VFAST Transactions on Software Engineering
http://vfast.org/journals/index.php/VTSE@ 2023, ISSN(e): 2309-3978, ISSN(p): 2411-6246
Volume 11, Number 2, April-June 2023 *pp: 160-171*

An empirical study of performance of block cipher algorithms in cloud computing environment

Mujeeb-ur-Rehman Jamali[1], Ghulam Nabi[2], Abdul Khalique Baloch[3], Abdul Rehman Baloch[4], Aadil Jamali[1], Riaz Ahmed Shaikh[2]

[1]Institute of Mathematics and Computer Science, University of Sindh, Jamshoro Pakistan
[2]Institute of Computer Science, Shah Abdul Latif University Khairpur, Sindh Pakistan
[3]Department of Artificial Intelligence and Data Science, Istanbul Aydin University, Turkiye
[4]Faculty of Engineering and Technology, University of Sindh, Jamshoro Pakistan
Corresponding author email: mujeebjamali@usindh.edu.pk

ABSTRACT

The security of private and sensitive data stored in the public domain is a major problem. It is critical for the user that data be safe both in transit and even after it has been stored on the server. The data owner must be guaranteed that the data held on the service provider site is safeguarded against data theft from outsiders, and the data must be protected even from the service providers. The secret key generation is one of the most important factors for the security of any cryptographic system because the length of the key directly affects the performance and prevents various cryptographic attacks such as brute force attacks. At the application level, our developed system efficiently secures sensitive, private, and personally identifiable information by ensuring privacy and confidentiality of data at rest in the public domain. This study also compares the performance of block cipher algorithms DES, 3DES, Blowfish, and AES. It was deduced from the result that AES consumes less time when compared to other symmetric algorithms with small consistent behavior for various cryptographic operations with small, medium and big datasets..

KEYWORDS

Security, Cryptography, Block Algorithms, Cloud Computing

JOURNAL INFO

HISTORY: Received: May 25, 2023 Accepted: June 27, 2023 Published: June 30, 2023

1. INTRODUCTION

Privacy refers to a user's capacity to manage access to personal information and keep that information secret. Privacy in the context of data transfer and storage refers to the protection of sensitive data from unauthorized access, disclosure, or change. Symmetric algorithms can be used to safeguard privacy by encrypting data using a secret key known only to the sender and intended recipient. Data may be safely sent over unsecured networks such as the internet and saved in databases or other storage systems using symmetric encryption methods. The key is kept private and is not sent over the Internet or stored in plaintext, which aids in the prevention of assaults and unauthorized access. Overall, symmetric encryption protects privacy while also maintaining the secrecy and availability of sensitive data. A brute force attack is a type of hacking that use trial and error to crack passwords, login credentials, and encryption keys. However, the suitable encryption technique and key size must be chosen depending on the individual demands and requirements of the data being safeguarded.

The Data Encryption Standard (DES) is designed as a standard encryption method for protecting electronic communication in the 1970s by the United States National Bureau of Standards (now the National Institute of Standards and Technology). DES is based on the Feistel cypher structure and encrypted and decrypted with a 56-bit key. A 64-bit block is used in DES. Triple DES is an encryption cypher evolved from the original Data Encryption Standard (DES). DES is a symmetric-key technique based on a Feistel network. As a symmetric key cypher, it employs the same key for both encryption and decryption. The Feistel network almost perfectly duplicates each of these operations, resulting in a more efficient technique to build. However, by the late 1990s, the DES and Triple DES algorithms are deemed insufficient for providing acceptable protection against brute-force assaults. As a result, the Advanced Encryption primary (AES) is created in 1998 to replace DES as the primary encryption method. AES employs a block cypher that can work on 128-bit data blocks and key sizes of 128, 192, or 256 bits, making it substantially more secure than DES and Triple DES. Bruce Schneier created the Blowfish algorithm, which features a fiestel structure, a key size in bits ranging from 32-448, a block size of 64, and 16 processing rounds. Because to the changing key length, blowfish required extra processing time. The time-consuming sub-key creation procedure complicates brute-force attacks. It ensures long-term data

security in the absence of any known backdoor vulnerability.

2. RELATED LITERATURE

This section compares previous studies and their strategies for data protection utilizing symmetric algorithms.

In [1] the authors published an empirical analysis of several symmetric algorithms, with execution time measured in Milliseconds (MS) and memory utilization measured in Megabytes (MB). Table 1 indicates that AES (256 bits key with 128 bits block size) used less time for encryption and decryption and consumed less memory than 3DES (168 bits key with 64 bits block size), Blowfish (448 bits with 64 bits block size), and Twofish (128 bits key with 128 bits block size). It is discovered that increasing the quantity of data also affected the execution time of encryption and decryption, as well as memory utilization.

Table 1: Symmetric algorithms performance and memory utilization

AES			3DES			Blowfish			Twofish		
ENC	**DEC**	**Memory**	**ENC**	**DEC**	**Memory**	**ENC**	**DEC**	**Memory**	**ENC**	**DEC**	**Memory**
1 KB Data											
52.7	2.2	6.88	67.1	12.1	7.28	159.2	2.9	8.74	176.1	19.5	8.78
100 KB Data											
58.4	3.0	7.8	63.9	15.3	7.78	165.6	7.4	9.1	270.8	112.9	12.86
1 MB Data											
64.3	7.9	10.62	114.8	49.9	12.92	194.2	51.3	16.24	614.0	392.9	17.38
10 MB Data											
113.9	58.2	71.38	547.3	315.3	83.72	398.9	480.1	118.02	3798.2	3755.1	107.52
100 MB Data											
542.3	549.2	840.6	4734.1	2490.7	674.9	2437.8	4833.9	1100	35174.0	35949.3	1100

According to the authors of [2] cloud computing processing has grown popular for handling a large number of customers. It provides massive storage and computational capacity to clients through the Internet. One of the most serious issues is information security. Table 2 shows a complete comparison of symmetric and asymmetric algorithms based on encryption and decryption performance measures. According to the results, the AES method required the least amount of time to encrypt and decode for all file sizes, followed by 3DES and Blowfish. Furthermore, the findings show that the execution time of all three methods for encryption and decryption rises as the given file size grows.

