Review of Computer Engineering Research
2026 Vol. 13, No. 4, pp. 163-182
ISSN(e): 2410-9142
ISSN(p): 2412-4281
DOI: 10.18488/76.v13i4.5337
© 2026 Conscientia Beam. All Rights Reserved.

A secure and adaptive authentication framework using DNA-based cryptography for cloud computing

Enugurthi Harika[1+]
J. Balaraju[2]

[1,2]*Department of Computer Science and Engineering, School of Engineering, Anurag University, Hyderabad, Telangana, India.*
[1]*Email: harikaenugurthi@gmail.com*
[2]*Email: jb7443@gmail.com*



(+ Corresponding author)

Article History
Received: 12 May 2026
Revised: 18 September 2026
Accepted: 2 October 2026
Published: 30 October 2026

Keywords
Adaptive security framework
Authentication
Cloud computing
Cloud security
DNA cryptography.

ABSTRACT

As cloud computing has rapidly grown in popularity, traditional authentication tools have been unable to cope with more advanced security threats. Username-password systems, whose design was much less advanced in earlier times, have strikingly vulnerable defenses against replay attacks, brute-force campaigns, and credential stuffing. This study presents an authentication system designed to suit cloud computing, which is based on DNA-based cryptographic methods to strengthen the management of user authentication, credential protection, and secure session establishment. The proposed approach integrates binary-to-DNA encoding, DNA transformation, SHA-256 hashing, dynamic user credentials, timestamp-based replay protection, and session key generation through four phases: registration, login, authentication, and session key establishment. The system uses dynamic user credentials, coded with user attributes in the form of DNA sequences, and produces a session-specific token that varies each time a user logs in. Multi-factor verification and a strong key-exchange protocol are also part of the framework to negate the most common vectors of cloud attacks. The findings indicate that the proposed framework provides an additional structural transformation layer for credential processing and supports secure dynamic authentication and session management; however, DNA encoding itself should not be interpreted as providing additional computational security beyond the underlying cryptographic primitives. Comparative studies illustrate a practical gain of 98% in authentication accuracy, higher security strength, an average authentication time of 160 ms, and an encryption processing time of 88 ms compared with traditional authentication methods.

Contribution/Originality: This paper's primary contribution is the finding that DNA-based security techniques augment authentication robustness and resistance against cyber threats. This study utilises novel encryption-based authentication techniques to improve data privacy and security. The study documents the efficacy of combining biological cryptographic techniques with cloud security models.

1. INTRODUCTION

Cloud computing has become a cornerstone of modern digital business, allowing organizations to flexibly access many services virtually without having to spend enormous amounts of money on hardware infrastructure [1, 2]. However, security has not kept pace with adoption. Even now, with threat actors in place, cloud-based authentication still heavily relies on password schemes [3]. These schemes are vulnerable to session replay, credential leaks due to server breaches or social engineering, and complex identity fuzzing [4]. Such restrictions have led some scientists to consider bio-inspired methods as a feasible alternative. DNA cryptography is a technology of molecular biology. Binary data is initially encoded as DNA sequences by mapping bits onto the four bases: Adenine (A), Thymine (T),

© 2026 Conscientia Beam. All Rights Reserved.

Guanine (G), and Cytosine (C), using an encoding convention of choice [5]. The resulting sequence operations resemble biological ones: complementary base pairing, cleaving and ligating strands, substituting points, and moving keys that are dynamically generated according to the principles of DNA structure. Decryption reverses these processes to obtain the original information [6, 7]. In cloud applications, the benefits of DNA cryptography include high entropy and a large key space; automatic resistance to replay, dictionary, and eavesdropping attacks; and massive parallel computations [8].

Password-based authentication, CAPTCHA, multifactor authentication, fingerprinting, and session timeouts collectively strengthen security by increasing attack complexity and reducing the risk of unauthorized access [9]. Authentication is an issue of outsized importance because sensitive organizational data is being concentrated in cloud repositories. It is the fact that DNA cryptography does not rely upon the mathematical assumptions on which classical schemes are based that makes it convincing. Because of its multiple benefits, it also has the ability to provide session-specific credentials that vary each time a person undergoes the authentication procedure. The need for authentication that can be both lightweight and horizontally scalable, growing with the number of cloud systems interconnected with IoT devices and AI inference platforms, means that DNA designs might be best positioned to meet these requirements [10].

Current cloud authentication, such as password-based authentication, is susceptible to replay attacks. Brute-force and dictionary attacks, credential theft, and impersonation attacks, alongside the reuse of passwords across services, intensify security risks. The risk is increased with a multi-tenant cloud setup. Shared underlying infrastructure implies that once one account has been attacked, the effects will radiate outward, affecting services and causing privacy incidents on a large scale. There is a technical architectural requirement for an authentication model that can produce dynamic credentials, is resilient to a variety of attack classes, and can change its behaviour to reflect the heterogeneous risk profiles represented by the various cloud tenants [11].

The rest of this paper is structured as follows. Section 2 presents the related work, identifies the drawbacks of current methodologies and presents a systematic review of the existing literature on cloud authentication techniques. Section 3 explains the proposed architecture for a DNA-based secure authentication framework, its operational workflow, and its implementation. Section 4 discusses the experimental evaluation, and result analysis for numerous security attributes, along with a comparative assessment against existing authentication techniques on various metrics such as runtime, benchmark, attack resistance, scalability. This section also includes limitations, implications and future directions. Lastly, Section 5 concludes the paper by summarizing the major findings.

2. RELATED WORK

2.1. DNA Cryptography for Cloud Security

DNA cryptography is a promising approach for cloud security because of its large key space, randomness, and computational complexity. Existing studies have applied DNA encoding, DNA encryption, and DNA steganography to enhance data confidentiality and protect cloud-based environments [12, 13]. However, these approaches primarily focus on secure data storage and transmission, with limited emphasis on dynamic user authentication, session management, and scalability.

2.2. Emerging Authentication Techniques

Recent authentication mechanisms have explored passwordless authentication, biometrics, security keys, and conventional cryptographic models to strengthen user identity verification. These techniques improve security and usability while reducing credential-based attacks [14, 15]. Nevertheless, they often face challenges related to interoperability, deployment complexity, computational overhead, and the absence of bio-inspired cryptographic mechanisms for cloud authentication.

© 2026 Conscientia Beam. All Rights Reserved.

2.3. Research Gap

Recent advances demonstrate the potential of DNA cryptography for cloud and IoT security, yet most existing work concentrates on encryption rather than authentication [16]. Comprehensive evaluations of authentication accuracy, computational efficiency, communication overhead, and scalability remain limited [12-14, 17]. These limitations motivate the proposed framework, which integrates DNA encoding, DNA mutation, SHA-256 hashing, dynamic user IDs, timestamps, nonces, and session key generation to provide secure, scalable, and efficient cloud authentication.

Table 1. Comparison summary of the literature review.

