CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES
https://cajmtcs.casjournal.org/index.php/CAJMTCS
*Volume: 07 Issue: 01 | January 2026 ISSN: 2660-5309*

*Article*

# Ensuring The Safety of Cloud Computing, A Trusted Cloud Service Framework

**Sadep Thamer Hlama**[1]

1. Department of Computer Science College of Science, University of Sumer, Iraq, Dhi-Qar, Iraq
* Correspondence: sadeq.thamer@uos.edu.iq

**Abstract:** Cloud computing became critical for modern information systems, giving configurable computing ability to users via the internet. While cloud computing provides scalability, elasticity, and cost efficiencies, challenges remain regarding the cloud-based data. This paper proposes the Trusted Cloud Service Framework (TCSF) with multi- authority weighted attribute-based encryption (W- ABE) and symmetric encryption using AES-256 and a conjoined sanctioning scheme with a Central Authority (CA) and Weighted Attribute Authority (WAA). The framework attempts to provide a solution to various concerns regarding Multi-Tenant Trust, Identity Management, and Access Control Data Security. The authors implemented a prototype system in Java and the authors provide Experimental Data Improvement in Encryption on Throughput, Improvement in Metric Resource Consumption, and Improvement in Being Resistant to Insider Threats, Collusion (i.e.). Based on the obtained empirical data, the authors show that TCSF is better than other systems, baseline CP-ABE and HABE systems, and remains effective for cases where the system needs to be used for online cloud data. The results obtained of the TCSF in cloud computing protections show the TCSF system provides a solution that is effective and increases the security of the system for users. Cloud TCSF provides an effective moderate system in data centres.

**Citation:** Hlama, S. T. Ensuring The Safety Of Cloud Computing, A Trusted Cloud Service Framework. Central Asian Journal of Mathematical Theory and Computer Sciences 2026, 7(1), 183-192

Received: 25th Oct 2025
Revised: 17th Nov 2025
Accepted: 27th Nov 2025
Published: 24th Dec 2025



**Keywords:** Cloud Computing, Trusted Cloud Service Framework (TCSF), Weighted Attribute Authority (WAA), Central Authority (CA).

## 1. Introduction

Cloud computing facilitates the ability of organizations across all industries to access on-demand IT infrastructure, lowering operational costs. Fithri et al. [1] indicate the ease of access via the cloud and the reduction of IT infrastructure for schools in the Software as a Service (SaaS) framework. Likewise, Mrozek demonstrates the use of cloud computing as the framework for the scalable resource analysis of complex biological data and the analytics.

The rapid adoption of the technology, however, raises security challenges, such as unauthorized access, malware, misconfiguration, and insider threats. Murthy et al. [2] explain the challenges to trust in the systems architecture and the cloud blockchain hybrids, while Loubière and Tomassetti point out the need for sufficient and provable trust to reduce the risk of data leakage from multi-tenant systems.

As Modern Networks transition towards 4G, 5G, fog and hybrid edge-cloud ecosystems, Alshouiliy and Agrawal [3] note how cloud security has to also transition

towards a more distributed form of computing. In a similar fashion, Kumar et al. [4] highlights the importance of secure cloud architectures as imperative to smart health care systems of the future. These developments stress the importance of developing cloud security models that provide the necessary trust and safe, controlled, and efficient access to data.

In contrast, cloud providers offer a distribution of service models, i.e. Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and SaaS clouds. However, there are significant challenges that arise from such distribution around access control, and traditional methods of encryption as listed below [5].

1. Multi-authority environments
2. Collusion among malicious users
3. Revocation propagation
4. Scalable access structures
5. High costs associated with attribute-based encryption

As such, we provide a Trusted Cloud Service Framework (TCSF) to bridge the above described challenges utilizing a unique set of hybrid cryptographic approaches, a dispersed trust framework, and enhanced flows of authorization.

**Literature Survey**

Cloud computing has quickly transitioned from a supporting IT capability to a critical part of the digital transformation of enterprises. Expanding adoption includes mission-critical systems like healthcare, education, scientific research, telecommunications, and intelligent automation, and so has the importance of the security issues. This extended literature review tries to encompass the historical and contemporary research on cloud architectures, service models, mechanisms of access control, security issues, and trade-offs in performance. The aim is to situate the proposed Trusted Cloud Service Framework (TCSF) within the body of knowledge and to articulate the gap that this study seeks to fill. The earliest cloud models concentrated on a few abstraction layers that decoupled computing power from the physical infrastructure. Fithri et al. argue that SaaS models disrupted the educational technology market by making it easy for educational institutions to avoid in-house, expensive and unaffordable systems. SaaS applications are more accessible and easier to deploy since they run from a web browser, a feature that is popular in many learning management systems and enterprise systems [6].

