				AES	3DES	Blowfish
	audio01.mp3	mp3	1026	1130	188	188
	audio02.mp3	mp3	2055	566	204	47
	audio03.mp3	mp3	5121	314	549	111
	audio04.mp3	mp3	10248	440	1036	220
	audio05.mp3	mp3	20953	612	2071	415
	audio06.MP3	MP3	30954	843	3076	612
	Average Execution Time (ms)			651	1187	266
	Throughput (bpms)			18.0172	9.8761	44.1664

Table 1's results indicate that the average encryption time for AES was 651 ms, corresponding to an 18.0172 unit throughput; the average encryption time for 3DES was 1187 ms, corresponding to a 9.8761 unit throughput; and the average encryption time for Blowfish was 266 ms, corresponding to a 44.1664 unit throughput for audio files. This can also be shown in Fig. 4 and Fig. 5 respectively.

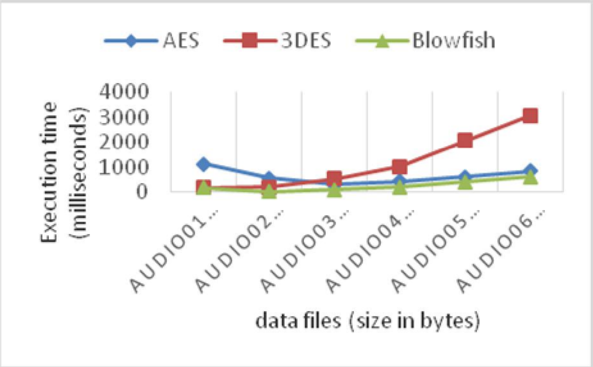


Fig. 4: Performance comparison based on execution time on audio data files

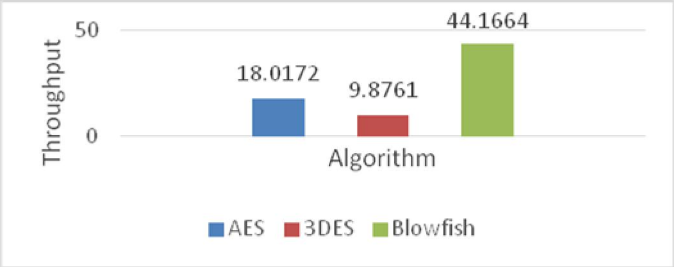


Fig. 5: Performance comparison based on throughput on audio data files

iv. The Effect of Text Files for Cryptography Algorithm on Encryption Time and Throughput

Table 2: Experimental result of encryption time and throughput for text files

S/N	File Name	File Type	File Size	Execution Time (milliseconds)		
				AES	3DES	Blowfish
	doc01.txt	Txt	1024	612	157	79
	doc02.txt	Txt	2119	235	251	47
	doc03.txt	Txt	5297	204	533	110
	doc04.txt	Txt	10630	267	1052	220
	doc05.txt	Txt	20754	613	3123	568
	doc06.txt	Txt	30947	1068	4551	878
	Average Execution Time (ms)			500	1611	317
	Throughput (kbpms)			23.5982	7.3209	37.2087

Table 2's results indicate that the average encryption time for AES was 500 ms, corresponding to a throughput of 23.5982 units; the average encryption time for 3DES was 1611 ms, corresponding to a throughput of 7.3209 units; and the average encryption time for Blowfish was 317 ms, corresponding to a throughput of 37.2087 units for text files. This can also be shown in Fig. 6 and Fig. 7 respectively.

