Webology (ISSN: 1735-188X)
Volume 18, Number 4, 2021

Optimization Based Key Generation Approach For Enhancing Authentication In Cloud Computing

D.Karthik[1] , Dr. A.Vinayagam[2]

[1]Research Scholar, PG & Research Department of Computer Science Government Arts College (Autonomous) (Affiliated to Bharathidasan University) Karur, Tamilnadu, India.

[2]Assistant Professor, PG & Research Department of Computer Science Government Arts College (Autonomous) (Affiliated to Bharathidasan University) Karur, Tamilnadu, India.

ABSTRACT
Computing applications and data are developing at such a rapid rate that larger servers and data centers are necessary to process data quickly and efficiently. Cloud computing is the outcome of a significant change in the way information technology (IT) and computer services are supplied and purchased. Cloud computing is gaining in popularity by the day. Many businesses and government agencies will migrate to the cloud if security parameters are appropriately addressed. Cloud computing's recent rise has drastically altered everyone's perception of network architectures, development processes, and software delivery. From a security standpoint, the transition to the cloud has presented various uncharted risks and challenges, undermining the efficiency of established protective systems. An improved Elliptical Curve Cryptography (ECC) based encryption technique is proposed in this research to provide assured safe communication among clouds by combining ECC with Dragan Fly Optimization search, which is utilized to create the ideal random number in the key generation phase of ECC. This method will cut down on encryption, decryption, and key generation time.

KEYWORDS: Cloud Computing, Encryption, Decryption, Key Generation, Elliptical Curve Cryptography, Dragon Fly Optimization

1. INTRODUCTION

NIST definition of cloud computing "Cloud computing is a model for providing universal, appropriate, on-demand network access to a mutual pool of configurable computing resources (different data storage, servers, networks, and applications) that can be rapidly provisioned and launched with minimal management effort or service provider interaction." [1][2].

The main reason for the existence of different perceptions of cloud computing is that cloud computing, contrasting other scientific terms, is not a new technology, but rather a new operations model that brings together a set of existing technologies to run big business in a

diverse (unusual) way [3]. Indeed, most of the technologies used by cloud computing, such as virtualization and as per service (utility) pricing, are not new. Instead, cloud computing leverages these existing technologies to meet the technological and economic requirements of today's demand for information technology [4].

Cloud Computing is a model for enabling universal, suitable, on-demand network access to a mutual pool of configurable computing resources (different networks, servers, data storage, services and applications) those may be rapidly provisioned and released with minimal management effort or service provider inter-action. Today Small and Medium Business (SMB) companies are increasingly realizing that simply by tapping into the cloud they can gain fast access to best business applications or significantly boost their infrastructure resources, all at very low cost. Cloud computing applications have broadly three areas known as cloud delivery models: Infrastructure as a service (IaaS), Platform as a service (PaaS), Software as a service (SaaS). So far, there have been little scientific definitions trying to develop a complete definition of the cloud computing phenomenon [5].

As the cloud computing is achieving popularity periodically, alarms (concerns) are being voiced about the security issues presented through the adoption of this new model. The usefulness and efficiency of traditional protection mechanisms are being reviewed, as the characteristics of this innovative deployment model, differ widely from them of traditional architectures.

Authentication plays an important role in protecting resources against unauthorized use. But still the most widely used authentication system is based on the use of text passwords [6][7]. Text based passwords are not secure enough for many applications that enforce security by access control mechanisms. Authentication based on text-based passwords has major drawbacks. More sophisticated authentication process is costly and may need additional equipment or hardware.

2. RELATED WORKS

Veerabathiran, Vijaya Kumar, et al [8] introduced a homomorphic proxy re-encryption (HPRE) in this paper that enables various CU to share INFO that they redistributed HPRE encrypted utilizing their PubKs with the plausibility by a close procedure such as INFO remotely. The test of giving secrecy, uprightness, and access control (AC) of INFO facilitated on cloud stages is not provided for by conventional AC models. CFCM models were created through the duration of numerous decades to satisfy the association's necessities, which accepted full authority over the physical structure of the assets. The hypothesis of the INFO proprietor, an INFO controller, and a supervisor is available in the equivalent trusted area. Besides, CCESR features like the essential unit, fuzzy set (FS) hypothesis, and EW strategy utilized to precisely measure the likelihood of CCE security risks (SR) and the subsequent damages of CCESR estimation.