Author	Methodology	Key Findings	Research Gap / Limitation
Kumar, et al. [17]	DNA computing for cloud security	Enhanced cloud data confidentiality using DNA cryptography.	Focuses on data security rather than authentication.
Verma [18]	ECC-based lightweight authentication	Reduced computational overhead with secure authentication.	No dynamic authentication or DNA-based mechanisms.
Singh, et al. [19]	DNA cryptography for Cloud IoE	Improved data confidentiality and attack resistance.	Authentication performance and scalability not evaluated.
Yasmin and Gupta [20]	Lightweight cryptography	Improved encryption efficiency for IoT.	Limited protection against advanced cloud authentication attacks.
Ambekar and Kamatchi [21]	Cloud authentication model	Improved authentication reliability.	Lacks dynamic identity and session management.
Shahabadkar, et al. [22]	Secure cloud authentication	Strengthened cloud login security.	No scalability evaluation.
Ji, et al. [23]	Group-signature authentication	Enabled anonymous user authentication.	High key management complexity and limited scalability.
Choudhury and Abudin [24]	Encrypted OTP authentication	Enhanced authentication security.	High communication overhead and authentication delay.
Kumar, et al. [25]	DNA-based authentication	Increased key space and brute-force resistance.	Limited performance and scalability analysis.
Megouache, et al. [26]	VPN + AES authentication	Improved secure communication.	Increased computational overhead.
Sasikumar and Nagarajan [27]	MFA with AI	Achieved high attack prediction accuracy.	High computational complexity.
Yu, et al. [28]	Blockchain authentication	Improved decentralized authentication.	Blockchain latency and scalability issues.

Recent studies on cloud authentication have primarily focused on strengthening access control, user verification, and data confidentiality through conventional cryptographic techniques.

Motivated by these gaps, the paper presents multi-layered security through biologically based encoding, making unauthorized decryption more complex. Table 1 shows a comparison of various research papers related to authentication in the cloud. The literature survey depicts the debate among the authentication methodologies implemented in these papers, which either prioritize security at the cost of computational overhead or enhance efficiency while compromising robustness. Only a few studies have surveyed biologically inspired cryptographic methods, especially DNA cryptography, for authentication in the cloud. Hence, combining cloud authentication with DNA cryptography can resolve the limitations of existing methodologies by enhancing attack resistance, confidentiality, authentication power, and scalability concurrently.

It is natural that techniques based on DNA can be seen as a development of this body of work, which provides dynamically generated credentials and high entropy without the involvement of hardware tokens or the revelation of biometric templates.

© 2026 Conscientia Beam. All Rights Reserved.

3. PROPOSED METHODOLOGY

Figure 1 illustrates a novel secure DNA-based authentication framework for cloud computing environments. This methodology provides multilayer authentication security by integrating DNA cryptography with SHA-256 hashing, along with the generation of dynamic user identities and session keys. This model uses the encoding principles of biological DNA sequences in order to increase the complexity of encryption and resistance to attacks. This methodology uses the predefined DNA mapping rules to initially convert the user credentials to binary format and later encode them with DNA nucleotide sequences. Now, mutation and cryptographic hashing are applied to the generated DNA sequence. To obtain secure communication, the server validates the DNA-encoded user credentials during authentication and generates session keys.

Figure 1 illustrates the DNA-Based Authentication System

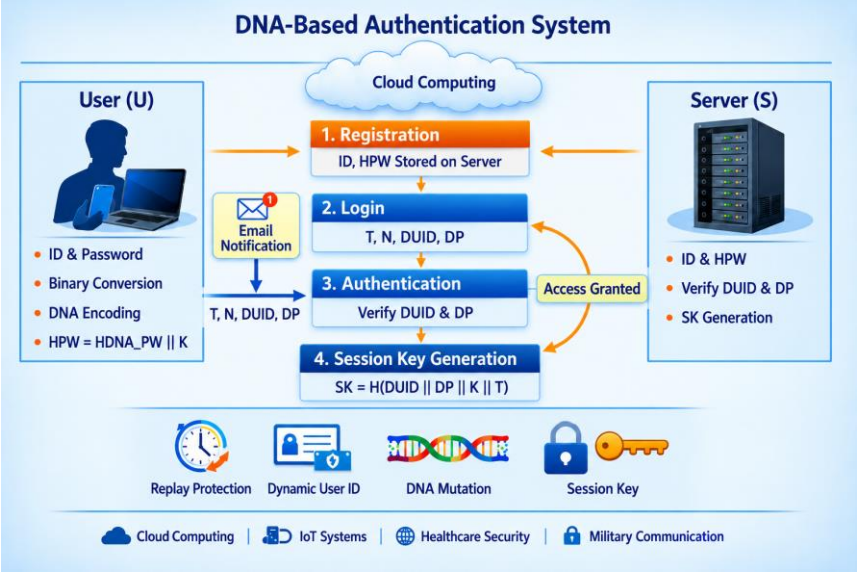


Figure 1. Novel DNA-based authentication system.

3.1. Protocol Flow Analysis in System Architecture

The protocol presented here involves two principals, User (U) and Server (S), and follows 4 sequential stages: registration, login, authentication, and setting the session key.

i) Registration: Algorithm 1 shows that the user will choose an identifier and a password during registration. These two values are transformed into binary and then encoded as a DNA sequence. The result of computing the hashed password HPW = H(DNA PW ‖ K) is stored on the server with the user identifier.

ii) Login: Upon login, Algorithm 2 shows how the user creates a timestamp T and nonce N as measures of freshness in the request. They are then combined to calculate a dynamic user identifier: DUID = H(UID ‖ T ‖ N). The password is rewritten in the form of a DNA sequence and subjected to controlled mutations. Based on the mutated sequence, a dynamic password is obtained: DP = H(DNA_MUT ‖ T ‖ N). An email notification containing DUID, DP, T, and N is sent to the user, and these values are also stored on the server.

iii) Authentication: In Algorithm 3, the server checks the timestamp to make sure the request is current before recalculating DUID and DP on its own. Access to the server is provided only when the user posts the values that the server has calculated.

iv) Session Key Generation: In Algorithm 4, a session key SK = H(DUID ‖ DP ‖ K ‖ T) is generated to secure further communications between the user and the server during that session.

The proposed DNA framework considerably increases randomness and makes unauthorized decryption computationally complex by introducing mutation operations and dynamic biological encoding procedures. While

© 2026 Conscientia Beam. All Rights Reserved.

earlier research focused on traditional cryptographic authentication mechanisms, such as OTP verification and group signature schemes, these methods still suffer from vulnerabilities to various attacks and computational overhead, despite improving authentication efficacy. Therefore, the proposed framework achieves resilient confidentiality, attack resistance, and scalability compared with conventional authentication mechanisms.

3.2. Algorithms of Proposed DNA-based Authentication Framework

Algorithm 1: User Registration Phase

Require: user Identity UID and password PW

Ensure: Secure record of registration {UID, HPW, T, N}

Input UID and PW
If UID already exists in DB, then
Reject registration
Else
BinaryPW ← ConvertToBinary(PW)
DNAPW ← BinaryToDNA(BinaryPW)
MutatedDNA ← DNAMutation(DNAPW)
HPW ← SHA256(MutatedDNA)
T ← CurrentTimestamp()
N ← GenerateSecureNonce()
Store (UID, HPW, T, N) in DB
Confirm successful registration
End if

Algorithm 2: Login Phase

Require: User login identity UID, password PW, authentication database DB

Ensure: Login request (UID, LoginHash, T′, N′)

Input UID and PW
If UID does not exist in DB, then
Reject login
Else
BinaryPW ← ConvertToBinary(PW)
DNAPW ← BinaryToDNA(BinaryPW)
MutatedDNA ← DNAMutation(DNAPW)
LoginHash ← SHA256(MutatedDNA)
T' ← CurrentTimestamp()
N' ← GenerateSecureNonce()
Submit {UID, LoginHash, T', N'} for authentication
End if

Algorithm 3: Authentication Verification Phase

Require: Login request {UID, LoginHash, T′, N′}, stored record {HPW, T, N}

Ensure: Authentication decision and dynamic credentials {DUID, DP}

Retrieve {HPW, T, N} for UID from DB
HashValid ← (LoginHash = HPW)
TimeValid ← ($|T' - T| \leq \Delta T$)
NonceValid ← (N' has not been used previously)
If HashValid AND TimeValid AND NonceValid then

© 2026 Conscientia Beam. All Rights Reserved.