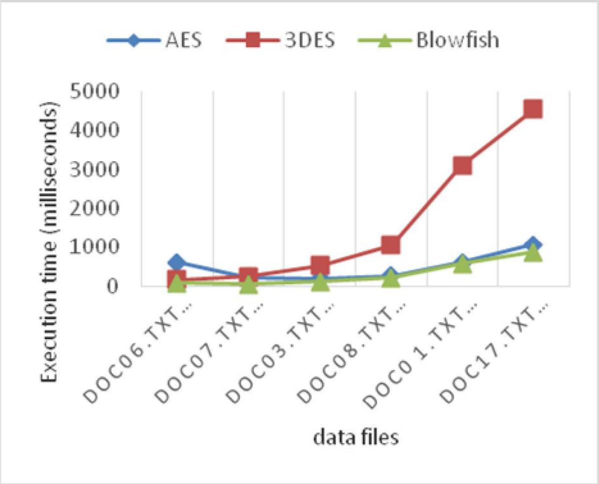


Fig. 6: Performance comparison based on execution time on text files

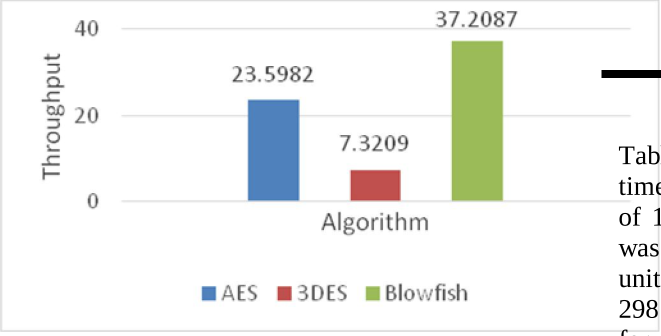


Fig. 7: Performance comparison based on throughput on text files

v. The Effect of Video Files for Cryptography Algorithm on Encryption Time and Throughput

Table 3: Experimental result for video files

S/N	File Name	File Type	File Size	Execution Time (milliseconds)		
				AES	3DES	Blowfish
	vid01.mp4	mp4	1030	565	188	78
	vid02.mp4	mp4	2058	393	220	47
	vid03.mp4	mp4	5130	314	548	110
	vid04.mp4	mp4	10252	472	1020	204
	vid05.mp4	mp4	20575	675	2056	423
	vid06.mp4	mp4	30753	2006	3106	926
	Average Execution Time (ms)			738	1190	298
	Throughput (bpms)			15.7736	9.7784	39.0369

Table 3's result demonstrates that the average encryption time for AES was 738 ms, corresponding to a throughput of 15.7736 units; the average encryption time for 3DES was 1190 ms, corresponding to a throughput of 9.7784 units; and the average encryption time for Blowfish was 298 ms, corresponding to a throughput of 39.0369 units for video files. This can be shown in Fig. 8 and Fig. 9 respectively.

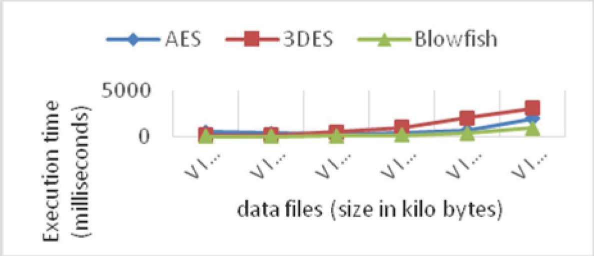


Fig. 8: Performance comparison based on execution time on video files

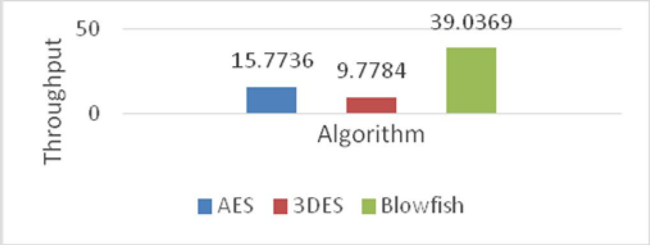


Fig. 9: Performance comparison based on throughput on audio data files

vi. The Effect of Image Files for Cryptography Algorithm on Encryption Time and Throughput