Table 2: The comprehensive results of symmetric algorithms with different file sizes

	AES		3DES		Blowfish	
Size of Data in KB	**ENC**	**DEC**	**ENC**	**DEC**	**ENC**	**DEC**
128 KB	**2.6**	2.6	3	3.5	3.3	4.2
256 KB	**3.5**	3	4.1	4.1	4.5	5.1
512 KB	**4.2**	3.3	5.1	4.5	5.4	5.4

The author noted in [3] that utilizing different key sizes with a hybrid algorithm (i.e., DES and AES) might greatly reduce security concerns. Figure 1 depicts the encryption process of several symmetric algorithms with varying key sizes graphically using a bar chart. It was discovered that AES algorithms with 128-bit key sizes required 139, 212, 249, 298, and 354 milliseconds to conduct encryption operations on varied dataset sizes. DES took 153, 292, 389, 449, and 505 milliseconds, whereas the Blowfish algorithm took 43, 69, 98, 108, and 139 milliseconds. It is discovered that the CPU

execution time of the Blowfish method is faster than that of DES and AES for dataset outcomes. According to the results, Blowfish used less time for encryption than DES and AES.

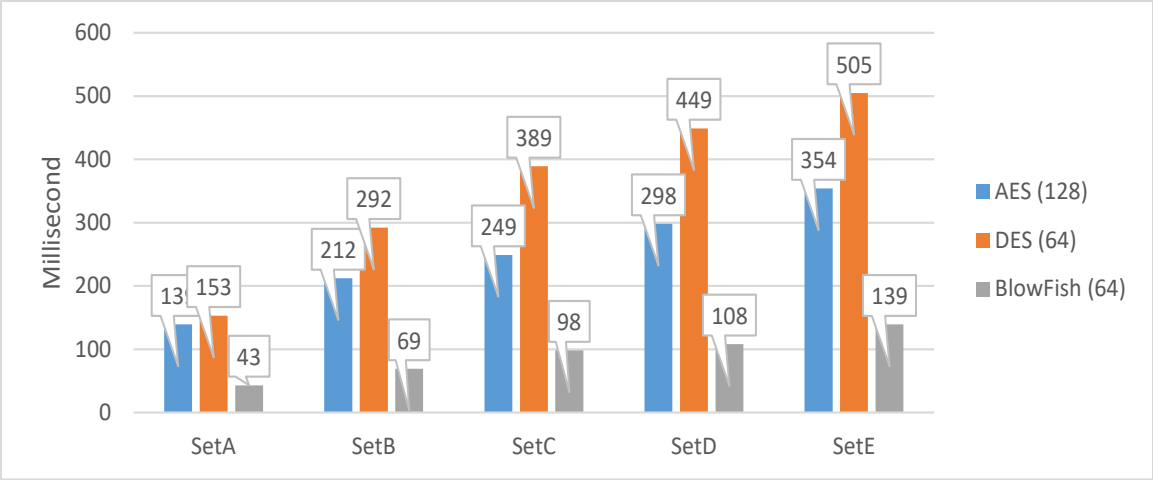

Figure 1: Symmetric algorithms encryption time

Figure 2 depicts the decryption operation of many symmetric methods with varying key sizes. It is discovered that AES algorithms with 128-bit key sizes conducted decryption operations on various dataset sizes in 97, 153, 190, 246, and 286 milliseconds. DES took 119, 204, 284, 303, and 362 milliseconds, whereas the Blowfish algorithm took 31, 46, 53, 73, and 87 milliseconds. It is discovered that the CPU execution time of the Blowfish method is faster than that of DES and AES for dataset outcomes. According to the results, Blowfish required less time for decryption than DES and AES.

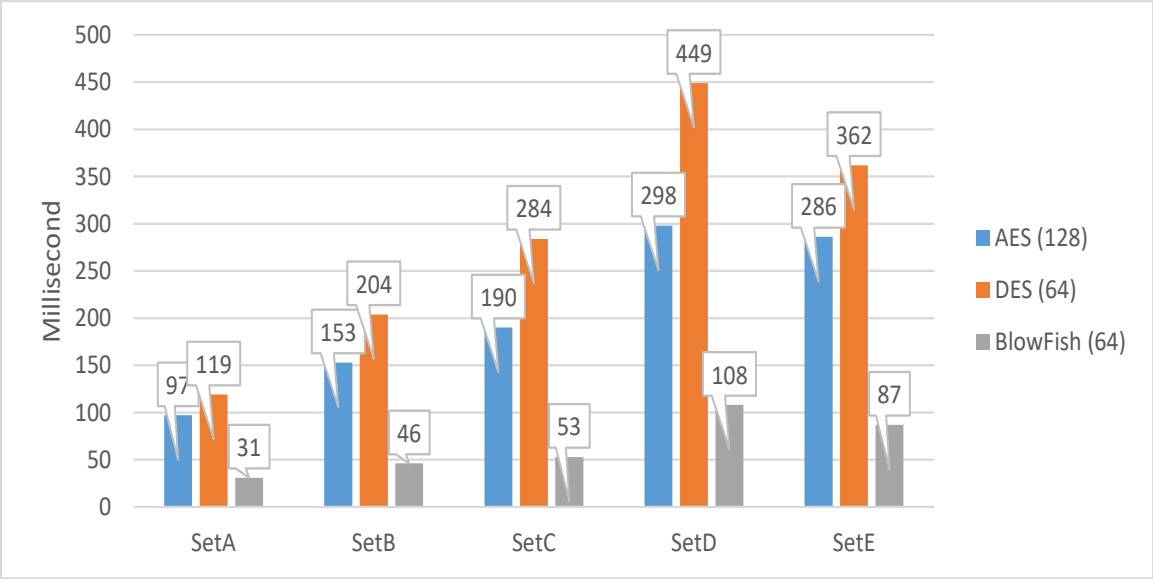

Figure 2: Symmetric algorithms decryption time

In [4] authors made a comparison of encryption and decryption execution times in milliseconds of the AES with 256 bits, Triple DES with 168 bits and Blowfish with 448 bits for various data file sizes which is shown in Table 3.