Anuradha, M., et al [9] designed a cancer prediction system using Internet of Things upon extracting the details of blood results to test whether it is normal or abnormal. In addition to this, encryption is done on the blood results of cancer affected patient and store it in cloud for quick reference through Internet for the doctor or healthcare nurse to handle the patient data secretly. This research work concentrated on enhancing the health care computations and

processing. It provides a framework to enhance the performance of the existing health care industry across the globe. As the entire medical data has to be saved in cloud, the traditional medical treatment limitations can be overcome. Encryption and decryption are done using AES algorithm in order to provide authentication and security in handling cancer patients.

Velmurugadass, P., et al [10] developed a novel framework that will monitor the activities that takes place on particular data evidence, the authors created a Cloud based Software Defined Network (SDN), its consists of 100 - mobile Nodes (IOT devices), open flow switch and Blockchain based controllers, cloud server, Authentication Server (AS) and investigator. Initially all users are registered with AS and obtain their secret key from AS based on the Harmony Search Optimization (HSO). In the mobile nodes the packets are encrypted by using Elliptic Curve Integrated Encryption Scheme (ECIES) algorithm and transfer to the cloud server. SDN controller maintains blockchain to preserve evidences collected from data and signature of the users based on the SHA-256 Cryptographic Hash Algorithm.

Shi, Canghong, et al [11] proposed a novel and effective authentication scheme for encrypted speech. At first, the host speech signal is first scrambled and encrypted by Advanced Encryption Standard (AES). Then, Integer Wavelet Transform (IWT) is performed to obtain the approximation coefficients and the detail coefficients. At last, Non-negative Matrix Factorization (NMF) is employed to generate perceptual hashing, which is embedded into the encrypted speech by differential expansion. In authentication section, the tampered region of encrypted speech is located by comparing the reconstructed perceptual hashing with the extracted perceptual hashing version.

Padmaja, K., and R. Seshadri [12] presented a Message Personalized E-Healthcare Services cyber secured authentication method for both at the storage and retrieval is presented for EHC systems generate Data via Cloud environment. The overall method is split into three stages. They are sign in stage, device authentication stage and device communication stage. First sign in is performed by applying Message Digest Hashing for each incoming medical device by cloud server, ensuring scalable and manageable architecture. Next, authentication is performed for each registered device using Message hash model-based encryption ensuring network latency and security during data storage at cloud server.

Safkhani, Masoumeh, et al [13] proposed RFID based authentication protocol for vehicular cloud computing whose authors claimed to be secure and efficient. Besides, despite the use of timestamps, we show how this protocol also suffers from a range of relay attacks. The complexity of any of the proposed attacks is negligible while the success probability is maximum (i.e., the adversary's success probability is '1' since all the proposed attacks are deterministic).

Kumar, Vinod, et al [14] proposed an elliptic curve cryptography (ECC) based authentication framework for Vehicular Cloud Computing (VCC), which is equipped with a radio frequency identification (RFID). The authors proved the claim of secure communication using formal security analysis in the random oracle model and information analysis.

Joseph, Teena, et al [15] proposed a multimodal authentication system image processing technique such as pre-processing, normalization and feature extraction. From the extracted features, a unique secret key is generated by fusing the traits in two stages. False

Acceptance Rate (FAR) and False Rejection Rate (FRR) metrics are used to measure the robustness of the system.

Saranya, A., and R. Naresh [16] presented a cloud based efficient Authentication for mobile payments using key distribution method. Based on the certificate less proxy re signature system, the authors have designed a different mobile payment procedure which not only attains secrecy; it also achieves the storage complexity by consuming fewer amounts of data. In the proposed method, the efficiency is particularly enhanced by retaining cost of computation in the payment area. Furthermore, by seeing that the payment area the Merchant Server wishes to achieve computation for every payment operation, the impression of batch authentication was implemented to remove the difficulties faced when a greater number of customers use the Payment area so that Merchant Server can solve the scalability dispute.

