The Scientific Temper (2023) Vol. 14 (3): 902-908

E-ISSN: 2231-6396, ISSN: 0976-8653

Doi: 10.58414/SCIENTIFICTEMPER.2023.14.3.53

https://scientifictemper.com/

RESEARCH ARTICLE

ECM: Enhanced confidentiality method to ensure the secure migration of data in VM to cloud environment

Raja Selvaraj*, Manikandasaran S. Sundaram

Abstract

Cloud is the technology behind all modern IT paradigms today. All kinds of applications run in the cloud environment and host their data. Enterprises show interest in migrating their data and servers to the cloud to benefit the cloud. The cloud is an open distributed network environment; hence, it is vulnerable to data security attacks. Migration in cloud computing permits the handover of resources, for example, virtual machine (VM) and data from off-premises to on-premises. During the migration, the data needs to be secured. This paper proposes an enhanced confidentiality technique (ECM) to ensure data security when migrated to the cloud. The proposed enhancement is on the advanced encryption standard (AES) to strengthen the protection level of the AES. The standard AES runs for rounds with four unique stages of data processing: Substitute bytes, add round key, shift rows and mix columns. The proposed ECM enhances the substitute bytes stage with dynamic substitution boxes for each round of the AES. The proposed ECM is evaluated for its efficiency by the avalanche effect. The result of the ECM improves data protection when the data is migrated to the cloud.

Keywords: Data migration, Data confidentiality, Authenticity, Data integrity, Cloud security, Hashing.

Introduction

A promising paradigm for computing called "cloud computing" makes computing resources available as online services. This new profitable model includes subscription-based or pay-per-use services provided over the Internet as an attractive, significant, large-scale venture (Arockiam L. *et al.,* 2017). It is the availability of virtualized IT services and IT resources on demand. Salesforce, Amazon, and Google offer these services, and customers are charged according to demand. Due to its ability to provide corporate environments with data storage, cloud computing has gained popularity in IT. Due to these benefits, enterprises are migrating their data to the cloud. Migration transfers a virtual machine (VM) and its data from off-premises to on-premises. VM migration delivers cloud job balancing and system maintenance topographies (Raja S. *et al.,* 2021). Cloud service providers stock VM disk images in scrambled form while at rest to prevent attacks. However, there is still an unresolved issue of ensuring that data is moved securely to the cloud and between clouds (VMs) and off-premises to on-premises. The major block to cloud implementation is a lack of security because users keep their sensitive data on clouds, which are public domains. The primary drawback of cloud storage is insecure storage (Arockiam L. *et al.,* 2013). Therefore, it is essential to create enough security. VM migration is mainly used for providing high availability, hardware maintenance, workload balancing and fault takeover in a cloud environment. However, it is vulnerable to active and passive safety attacks during the migration process, which makes the IT industry hesitant to accept this feature in the cloud.

Conciliatory VM migration procedures may result in DOS attacks and data integrity and confidentiality loss. Cloud providers store images in encrypted form to cater to attacks such as illegal access to images and vaccinating malevolent code on VM images. Therefore, the security of VM migration along encrypted disk image keys becomes necessary. Previously, research focused on VM migration's performance,

Department of Computer Science, Adaikalamatha College, Vallam, Thanjavur. Affiliated to Bharathidasan University, Trichy, Tamil Nadu, India.

***Corresponding Author:** Raja Selvaraj, Department of Computer Science, Adaikalamatha College, Vallam, Thanjavur. Affiliated to Bharathidasan University, Trichy, Tamil Nadu, India., E-Mail: rajasjc@gmail.com

How to cite this article: Selvaraj, R., Sundaram, M.S. (2023). ECM: Enhanced confidentiality method to ensure the secure migration of data in VM to cloud environment. The Scientific Temper, **14**(3): 902-908.

Doi: 10.58414/SCIENTIFICTEMPER.2023.14.3.53

Source of support: Nil

Conflict of interest: None.