Table 3: Encryption vs decryption execution time in MS

Algorithms	1K	100K	1M	10M	100M
AES (ENC)	52.76	58.35	64.30	113.93	542.29
AES (DEC)	2.18	3.00	7.88	58.18	549.16
3DES (ENC)	67.12	63.92	114.78	547.33	4734.13
3DES (DEC)	2.89	7.42	51.33	480.12	4833.94
Blowfish (ENC)	159.19	165.63	194.21	398.89	2437.84
Blowfish (DEC)	12.12	15.25	49.99	315.29	2490.69

In [5] empirical study is carried out using secret key algorithms such as AES, DES, and Blowfish. The execution performance of the following algorithms is compared based on CPU time. Small and huge dataset of files used for execution time and space trade-off which is shown in Table 4. According to the authors' conclusions, based on actual results of time limitations, AES took 8% less time / slower than DES and 7% less time / slower than Blowfish. As a result, for small dataset encryption, DES was quicker than both techniques. Meanwhile, for big datasets, three techniques required almost the same time to encrypt and decrypt. In terms of memory constraints, it is discovered that DES is 1.01% slower than Blowfish but used 0.63% less memory to execute the AES algorithm. AES took 2.48% more memory than Blowfish for big datasets, and 1.34% more RAM than DES. As a result, Blowfish outperformed other symmetric algorithms. With time limits, DES may be a better alternative than Blowfish and AES for the use of algorithms, financial applications such as capital markets and e-commerce, and so on. Smartphones and tablets have limited storage space. Blowfish may outperform DES and AES. The authors suggested that future study focus on multimedia contents, such as photos, audio, and video, as well as CPU execution time and memory constraints, because the majority of application requirements are to store multimedia contents.

Table 4: Symmetric Algorithms Time and Space Trade-Off in MS

Algorithms	Small Dataset	Huge Dataset
AES (ENC)	198.5	167.00
AES (DEC)	1,263.030	1,309.141
DES (ENC)	184.4	465.36
DES (DEC)	1,271.082	1,291.483
Blowfish (ENC)	185.1	459.22
Blowfish (DEC)	1,258.205	1,276,567

The security weaknesses outlined in [6] that well-known secret key methods experimental results demonstrate that the DES algorithm's efficiency/performance is higher than AES, as indicated by the author. Meanwhile, Blowfish performed well in terms of execution speed on short text data files (less than 1000 bytes in size). The results reveal that the performance of symmetric algorithms, such as AES, DES, and Blowfish is nearly comparable on big text data files (size more than 10,000 bytes). The author concluded that the testing results clearly suggest that Blowfish outperforms AES in terms of encryption performance. Meanwhile, DES is superior for memory-limited programmes that relied on memory requirements during execution. Blowfish can be used to guarantee adequate/strong data security.

In [7] symmetric algorithms are compared and analyzed. Performance assessment of the DES, 3DES, AES, and Blowfish algorithms in Java. The symmetric methods are used to create a block cypher with various block sizes and key sizes. The authors determined that when the block size is big, the result is faster since it takes less time to encrypt data.

Meanwhile, if the block size is short, additional execution time is required. The authors determined that Blowfish took the least amount of time and is the quickest of the secret key algorithms, but 3DES required more time for encryption and decryption and is a slower approach.

The authors [8] compare and contrast secret key encryption and decryption methods (AES, DES, 3DES, RC2, Blowfish, and RC6). Each method has distinct components, different plaintext sizes, and different kinds with varied key sizes. The computing resources needed to conduct encryption and decryption. According to the author's findings, Blowfish took less time than RC6. DES took longer to complete and is slower than the DES algorithm. The authors stated that AES outperformed alternative symmetric algorithms such as DES, 3DES, and RC2 in terms of performance.

In [9] performance of well-known symmetric algorithms evaluated and different data block sizes are utilized for encryption and decryption to record the performance of the algorithms on various systems utilizing various software and hardware. C++ is used to build the system. It is discovered that Blowfish outperformed other symmetric algorithms with a strong and maximum key size of 448. With a huge data set, AES required additional computing time. Because 3DES consumed more time, DES is quicker.

The authors of [10] compared the secret key algorithms DES, AES, and Blowfish. The performance is recorded. The authors discovered that Blowfish had the fastest-varying block size. It is discovered that AES performed poorly when compared to other symmetric algorithms. Symmetric algorithms are quicker than asymmetric algorithms, but they are susceptible since the secret key is shared. The authors also determined that the strength of asymmetric algorithms is that they employ two separate keys, although they take more processing time than symmetric algorithms [15].

The author developed a method for concealing data in [11]. The suggested approach concealed a data file in two steps by combining audio steganography and cryptography. Before concealing in the audio file, the symmetric algorithm Blowfish is employed for encryption. Steganography is a technique for concealing and hiding data in audio files. After embedding, there is no need to modify the file format or size of the data file, which is fantastic for data protection from both passive and active attackers. The same strategy will work for data files of any size.

The authors of [12] presented an efficient solution for securing data from an intruder using cryptography and steganography techniques. The use of steganography techniques conceals data in an audio recording. Meanwhile, the Data Encryption Algorithm is utilized to convert data from plaintext to ciphertext. The audio stated that the size of the audio file should be the same after the embedded in the file and that the same technique should be applicable to all types of audio file formats.

In [13][14] autors compared symmetric and asymmetric algorithms i.e., AES, DES, Blowfish, and RSA. The system is built in Java to keep data safe in the cloud. AES is discovered to need less calculation time for execution. Meanwhile, Blowfish required the least amount of RAM. The DES also took less time to encrypt. The asymmetric methods, such as RSA for encryption with large memory sizes, took the most time.

To differentiate the proposed research from other systems, the many authors presented their work on vulnerabilities of security concerns employing encryption data at rest with symmetric key cryptography, where the same secret key is used for encryption and decryption. The proposed work takes a different strategy than previous researchers have. Instead of data at rest encryption, application-level security is advocated in this study. The application-level allows encryption of sensitive data as well as total data control. It is discovered through literature that existing systems are from a time when there are improved security challenges that must be handled using enhanced solutions.

3. METHODOLOGY

This study made use of empirical approaches and concepts. Issues have been investigated and resolved utilizing quantitative data obtained. Adoption of modern symmetric cryptography, which is the most important technique for protecting sensitive and private data in the suggested approach. The proposed research design acts as a framework for organizing the whole research endeavor. It provides the theoretical foundation for the inquiry. It is critical because it links the entire study project together. Its primary objective is to facilitate the collection of pertinent data. Primary and secondary sources were used to acquire data. The experiment directly collects primary data in empirical study, which is the first-ever data collecting in quantitative format. Secondary data is information obtained or acquired by other researchers that makes data found in magazines and other published sources available. A statistical analysis tool that is used to assess the performance of several algorithms and discover their strengths and faults. The study findings are determined by data analysis and presentation utilizing the established approach. A tabular format is also utilized to present the frequency's summarized mean, percentage, standard deviation, and percentile, which will be addressed in further depth in the results and comments sections. The Secure Socket Layer (SSL) provides transaction-level security. SSL ensures the privacy, integrity, and authenticity of data in transit from beginning to end. This provides transaction level security on both the transmitting and receiving sides. The system is created and used for data synthesis. Each process is repeated at least ten times to ensure proper data collection. Following that, the outcome synthesis is analyzed. According to the findings of the study, the choice of algorithm is determined by the unique use case, with different algorithms performing better under different

situations. The key length in bits is incorporated in the symmetric algorithm metric to offer a comparison value of the synthesis outcomes. Our designed system employs four (04) secret-key algorithms in which the same key is utilized to execute plaintext encryption and decryption operations to return ciphertext.