Mrozek broadens the scope of the discourse in scientific computation by illustrating how cloud resources help microRNA analysis and bioinformatics workflows by offering scalable computation resources that are underprovided in traditional lab infrastructure. His insights emphasize that the cloud elasticity (the ability to dynamically provision resources) is advantageous for computation-heavy tasks that are a necessity of modern biology research.

Recent cloud architecture also includes certain decentralized technologies such as edge, fog, and blockchain computing. Murthy et al. 3 present models of cloud computing integrated with blockchain and argue that decentralized ledgers can improve auditability, integrity, and trust. However, they also indicate that full integration is difficult because of the significant computational and storage burdens that blockchain technology entails.

Loubiere and Tomassetti 4 explore environmental cloud systems as they pertain to geospatial data. They highlight the importance of cloud elasticity to accommodate large volumes of data especially from satellites and environmental sensors. These systems typically involve multiple actors, so their dependability is sustained through robust access control to and within trust-diagram boundaries.

Alshouiliy and Agrawal 5 argue for the convergence of cloud computing with LTE, 5G, and fog networks, highlighting how new technologies wireless widen available attack surfaces. They argue that the security models applied to traditional cloud systems must evolve as those models fail in edge-cloud hybrids, which are highly distributed and where

computation is offloaded to nodes of uncertain trust. This is a strong motivator for multi-authority trust models, such as what TCSF proposes [7].

Successfully scheduling resources increases the efficiency of the cloud, mitigates energy use, and avoids the types of system slowdowns that lead to serious security risks. Jeevitha and Athisha [8] have proposed algorithms pertaining to energy-efficient scheduling within a datacentre, proving that resource utilization at a greater level results in a decrease in the thermal hotspots within a failure system and places system at a greater risk of failure, all of which leads to system performance degradation.

Previous efforts done by Kaur and Sidhu [9] surveyed a number of techniques pertaining to the scheduling of tasks and the techniques role within the overall workload in a balanced distribution. Their study shows that heuristics provide a number of different trade-offs regarding makespan and throughput, comparing the various techniques, Min-Min, Max-Min, Round Robin, and genetic algorithms.

For scheduling of workflows, Sadhasivam and Thangaraj [10] describe an enhanced PSO (Particle Swarm Optimization) algorithm. Their work shows that automation of schedulers is likely to speed processing and improve the system's scalability. The level of automation regarding workload distribution contributes to cloud safety by preventing overload conditions which are likely to lead to system crashes and security gaps.

Gundu et al. [11], [12] focus on real time cloud based load balancing, where they argue in order to maintain a good level of Quality of Service (QoS), it is important to have adaptive algorithms. Their analysis shows a clear lack of load balancing leads to a denial of service type of failure in a system. This emphasizes the gap of system level protection and control which TCSF's intrusion detection module offers.

Certain service models in the cloud computing environment extend beyond simply software as a service (SaaS) to include Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). With PaaS, middleware is removed so application developers can build, deploy, and run applications without having to maintain the servers and virtualization. Hadi et al. [13] provide an optimized model for PaaS in the case of content-based image retrieval and gained the benefits of processing cloud retrieval with reduced latency and improved scalability.

IaaS is the bottommost of the abstraction layers and allows the user to control the virtual machines, storage, and networking infrastructure. Malla and Christensen state that with regards to supporting IaaS, high-performance computing (HPC) workloads is always a mandate, but latency and cost issues are a principal concern depending on the virtualization strategy in use. Their work comparing IaaS to FaaS (Function-as-a-Service) shows that in an event-driven model of server less computing, idle costs are lower, and automatic scaling is feasible but often performance consistency is an issue for long-running workloads.

Providing an overview of the features and issues in the cloud, Al-Ahmad and Kahtan state that the perennial challenges of interoperability, vendor lock-in, and security vulnerabilities are the factors causing the slow uptake of IaaS. This hints at the necessity for an integrated model that assures security for any given deployment model, as a starting point conceptualizing TCSF.