Table 4: Experimental result of encryption time and throughput for image files

				Execution Time (milliseconds)		
S/N	File Name	File Type	File Size	AES	3DES	Blowfish
	img 01.jpg	Jpg	1009	597	157	63
	img 02.jpg	Jpg	2052	251	189	47
	img 03.jpg	Jpg	5143	314	534	125
	img 04.jpg	Jpg	10260	392	1036	220
	img 05.jpg	Jpg	20847	675	2087	408
	img 06.jpg	Jpg	30067	832	2964	596
	Average Execution Time (ms)			510	1161	243
	Throughput (bpms)			22.6651	9.9581	47.5517

According to table 4's results, the average encryption time for AES was found to be 510 ms, with a corresponding throughput of 22.6651 units; the average encryption time for 3DES was found to be 1161 ms, with a corresponding throughput of 9.9581 units; and the average encryption time for Blowfish was found to be 243 ms, with a corresponding throughput of 47.5517 units for image files. This can be shown in Fig. 10 and Fig. 11 respectively.,

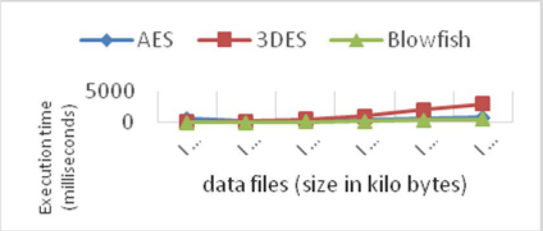


Fig. 10: Performance comparison based on execution time on image files

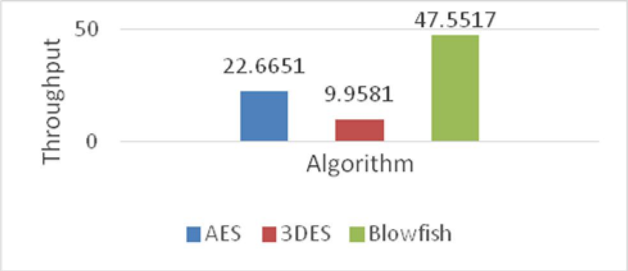


Fig. 11: Performance comparison based on throughput on image files

vii. The Effect of PDF Files for Cryptography Algorithm on Encryption Time and Throughput

Table 5: Experimental result of encryption time and throughput for PDF files

			Execution Time (milliseconds)		
File Name	File Type	File Size	AES	3DES	Blowfish
file 01.pdf	Pdf	1027	863	174	63
file 02.pdf	Pdf	2110	518	236	54
file 03.pdf	Pdf	5259	306	534	125
file 04.pdf	Pdf	10454	456	1067	219
file 05.pdf	Pdf	20906	659	2087	440
file 06.pdf	Pdf	31363	832	3153	974
Average Execution Time (ms)			606	1209	313
Throughput (bpms)			19.5704	9.8082	37.9301

The findings presented in Table 5 demonstrate that the average encryption time for AES was 606 ms, corresponding to a throughput of 19.5704 units; the average encryption time for 3DES was 1209 ms, corresponding to a throughput of 9.8082 units; and the average encryption time for Blowfish was 313 ms, corresponding to a throughput of 37.9301 units for PDF files. This ca be shown in Fig. 12 and Fig. 13 respectively.

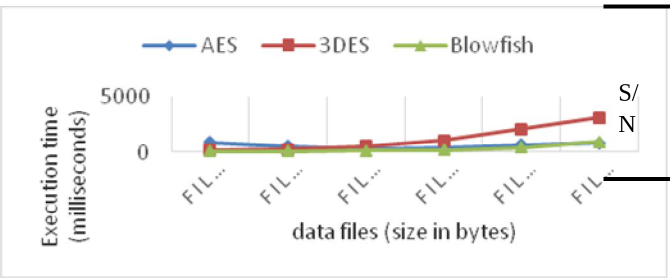


Fig. 12: Performance comparison based on execution time on PDF files

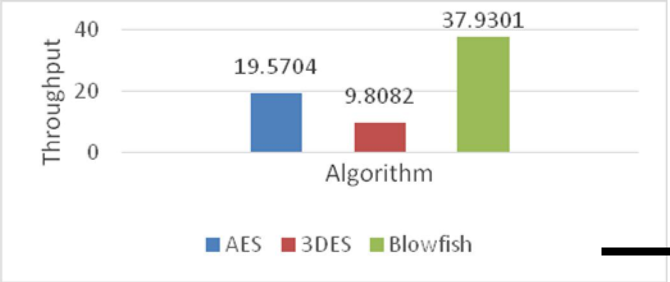


Fig. 13: Performance comparison based on throughput on PDF files

B. Memory Utilization

This subsection computes memory utilization/usage to demonstrate the superior performance of each encryption scheme.