The approach of the system as represented by the UML Class diagram the characteristics operations and parameters of the operations for the creation secret keys, encryption of the plaintext, and decoding of the ciphertext are shown in Figure 3.

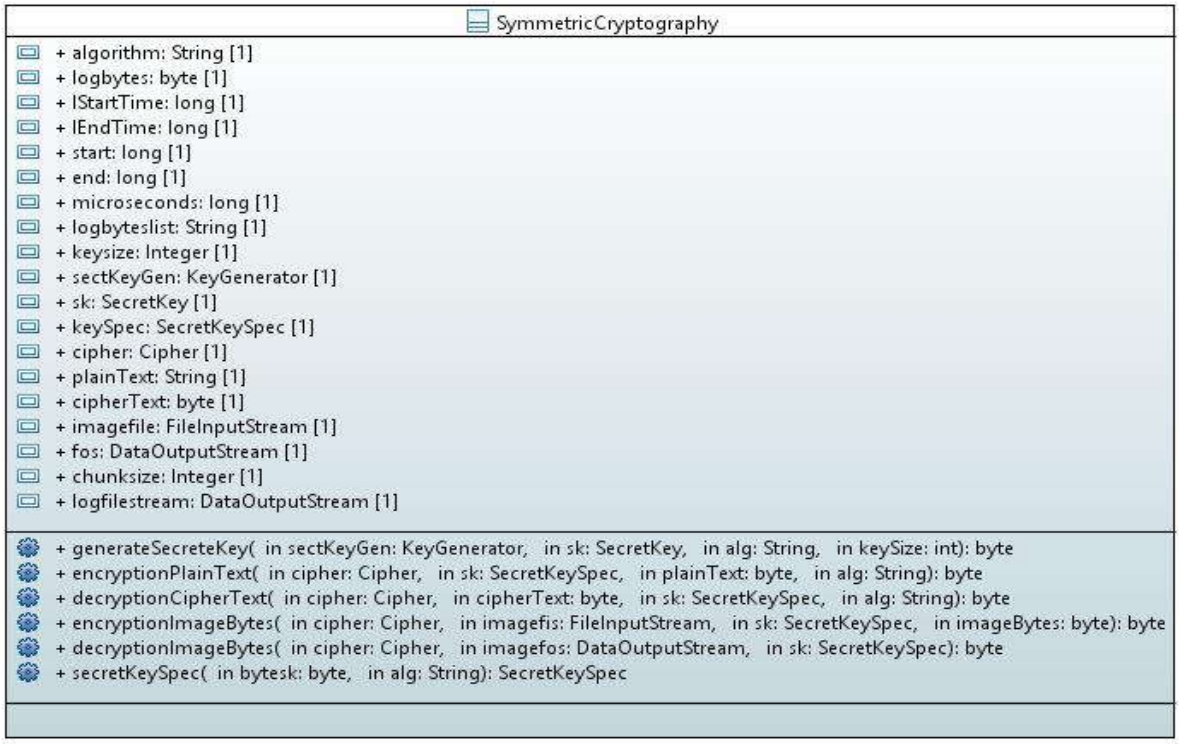

Figure 3: Symmetric Cryptography attributes and operations

The following is the generation of key of the Blowfish algorithm in size of bits and key encoding in Raw. The original content/data presented below is recovered using the same secret key:

Blowfish (32) bits Key	-21 -63 -53 -98
Blowfish (128) bits Key	-4 71 4 91 54 82 66 -24 -120 86 54 -65 126 -80 -42 77
Blowfish (192) bits Key	72 -105 14 -46 -57 -95 17 -77 -104 -30 20 -4 -71 -16 -119 21 56 -1 -56 109 16 -114 61 -31
Blowfish (256) bits Key	7 30 90 77 127 69 -88 41 117 68 87 -104 -23 37 -102 -12 95 -54 -78 -74 0 40 -66 -12 30 104 32 42 102 -124 -113 113
Blowfish (448) bits Key	30 0 -119 -33 -106 -81 -29 65 -53 -43 1 -11 -50 -5 111 71 -95 100 -19 -109 89 126 -111 100 -27 50 -118 41 -15 35 -52 -46 -107 70 -108 -126 -96 71 -20 102 53 -32 -13 97 65 19 12 -65 -76 25 53 -93 1 36 -94 11

The length of the Blowfish secret key generation, which can be from 32, 128, 192, 256, 448 bits long, increases the level of protection against brute force attacks. Blowfish generated 280 bytes ciphertext from 276 bytes plaintext, which is presented as a Base64 encoded encrypted field in a document database:

{

_id : ObjectId("482b31385e0f4c0fcc98cc5e")

UserId : 3310,

Name : Binary ('
6noNJuKH4vM14/f+MgyQNuutKL9bX9nxsxcM+ZZXI
BOu8Mo1IY52FwiJGbANYkY8JzsvtX/pVwHI

VN9idkhVTAzH9gpPaHIRSlYrzGeHPLKUEwrcZajxPr
JMxzFg2PsdFP01ReaG90PJwGIkB8+xLKzU

QBOVkv5t4XcOoO4cvOxabrvheeYMyZH4ENxuvsSE
mmwRFn4UvKxvxPzQvuvQV4b5tKOQyop5a/3U

qhD9/WX9L0tV8Nin6Op6DSbih+LzNeP3/jIMkDbrrSi/
W1/Z8bMXDPmWVyATrvDKNSGOdhcIiRmw

DWJGPCc7L7V/6VcByFTfYnZIVUwMx/YKT2hyEUp
WK8xnhzyylBMK3GWo8T6h6hdQ2g6kVA== ')

}

The following is the generation of key of the AES algorithm in size of bits and key encoding in Raw. The original content/data presented below is recovered using the same secret key:

AES (128) bits Key	12 96 -21 -38 55 70 -125 30 85 -99 -90 39 -11 -8 107 -102
AES (192) bits Key	-76 -112 -126 7 -6 123 -8 40 -44 37 -112 27 112 44 -111 -127 31 -53 -42 107 -84 9 -63 93
AES (256) bits Key	110 -109 -26 73 91 38 -28 -83 -13 -103 -29 55 -81 40 -102 -34 0 41 111 1102-84 -28 101 -66 54 -56 75 -64 -83 -4 -87

The increase in the size of AES secret key generation, from 128, 192, and 256 in bits, gives protection against brute force attacks. The AES encryption method converts data into random bits, which are known as ciphertext, with various key size in bits. AES generated 288 bytes ciphertext from 276 bytes plaintext, which is presented as a Base64 encoded encrypted field in a document database:

{

_id : ObjectId("8cc5e0fc482b3138c95e0f4c")

UserId : 2467,

Name : Binary
(9w3uA2GOyK7MfTyoBJZCFA3HuPgpDyuG23c129H
TgNpyaloo94ce7CimTujjN3ls3qBBk7KRH1Yt

U9w+H4EwLvJf9e2wjRqG00W6LQwwy+KzBO7UiuH
D5V8teqjzk9svnW+1d9JytEC2AVWH+Yq7ZMDXZt+h
BNiFPtljrk7OV08yVEZ8JjFY/gUIkDhaQzTrSKJlKeO5
0t1eYiUV/eOXolNWvtKrFosiZuIL

z0xAXIICkn7gOTVaXn4U2Lij1oGkQOGiWxQ4rifGHL
NEDrCl9Ka0bufTCG0pbPXz89xbdfFi4+Ii

cK1L31Md2Y9YN0UB4WuZyLtXIT5BE+/3Pd/WIyQSl
oTYdqyLU0wakq2SkowwIxoucyAIYMvFXusX

b71A')

}

The DES (56 bit) secret key is generated which is used to retrieve the original data:

DES (56) bits Key	62 13 -42 21 124 -88 -53 -101

To convert data to ciphertext length, use the DES encryption process. Using the same key, the decryption procedure is used to retrieve the original data. The data is BASE64 encrypted as reproduced below DES 56-bit long key generated 280 bytes ciphertext from original plaintext of 276 bytes that is in the document database.

{

_id : ObjectId("3138c5e0f8cc5e4c0fc482b9")

UserId : 4496,

Name : Binary
('liMYw/p51L5vvCIsJCYfm7tAp9zHfLexOlluNTLnVw
KTnpzI6aj/kRDM8kY/szQkZg6I6KJYeyXq

V5CjaknxUbGGBJ9AuvCS+yvBjzgbqSFg+ySsvzopZb9
w1JTGXNxptueWrL/5iQIf96RTTMf1NQKQ

QZIbQ3RUHYPyr9VKq/gq7PPcE7BQJVBvOrvI0OKG
7WEyyT6C/sFYru29gxZ1m+sE+eZUnNwsrIXz

Xe9M43v61SgC5rwbopYjGMP6edS+b7wiLCQmH5u7
QKfcx3y3sTpZbjUy51cCk56cyOmo/5EQzPJG')

}

The generated Triple DES 168-bit secret key (Key Encoding as Raw) which is used retrieve the original data is shown below.:

Triple DES (168) bits Key	-118 98 -32 -89-42 88 -105 -70 97 127 91 -99 109 107 59-128 1 37 2138 -105 -42 -80 94

The increase in the size of the Triple DES secret key generation to 168 bits from DES's 56-bits gives higher security against brute force attacks than DES.

Triple DES 168-bit key generated 280 bytes ciphertext from plaintext of 276 bytes, which is shown as a Base64 encoded encrypted field in a document database.

{

_id : ObjectId("98cc5ec482b5e0f4c0f 3138c")

UserId : 1535,

Name : Binary
('QZIbQ3RUHYPyr9VKq/gq7PPcE7BQJVBvliMYw/p5

1L5vvCIsJCYfm7tAp9zHfLexOlluNTLnVwKTnpzI6aj/kRDM8kY/szQkZg6I6KJYeyXqsvzopZb9w1JTGXNxptueWrL/5iQIf96RTTMf1NQKQOrvI0OKG7WEyyT6C/sFYru29gxZ1m+sE+eZUnNwsrIXzXe9M43v61SgC5rwbopYjGMP6edS+b7wiLCQmH5u7QKfcx3y3sTpZbjUy51cCk56cyOmo/5EQzPJGV5CjaknxUbGGBJ9AuvCS+yvBjzgbqSFg+yS')

}

4. HARDWARE AND SOFTWARE REQUIREMENT

For the experiments carry out the system are used i.e., Windows 10, running on Processor Intel (Core) M-5 Y 10c (4 CPUs) 1 GHz, RAM 4096 MB, 500 GB and JDK with cryptographic libraries (JCA and JCC). The system's prototype has been completed. Small, medium, and large datasets are the three primary classifications. A symmetric algorithm is utilized for key pair creation, encryption, and decryption. Our technology derived quantitative results from empirical data. A statistically sound examination of the outcomes that are founded is offered in detail in the results and discussion sections.

5. RESULTS AND DISCUSSIONS

Empirical comparative analysis of symmetric algorithms and evaluating the performance of several symmetric encryption operations under diverse settings. When comparing symmetric algorithms, some significant elements to consider are speed and security from prevention brute force attack. To assess the performance of the algorithms under various metrics/scenarios, such as plaintext widths and key lengths in bits. There is a comparison of maximum size symmetric secret key generation, encryption and decryption of plaintext with small, medium, and big datasets. The performance of secret key generation of various block cipher algorithms i.e., 31.4, 27.6, 31.7, and 32.0 respectively (mentioned in Table 4 with maximum key size in bits) are presented. AES takes less time to generate secret keys than alternative symmetric algorithms with less consistent behaviour.

Table 4: Comparative Analysis of Block Cipher Algorithms (Source: primary data)

	Blowfish			AES			DES			Triple DES		
	Secret Key GEN	ENCRYPTION	DECRYPTION	Secret Key GEN	ENCRYPTION	DECRYPTION	Secret Key GEN	ENCRYPTION	DECRYPTION	Secret Key GEN	ENCRYPTION	DECRYPTION
	Dataset (Small) Encryption and Decryption											
Average	31.4	11	2.6	27.6	9.4	2.8	31.7	11	1.7	32.0	11.2	2.1
Dispersion	1.799	1	0.823	1.252	3.062	0.316	1.135	1.155	0.422	2.885	1.16	0.919
	Dataset (Medium) Encryption and Decryption											
Average	-	30	5	-	70	4	-	32	4	-	34	8
Dispersion	-	3	2	-	3	1	-	3	1	-	3	2
	Dataset (Large) Encryption and Decryption											
Average	-	29	6	-	71	4	-	33	5	-	38	6
Dispersion	-	2	1	-	4	2	-	3	1	-	2	2

Figure 4 depicts the cooperative research for symmetric secret key generation of Blowfish, AES, DES, and Triple DES at the application level. AES is discovered to be 13.240% quicker than Blowfish, 14.29% faster than DES, and 15.33% faster than Triple DES. Concluding, AES is found to be the most effective method for generating secret keys.