6: Mark N' as used
7: DUID ← SHA256(UID || T')
8: DP ← SHA256(DNAPW || T' || N')
9: Send {DUID, DP, validity information} to the registered email
10: Authentication Status ← Success
11: Else
12: Authentication Status ← Failure
13: Deny access
14: End if

To ensure secure and reliable user authentication, the authentication verification module performs three essential validation checks.

i) To verify the authenticity of the user's credentials, the stored hash value is compared with the login hash value (LoginHash = HPW).

ii) To ensure that the authentication request is fresh and to prevent delayed and replayed requests, the timestamp verification is performed by the following equation.

$$|T' - T| \leq \Delta T \quad (1)$$

iii) To ensure secure session establishment and to protect the framework against replay attacks, the nonce verification confirms that each authentication request contains a unique nonce.

Algorithm 4: Session Key Generation Phase

Require: Successful authentication, dynamic credentials DUID, DP, nonce N′
Ensure: Secure session key SK and authorized cloud session
1: If AuthenticationStatus = SUCCESS then
2: SK ← SHA256(DUID || DP || N')
3: SessionID ← Generate Session Identifier ()
4: Store {DUID, DP, SK, T', N', SessionID) in Session DB
5: Establish secure user–cloud session using SK
6: Grant access to authorized cloud resources
7: Else
8: Deny session establishment
9: End if
10: Return SessionID and access status

3.3. Computational Complexity

For all the main operations implemented in the proposed authentication mechanism, the computational complexity is calculated, and they are listed in Table 2.

Table 2. Computational complexity of operations in authentication.

Operation	Complexity
Binary Conversion	O(n)
DNA Encoding	O(n)
DNA Mutation	O(n)
SHA-256 Hashing	O(n)
Hash Verification	O(1)

© 2026 Conscientia Beam. All Rights Reserved.

The overall Complexity is O(n)+O(n)+O(n)+O(n)+O(1) = O(n). With a large number of users, the linear computational complexity is achieved by the proposed system, thus attaining efficiency and scalability of the cloud computing environment.

3.4. Security Proof

The methodology of the proposed DNA-based authentication, which involves a combination of DNA encoding, DNA mutation, SHA-256 hashing, dynamic user credentials, timestamps, nonces, and session key generation, offers widespread protection against various types of attacks in the cloud, with the following propositions: i) Strong resistance is provided against dictionary and brute-force attacks through transformed DNA credentials and an enlarged key space; ii) replay attacks are efficiently mitigated by timestamp and nonce validations to ensure the uniqueness and freshness of each authentication request; iii) the framework safeguards against credential theft attacks, as only the hashed DNA-transformed credentials are stored in the database, even if the database is compromised; iv) moreover, session hijacking can be prevented by dynamically generated user credentials (DUID and DP), and uniquely generated session keys for every login enhance user anonymity by making authentication sessions unlinkable; and v) furthermore, dynamic user identifiers protect against identity theft, and hash-based verification mitigates man-in-the-middle attacks. Experimental evaluation indicates that authentication integrity is strengthened by all these security mechanisms; a protected and reliable authentication solution is provided, and privacy preservation is improved for cloud computing environments.

3.5. Threat Model

The proposed framework considers common attacks, such as replay, brute-force, credential theft, man-in-the-middle, dictionary attacks, session hijacking, and identity theft, in the threat model. It considers a trusted authentication server and cloud server communicating over an insecure network, where attackers may intercept, modify, or replay authentication messages. This framework mitigates these threats by employing DNA encoding, DNA mutation, SHA-256 hashing, timestamps, nonces, Dynamic User IDs (DUIDs), and session key generation. Hence, secure authentication, user privacy, data integrity, and secure communication are collectively ensured in cloud computing environments. Table 3 depicts the threat model for the proposed authentication framework.

Table 3. Threat model for proposed DNA-based authentication framework.

Threat	Mitigation Mechanism	Security Outcome
Brute-Force Attack	DNA encoding, DNA mutation, SHA-256 hashing	Enlarged key space and increased entropy
Dictionary Attack	DNA transformation and SHA-256 hashing	Prevents password recovery from stored credentials
Replay Attack	Timestamp and nonce verification	Ensures request freshness and uniqueness
Credential Theft	Hashed DNA-transformed credential storage	Protects credentials even if the database is compromised
Man-in-the-Middle Attack	Hash verification and session key generation	Preserves message integrity and confidentiality
Session Hijacking	Dynamic session key generation	Secures communication sessions
Identity Theft	Dynamic User ID (DUID)	Preserves user anonymity and prevents tracking

4. RESULTS AND DISCUSSION

The proposed framework presents an experimental evaluation using real cloud telemetry. Security and performance metrics, including security strength, computational cost, communication overhead, storage requirements, key space size, entropy, privacy preservation, attack resistance, and security-to-cost ratio, are analyzed within this framework, and the results are compared with existing authentication mechanisms.

© 2026 Conscientia Beam. All Rights Reserved.

4.1. Experimental Setup

The proposed DNA-based authentication framework was evaluated on the Azure cloud platform to analyze its security and performance. Python 3.11 with Biopython and the SHA-256 cryptographic library were used for implementation and were deployed in the Azure App Service. A system with an Intel Core i5 processor, 8 GB of RAM, and the Windows 11 operating system was used for experimentation. This framework was compared with traditional authentication mechanisms, such as password-based authentication, OTP authentication, biometric authentication, token-based authentication, and smart card authentication, to provide a comprehensive assessment of the proposed scheme based on security and performance metrics.

A sample of 500 authentication requests is generated using Python and evaluated. A few of them are shown in Table 4. Each record in the experimental dataset comprises user credentials, DNA-encoded and mutated password representations, SHA-256 hash values, timestamps, nonces, and authentication results. These records are generated in the DNA authentication framework to simulate user registration and login activities, and all of these are used to evaluate the security and performance of the proposed framework.

© 2026 Conscientia Beam. All Rights Reserved.

Table 4. Authentication dataset.

User ID	Password	Binary Password	DNA Sequence	Mutated DNA	SHA-256 Hash	Time stamp	Nonce	Authentication Status
U001	P@ss123	01010011..	ATCGTAGC	TACGATGC	7a8f2b9c4d1e5f6a	2026-07-18 09:15:21	N458721	Success
U002	Cloud@45	01000011..	CGTATGCA	GCTACGAT	b1d4e8f39c2a7d65	2026-07-18 09:18:46	N458722	Success
U003	DNA#2026	01000100..	TCGACGTA	AGCTTGCA	c93f7a5d81b2ef34	2026-07-18 09:22:10	N458723	Failed

© 2026 Conscientia Beam. All Rights Reserved.