Received: 10/07/2023 **Accepted:** 20/08/2023 **Published** : 25/09/2023

leaving security aspects of the migration process completely explored. Data migration and management are enterprises' most challenging and complicated tasks. It involves high costs because it requires many resources, human power, and management procedures. Cloud computing offers great potential to migrate, store and maintain data with significant cost-savings to users. Many enterprises migrate to data outsourcing in the cloud environment (Raja Selvaraj *et al.,* 2023). As cloud storage has benefits, data security is the concern of cloud storage. Due to the multi-tenancy nature of the cloud, the data may be accessed by other users of the same cloud, and hence, two types of illegal users can access the data, such as insiders and outsiders. As a result, the confidentiality of cloud-stored data may be compromised. Confidentiality is a criterion used to measure data security in the cloud.

Encryption techniques ensure the confidentiality parameter. The most used symmetric encryption in the cloud is AES encryption (Vidya S. *et al.,* 2022). Due to the open nature of the cloud, AES needs to strengthen itself to protect the data during migration and in cloud servers. The enhanced cryptography symmetric technique is proposed in this paper. The enhancement is carried out in the AES encryption method. The AES is a symmetric cryptography approach. It takes a 128-bit key and 128-bit plaintext for encryption and decryption (Vaidehi M. *et al.,* 2015). The AES-128 runs 10 rounds, AES-192 runs 12 rounds, and AES-256 runs 14 rounds to generate the cipher text. Each round in AES consists of four distinct bits of transformation. These are subbytes, shift row, mix column and add round key. the four transformations are carried out n-1 round, and the mix column is omitted in the Nth round. Figure 1 portrays the block diagram of the AES.

This paper proposes the enhanced confidentiality method. The ECM enhances the SubByte stage to improve the quality of the encrypted data so as not to be hacked. Data in the VM is encrypted using the proposed approach and migrated without any modification or disclosure.

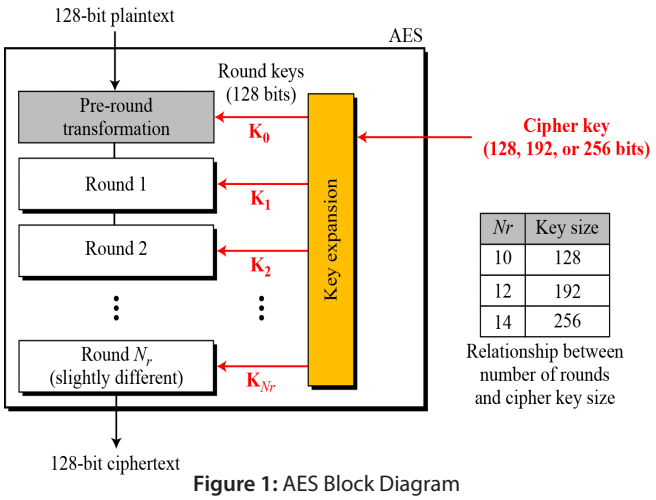

Figure 1: AES Block Diagram

Related Work

Cryptography techniques can only ensure data security (Monikandan S. *et al.,* 2015). Some of the earlier research ideas to handle data security in cloud computing are presented in this section. Sam Njuki and colleagues (Sam Njuki *et al.,* 2017) investigated the security issues raised by cloud migration and created a more effective architecture to address them. They examined various cloud migration security models concerning the security requirements during migration and listed their advantages and disadvantages. They reported using a more effective architectural design to address the security matters connected with cloud migration.

A wide-ranging resolution for Secure VM Migration (SV2M) in a cloud environment was put forth by Shibli M. A. *et al.,* (Shibli M. A. *et al.,* 2014), which ensured authorization, mutual authentication, confidentiality, replay protection, integrity, and nonrepudiation with only minor changes to existing infrastructure. They have included additional tools for the management and storage of keys used in their SV2M solution and enhanced the capabilities of the cloud provider's key manager. Additionally, they have linked the system with OpenStack, a widely used open-source cloud computing platform for research. Additionally, they used the well-known automatic protocol verification programme AVISPA to assess the security of the SV2M system.