i. The Effect of Audio Files for Cryptography Algorithm on Memory Utilization

Table 6: Experimental result of memory utilization for audio files

S/N	File Name	File Type	File Size	Memory Usage (kilobytes)		
				AES	3DES	Blowfish
	audio01.mp3	mp3	1026	14801	3215	5629
	audio02.mp3	mp3	2055	3915	15360	25054
	audio03.mp3	mp3	5121	35840	32768	32768
	audio04.mp3	mp3	10248	20950	6145	22368
	audio05.mp3	mp3	20953	41962	13817	7644
	audio06.MP3	MP3	30954	31154	179200	179200
Average Memory Usage (kilobytes)				148622	250506	272663

Table 6's result indicates that the average memory utilized for AES was 148622 kb, the average memory utilized for 3DES was 250506 kb, and the average memory utilized for Blowfish was 272663 kb for audio files. This can be shown in Fig. 14.

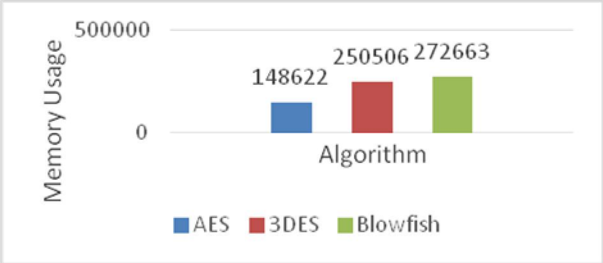


Fig. 14: Performance comparison based on memory utilization on audio files

ii. The Effect of Text Files for Cryptography Algorithm on Memory Utilization

Table 7: Experimental result of memory utilization for text files

				Memory Usage (kilobytes)		
S/N	File Name	File Type	File Size	AES	3DES	Blowfish
1	doc01.txt	Txt	1024	14725	3053	6177
2	doc02.txt	Txt	2119	3881	15360	28253
3	doc03.txt	Txt	5297	17909	32768	32768
4	doc04.txt	Txt	10630	65536	7208	7677
5	doc05.txt	Txt	20754	40409	7652	7625
6	doc06.txt	Txt	30947	31256	20367	11361
Average Memory Usage (kilobytes)				173717	86407	93861

Table 7's result demonstrates that the average memory utilized for AES was 173717 kb, the average memory utilized for 3DES was 86407 kb, and the average memory utilized for Blowfish was 93861 kb for audio files. This can be shown in Fig. 15.

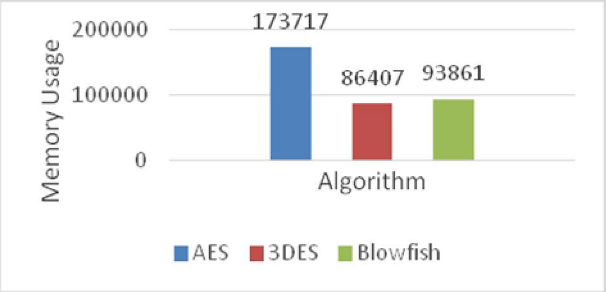


Fig. 15: Performance comparison based on memory utilization on text files

iii. The Effect of PDF Files for Cryptography Algorithm on Memory Utilization

Table 8: Experimental result of memory utilization for PDF files

				Memory Utilization (bytes)		
S/N	File Name	File Type	File Size	AES	3DES	Blowfish
1	file01.pdf	Pdf	1027	14720	2816	5344
2	file02.pdf	Pdf	2110	3882	15360	15360
3	file03.pdf	Pdf	5259	9778	32768	32768
4	file04.pdf	Pdf	10454	65534	39795	62464
5	file05.pdf	Pdf	20906	33422	13845	7623
6	file06.pdf	Pdf	31363	31202	21986	21982
Average Memory Usage (kilobytes)				158538	126570	145540

The results in Table 8 demonstrate that the average memory used for AES was 158538 kb, the average memory used for 3DES was 126570 kb, and the average memory used for Blowfish was 145540 kb for audio files. This can be shown in Fig. 16.