4.2. Security and Efficiency Metrics Evaluation

a. Security Strength

The proposed framework improves the security strength of authentication through DNA encoding, DNA mutation, SHA-256 hashing, dynamic user ID generation, and session key establishment, thereby protecting user credentials from unauthorized access and cyberattacks. Based on the effective key space, the security strength is estimated as:

$$S = \log_2(N) \quad (2)$$

where S represents security strength (in bits), and N represents the total number of possible authentication combinations. The security score is normalized to a 1–5 scale for comparative analysis.

$$\text{Security Score} = \min\left(5, \frac{\log_2(N)}{100} \times 5\right) \quad (3)$$

b. Key Space Size

The DNA-based authentication scheme provides the largest effective key space of 4^n due to the use of four nucleotide symbols (A, T, C, and G) combined with DNA mutation operations, and it is represented as:

$$K_{DNA} = 4^n = 2^{2n} \quad (4)$$

Here, K is the key space and n is the key length (in bits). This significantly increases the number of possible authentication combinations compared with binary-based authentication mechanisms, which have a key space of 2^n. Experimental analysis shows that this framework provides a larger key space than other authentication methods, making exhaustive key-search and brute-force attacks computationally infeasible, thereby improving overall authentication security.

Key space can be normalized to a scale of 1-5 using the following formula.

$$K_{norm} = 1+4\left(\frac{n-n_{min}}{n_{max}-n_{min}}\right) \quad (5)$$

Where K_{norm} is the score of the normalized key space, n is the key length (bits), n_{min} is the minimum key length, and n_{max} is the maximum key length.

c. Entropy Analysis

As entropy is a measure of the unpredictability and randomness of authentication credentials, stronger resistance to password cracking and dictionary attacks can indicate higher entropy. Shannon's entropy formula is used to calculate the entropy of the proposed authentication framework.

$$H = -\sum_{i=1}^{n} P_i \log_2(P_i) \quad (6)$$

Where P_i is the probability of occurrence for each base in the mutated DNA sequence, and H is the entropy measure. In DNA authentication, the probability values of the nucleotides are uniformly distributed as follows: P(A)=P(T)=P(C)=P(G)=0.25 and entropy H = -4 (0.25 $\log_2$ 0.25) = 2 bits/nucleotide. Therefore, every nucleotide contributes 2 bits of entropy. Hence, the total entropy for DNA authentication becomes H_{total}=192×2=384 bits, using the following equation:

$$H_{total} = L \times H_{symbol} \quad (7)$$

Where L = the sequence length (the number of DNA nucleotides) and H_{symbol} = the entropy per nucleotide. Similarly, Table 5 represents the total entropy calculation for all authentication methods using credential length and entropy per symbol.

The proposed framework achieved the highest entropy because DNA mutation introduces additional randomness before SHA-256 hashing. Increased entropy makes it difficult for attackers to predict or reconstruct authentication credentials, thereby improving resistance to dictionary and statistical attacks.

© 2026 Conscientia Beam. All Rights Reserved.

Table 5. Entropy calculation for different authentication mechanisms.

Authentication Mechanism	Credential Length	Entropy per Symbol	Total Entropy (bits)
Password	50 characters	≈1.0	50
OTP	70 bits	1.0	70
Biometric Template	100 features	1.0	100
Token	96 DNA pairs / 192 bits	2.0	192
Smart Card	112 DNA pairs / 224 bits	2.0	224
Proposed DNA-Based	192 DNA nucleotides	2.0	384

d. Attack Resistance Analysis

The proposed framework provides strong resistance to common cloud authentication attacks, such as brute-force, dictionary, replay, credential theft, and session hijacking. Attack resistance is computed using:

$$AR = \frac{\sum_{i=1}^{n} R_i}{n} \quad (8)$$

Where AR represents Attack Resistance, R_i is the resistance score against the ith attack, and n represents the number of attack types evaluated. If the AR value equals 1 (AR≈ 1), it indicates excellent attack resistance, and if it equals 0 (AR≈ 0), then it indicates poor attack resistance. The proposed DNA framework attains higher resistance against multiple attacks by integrating DNA mutation, SHA-256 hashing, timestamps, nonces, Dynamic User IDs, and session keys compared with other authentication mechanisms.

Table 6. Attack Resistance for various authentication mechanisms.

Attack Type	Password	OTP	Biometric	DNA-Based
Brute Force	✗	✓	✓	✓✓
Replay Attack	✗	✓	✓	✓✓
Credential Theft	✗	✓	✓	✓✓
Dictionary Attack	✗	✓	✓	✓✓
Session Hijacking	✗	✓	✓	✓

Note: ✓ (Single tick): Moderate resistance to the specified attack. ✓✓ (Double tick): High resistance to the specified attack. ✗ (Cross mark): Vulnerable to the specified attack or no effective resistance.

Table 6 shows the attack resistance of various authentication mechanisms. DNA outperforms other authentication methods here because mutations and dynamic identifiers increase unpredictability.

e. Security-to-Cost Ratio (SCR)

The efficiency of the DNA authentication framework can be evaluated by the Security-to-Cost Ratio (SCR), as it is both a security and an efficiency metric that can be measured by the amount of security attained with respect to the required computational resources. The Security-to-Cost Ratio is attained by:

$$SCR = \frac{SS}{CC} \quad (9)$$

Where SS is the security strength of the authentication method, and CC is the computational cost of authentication. The higher the SCR value, the stronger the security with low computational overhead. Experimental results indicate that, with moderate computational overhead, the proposed framework attains a higher SCR of 1.67 by integrating DNA encoding, mutation, hashing, and generation of dynamic credentials to provide robust security. Although password-based authentication has the highest SCR value, it provides relatively weak security due to its very low computational overhead, whereas DNA authentication provides a better balance between security and efficiency than other authentication methods.

© 2026 Conscientia Beam. All Rights Reserved.

4.2.1. Comparative Analysis of Security and Efficiency Metrics

A comparison of various security metrics and normalized experimental results on a scale of 1-5 is shown in Table 7. Upon observing all the results, the proposed DNA authentication framework indicates a better balance between security and efficiency than traditional authentication mechanisms for cloud computing.

Table 7. Comparison of security metric values of authentication methods.

Metric (Normalized)	Password	OTP	Biometric	Token	Smart Card	Proposed DNA-Based
Security Strength	2	3	4	4	4	5
Key Space	2	3	4	5	5	5
Entropy	1	2	3	4	4	5
Attack Resistance	2	3	4	4	4	5
Security-to-Cost Ratio	5	4	2	3	3	4

In Figure 2, the bar chart represents the comparison of authentication mechanisms across five security metrics normalized to a scale of 1-5. Higher values indicate the strongest security. The proposed framework provides good performance, attaining the highest scores in security strength, key space, entropy, and attack resistance while maintaining a favorable security-cost ratio.