Goumidi, Hadjer, et al [17] proposed an efficient algorithm to ensure VCC security and privacy. The authors used Pseudo-ID instead of vehicles' real ID to provide conductors' privacy, Identifier-Based Signature mechanism is used to guarantee vehicles' authentication, and Ciphertext-Policy Attribute-Based Encryption (CP-ABE) algorithm is used for key distribution. The liGhtweight secURe AutheNticaTion and keY distribution scheme for vehicular cloud computing (GUARANTY) ensures a secure keys distribution to minimize the encryption and decryption computation cost. Vehicles use a symmetrical cryptography in their communication.

3. DRAGON FLY OPTIMIZATION ALGORITHM

Xin-She Yang developed the Dragon Fly algorithm [21] in the year 2008 by the inspiration got from the fireflies. Three main assumptions were made here; they are (a) all FF are unisex (b) Attractiveness is directly proportional to brightness, and attractiveness is inversely proportionally to distance. (c) The objective function defines the brightness of FF. Each FF has its attractiveness, which is represented as ρ, and it decreased with distance x. Equation (3) represents the attractiveness between two FF in which ρ_0 denotes maximum attractiveness, and it is referred to as the light absorption coefficient. Further, g and h are the two FF at position K_g and K_h, their distance is evaluated using the mathematical equation (4) in which b represents the count of dimensions. The movement of FF is represented in Eq. (5). The light intensity M_h of FF is evaluated based on the distance between the fireflies. The mathematical equation of FF is shown in Eq. (2) in which M_0 represents the original light intensity [18].

$$M = M_0 e^{-x} \quad (2)$$

$$\rho(x) = \rho_0 e^{-x},\ v \geq 1 \quad (3)$$

$$x_{gh} = \left\| K_g - K_h \right\| = \sqrt{\sum_{w=1}^{b} (K_{g,w} - K_{h,w})^2} \quad (4)$$

$$K_{best} = K_g + \rho_0^{-\gamma x_{gh}^2} (K_h - K_g) + \omega \left(rand - \frac{1}{2} \right) \quad (5)$$

The first term denotes the current position of FF, and the second term denotes the attractiveness of FF. The last term describes the random movement of FF. The initial position of FF is denoted as per Eq. (5).

4. ELLIPTICAL CURVE CRYPTOGRAPHY ALGORITHM FOR AUTHENTICATION

Elliptical curve cryptography is a public key encryption strategy that depends on the hypothesis of elliptical curves. This encryption strategy utilizes the properties of the elliptic curve to create keys as opposed to utilizing the standard approach of the age of keys utilizing the result of two enormous prime numbers. At first, the elliptic curves for cryptography were utilized in H.W. Lenstra's elliptical curve factoring algorithm. Roused by this high utilization of elliptic curves, the elliptical curve cryptography was proposed by N.Kobiltz and V.Miller independently in 1985. The most significant preferred position of elliptical curve cryptography is the utilization of littler keys giving a similar degree of security. ECC can give a similar security a 164-piece key that different frameworks furnish with a 1024-piece key. It is generally valuable for mobile applications as it can furnish elevated level security with low computing force and battery resources. ECC is a public-key cryptosystem that is utilized to create the public key and the private key to encrypt and decrypt the data. It depends on the multifaceted scientific nature of unraveling the elliptic curve discrete logarithm issue, which manages the issue of calculating the number of steps or hops it takes to move to start with one point then onto the next point on the elliptic curve [19].