Chandrakala N. *et al.,* (2018) proposed virtual machine migration to increase cloud computing security. An individual computer system is simulated by a virtual machine (VM). Virtual machine migration is helpful in cloud computing for moving operating system instances between different physical machines. Data centre operators can modify VM placement to meet performance goals better, enhance resource usage and communication locality, achieve fault tolerance, consume less energy, and make it easier to perform system maintenance tasks. The placement of VMs is suggested to have a substantial impression on safety levels in the migration-based security method. They create a method that provides a secure placement arrangement based on the survivability study of VMs and the Discrete Time Markov Chain (DTMC) study, allowing the guest VMs to move before the attack is successful.

By adjusting the sub-byte operation, Santhanalakshmi M. *et al.,* (Santhanalakshmi M. *et al.,* 2023) suggested a modified conventional AES. The traditional AES is changed by doing the following actions: XORk0, XORk1, XORk2, and XORk3 are four 8-bit keys created using the 16-byte round key. In the key matrix for that round, each byte in the associated row is XORed. Before s-box substitution, each XORki is also XORed to the associated byte in the row of the state matrix. The results show that the early and full rounds of the encryption section have improved the suggested AES's diffusion properties. This improvement employs newly added elementary operations, including exclusive OR, modulo arithmetic, and the sub-byte operation to inject more key variants into the cypher round.

Key aspects of the framework proposed by Ijaz Ahmad Awan *et al.,* (Ijaz Ahmad Awan *et al.,* 2020) include improved security and owner's data privacy. The double-round key feature alters the 128 AES method, accelerating encryption at 1000 blocks per second. However, a single round key with 800 blocks per second is traditionally used. This method uses less energy, improves load balancing, and improves network trust and resource management. The deployment of AES with 16, 32, 64, and 128 plain text bytes is part of the framework. The visualization of simulation results shows the algorithm's suitability for obtaining specific quality attributes. Results reveal that the suggested architecture reduces network usage by 11.53 %, delay by 15.67 %, and energy consumption by 14.43%. As a result, the suggested framework improves security, decreases resource usage, and shortens latency when delivering computational cloud services.

Nahom Gebeyehu Zinabu *et al.,* (2022) developed the enhanced efficiency of advanced encryption standard (EE-AES) for data security in the cloud. Sub-bytes and Mix Column, among the four stages of traditional AES utilized for encryption and decryption, cause the most latency. The mix column accounts for 60% of the delay. The bitwise reverse transposition approach replaces the mixed column stage of the intended symmetrical cryptography technique to address these issues. The current advanced encryption standard (AES) and modified advanced encryption standard (MAES) algorithm benefit from increased speed efficiency. Compared to the original advanced encryption standard (AES) and modified advanced encryption standard, our bitwise reverse transposition technique's simulation results showed faster encryption and decryption times (MAES). With this process, the security level of the AES algorithm is maintained, but the calculation requirements of the original design mix column step are reduced.

Migration System Procedure

Cloud migration is a vital procedure that involves moving data, programs, or other components into a cloud computing environment. The data is transported with the virtual machine. In addition to actual data (customer and user data), configuration data, metadata about the data, and data about the data, VM also houses several other types of data. Different forms of cloud migration are required to transfer to a modern cloud computing environment. Transferring data to the cloud or another location must be safeguarded. The recommended Cloud Architecture gives security and reliability to migrate the workload to cloud computing. Figure 2 shows the migration procedure involved in the proposed system design.

The sender machine generates a message size of 128-bit and a 128-bit number to be used as a symmetric key Sk for this message only. The message is encrypted using the proposed ECM with the symmetric key. The symmetric key is scrambled with a public key cryptosystem using the