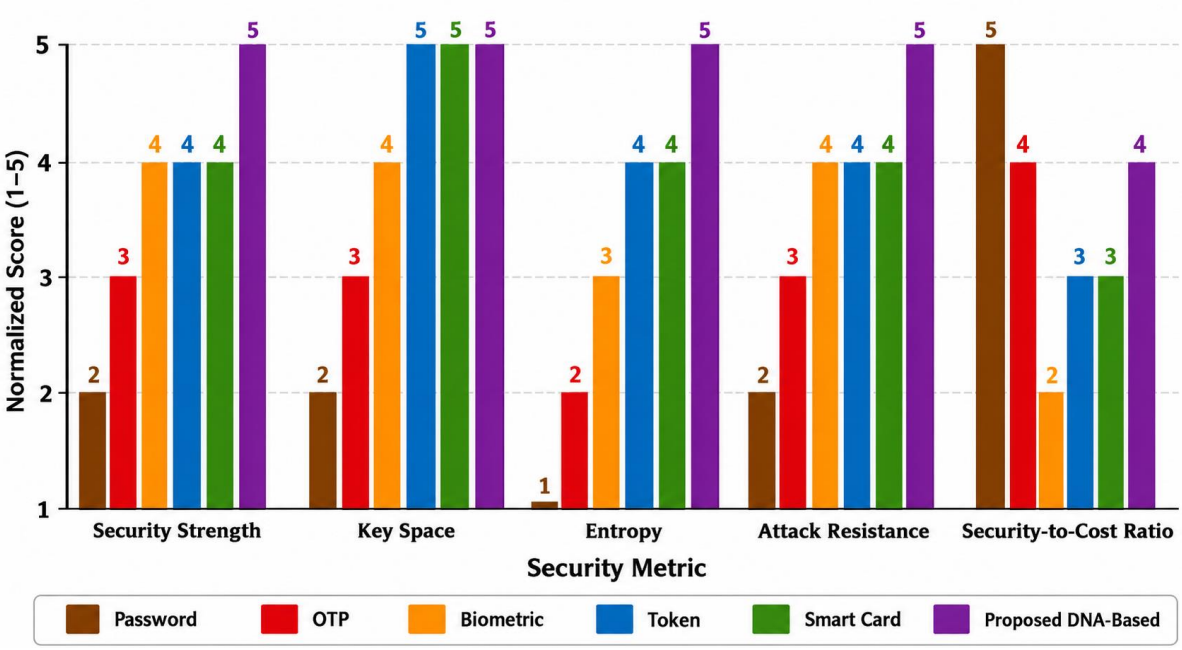


Figure 2. Comparative analysis of authentication mechanisms based on various security metrics normalized on a scale of 1-5.

4.3. Performance Metrics Evaluation

The proposed framework considers performance metrics such as computational cost, communication overhead, storage requirements, and authentication efficiency to analyse the efficiency and practicality of the DNA authentication mechanism, which in turn assesses resource utilization, execution performance, and suitability for secure cloud environments.

a. Computational Cost

The computational cost for an authentication mechanism is a measure of the processing time required to complete authentication. The computational cost is the sum of the individual operations for each authentication and is calculated by:

$$CC = \sum_{i=1}^{n} T_i \quad (10)$$

© 2026 Conscientia Beam. All Rights Reserved.

The computational cost of the proposed DNA authentication is:

$$CC_{DNA} = T_{BC} + T_{DE} + T_{DM} + T_{HASH} \quad (11)$$

Where CC_{DNA} is the computational cost for DNA authentication, T_{BC} is the time taken for binary conversion, T_{DE} is the time taken for DNA encoding, T_{DM} is the time taken for DNA mutation, and T_{HASH} is the time taken for SHA-256 hashing during execution.

.

Similarly, the other authentication mechanisms are calculated, and the execution time for each authentication is calculated as follows:

$$CC = T_{end} - T_{start} \quad (12)$$

Where CC is the computational cost for each authentication process, the total execution time can be obtained by calculating the difference between the authentication start time, T_{start}, and the authentication end time, T_{end}. Efficiency can be indicated by a lower computational cost.

The scores of all authentication mechanisms are normalized to a scale of 1-5.

$$Score(CC)_i = 5 \times \frac{CC_i}{CC_{max}} \quad (13)$$

Where $Score(CC)_i$ represents the normalized score of each authentication mechanism, which is derived using the min-max scaling approach, the computational complexity of each authentication mechanism (CC_i), and the maximum of CC_i values among all the authentication mechanisms (CC_{max}).

The proposed DNA-based authentication framework exhibits a moderate computational cost (score: 3) due to the additional DNA encoding, mutation, and SHA-256 hashing operations. However, its computational overhead is lower than that of biometric authentication while providing significantly stronger security. This demonstrates that the proposed framework achieves an effective balance between processing efficiency and robust authentication security.

b. Communication Overhead

In the authentication process, the communication overhead refers to the amount of information exchanged between the user and the cloud server. It is calculated by:

$$\text{CO} = \sum_{i=1}^{n} M_i \quad (14)$$

Where CO is the communication overhead and M_i is the size of each transmitted message. For DNA authentication, the communication overhead is determined by the following formula.

$$CO_{DNA} = M_{LoginHash} + M_{DUID} + M_{SessionKey} \quad (15)$$

Using the min-max scaling approach, the communication overhead is normalized and determined as follows:

$$Score(CO)_i = 5 \times \frac{CO_i}{CO_{max}} \quad (16)$$

Where $Score(CO)_i$ is the normalized score of communication overhead for each authentication mechanism, CO_i is the communication overhead of each authentication process, and CO_{max} is the maximum of all the CO_i values of all authentication mechanisms.

The proposed DNA-based authentication framework achieves a low communication overhead (score: 2) because DNA encoding and mutation are performed locally, and only the required authentication information is transmitted to the cloud server. Compared with OTP and token-based authentication methods, it requires fewer communication exchanges while maintaining strong security. This makes the proposed framework efficient for cloud environments with limited network bandwidth and latency constraints.

c. Storage Requirement

The amount of memory needed to store authentication credentials and session details is termed the storage requirement, which is evaluated using:

$$SR = \sum_{i=1}^{n} D_i \quad (17)$$

© 2026 Conscientia Beam. All Rights Reserved.

Where SR is the storage requirement and D_i represents the data elements stored for authentication. The DNA authentication framework's storage requirement is calculated by:

$$SR = S_{HPW} + S_{TS} + S_{Nonce} + S_{DUID} + S_{DP} + S_{SK} \quad (18)$$

Where S_{HPW}, S_{TS}, S_{Nonce}, S_{DUID}, S_{DP}, and S_{SK} represent the storage requirements for the SHA-256-hashed password, timestamp, nonce, dynamic user ID, dynamic password, and session key, respectively. The storage requirement remains moderate (score: 3) because the proposed framework stores only hashed credentials rather than DNA sequences or plaintext passwords, thereby ensuring secure credential management. This provides an efficient balance between storage utilization and authentication security in cloud environments.

d. Security Efficiency

The proposed framework's ability to provide strong security by reducing computational overhead is evaluated by Security Efficiency. Greater security efficiency can be attained through higher values of security strength and attack resistance with lower computational cost. Security Efficiency is evaluated by:

$$SE = \frac{SS+AR}{CC} \quad (19)$$

Here, SE is security efficiency, SS is the security strength, AR is the attack resistance, and CC is the computational cost. Normalization on a scale of 1-5 can be done using the formula.

$$SE_{norm} = \frac{SE}{SE_{max}} \times 5 \quad (20)$$

The above formula is applied to all other authentication mechanisms, and the security efficiency values are calculated and represented in Table 8.