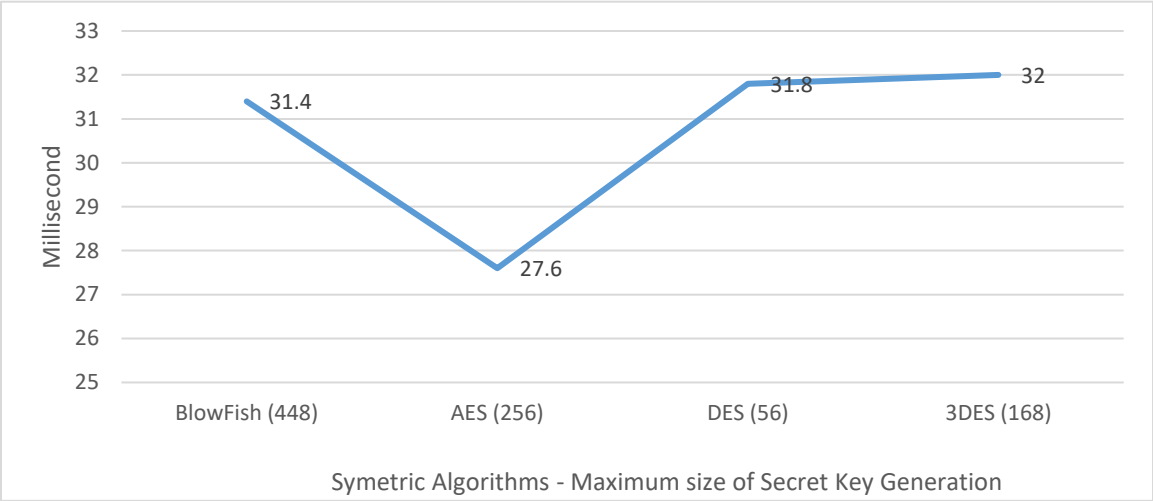

Figure 4: Comparative analysis of secret key generation

Figure 5 depicts a comparative results of encryption operation of symmetric algorithms. It is found that duration of time in a millisecond is 11, 9.4, 11 and 11.2 respectively for the Blowfish, AES, DES, and Triple DES with maximum key size in bits. The acquired results for the decryption operation are 1.9, 2.8, 1.7, and 2.1 ms. In terms of encryption, AES is found to be 14.285% quicker than Blowfish and DES. AES outperforms Triple DES by 17.14%. Meanwhile, for decryption, DES is shown to be 61.11% quicker than AES, 22.22% faster than Triple DES, and 11.11% faster than Blowfish. It may be deduced from the results that AES's encryption procedure took less time than other symmetric algorithms with less consistent behavior. Meanwhile, DES outperformed other symmetric algorithms with minimal consistent deviations from the mean.

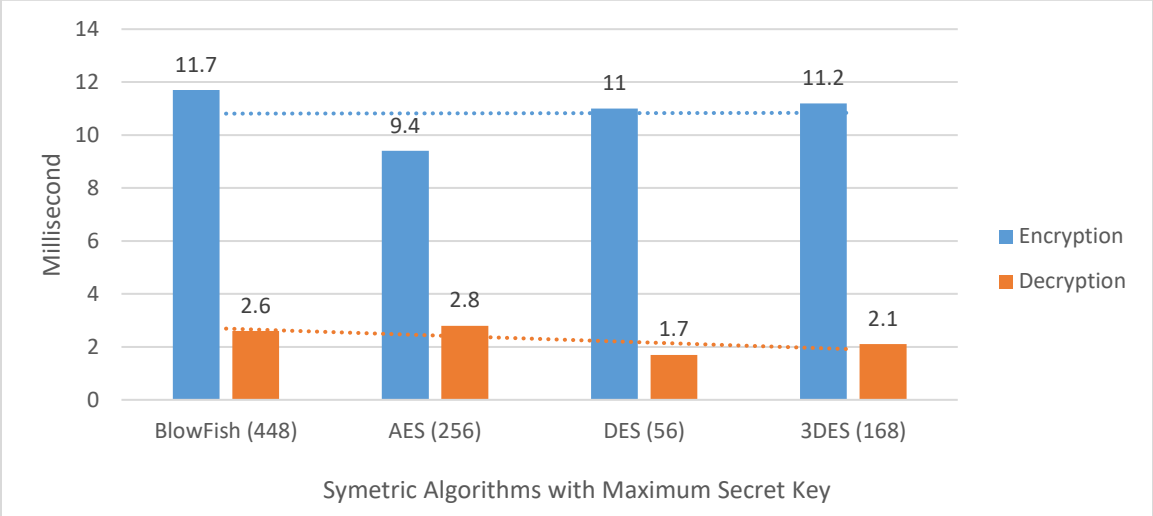

Figure 5: Comparative Analysis of small plaintext encryption and decryption

In Figure 6, it is found that duration of time in a millisecond is 30, 70, 32 and 34 respectively for the Blowfish, AES, DES, and Triple DES with maximum key size in bits for encryption operation. The experimental results for decryption operation are 5, 4, 4 and 8. Blowfish is shown to be 6.452% quicker than DES, 12.9% faster than Triple DES, and 129% faster than AES for encryption. Meanwhile, it is discovered that AES with a bigger key size and DES with a small key size are both 20% quicker than Triple DES and 80% faster than Blowfish in terms of time and scatter from mean. It may be deduced from the results that Blowfish required less time for encryption operations of AES and DES than other symmetric algorithms with modest consistent behavior.