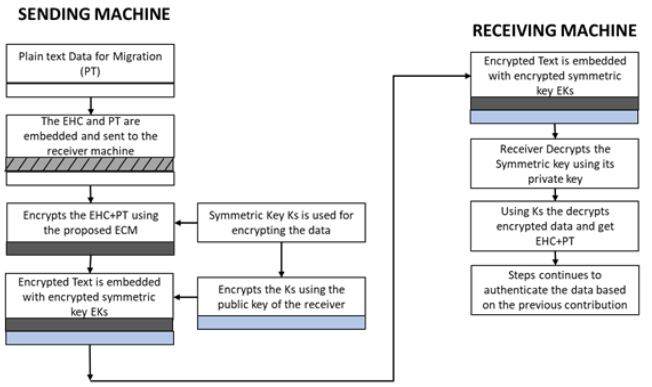

Figure 2: Proposed secured migration procedure

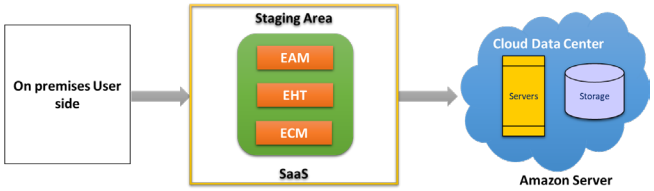

Figure 3: Abstract View of Migration System Design

recipient's public key and is prepended to the message. The receiver uses its private key to decrypt and recover the symmetric key. The symmetric key is used to decrypt the message. The data and key are encrypted and forwarded to the recipient to maintain confidentiality.

The procedures depicted in Figure 2 are done in the architecture's staging area. The staging area is one of the various architectural components proposed in the previous research (Manikandasaran S. S. *et al.,* 2018). The staging area represents the movement of workloads from the on-premises data centre to the cloud data centre. It separates several tenants to segregate and transmit tasks securely. The staging area stops threats and malicious attacks from the cloud data centre. The VMs' data is moved to the staging area. The VM migration is controlled and watched over by the staging area. Figure 3 shows the abstract representation of the proposed research, where ECM is one of the parts of the research.

According to the migration procedure, the proposed ECM encrypts the enhanced hash code (EHC) with plaintext (PT) being migrated to the cloud. The key used for ECM encryption is also encrypted and embedded with the encrypted data. Figure 2 depicts the proposed research's overall procedure of the migration design. In the overall procedure, ECM encryption provides security to the data when migrated from off-premises to on-premises.

Proposed ECM Methodology

The symmetric cipher's most popular encryption is AES. The four procedures in each round of the secret writing method are sub byte, shift rows, mix column, and add round key. There are four operations in each round, which is repetitive.

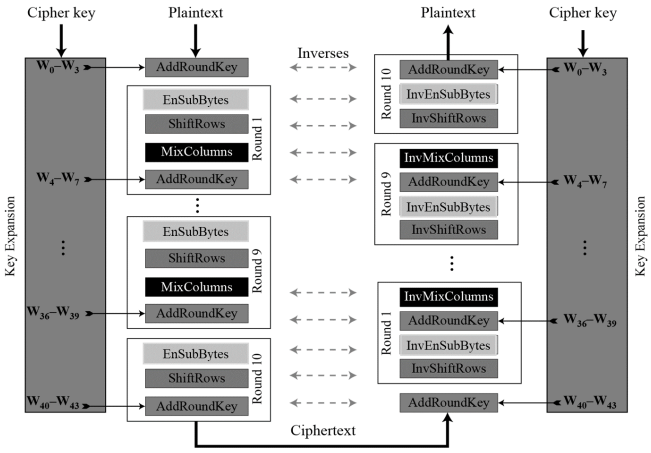

Figure 4: Proposed ECM Block Diagram

As a result, the output of the first round is given as input to the second round, which uses a different set of keys to conduct constant operations. Up until the last round, this procedure is used. There is no mix column operation in the final round. The state array is obtained when the final round is encryption text for transmission. Figure 4 demonstrates the proposed ECM block diagram where the subbytes stage is enhanced, and other stages are carried out as the same procedure as standard AES.

Initially, the input data is considered in a 4X4 matrix. The 128 bits are taken as 16 bytes or 32-bit words. Before starting the rounds, a pre-round computation is performed called AddRoundKey. The main key size is 128-bit, expanded to 44 32-bit words. A 4-word key is given to the add round key transformation for each round. Enhancement in the existing AES is considered in the subbytes stage, where the enhanced ECM uses dynamic S-Boxes each round. ECM also uses an additional 128-bit key for generating dynamic S-boxes. The degree of security is improved in the enhanced subbytes stage. In the traditional AES, there are four steps in each round; among them, subbytes uses a fixed S-box to substitute the bytes. Using a fixed S-box will lead to cipher attacks. Hence, the proposed method uses a dynamic s-box for each round. The procedure of ECM is as follows.