Table 8. Security efficiency values for all the authentication mechanisms.

Method	SS	AR	CC	SE = (SS+AR)/CC
Password	2	2	2	2.00
OTP	3	3	3	2.00
Biometric	4	4	4	2.00
Smart Card	4	4	3	2.67
Proposed DNA	5	5	3	3.33

4.3.1. Comparative Analysis of Performance Metrics

a. Performance Analysis

In the first case, performance analysis of three metrics, computational cost, communication overhead, and storage requirement, is considered because lower values of these metrics provide the best performance. These metrics are shown in Table 9 and in the bar chart in Figure 3.

Table 9. Comparative performance of authentication mechanisms.

Method	Computational cost	Communication overhead	Storage requirement
Password	1	1	2
OTP	2	3	2
Biometric	4	2	5
Smart Card	3	2	4
Proposed DNA	3	2	3

© 2026 Conscientia Beam. All Rights Reserved.

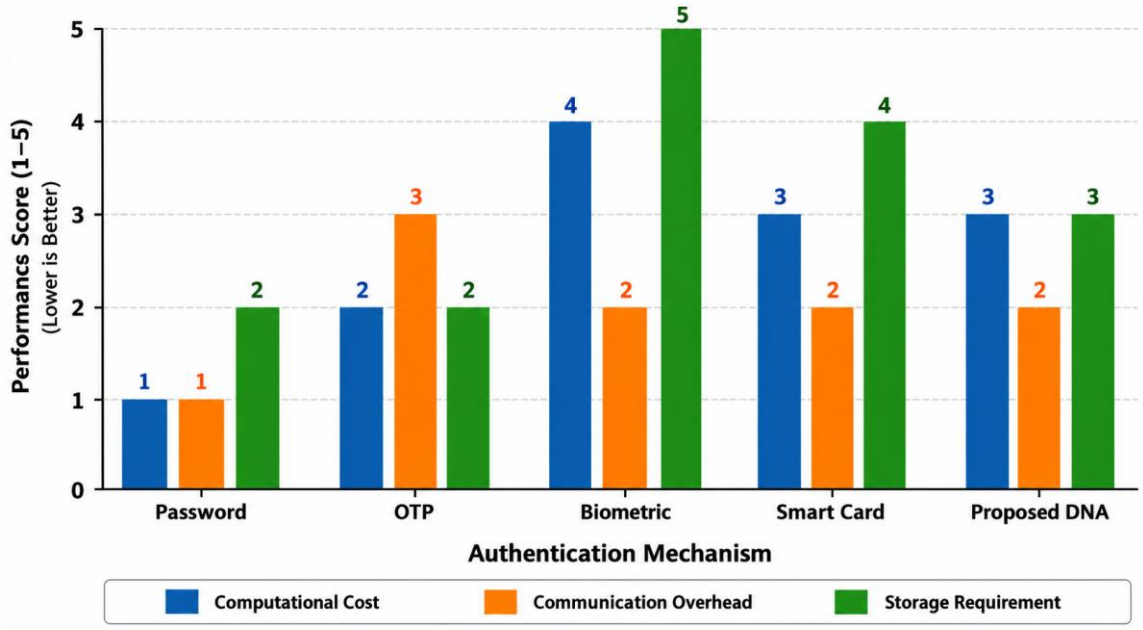


Figure 3. Comparative analysis of the performance metrics.

b. Security Efficiency Analysis

Observation of all the security efficiency values reveals that the proposed DNA framework achieves the highest SE value of 3.33 compared with other authentication methods represented in Figure 4. This indicates that the proposed framework offers better security strength and attack resistance at a moderate computational cost.

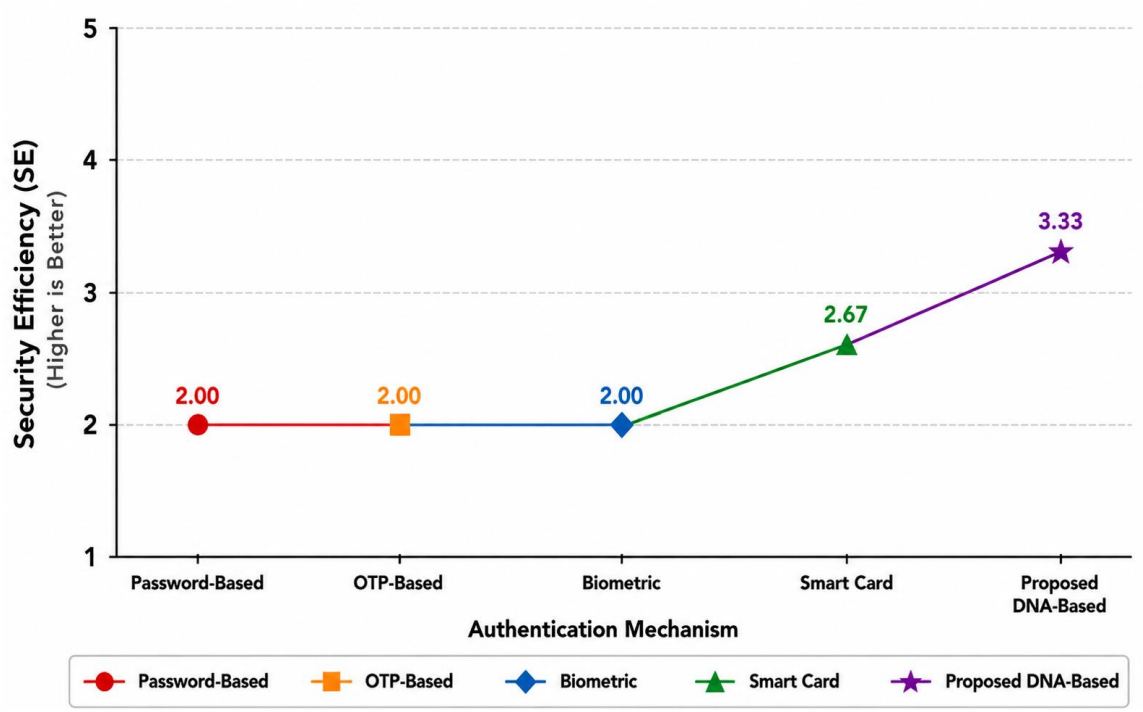


Figure 4. Comparison of security efficiency for the various authentication mechanisms.

4.4. Runtime Analysis

The proposed framework's runtime performance is analysed by measuring average authentication time, encryption processing time, and computational complexity. A sandbox cloud environment is used to carry out an experimental authentication simulation using Python and Azure to emulate multiple user login requests. The DNA

© 2026 Conscientia Beam. All Rights Reserved.

authentication mechanism involving all the operations was executed continuously to create authentication logs and validate security and performance metrics.

More than 100 authentication times were measured, and the sums of all the operations involved in the authentication and encryption mechanisms were calculated and are shown in Table 10. The average authentication time and encryption processing time were computed using the following formulas.

$$\text{Average Authentication Time} = \frac{\sum_{i=1}^{N} T_i}{N} \qquad (21)$$

Where T_i indicates the authentication time incurred during the i^{th} authentication attempt (in milliseconds), and N indicates the total number of authentication attempts during the experiment conducted.

$$\text{Encryption Processing Time} = \frac{\sum E_i}{N} \qquad (22)$$

Where E_i is the encryption processing time.