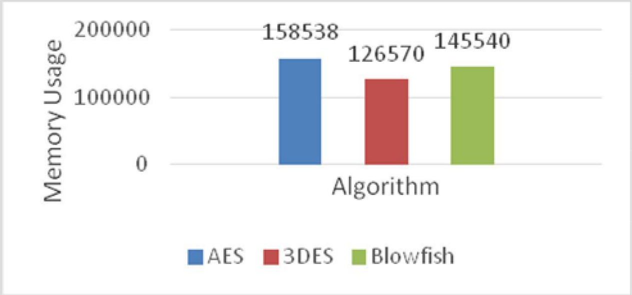


Fig. 16: Performance comparison based on memory utilization on PDF files

iv. The Effect of Image Files for Cryptography Algorithm on Memory Utilization

Table 9: Experimental result of memory utilization for image files

				Memory Utilization (bytes)		
S/N	File Name	File Type	File Size	AES	3DES	Blowfish
1	img01.jpg	Jpg	1009	11644	7168	5252
2	img02.jpg	Jpg	2052	554	15360	15360
3	img03.jpg	Jpg	5143	36350	18269	11035
4	img04.jpg	Jpg	10260	16993	5633	22367
5	img05.jpg	Jpg	20847	40966	14317	7630
6	img06.jpg	Jpg	30067	27157	174080	174080
Average Memory Usage (kilobytes)				133665	133665	235723

The results in Table 9 demonstrate that the average memory used for AES, 3DES, and Blowfish for audio files was measured to be 133665kb, 235723kb, and 133665kb, respectively. This can be shown in Fig. 17.

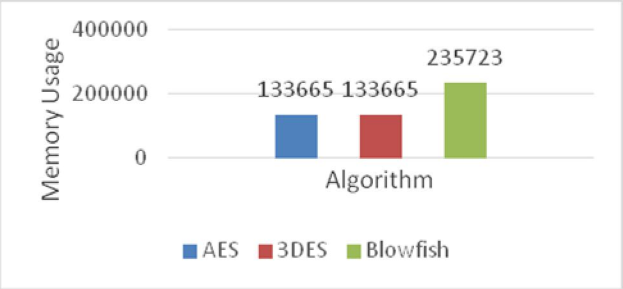


Fig. 17: Performance comparison based on memory utilization on image files

v. The Effect of Video Files for Cryptography Algorithm on Memory Utilization

Table 10: Experimental result of memory utilization for video files

				Memory Utilization (bytes)		
S/N	File Name	File Type	File Size	AES	3DES	Blowfish
1	video01.mp4	mp4	20992	14671	2868	5566
2	video02.mp4	mp4	7307	3961	15360	25620
3	video03.mp4	mp4	35587	18358	32768	32768
4	video04.mp4	mp4	42627	65536	6650	5662
5	video05.mp4	mp4	21456	39414	8670	9186
6	video06.mp4	mp4	30630	34292	20435	20482
Average Memory Usage (kilobytes)				176231	86751	99283

Table 10's result indicates that the average memory utilized for AES was 176231 kb, the average memory utilized for 3DES was 86751 kb, and the average memory utilized for Blowfish was 99283 kb for audio files. This can be shown in Fig. 18.