The procedure of the enhanced cryptography method

- The user inputs the data, i.e., plaintext (PT) and keys (K_i), to ECM.
- ECM accepts 128-bit block array, i.e., 16 bytes array.
- The 16-byte array is copied in a 4x4 matrix format known as the state matrix.
- Accepts 128-bit length key in the form of 4-words.
- The 4-word key is expanded into a W matrix with 44 words.
- The state matrix is XORed with the AddRoundKey stage, which is the pre-round calculation.
- Rounds are started and run for 9 rounds,
  - Enhanced_SubBytes with a new S-Box stage performs byte-wise substitution in the original message.
  - ShiftRows stage performs row-wise permutation by left circular shifting operation on the Enhanced_SubBytes output.
  - MixColumns mixes the bytes of the previous stage output by column-wise substitution that uses addition in Galois Fields GF(2^8).
  - AddRoundKey XORs the output of the Enhanced_MixColumns stage bits with expanded keys.
- After 9 rounds, the 10th round is performed without the MixColumns stage.
- The output from the 10-round MixColumn is 128-bit cipher text.

EnSubByte Stage

The 16 × 16 matrix look-up table S-box is used in the substitute bytes stage, also known as SubBytes. An S-box is a 16 × 16 bytes look-up table with 256 bits. Galois field GF (28) arithmetic operation and the bit scrambling method employing invertible affine translation are used to retrieve the values of the S-box. The S-box look-up table values are produced using GF (28)'s multiplicative inverse and the degree 8 irreducible polynomial x8 + x4 + x3 + x+1.

The state matrix undergoes bit scrambling and substitution modifications in the SubBytes step. In a replacement transformation, an additional byte drawn from the S-box look-up table replaces each 4x4 state matrix input. The state matrix is split into two 4-bit segments for each byte. In an S-box, the first component receives the row position, while the second receives the column position. Using this technique, you may retrieve the S-substitute box's value for any input byte. The input byte 00 remains the same if a byte value of the state matrix is 00 because 00 has no multiplicative inverse.

The second transformation of the subbytes stage uses bit scrambling to break the connection between the state matrix's bytes. The bit scrambling process in the affine translation employs the XOR and mod operations. The following formula is used to determine how each bit of a byte in the state matrix has been transformed:

$$d'i = di \otimes d_{(i+4)} \bmod 8 \otimes d_{(i+5)} \bmod 8 \otimes d_{(i+6)} \bmod 8 \otimes d_{(i+7)} \bmod 8 \otimes c_i$$

Where the c_i value is 0x63 (01100011).

Consequently, each input byte is replaced with a new byte produced by GF at the SubBytes. After a new byte has been substituted, bit scrambling is used on the input to uncover the relationship between the bits. 63 is the mapping value for 00 bytes for 00 input.

The inverse Substitute Bytes stage, or InvSubBytes, minimizes the association between the bytes and the decryption process. The look-up table, in this instance, is built in reverse. Before performing the multiplicative inverse

Table 1: S-box First Row

Key	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
s-value	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76

Table 2: ECM S-Box for EnSubBytes Round1

	0	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*a*	*b*	*c*	*d*	*e*	*f*
0	63	7d	75	78	f6	6e	69	C2	38	08	6d	20	f2	da	a5	79
1	ca	83	cb	7e	fe	5c	41	f7	a5	dd	a8	a4	90	a9	7c	cf
2	b7	fc	91	25	32	3a	f1	cb	3c	ac	ef	fa	7d	d0	3f	1a
3	04	c6	21	c0	1c	93	03	9d	0f	1b	8a	e9	e7	2a	bc	7a
4	09	82	2e	19	1f	6b	5c	a7	5a	32	dc	b8	25	ee	21	8b
5	53	d0	02	ee	24	f9	b7	5c	62	c2	b4	32	46	41	56	c0
6	d0	ee	a8	f8	47	48	35	82	4d	f0	08	74	5c	31	91	a7
7	51	a2	42	8c	96	98	3e	f2	b4	bf	d0	2a	1c	f2	fd	dd
8	cd	0d	11	ef	5b	92	42	10	cc	ae	74	36	68	50	17	7c
9	60	80	4d	df	26	2f	96	8f	4e	e7	b2	1f	d2	53	05	d4
a	e0	33	38	09	4d	03	22	5b	ca	da	a6	69	9d	98	ea	76
b	e7	c9	35	6e	89	d0	48	ae	64	5f	fe	e1	69	77	a0	07
c	ba	79	27	2d	18	a3	b2	c1	e0	d4	7e	14	47	b0	85	85
d	70	3f	b7	65	4c	06	f0	09	69	3c	5d	b2	8a	cc	13	91
e	e1	f9	9a	12	6d	dc	88	93	93	17	8d	e2	c2	58	26	d0
f	8c	a0	8b	0e	bb	e3	44	6f	49	90	27	04	bc	59	b5	19