Elliptic curves are the binary curves and are symmetrical over x- axis. The function defines these:

$$y^2 = x^3 + ax + b \quad (1)$$

where x and y are the standard variables that define the function while as a and b are the constant coefficients that define the curve. As the values of a and b change, the elliptical curve also alters. For elliptical curves, the discriminant $\Delta = 4a^3 + 27b^2$ is non-zero. The operations used on elliptical curves in cryptography are point addition, point multiplication, and point doubling. The important characteristic of the elliptic curve is the finite field concept. This means that there is a way to limit the values on the curve. This "max" value established on the x-axis is represented by "p". It is also called "modulo value" for any ECC cryptosystem. This point depicts the finite length upon which the operations can be performed on the curve. In ECC, the modular value depicts the key size for the system. Thus, the parameters that fully define the ECC cryptosystem are:

p – Specification of the finite field.
a,b: Coefficients for defining a curve
G – Generator points on the curve where the operations start.
n – Order of G.
h – Division of the total points on the curve and the order of G.

Encryption Process:

Step 1: Obtain the text to be sent.
Step 2: It converts to corresponding ASCII values.

Step 3: The ASCII value partition as [ASCII values, group size 1, this operation groups the ASCII values with size known by group size with no overlapping and the sublists that have size lesser than group size are left as it is without padding]

Step 4: Each group obtained from the above step is converted into big integer values taking base as 65536. From Digits [Group of ASCII values, 65536].

Step 5: Pad with 32 to the end of the list from the above step if the count of the above list is odd, to make it even for performing complete pairing. Every single pair will be an input to the ECC system as „Pm". It is a pad with 32 because 32 represent blank space in ASCII code.

Step 6: Choose random k value, k = Random value with range 1 to n−1. Calculate kG and kPb using Point multiplication operation.

Step 7: Compute Pm + kPb using point doubling or point addition as needed.

Step 8: Transmit Pc = {kG, Pm + kPb} as cipher text to the receiver side.

Decryption Process:

Step 1: The ciphertext Pc is obtained.

Step 2: The left part kG and right part of the Pc separately (Pm + kPb).

Step 3: Multiply with nB to the left part and subtract it from the right part to obtain Pm. {Pm + kPb} − nBkG = Pm since Pb = nBG.

Subtraction operation can be converted to addition by multiplying with −1 to the y coordinate. By applying point addition operation can be validated to obtain the mirror-image point over the x-axis.

Step 4: By forming a set of ASCII values, the above operation will provide the integer value and then convert it back to a list of ASCII values. Integer Digits [n, b] in Mathematics provides a set of the base b digits in the integer n. Digits function and Integer Digits are opposite to each other so that the ASCII values are secured during encryption and decryption.

Step 5: The list of ASCII values converts to its corresponding characters.

5. PROPOSED ENHANCED AUTHENTICATION PROCESS WITH OPTIMIZATION ALGORITHM

In this proposed DF-ECC algorithm is a public key cryptosystem where every user possesses two keys: a public key and private key. The public key is used for encryption and signature verification, while a private key is used for decryption and signature generation. This proposed Optimized ECC is composed of the following phases: (i) Key Generation: using the DF Optimization algorithm, (ii) Signature Generation, (iii) Encryption algorithm, (iv) Decryption algorithm, and (v) Signature Verification.

In the proposed DF-ECC algorithm, key values are optimally selected using the DF optimization algorithm. The public and private keys generated by the ECC method make the encrypted data transfer. The generation equation for defining the ECC is given equation (1). The key generation is an important part which has to generate both public key and private key. The sender will be encrypting the message with the receiver's public key, and the receiver will decrypt its private key. The key generation and formation are explained underneath:

It is the most important step in which an algorithm is used to generate both public and private keys. The sender encrypts the message data with the help of the receiver's public key, and the receiver decrypts the data using its private key.

Step 1: Key Generation by Dragon Fly Algorithm

Step 1.1: The random values are selected using the Dragon Fly Optimization algorithm. Initialize Maximum generation Max_g and intensity of light M_g Light Absorption coefficient is defined.

Step 1.2: While ($t<Max_g$)

Step 1.3: For g=1: n_1 for all DF.

Step 1.4: For h=1: n_2 for all DF

Step 1.5: IF ($M_h > M_g$)

Step 1.6: FF g is moved towards h

Step 1.7: End if

Step 1.8: Attractiveness varies with distance x.