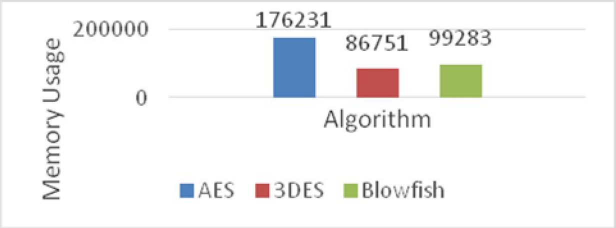


Fig. 18: Performance comparison based on memory utilization on video files

C. The Total Average of Memory Utilized

Table 11: Experimental result of total average of memory utilized by each algorithm

File Type	AES	3DES	Blowfish
Audio	**148622**	**250506**	**272663**
Text	**173717**	**86407**	**93861**
PDF	**158538**	**126570**	**145540**
Image	**133665**	**133665**	**235723**
Video	**176231**	**86751**	**99283**
Total	**790773**	**683899**	**847070**

Table 11 displays the results, which indicate that the total average memory utilized for AES was 790773 kb, the total average memory utilized for 3DES was 683899 kb, and the total average memory utilized for Blowfish was 847070 kb for audio files. This can be shown in figure 19.

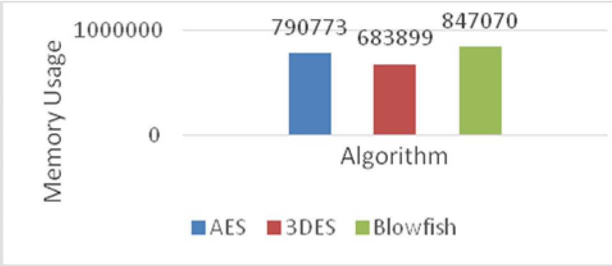


Figure 19: Performance comparison based on total memory utilized by each

D.Security Strength

Using CrypTool 2, an examination was performed to further evaluate the security strength of AES, 3DES, and Blowfish. In order to determine how resilient the algorithms were to attacks, brute force approaches were used. This produced insights on key entropy and the possible amount of time needed for a brute force attack.

i. AES Security Strength

Strong security strength was shown by AES in every analysis. With key sizes of 128 bits, 192 bits, and 256 bits, the technique demonstrated exceptional resilience against cryptographic assaults. The set number of rounds and substitution-permutation network (SPN) structure made it a dependable method for protecting the privacy of data.

According to CrypTool 2's investigation, AES demonstrated significant key entropy while maintaining a high level of security. The technique showed an unfeasible timeframe for a successful key discovery, illustrating the resilience of AES against brute force attacks. According to the experiment's key entropy, brute-forcing an AES key is expected to take roughly 2.7 * 1025 years. This astronomical duration confirms AES's status as a highly secure encryption system and highlights its resilience to brute force attacks.

ii. 3DES Security Strength

Even though 3DES kept some security, it's crucial to be aware of its key size limitations. Certain vulnerabilities are introduced by the use of three distinct 56-bit keys, particularly when compared to AES and Blowfish.
Due to 3DES's key size constraints, some vulnerabilities were identified via the CrypTool 2 investigation. Although the technique exhibited a certain degree of security, the brute force study suggested a significantly shorter period for possible key finding. This observation is consistent with the aging architecture of the algorithm and the industry's move toward safer alternatives.

Based on the key entropy found in the experiment, it is predicted that brute forcing a 3DES key would take 500 billion years or more. This astronomical duration confirms that 3DES is a less secure encryption method than AES, but it still stands well against brute force attacks.

iii. Blowfish Security Strength

The security strength of Blowfish was found to be adequate. Though flexible, its various key length (32–448 bits) might also raise questions about how resistant the algorithm is to particular cryptographic assaults.
Blowfish's examination by CrypTool 2 demonstrated acceptable key entropy and resistance to brute force attacks. The algorithm's security strength was enhanced by the degree of flexibility it offered due to its varied key sizes.

A Blowfish key can be brute-forced successfully in an estimated 1.7 * 1021 years, based on the key entropy found in the experiment. This enormous amount of time highlights how resistant Blowfish is to brute force assaults and confirms that it is a more secure encryption technology than AES but not as secure as 3DES.