Table 3: ECM S-Box for EnSubBytes Round2

	0	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*A*	*B*	*C*	*D*	*E*	*F*
0	62	7f	76	7c	f3	68	6e	ca	31	02	66	2c	ff	d4	aa	79
1	cb	81	c8	7a	fb	5a	46	ff	ac	d7	a3	a8	94	a7	73	cf
2	b6	fe	92	c4	37	3c	fb	c3	35	a6	e4	f6	70	de	30	1a
3	05	c4	22	c4	19	95	04	95	06	11	81	e5	ea	24	b3	7a
4	08	80	2d	1d	1a	6d	5b	af	53	38	d7	b4	28	e0	2e	8b
5	52	de	01	ea	21	ff	b0	54	6b	c8	bf	3e	4b	4c	59	c0
6	d1	ec	ab	fc	42	4e	32	8a	44	fa	03	78	51	3f	9e	a7
7	50	a0	41	88	93	9e	39	fa	bd	b5	db	26	11	fc	f2	dd
8	cc	0f	12	eb	5e	94	45	18	c5	a4	7f	3a	65	5e	18	74
9	61	82	4e	db	23	29	91	87	47	ed	b9	13	df	5d	0a	d4
A	e1	31	3b	0d	48	08	25	53	c3	d0	ad	65	90	96	e5	76
B	e6	cb	36	6a	8c	d6	4f	a6	6d	55	f5	ed	64	79	af	07
C	bb	7b	24	29	1d	a5	b5	c9	e9	de	75	18	4a	be	8a	85
D	71	3d	b4	61	49	00	f7	01	60	36	56	be	87	c2	1c	91
E	e0	fb	99	1b	68	da	8f	9b	9a	1d	86	ee	cf	56	29	d0
F	8d	a2	88	0a	be	e5	43	67	40	9a	2c	08	b1	57	ba	19

in GF(28) operations, the byte-wise reverse process of bit scrambling is carried out. The following formula is used to do bit scrambling's opposite operation:

$$d'_i = d_{(i+2)} \bmod 8 \otimes d_{(i+5)} \bmod 8 \otimes d(_{i+7)} \bmod 8 \otimes de_i$$

Where the de_i hexadecimal value is 0x05 (00000101), encryption c_i and decryption de_i are the bits of chosen bytes c and d to construct the S-box.

The look-up table S-Box in AES is static yet exhibits non-linearity characteristics. To avoid attacks, the S-box is constructed using an invertible affine transformation and a multiplicative inverse function. The values in the S-box are substituted for each input byte of the state matrix during the SubBytes stage. ECM strengthens the S-box, the only non-linearity feature, to increase the AES algorithm's security. The ECM suggests S-box is dynamic (dynamic S-box). It is a dynamic look-up table that is key-dependent. The Enhanced SubBytes stage of ECM increases the complexity of the SubBytes stage, making an algebraic attack impossible. A

Table 4: Result of Avalanche Effect After Changing a Bit in the Secret Key by AES

Methods	*Secret Key*	*Secret Key (Hexa)*	*Cipher Text (Hexa)*	*Avalanche Effect*
AES	dKr09Wahme#dHrn7	64 4B 72 00 39 57 61 68 6D 65 23 64 68 73 6E 37	83 4A A0 BC 25 78 FD FB 5D 14 24 BD 32 CD E0 00	0.517 (50.7%)
	dKr09Wahme#dHsn7	64 4B 72 00 39 57 61 68 6D 65 23 64 68 72 6E 37	BB 87 74 F9 78 20 28 D8 40 1D DE 6C F7 41 3A E7	