Step 1.9: New solutions are evaluated, and light intensity is updated

Step 1.10: End for h

Step 1.11: End for g

Step 1.12: DF are ranked, and the best DF is predicted

Step 1.13: End while

Step 1.14: Similarly, the receiver selects a private key dB and generates its public key PB =dB*G.

Step 1.15: The sender generates the security key "K= dA*PB," and the receiver also generates the security key "K= dB*PA".

Step 2: Signature Generation

To sign a message m by the sender, it performs the following steps: -

Step 2.1: It calculates a cryptographic hash function, $e = hash\ (m)$.

Step 2.2: The sender then selects a random integer k from [1, n-1].

Step 2.3: Then, it computes a pair (r,s).

Step 2.4: $r = x1\ (\mathrm{mod}\ n)$ where $(x1, y1) = k * G$

Step 2.5: s= k-1(e+ dA*r)

Step 2.6: This pair (r,s) defines the signature.

Step 2.7: This signature is sent to the receiver.

Step 3: Encryption Algorithm

Suppose the sender wants to send a message m to the receiver.

Step 3.1: Let m has any point M on the elliptic curve.

Step 3.2: The sender selects a random number k from [1,n-1].

Step 3.3: The ciphertexts generated will be the pair of points (B1, B2) where

B1= k*G

B2 = M + (k*G)

Step 4: Decryption Algorithm

To decrypt the ciphertext, the following steps are performed: -

Step 4.1: The receiver computes the product of B1 and its private key.

Step 4.2: Then, the receiver subtracts this product from the second point B2.

$$M = B2 - (dB * B1)$$

Where M is the original data sent by the sender.

Step 5: Signature Verification

To authenticate the sender's signature, the receiver must know the sender's public key PA.

Step 5.1: For authentication, the receiver needs to verify the pair (r,s) are in the range of [1,n-1].

Step 5.2: The receiver again then calculates the hash function e as in signature generation.

Step 5.3: Then the receiver calculates w =s-1 mod(n).

Step 5.4: Then calculate u1= e*w (mod n) and u2 = r*w (mod n).

Step 5.5: Calculate (x1,y1)= u1*G + u2*PA.

Step 5.6: If x1 = r (mod n), then the signature is valid.

6. RESULT AND DISCUSSION

The performance metrics like Key Generation time (in milliseconds), key updation time (in milliseconds), Encryption time (in milliseconds), Decryption time (in milliseconds), Total time is taken for encryption and decryption time (in ms) and throughput (in mbps) for the encryption and decryption. The performance of the proposed Enhanced Authentication Process which combines the Dragon Fly and Elliptical Curve Cryptography (DF-ECC) with existing public key encryption techniques like ECC, RSA, and Digital Signature Algorithm (DSA).

Table 1 depicts the key generation time (in milliseconds) by proposed DF-ECC, ECC, RSA and DSA encryption techniques against the varying file size (in MB). From the table 1, it is clear that the proposed DF-ECC requires less key generation time than ECC, RSA and DSA algorithms for varying file sizes.

Table 1: Key Generation (in Milliseconds) by the Proposed DF-ECC, ECC, DSA and RSA

File Size (in MB)	Key Generation Time (in milliseconds)			
	Proposed DF-ECC	**ECC**	**DSA**	**RSA**
100	738	846	966	942
200	856	932	1017	996
300	947	1081	1173	1182
400	1139	1248	1328	1321
500	1326	1451	1586	1502
600	1468	1534	1644	1651
700	1648	1744	1868	1734
800	1781	1853	1954	1987
900	1933	2057	2134	2232
1000	2118	2372	2516	2419

Table 2 depicts the key updation time (in milliseconds) by proposed DF-ECC, ECC, RSA and DSA encryption techniques against the varying file size (in MB). From the table 2, it is clear that the proposed DF-ECC requires less time for updating the key than ECC, RSA and DSA algorithms for varying file sizes.