V. DISCUSSION

In light of the reported performance characteristics, especially with regard to speed and throughput, Blowfish becomes a viable option for applications that prioritize speedy cryptographic operations, particularly when working with larger files. Notably, the consistent performance hierarchy across varying file sizes suggests that Blowfish maintains its efficiency and adaptability, making it a versatile option. These practical implications have substantial relevance for the real-world implementation of symmetric encryption algorithms: AES, 3DES, and Blowfish.

Blowfish's surprising discovery of increased memory use raises a subtle point for practical applications to take into account. This study calls into question Blowfish's use in situations where memory efficiency is crucial, despite its past application. In these kinds of situations, 3DES or AES might be considered better options because they better fit the real-world resource restrictions that are frequently present.

Security is still of the utmost importance, and the industry has recognized AES as a standard for protecting sensitive data as it is the most secure algorithm, followed by Blowfish and 3DES. The practical impact is obvious: AES is the suggested algorithm when security is a top priority. But Blowfish strikes a good mix between security strength and particular performance requirements, making it a worthy substitute that provides a sophisticated option for situations where a customized strategy is crucial.

The paper proposes Blowfish as an implementation for scenarios that prioritize high-speed cryptographic operations, especially when dealing with larger files. AES continues to be the preferred algorithm for applications where the highest level of security is required. Additionally, companies may find that AES or Blowfish are a better fit than 3DES in scenarios requiring a balance between security and memory efficiency.

Beyond the near term uses, the conversation emphasizes the necessity of continuing to monitor and modify cryptographic procedures. The practical implications discussed here give practitioners and decision-makers useful information that they may use to choose encryption algorithms that best meet the unique requirements of practical situations. These factors will become even more important as technology develops in order to preserve a safe and resilient digital environment. The flexibility of these algorithms to adjust to new technology may be the subject of future research to make sure that cryptography procedures continue to meet changing security requirements.

VI. CONCLUSIONS

This study compared the security features and performance of three well-known symmetric encryption algorithms: AES, 3DES, and Blowfish. Blowfish outperformed AES and 3DES, consistently displaying the fastest performance and throughput for both large and small files. With a balance between performance and security, AES came in second in terms of speed and throughput. 3DES performed the worst in terms of speed and throughput, which makes it less appropriate for situations where efficiency is crucial. The most memory-intensive algorithm was Blowfish, which was followed by AES. Because 3DES showed the smallest memory footprint, it was a good choice for situations with limited resources. The most secure algorithm is AES, which provides strong defense against a variety of threats.

Future research can examine or expand on this work by examining the efficacy of these algorithms in distributed systems and cloud-based environments, examining the effect of hardware acceleration on algorithm performance, assessing the viability of hybrid encryption schemes that combine symmetric and asymmetric techniques, and examining the algorithms' resistance to new threats associated with quantum computing.

REFERENCES

[1] Assa-Agyei, K., & Olajide, F. (2023). A Comparative Study of Twofish, Blowfish, and Advanced Encryption Standard for Secured Data Transmission. International Journal of Advanced Computer Science and Applications, 14(3), 393-98.

[2] Patel, K. (2019). Performance analysis of AES, DES and Blowfish cryptographic algorithms on small and large data files. International Journal of Information Technology, 11(4), 813-819.

[3] Buhari, B. A., Obiniyi, A. A., Sunday, K., & Shehu, S. (2019). Performance evaluation of symmetric data encryption algorithms: Aes and blowfish. Saudi Journal of Engineering and Technology, 4(10), 407-414.

[4] Alabdulrazzaq, H., & Alenezi, M. N. (2022). Performance evaluation of cryptographic algorithms: DES, 3DES, blowfish, twofish, and threefish. International Journal of Communication Networks and Information Security, 14(1), 51-61.

[5] Dibas, H., & Sabri, K. E. (2021, July). A comprehensive performance empirical study of the symmetric algorithms: AES, 3DES, Blowfish and Twofish. In 2021 International Conference on Information Technology (ICIT) (pp. 344-349). IEEE.