Table 5: Result of Avalanche Effect After Changing a Bit in the Secret Key by ECM

Methods	*Secret Key*	*Secret Key (Hexa)*	*Cipher Text (Hexa)*	*Avalanche Effect*
ECM	dKr09Wahme#dHrn7	64 4B 72 00 39 57 61 68 6D 65 23 64 68 73 6E 37	D6 BA 33 A8 C3 61 3A 74 B5 FB EB A4 EA 97 B1 10	0.587 (58.7%)
	dKr09Wahme#dHsn7	64 4B 72 00 39 57 61 68 6D 65 23 64 68 72 6E 37	5D 2D A4 93 11 04 95 C8 2E 17 D3 7F 5C 43 22 86	

Table 6: Result of Avalanche Effect After Changing a Bit in the Plaintext by AES

Methods	Secret Key	Plain Text (Hexa)	Cipher Text (Hexa)	Avalanche Effect
AES	dKr09Wahme#dHrn7	53 27 4C 6F 76 65 20 55 6E 69 6C 6F 72 69 6E 21	38 A4 A0 CB 25 78 FD FB 5D 14 24 BD 32 CD E0 00	0.503 (50.3%)
	dKr09Wahme#dHsn7	53 27 4C 6F 76 65 20 55 6E 69 6D 6F 72 69 6E 21	2F 69 36 B1 6A FA 68 D2 C4 4A DF 2D BA 64 CA A9	

Table 7: Result of Avalanche Effect After Changing a Bit in the Plaintext by ECM

Methods	*Secret Key*	*Plain Text (Hexa)*	*Cipher Text (Hexa)*	*Avalanche Effect*
ECM	dKr09Wahme#dHrn7	53 27 4C 6F 76 65 20 55 6E 69 6C 6F 72 69 6E 21	6D AB 33 8A C3 61 3A 74 B5 FB EB A4 EA 97 B1 10	0.574 (57.4%)
	dKr09Wahme#dHsn7	53 27 4C 6F 76 65 20 55 6E 69 6D 6F 72 69 6E 21	B6 CC 92 7D 1E C2 74 B4 E7 EB 7E 0A D1 CA 67 6F	

key schedule technique creates a 16-byte array key in the proposed ECM. Python's random method is used to produce the random number. Table 1 shows the first row of the S-box.

When we XOR the values of 03 & 7b

0000 0011

0111 1011 $\otimes$

0111 1000

Therefore, the new value of the dynamic S-Box in the same place is 78.

Similarly, new S-boxes are generated for all 10 iterations of the encryption processes. This dynamic S-box of ECM offers more security than the existing AES algorithm. Table 2 and Table 3 are tables generated for round 1 and round 2 of ECM encryption.

The remaining rounds of S-boxes are generated in the same way. The dynamic S-boxes for each round of the ECM make the cipher text harder. The adversarial is not able to make any guess on the cipher text. Hence, the ECM creates more confusion about the data and protects it from data attacks.

Evaluation and Results

The Proposed research work is implemented in Python. A Software-as-a-service cloud-based application is developed. The cloud application is coded for ECM alone with other implementations of research. The sample text is considered for the proposed research. The text is encrypted using ECM. The efficiency is tested with the avalanche effect. The formula for the avalanche effect is,

Avalanche Effect= (Number of Changed bits in ciphertext) /(Number of bits in ciphertext)

An avalanche > 50% should always be satisfied by a good cipher. Additionally, it is advised to carry out this analysis for numerous cipher test cases and track the average avalanche effect.

The plaintext is represented in hexadecimal values, 53 27 4C 6F 76 65 20 55 6E 69 6C 6F 72 69 6E 21. This hexadecimal plaintext is measured with the avalanche effect of the ECM and conventional AES. Table 4 and 5 shows the result of the avalanche effect of AES and ECM, respectively, after changing a bit in the secret key.

The results of the avalanche effect show that the proposed ECM has a 7% improved avalanche effect compared to the AES. Tables 6 and 7 show the result avalanche effect after changing a bit in the plaintext. The result shows that the efficiency of the ECM is increased by 7% compared to the AES encryption technique.