Table 2: Key Updation Time (in Milliseconds) by the Proposed DF-ECC, ECC, DSA and RSA

File Size (in MB)	Key Updation Time (in Milliseconds)			
	Proposed DF-ECC	**ECC**	**DSA**	**RSA**
100	742	817	941	984
200	855	971	1076	1181
300	1048	1036	1136	1221
400	1218	1387	1401	1524
500	1357	1415	1583	1592
600	1484	1520	1653	1614
700	1653	1778	1812	1898
800	1780	1971	2029	2089

900	1964	2198	2342	2448
1000	2109	2233	2520	2512

Table 3 depicts the encryption time (in seconds) by proposed DF-ECC, ECC, RSA and DSA encryption techniques against the varying file size (in MB). From the table 3, it is clear that the proposed DF-ECC requires less time for encryption than ECC, RSA and DSA algorithms for varying file sizes.

Table 3: Encryption Time (in Seconds) by the Proposed DF-ECC, ECC, DSA and RSA

File Size (in MB)	Encryption Time (in Seconds)			
	Proposed DF-ECC	**ECC**	**DSA**	**RSA**
100	121	144	223	286
200	157	207	286	354
300	213	292	362	427
400	281	323	397	433
500	365	485	624	726
600	411	594	678	823
700	562	678	741	915
800	617	772	853	1091
900	818	941	1081	1112
1000	923	1008	1191	1228

Table 4 depicts the decryption time (in seconds) by proposed DF-ECC, ECC, RSA and DSA encryption techniques against the varying file size (in MB). From the table 4, it is clear that the proposed DF-ECC requires less time for decryption than ECC, RSA and DSA algorithms for varying file sizes.

Table 4: Decryption Time (in Seconds) by the Proposed DF-ECC, ECC, DSA and RSA

File Size (in MB)	Decryption Time (in Seconds)			
	Proposed DF-ECC	ECC	DSA	RSA
100	131	228	293	334
200	167	303	369	415
300	234	352	427	481
400	296	413	527	537
500	344	483	581	617
600	425	546	662	682
700	492	634	749	802
800	559	706	832	861
900	657	764	978	996
1000	776	881	1035	1128

Table 5 gives the total time taken (in seconds) for encryption and decryption by proposed DF-ECC, ECC, RSA and DSA encryption techniques against the varying file size (in MB). From the table 5, it is clear that the proposed DF-ECC consumes less time for encryption and decryption than ECC, RSA and DSA algorithms for varying file sizes.

Table 5: Total Time Taken for Encryption and Decryption (in Seconds) by the Proposed DF-ECC, ECC, DSA and RSA

File Size (in MB)	Total Time taken for Encryption and decryption (in Seconds)			
	Proposed DF-ECC	ECC	DSA	RSA
100	252	372	516	620
200	324	510	655	769
300	447	344	789	908
400	577	736	924	970
500	709	848	1205	1343
600	836	1140	1340	1505
700	1054	1312	2204	1717
800	1176	1418	1685	1952
900	1475	1705	2059	2108
1000	1699	1889	2226	2356

Table 6 gives the Throughput (in mbps) for encryption and decryption by proposed DF-ECC, ECC, RSA and DSA encryption techniques against the varying file size (in MB). From the table 6, it is clear that the proposed DF-ECC increased throughput ECC, RSA and DSA algorithms for varying file sizes.

Table 6: Throughput (in mbps) obtained by the Proposed DF-ECC, ECC, DSA and RSA

File Size (in MB)	Throughput (in mbps)			
	Proposed DF-ECC	ECC	DSA	RSA
100	1291	1028	935	922
200	1319	1181	991	997
300	1392	1249	1023	1055
400	1437	1385	1158	1182
500	1578	1431	1233	1248
600	1613	1577	1352	1343
700	1747	1623	1413	1409
800	1877	1767	1545	1541
900	1937	1811	1620	1606
1000	2086	1958	1754	1743

7. CONCLUSION

Authentication is one of the main important challenges in security of cloud computing. Single level authentication has many problems mainly with sensitive data, as passwords are easy to break. In this paper, an enhanced encryption technique has proposed to ensure security among

the data communication between the cloud servers. This proposed technique is the combination of Dragon Fly optimization and ECC algorithm. The random number for generating the secret keys by the ECC is calculated with the DF algorithm. The proposed DF-ECC technique performed better than the other existing techniques like RSA, ECC, and DSA. The performance of the proposed DF-ECC is evaluated with a key generation time, key updation time, time is taken for encryption and decryption and its total time is taken and throughput. From the results obtained, it is evident that the proposed DF-ECC method performs better than existing techniques.