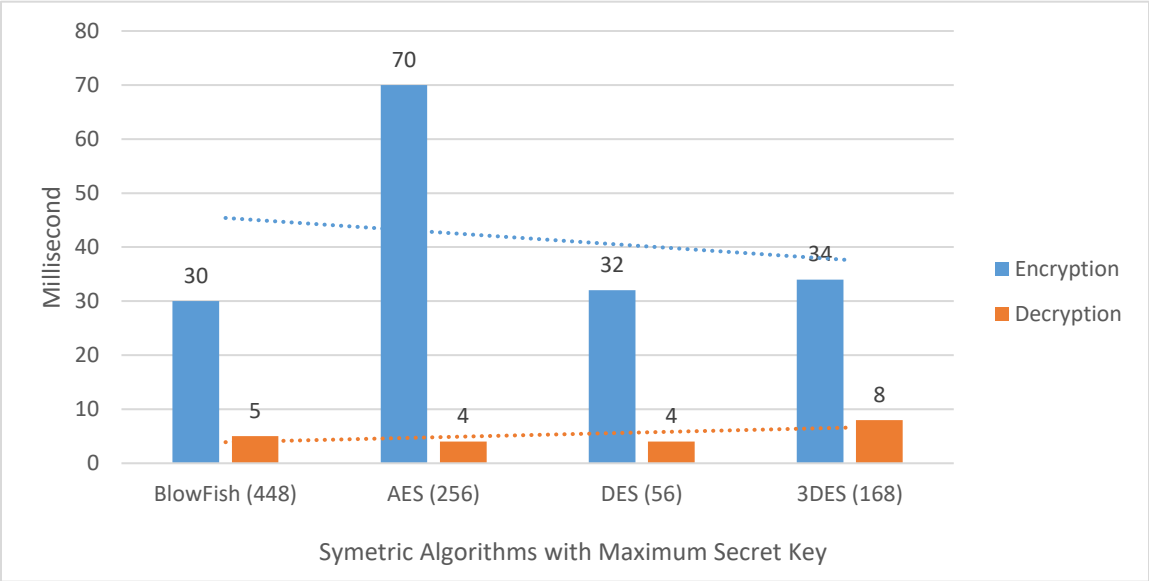

Figure 6: Comparative Analysis of medium size plaintext encryption and decryption

It is found that duration of time in a millisecond is 29, 71, 33 and 38 respectively for the Blowfish, AES, DES, and Triple DES with maximum key size in bits. The experimental results for decryption operation are 6, 4, 5, and 6 (Figure 7). Blowfish is shown to be 13.33% quicker than DES, 30% faster than Triple DES, and 140% faster than AES for encryption. Meanwhile, it is discovered that AES decryption is 20% quicker than DES and 40% faster than Triple DES and Blowfish. It is clear from the results that AES takes less time to encrypt and decode than alternative symmetric algorithms with less consistent behavior.

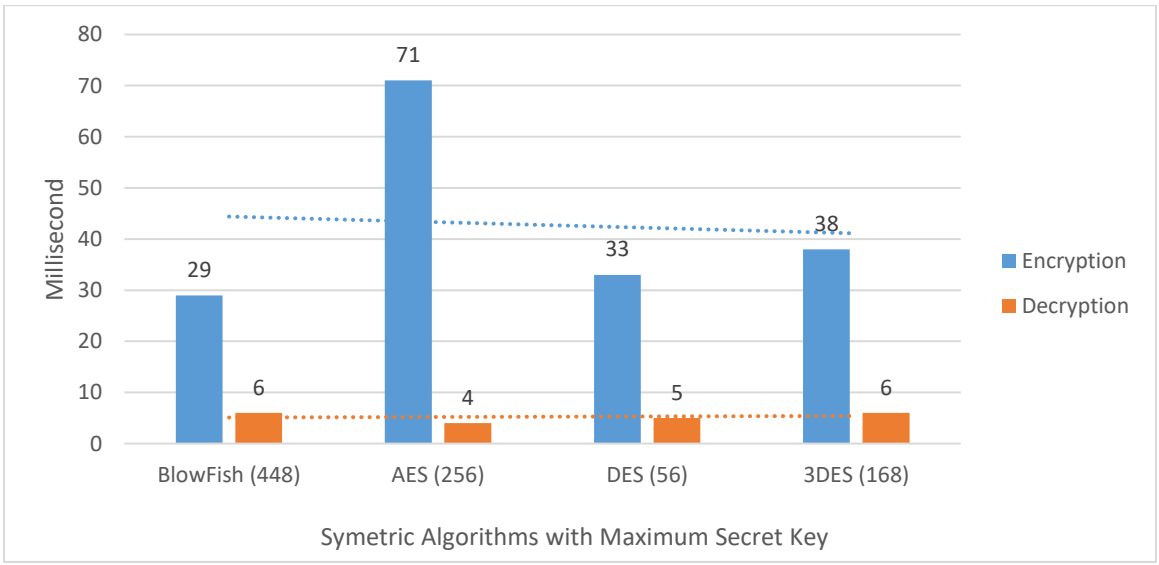

Figure 7: Comparative Analysis of large size of plaintext encryption and decryption

THE MEMORY AND STORAGE COMPARISON OF THE BLOCK CIPHER ALGORITHMS

To accomplish the encryption operations and change the plaintext data of various sizes, symmetric algorithms (DES, 3DES, AES, and Blowfish) with minimum and maximum key sizes are utilized. Table 5 compares memory space and storage of ciphertext generated by symmetric algorithms on small, medium, and big datasets. When compared to other symmetric algorithms, AES used more space for ciphertext (32, 288 and 544 bytes) than plaintext (24, 276 and 529 bytes) respectively.

Table 5: Empirical Comparison of memory and storage space utilization (Source: primary data)

Bytes/KB/MB	DES	3DES	AES	BLOWFISH
23 bytes	24	24	32	24
276 bytes	280	280	288	280
529 bytes	536	536	544	536

In literature, researchers explored a variety of experimental environments and system configurations. They employed various dataset sizes and reported their findings in the 50% or 2nd quartile (Q2) average/median. Thus, outcomes cited in the literature are not given real results but analysis of data, i.e., average/median were considered. Table 6 compares the percentiles of symmetric methods for secret key creation, encryption, and decryption with different dataset sizes.