Table 10. Authentication runtime comparison.

Authentication method	Average authentication time (ms)	Encryption processing time (ms)	Computational complexity
OTP-based authentication	210	120	O(n)
Group signature method	340	260	$O(n^2)$
Traditional SHA authentication	185	95	O(n)
Proposed DNA-based method	160	88	O(n)

4.4.1. Discussion

The operations of the proposed DNA-based authentication mechanism are computationally lightweight and effective, hence achieving lower authentication time than existing methods. Although group signature approaches provide strong anonymity, their complex key management results in higher runtime overhead. The proposed framework preserves linear computational complexity and maintains robust security.

4.5. Scalability Testing

Scalability testing analysis was carried out as the number of cloud users increased from 100 to 1,000. Table 11 represents the scalability results.

The authentication delay is obtained using ($T_{response}$-$T_{request}$); it is the difference between the time when the authentication is sent and the time when the authentication result (success/failure) is received.

Average delay is calculated when multiple users authenticate simultaneously.

$$\text{Hence, Average Authentication Delay} = \frac{\sum_{i=1}^{N} D_i}{N} \qquad (23)$$

Where D_i is the authentication delay for the i^{th} user (ms), and N is the total number of authentication requests.

Table 11. Scalability results.

Number of concurrent users	Password-Based Delay (ms)	OTP-Based Delay (ms)	Biometric Delay (ms)	Smart Card Delay (ms)	Proposed DNA-Based Delay (ms)
100	180	210	240	195	150
250	265	305	340	285	195
500	390	455	510	420	260
750	520	610	690	565	340
1000	690	810	920	760	420

© 2026 Conscientia Beam. All Rights Reserved.

4.5.1. Discussion on Scalability Analysis

The proposed framework represents lower authentication-delay values, indicating stable performance as the number of users increases. Few researchers have debated whether advanced cryptographic techniques reduce scalability and whether stronger encryption increases computational complexity. The DNA-based encoding, however, achieves stronger security while maintaining scalability and manageable runtime performance.

4.6. Comparative Benchmark Analysis

Comparative benchmark analysis of traditional authentication mechanisms with the proposed DNA authentication methodology is shown in Table 12.

4.6.1. Authentication Accuracy

Authentication accuracy is evaluated using the results of multiple authentication attempts in the testing process. For the DNA authentication framework, calculating the percentage of authentication requests that are correctly classified as either authorized or unauthorized provides the authentication accuracy, which is calculated using the formula.

$$\text{Authentication Accuracy } (\%) = \frac{TP+TN}{TP+TN+FP+FN} \times 100 \qquad (24)$$

Where TP denotes true positives, representing correctly authenticated legitimate users; TN denotes true negatives, representing correctly rejected unauthorized users; FP denotes false positives, representing unauthorized users incorrectly granted access; and FN denotes false negatives, representing legitimate users incorrectly denied access.

Approximately 1,000 authentication attempts are considered to identify legitimate and malicious login requests, and authentication accuracy is calculated.

Table 12. Comparative benchmark analysis of authentication methods.

Benchmark metric	OTP-based method	Group signature method	Traditional SHA-256 authentication	Proposed DNA-based authentication
Authentication accuracy (%)	91	94	90	98
Average authentication time (ms)	210	340	185	160
Encryption processing time (ms)	120	260	95	88
Security strength	Medium	High	Medium	Very High
Attack resistance	Moderate	High	Moderate	Very High
Scalability	Moderate	Low	High	Very High
Communication overhead	High	Moderate	Low	Low

4.6.2. Discussion on Comparison of Benchmark Analysis

The proposed DNA framework outperforms existing authentication systems in terms of all the benchmark metrics provided in the table. Existing methodologies improved either security or reduced computational overhead, but the DNA framework achieved both objectives.

4.7. Overall Discussion of Findings

The proposed DNA authentication framework therefore provides a balanced solution.

- Lightweight DNA encoding
- Dynamic authentication
- Efficient session key generation
- Strong attack resistance

© 2026 Conscientia Beam. All Rights Reserved.

- Scalable cloud security architecture

4.8. Limitations of the Study

The evaluation of the proposed DNA authentication framework is restricted to a controlled cloud environment, although it provides enhanced authentication accuracy, lower authentication delay, and improved security. Advanced AI-driven attacks, cryptographic techniques that resist quantum attacks, and multifactor authentication were beyond the scope of the study. The lack of large-scale production deployment and the fact that its scope is limited to authentication rather than authorization are also limitations. Scalability is limited, as the framework has not been tested in enterprise-scale environments involving millions of users.

4.9. Implications and Future Research Directions

Improved cloud security, improved authentication reliability, and a scalable authentication mechanism have been achieved by integrating DNA-inspired cryptographic transformations that utilize hashing and dynamic credential management. Future research will focus on improving the framework through post-quantum cryptography, artificial intelligence, multi-factor authentication, large-scale cloud deployment, and identity management with blockchain, which would further enhance its scalability, practical applicability, and security. Detection of intelligent attacks and adaptive authentication might be attained by integrating machine learning techniques.

5. CONCLUSION

The research presented a DNA-based authentication framework to provide robust security for cloud environments by combining DNA cryptography with DNA encoding, mutation operations, hashing, dynamic user IDs, timestamps, nonces, and session key generation. In comparison with existing password-based, OTP-based, and group signature authentication methods, the experimental results of the proposed framework achieved enhanced authentication security, attack resistance, and scalability by attaining 98% authentication accuracy, 88ms encryption processing time, and 160ms average authentication time. The framework achieved lower authentication runtime and stronger resistance against replay, brute-force, phishing, and credential theft attacks. Furthermore, comparative benchmark analysis showed a better balance between computational efficiency and security when compared to conventional authentication mechanisms. Nevertheless, this research does not claim that DNA-based authentication completely eradicates all cybersecurity risks. The proposed framework does not address deployment challenges such as integration complexity, network latency, and storage overhead in real-world cloud environments on a large scale. Hence, further study is required before practical enterprise-level adoption.

Funding: This study received no specific financial support.
Institutional Review Board Statement: Not applicable.
Transparency: The authors state that the manuscript is honest, truthful, and transparent, that no key aspects of the investigation have been omitted, and that any differences from the study as planned have been clarified. This study followed all writing ethics.
Competing Interests: The authors declare that they have no competing interests.
Authors' Contributions: Both authors contributed equally to the conception and design of the study. Both authors have read and agreed to the published version of the manuscript.
Disclosure of AI Use: In preparing this manuscript, AI-assisted technologies, specifically Grammarly, were utilised to enhance language clarity, grammar, and readability. No generative AI tools were employed to generate original content, data, or analysis.

REFERENCES

[1] Y. Zou, "Automated reasoning and technological innovation in cloud computing security," *Economics and Management Innovation*, vol. 2, no. 6, pp. 25-32, 2025.

© 2026 Conscientia Beam. All Rights Reserved.

[2] A. Sharma and U. K. Singh, "Cloud computing security through detection & mitigation of zero-day attack using machine learning techniques," in *Natural Language Processing for Software Engineering*, R. K. Chakrawarti, R. Sikarwar, S. K. Sarangi, S. A. R. Albert Raj, S. Gupta, K. S. Sankaran, and R. Rawat, Eds. Hoboken, NJ, USA: Wiley, 2025, pp. 357–388.