REFERENCES

[1] Bohn, Robert B., Craig A. Lee, and Martial Michel. "The NIST cloud federation reference architecture." (2020).

[2] Tabrizchi, Hamed, and Marjan Kuchaki Rafsanjani. "A survey on security challenges in cloud computing: issues, threats, and solutions." The journal of supercomputing 76.12 (2020): 9493-9532.

[3] Singh, Jaswinder, and Gaurav Dhiman. "A survey on cloud computing approaches." Materials Today: Proceedings (2021).

[4] Sunyaev, Ali. "Cloud computing." Internet computing. Springer, Cham, 2020. 195-236.

[5] Wulf, Frederik, et al. "IaaS, PaaS, or SaaS? The Why of Cloud Computing Delivery Model Selection." (2021).

[6] Anandhi, S., R. Anitha, and Venkatasamy Suresh kumar. "An authentication protocol to track an object with multiple RFID tags using cloud computing environment." Wireless Personal Communications 113.4 (2020): 2339-2361.

[7] Liu, Chia-Hui, et al. "A reliable authentication scheme of personal health records in cloud computing." Wireless Networks (2021): 1-11.

[8] Veerabathiran, Vijaya Kumar, et al. "Improving secured ID-based authentication for cloud computing through novel hybrid fuzzy-based homomorphic proxy re-encryption." Soft Computing 24.24 (2020): 18893-18908.

[9] Anuradha, M., et al. "IoT enabled cancer prediction system to enhance the authentication and security using cloud computing." Microprocessors and Microsystems 80 (2021): 103301.

[10] Velmurugadass, P., et al. "Enhancing Blockchain security in cloud computing with IoT environment using ECIES and cryptography hash algorithm." Materials Today: Proceedings 37 (2021): 2653-2659.

[11] Shi, Canghong, et al. "A novel NMF-based authentication scheme for encrypted speech in cloud computing." Multimedia Tools and Applications (2021): 1-26.

[12] Padmaja, K., and R. Seshadri. "A real-time secure medical device authentication for personal E-Healthcare services on cloud computing." International Journal of System Assurance Engineering and Management (2021): 1-11.

[13] Safkhani, Masoumeh, et al. "RSEAP2: An enhanced version of RSEAP, an RFID based authentication protocol for vehicular cloud computing." Vehicular Communications 28 (2021): 100311.

[14] Kumar, Vinod, et al. "RSEAP: RFID based secure and efficient authentication protocol for vehicular cloud computing." Vehicular Communications 22 (2020): 100213.

[15] Joseph, Teena, et al. "A multimodal biometric authentication scheme based on feature fusion for improving security in cloud environment." Journal of Ambient Intelligence and Humanized Computing 12.6 (2021): 6141-6149.

[16] Saranya, A., and R. Naresh. "Cloud based efficient authentication for mobile payments using key distribution method." Journal of Ambient Intelligence and Humanized Computing (2021): 1-8.

[17] Goumidi, Hadjer, et al. "Lightweight Secure Authentication and Key Distribution Scheme for Vehicular Cloud Computing." Symmetry 13.3 (2021): 484.

[18] Shirani, Mohammad Reza, and Faramarz Safi-Esfahani. "Dynamic scheduling of tasks in cloud computing applying dragonfly algorithm, biogeography-based optimization algorithm and Mexican hat wavelet." Journal of Supercomputing 77.2 (2021).

[19] Khan, Mohammad Ayoub, et al. "A secure framework for authentication and encryption using improved ECC for IoT-based medical sensor data." IEEE Access 8 (2020): 52018-52027.