Table 6: Comparative Percentile of Symmetric algorithms

	Blowfish (448)			AES (256)			DES (56)			3DES (168)		
	GEN	**ENC**	**DEC**	**GEN**	**ENC**	**DEC**	**GEN**	**ENC**	**DEC**	**GEN**	**ENC**	**DEC**
	Plaintext Small Dataset											
Q1	30.75	12	2	27.75	9	3	31.75	11	1.75	31	11.75	1.75
Q2	32.5	12	2.5	28.5	9	3	33	12	2	32	12	2
Q3	34.25	13.25	3.25	30	9.25	3	34	12.5	2	32.75	12	2
	Plaintext Medium Dataset											
Q1	-	29.75	4	-	69.75	4	-	31.5	4	-	32.75	6.75
Q2	-	31	6	-	70.5	5.5	-	33	4	-	34.5	7.5
Q3	-	32.25	8	-	72.5	6	-	34.25	5.25	-	35.5	6
	Plaintext Large Dataset											
Q1	-	28.75	6.5	-	70.5	4	-	30.75	5	-	36.75	6
Q2	-	30.5	7.5	-	71	5	-	33.5	5.5	-	37.25	7
Q3	-	31.25	8.25	-	72.25	6.5	-	36	7	-	38.75	7.75

6. CONCLUSION

The empirical comparative study of symmetric algorithms, specifically time and space complexity is performed. It gives useful insights into the performance, security, and implementation aspects of numerous frequently used encryption methods. Data privacy and confidentiality, comparison of synthesis of encryption and decryption results with maximal key size of symmetric algorithms (Blowfish, AES, DES, and Triple DES) are analyzed. AES required shorter time to encrypt a tiny dataset than other methods. Meanwhile, DES required less time to decrypt. The findings for a medium dataset demonstrated that AES with the maximum key size is time efficient. According to the results, Blowfish required less time for encryption operations than other symmetric algorithms with minimal consistent behavior and dispersion of value from mean. Meanwhile, AES and DES required the same amount of time to decode with less consistent behavior and value dispersion from the mean than other symmetric methods. The findings for big datasets indicated that Blowfish with the maximum key size is time efficient. According to the results, Blowfish required less time for encryption operations than other symmetric algorithms with minimal consistent behavior and scatter of value from mean. Meanwhile, AES is found to be the most efficient decryption method with the least consistent behavior when compared to other symmetric algorithms. Asymmetric cryptography is slower than block cypher techniques. The study's goal is to avoid data leaks and expose unauthorized users. It is also hoped to lessen the difficulty in resolving the unsolvable problem of confidentiality and privacy of private, sensitive, and personally identifiable information. As well as security is resolved. These are the key challenges encountered when data is stored in distant and public domains. This work concludes that assessing the security of algorithms by analyzing their resistance to various sorts of assaults, including as brute-force attacks, side-channel attacks, and chosen-plaintext attacks, may be averted by using a proper key size.

In terms of future prospects, the advent of quantum computing has posed a possible danger to symmetric encryption methods, as they may be subject to quantum computer assaults. As a result, researchers are creating and testing post-quantum symmetric encryption methods that are resistant to assaults by quantum computers.

CREDIT AUTHOR STATEMENT

Dr. Mujeeb-ur-Rehman Jamali: Research Experiments, Implementation and Data Curation, **Ghulam Nabi:** Conceptualization and Methodology, **Abdul Khalique Baloch**: Writing Original Draft, **Abdul Rehman Baloch:** Reviewing and Editing, **Aadil Jamali**: Acquisition of data, Analysis and Interpretation and **Riaz Ahmed Shaikh**: Evaluation and Results Comparison.

COMPLIANCE WITH ETHICAL STANDARDS

It is stated that all authors don't have any conflict of interest. Furthermore, informed consent was obtained from all individual participants included in the research work.

REFERENCES:

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

[2]. K. Ali, F. Akhtar, S. A. Memon, A. Shakeel, A. Ali, and A. Raheem, "Performance of Cryptographic Algorithms based on Time Complexity," in 2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), IEEE, 2020, pp. 1-5.

[3]. S. Tamane, "Non-Relational Databases in Big Data," ACM, 2016.

[4]. H. Dibas and K. E. Sabri, "A comprehensive performance empirical study of the symmetric algorithms: AES, 3DES, Blowfish and Twofish," in 2021 International Conference on Information Technology (ICIT), IEEE, 2021, pp. 344-348.

[5]. K. Patel, "Performance analysis of AES, DES and Blowfish cryptographic algorithms on small and large data files," Springer Int. j. inf. Tecnol., 2019. [Online]. Available: https://doi.org/10.1007/s41870-018-0271-4.

[6]. O. C. Abikoye, K. S. Adewole, and A. J. Oladipupo, "Efficient data hiding system using cryptography and steganography," Int J Appl Inf Syst, vol. 4, no. 11, pp. 6-12, 2012.

[7]. Z. Hercigonja, "Comparative Analysis of Cryptographic Algorithms," International Journal of Digital Technology & Economy, vol. 1, pp. 127-134, 2016.

[8]. M. Streedevi, "Threshold Sr2n Public Key Cryptography," International Journal of Engineering Trends and Technology (IJETT), vol. 31, no. 1, pp. 15-17, 2016.

[9]. S. Gupta and S. Vashisht, "Implementation of ECC Using Socket Programming In Java," IOSR Journal of Computer Engineering (IOSR-JCE), vol. 16, no. 4, pp. 65-68, 2014. [Online]. Available: www.iosrjournals.org.

[10]. D. S. A. Elminaam, H. M. A. Kader, and M. M. Hadhoud, "Performance evaluation of symmetric," 2008.

[11]. P. G. and V. PS, "Audio steganography and cryptography: using LSB algorithm at 4th and 5th LSB layers," Int J Eng Innov Technol, vol. 2, no. 4, pp. 177-181, 2012.

[12]. S. VK, P. Neelima, P. Sruthi, D. P. Sai, and B. Manasa, "Implementation of Blowfish algorithm for efficient data hiding in audio," Int J Comput Sci Inf Technol, vol. 5, no. 1, pp. 748-750, 2014.

[13]. M. U. R. Jamali, A. G. Memon, N. A. Kanasro, and M. U. R. Maree, "Data integrity issues and challenges in next generation non-relational document-oriented database outsourced in public cloud," International Journal of Emerging Trends in Engineering Research, vol. 9, no. 4, pp. 416-420, 2021. [Online]. Available: https://doi.org/10.30534/ijeter/2021/13942021.

[14]. V. Abramova and J. Bernardino, "NoSQL Databases: MongoDB vs Cassandra," in Proceedings of the International Conference on Computer Science and Software Engineering, ACM, 2013.

[15]. M. R. Jamali, A. G. Memon, and M. R. Maree, "Security issues in data at rest in a non-relational Document Database," SindhUniv. Res. Jour. (Sci. Ser.), vol. 52, no. 03, 2020. [Online]. Available: http://doi.org/10.26692/sujo/2020.09.41.