[6] Vyakaranal, S., & Kengond, S. (2018, April). Performance analysis of symmetric key cryptographic algorithms. In 2018 international conference on communication and signal processing (ICCSP) (pp. 0411-0415). IEEE.

[7] Radhi, S. M., & Ogla, R. (2023). In-Depth Assessment of Cryptographic Algorithms Namely DES, 3DES, AES, RSA, and Blowfish. Iraqi Journal of Computers, Communications, Control and Systems Engineering, 23(3), 125-138.

[8] Kureshi, R. R., & Mishra, B. K. (2022). A comparative study of data encryption techniques for data security in the IoT device. In Internet of Things and Its Applications: Select Proceedings of ICIA 2020 (pp. 451-460). Singapore: Springer Nature Singapore.

[9] Commey, D., Griffith, S., & Dzisi, J. (2020). Performance comparison of 3DES, AES, Blowfish and RSA for Dataset Classification and Encryption in Cloud Data Storage. International Journal of Computer Applications, 177(40), 17-22.

[10] Semwal, P., & Sharma, M. K. (2017, September). Comparative study of different cryptographic algorithms for data security in cloud computing. In 2017 3rd international conference on advances in computing, communication & automation (ICACCA)(Fall) (pp. 1-7). IEEE.

[11] Mota, A. V., Azam, S., Shanmugam, B., Yeo, K. C., & Kannoorpatti, K. (2017, September). Comparative analysis of different techniques of encryption for secured data transmission. In 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI) (pp. 231-237). IEEE.

[12] Mushtaq, M. F., Jamel, S., Disina, A. H., Pindar, Z. A., Shakir, N. S. A., & Deris, M. M. (2017). A survey on the cryptographic encryption algorithms. International Journal of Advanced Computer Science and Applications, 8(11).

[13] OKOLIE, S. O., & ADETOBA, B. T. (2016). Comparative Analysis of Performance Characteristics of well-known Symmetric Key Encryption Algorithms. International Journal of Scientific Research in Network Security and Communication, 4(3), 1-6.

[14] Awotunde, J. B., Ameen, A. O., Oladipo, I. D., Tomori, A. R., & Abdulraheem, M. (2016). Evaluation of four encryption algorithms for viability, reliability and performance estimation. Nigerian Journal of Technological Development, 13(2), 74-82.

[15] Rouaf, M. T., & Yousif, A. (2021, February). Performance Evaluation of Encryption Algorithms in Mobile Devices. In 2020 International Conference on

Computer, Control, Electrical, and Electronics Engineering (ICCCEEE) (pp. 1-5). IEEE.

[16] Alenezi, M. N., Alabdulrazzaq, H., & Mohammad, N. Q. (2020). Symmetric encryption algorithms: Review and evaluation study. International Journal of Communication Networks and Information Security, 12(2), 256-272.

[17] Advani, N. A., & Gonsai, A. M. (2019, March). Performance analysis of symmetric encryption algorithms for their encryption and decryption time. In 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom) (pp. 359-362). IEEE.

[18] Al Sibahee, M. A., Lu, S., Hussien, Z. A., Hussain, M. A., Mutlaq, K. A. A., & Abduljabbar, Z. A. (2017, April). The best performance evaluation of encryption algorithms to reduce power consumption in WSN. In 2017 International Conference on Computing Intelligence and Information System (CIIS) (pp. 308-312). IEEE.

[19] Yassein, M. B., Aljawarneh, S., Qawasmeh, E., Mardini, W., & Khamayseh, Y. (2017, August). Comprehensive study of symmetric key and asymmetric key encryption algorithms. In 2017 international conference on engineering and technology (ICET) (pp. 1-7). IEEE

[20] Raigoza, J., & Jituri, K. (2016, December). Evaluating performance of symmetric encryption algorithms. In 2016 international conference on computational science and computational intelligence (CSCI) (pp. 1378-1379). IEEE.

[21] Panda, M. (2016, October). Performance analysis of encryption algorithms for security. In 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES) (pp. 278-284). IEEE.