The evaluation of the proposed ECM and AES uses the avalanche effect. The result of the avalanche effect shows

the proposed ECM has secured a percentage of efficiency compared to the AES. Hence, the proposed ECM is more secure and efficient in protecting the migrated data in VMs.

Conclusion

Security of the cloud environment and data in the public cloud is more vulnerable to attack during migration. There are possibilities of losing the data in the VM when migrated. An enhanced architecture proposes different components for migrating VMs and data to the cloud. One of the architecture's components is ECM, an Enhanced Cryptography Method proposed in this paper. The ECM is the enhanced method of AES. The SubBytes stage of the AES is enhanced with the dynamic S-Boxes. The substitution of bytes in ECM in the SubByte stage is enhanced, and different S-boxes are used for each round of the ECM. Changes in the S-Box of ECM confuse the hacker when trying to attack the data. It ensures the confidentiality of the data being migrated. The key generation technique generates the key used for ECM. The proposed ECM is evaluated using the avalanche Effect. Results show that the proposed methods produce better security than the existing method.

Reference

Arockiam, L., & Monikandan, S. (2013). Data security and privacy in cloud storage using hybrid symmetric encryption algorithm. *International Journal of Advanced Research in Computer and Communication Engineering*, *2*(8): 3064-3070.

Arockiam, L., Monikandan, S., & Parthasarathy, G. (2017). Cloud computing: a survey. *Journal of Computer and Communication Technology*, *8*(1): 21-28.

Chandrakala, N., B. Thirumala Rao. (2018). Migration of Virtual Machines to Improve the Security in Cloud Computing. *International Journal of Electrical and Computer Engineering*, *8*(1): 210~219.

Ijaz Ahmad Awan, Muhammad Shiraz, Muhammad Usman Hashmi, Qaisar Shaheen, Rizwan Akhtar, and Allah Ditta. (2020). Secure Framework Enhancing AES Algorithm in Cloud Computing, *Hindawi, Security and Communication Networks*, *2020*:1-16.

Manikandasaran, S. S., and Raja S. (2018). Security architecture for multi-tenant cloud migration. *International Journal Future Computer Communication*, *7*(2): 42-45.

Monikandan, S., and Arockiam, L. (2015). Confidentiality Technique to Enhance Security of Data in Public Cloud Storage Using Data Obfuscation. *Indian Journal of Science and Technology*, *8*(24): 1-10.

Nahom Gebeyehu Zinabu, Samuel Asferaw. (2022). Enhanced Efficiency of Advanced Encryption Standard (EE-AES) Algorithm. *American Journal of Engineering and Technology Management*, *7*(3): 59-65.

Raja S., Dr. S.S. Manikandasaran. (2021). Enhanced Framework To Migrate Virtual Machines To The Container In Cloud Environment. *Webology*, *18*(6): 7507-7515.

Raja Selvaraj and Manikandasaran S. S. (2023). EAM: Enhanced authentication method to ensure the authenticity and integrity of the data in VM migration to the cloud environment. *The Scientific Temper, 14*(1): 227-232.

Sam Njuki, Jianbiao Zhang and Edna Too. (2017). Analysis of Virtual Machine Migration Security Architectures in Cloud Computing. International Journal of Scientific & Engineering Research, *8*(10): 1753-1763.

Santhanalakshmi, M., Ms Lakshana, K., Shahitya, G. M. (2023). Enhanced AES-256 cipher round algorithm for IoT applications. *The Scientific Temper*, *14*(1): 184-190.

Shibli, M. A., N. Ahmad, A. Kanwal and A. Ghafoor. (2014). Secure virtual machine migration (SV2M) in cloud federation. *IEEE International Conference on Security and Cryptography*: 1-6.

Vaidehi M. and B. Justus Rabi. (2015). Enhanced MixColumn Design for AES Encryption. *Indian Journal of Science and Technology*, *8*(35): 1-7.

Vidya S. and Deepa T. (2022). Security Enhancement Using AES Algorithm forEmergency Situation Detection System. *International Journal of Innovative Science, Engineering & Technology*, *9*(7): 15-15.