[3] R. Ali *et al.*, *Authentication of user data for enhancing privacy in cloud computing using security algorithms "Securing the digital realm.* Boca Raton, FL, USA: CRC Press, 2025.

[4] K. K. S. V. A. Satheesh and T. K. Sree, "Enhancing cloud security with authentication for securing sensitive information," *Pattern Recognition*, vol. 171, p. 112345, 2026. https://doi.org/10.1016/j.patcog.2025.112345

[5] S. Rawther and S. Sivaji, "Protecting cloud computing environments from malicious attacks using multi-factor authentication and modified DNA cryptography," *Recent Patents on Engineering*, vol. 19, no. 1, p. E050923220730, 2025. https://doi.org/10.2174/1872212118666230905141926

[6] I. Qiqieh, J. Alzubi, and O. Alzubi, "DNA cryptography based security framework for health-cloud data," *Computing*, vol. 107, no. 1, p. 35, 2025. https://doi.org/10.1007/s00607-024-01393-9

[7] S. Aashiq Banu, L. K. Rao, P. S. Priya, H. Thanikaiselvan, M. Hemalatha, R. Dhivya, and A. Rengarajan, "A review of genome to chaos: Exploring DNA dynamics in security," *Multimedia Tools and Applications*, vol. 84, no. 22, pp. 24859-24886, 2025. https://doi.org/10.1007/s11042-024-20074-5

[8] S. Aqeel, A. S. Khan, I. A. Abbasi, F. Algarni, and D. Grzonka, "Enhancing IoT security with a DNA-based lightweight cryptography system," *Scientific Reports*, vol. 15, no. 1, p. 13367, 2025. https://doi.org/10.1038/s41598-025-96292-0

[9] K. Sasikumar and S. Nagarajan, "Enhancing cloud security: A multi-factor authentication and adaptive cryptography approach using machine learning techniques," *IEEE Open Journal of the Computer Society*, vol. 6, pp. 392-402, 2025. https://doi.org/10.1109/OJCS.2025.3538557

[10] B. Shannaq, "Advancing user authentication in cloud computing: Automated smart application as a multi-factor approach beyond extended-Levenshtein distance," *TEM Journal*, vol. 14, no. 2, pp. 1140-1151, 2025. https://doi.org/10.18421/TEM142-15

[11] E. Harika and J. Balaraju, "Next-generation cryptographic approaches for cloud data security: A comprehensive review of techniques and innovations," presented at the 2025 International Conference on Computational Innovations and Sustainable Technologies (ICCIST), IEEE, 2025.

[12] A. Kairi and T. Bhadra, "A cloud-based DNA cryptography (CBDC) for enhanced data protection," *Security and Privacy*, vol. 8, no. 6, p. e70126, 2025. https://doi.org/10.1002/spy2.70126

[13] S. Namasudra, "A secure cryptosystem using DNA cryptography and DNA steganography for the cloud-based IoT infrastructure," *Computers and Electrical Engineering*, vol. 104, p. 108426, 2022. https://doi.org/10.1016/j.compeleceng.2022.108426

[14] M. I. M. Yusop, N. H. Kamarudin, N. H. S. Suhaimi, and M. K. Hasan, "Advancing passwordless authentication: A systematic review of methods, challenges, and future directions for secure user identity," *IEEE Access*, vol. 13, pp. 13919-13943, 2025. https://doi.org/10.1109/ACCESS.2025.3528960

[15] M. Kumari and R. Nath, *Data security model in cloud computing environment. In M. U. Bokhari, N. Agrawal, & D. Saini (Eds.), Cyber security: Proceedings of CSI 2015 **(Advances in Intelligent Systems and Computing.* Singapore: Springer, 2022.

[16] N. N. Ahamed and T. Murugan, "Research advances in DNA cryptography: Integration with cloud computing, Internet of Things, and DNA computing," *Discover Applied Sciences*, vol. 8, p. 624, 2026. https://doi.org/10.1007/s42452-026-08554-9

[17] T. Kumar, S. Namasudra, and P. Kumar, "Providing data security using DNA computing in the cloud computing environment," *International Journal of Web and Grid Services*, vol. 19, no. 4, pp. 463-486, 2023. https://doi.org/10.1504/IJWGS.2023.135587

[18] V. Verma, "Lightweight mutual authentication protocol for IoT devices using elliptical curves," *International Journal of Electronics and Telecommunications*, vol. 70, no. 4, pp. 785-790, 2024.

© 2026 Conscientia Beam. All Rights Reserved.

[19] A. Singh, A. Kumar, and S. Namasudra, "DNACDS: Cloud IoE big data security and accessing scheme based on DNA cryptography," *Frontiers of Computer Science*, vol. 18, no. 1, p. 181801, 2024. https://doi.org/10.1007/s11704-022-2193-3

[20] N. Yasmin and R. Gupta, "Modified lightweight cryptography scheme and its applications in IoT environment," *International Journal of Information Technology*, vol. 15, no. 8, pp. 4403-4414, 2023. https://doi.org/10.1007/s41870-023-01486-2

[21] K. Ambekar and R. Kamatchi, "Enhanced user authentication model in cloud computing security," presented at the The International Symposium on Intelligent Systems Technologies and Applications, Springer, 2016.

[22] R. Shahabadkar, S. Sai Satyanarayana Reddy, C. Manjunath, U. Channabasava, and K. R. Shahabadkar, "Secure framework of authentication mechanism over cloud environment," presented at the Computer Science On-line Conference, Springer, 2017.

[23] S. Ji, D. Liu, and J. A., *Exploiting group signature to implement user authentication in cloud computing*. China: Springer Nature Singapore PTE LTD, 2024.

[24] G. Choudhury and J. Abudin, "Modified secure two way authentication system in cloud computing using encrypted one time password," *International Journal of Computer Science and Information Technologies*, vol. 5, no. 3, pp. 4077–4080, 2014.

[25] T. Kumar, P. Kumar, and S. Namasudra, "A DNA-based authentication system for securing cloud data storage and transactions," presented at the 2024 11th International Conference on Computing for Sustainable Global Development (INDIACom), IEEE, 2024.

[26] L. Megouache, S. Sadouni, O. Sadouni, M. Djoudi, and A. Zitouni, "Secure cloud authentification using AES encryption," in *Proceedings of the International Conference (CEUR Workshop Proceedings), CEUR-WS*, 2024.

[27] K. Sasikumar and S. Nagarajan, "Comprehensive review and analysis of cryptography techniques in cloud computing," *IEEE Access*, vol. 12, pp. 52325-52351, 2024. https://doi.org/10.1109/ACCESS.2024.3385449

[28] L. Yu, M. He, H. Liang, L. Xiong, and Y. Liu, "A blockchain-based authentication and authorization scheme for distributed mobile cloud computing services," *Sensors*, vol. 23, no. 3, p. 1264, 2023. https://doi.org/10.3390/s23031264

Views and opinions expressed in this article are the views and opinions of the author(s), Review of Computer Engineering Research shall not be responsible or answerable for any loss, damage or liability etc. caused in relation to/arising out of the use of the content.

© 2026 Conscientia Beam. All Rights Reserved.