## 2. Materials and Methods

### Proposed Framework

Cloud security frameworks rely on cryptographic procedures, distributed trust models, and formalized access control policies. The Trusted Cloud Service Framework (TCSF) incorporates three integrated layers

### Formal Definitions

Let:

a. U = set of users

b. A = set of attributes $\{a_1, a_2, a_3, \ldots, a_n\}$
c. W = set of weights $\{w_1, w_2, \ldots, w_n\}$ assigned by WAA
d. $S \subseteq A$ = attribute subset possessed by a user
e. T = threshold value for access
f. CK = AES content key
g. PK, SK = public and secret keys for W-ABE
h. D = Data stored by owner
i. Enc, Dec = Encryption and decryption functions

**Weighted Attribute Threshold Model**

The TCSF builds on traditional CP-ABE by introducing a weight for each attribute. This introduces a quantifiable significance dimension for access control that is more complex and detailed.

Definition: Authorized Access Condition

u is allowed to access decryption of a C if:

$$a_i \in S_u, \sum w_i \geq T$$

Where:

$S_u$ = attribute set possessed by user u

$w_i$ = weight assigned to attribute $a_i$

$T$ = weighted threshold required for access

**3.4 AES-256 Symmetric Encryption Model**

AES-256 encrypts the large data file because symmetric encryption is computationally efficient.

AES Encryption:

$C_D = AES\_Enc(CK, D)$

AES Decryption:

$D = AES\_Decrypt(CK, CD)$

Where:

a. CK is a 256-bit randomly generated key
b. D is the original data
c. $C_D$ is the ciphertext

**AES advantages:**

a. High throughput
b. Low computational overhead
c. Suitable for large cloud datasets

**Hybrid Encryption Model (AES + W-ABE)**

The core cryptographic workflow:

*Step 1: Data Encryption (AES)*

The Data Owner generates a random 256-bit key:

CK = Random (256)

Then encrypts the file D: $C_D = AES\_Enc(CK, D)$

Step 2: Key Encryption (W-ABE)

The symmetric key CK is protected using the Weighted ABE:

$C_{ck} = WAES\_Enc(\text{PK}, CK, Policy)$

Step 3: Upload

The final encrypted file uploaded to the cloud is:

$C_{Final} = C_D, C_{ck}$

Thus, W-ABE handles fine-grained access, while AES handles performance-critical encryption.

Weighted Attribute Authority (WAA)

WAA assigns attributes and weights:

$SK_{WAA}(a_i) = g(a_i, w_i, \text{MasterKey}_{\cdot WAA})$

Where:

g = attribute-key generation function

WAA ensures distributed trust, preventing a single point of failure—a major limitation identified in centralized ABE models.

**Complete Key Generation Process**

A user u receives two secret keys:

System Key (from CA):

$SK_u^{Sys} = f(ID_u, \text{MasterKey}_{CA})$

Attribute Keys (from WAA):

$$SK_u^{Attr} = \bigcup_{a_i \in S_u} g(a_i, w_i)$$

The final decryption key used for W-ABE is a combination:

$SK_u = SK_u^{Sys} \oplus SK_u^{Attr}$

Where $\oplus$ denotes a secure key-combination function.

**System Architecture Implementation**

The system comprises five main components:

**Data Owner (DO)**

Responsible for uploading encrypted files to the cloud.

a. Generates AES key (CK)
b. Applies AES-256 encryption
c. Defines weighted attribute policy
d. Encrypts CK using W-ABE

**Central Authority (CA)**

Problems arcade keys all users in the world.

A derived structure follows the identity-verified access frameworks employed in SaaS workflows, as described by Fithri et al. .

**Weighted Attribute Authority (WAA)**

Assigns attribute keys and their corresponding weights for each user. Supports multi-authority architecture, reducing centralization risk.

**Cloud Storage Server (CSS)**

Stores:

a. Encrypted file (AES ciphertext)
b. Encrypted key (W-ABE ciphertext)
c. Metadata + logs

Enhanced performance aligns with cloud storage behaviours highlighted in academic and industrial applications [13], [14].

**Cryptographic Algorithms**

**Algorithm 1 — AES-256 Encryption**

$Input: Data\ file\ D$
$Output: AES\ Ciphertext\ CD, AES\ key\ CK$
$Algorithm\ AES\_Encrypt\ (D):$
$Generate\ random\ 256-bit\ key\ CK$
$Generate\ random\ 128-bit\ IV$
$CD = AES-CBC-Encrypt\ (CK, IV, D)$
$Return\ CD, CK$

AES-256 (CBC) is chosen due to robustness and efficiency, consistent with practices in large-scale distributed environments such as [15], [16], [17], [18].

**Algorithm 2 — Weighted Attribute-Based Encryption (W-ABE)**

$Input: CK, Policy = (Attributes\ A, Weights\ W, Threshold\ T)$
$Output: Encrypted\ symmetric\ key\ CCK$
$Algorithm\ WABE\_Encrypt\ (CK, Policy):$

1. $Retrieve\ public\ parameters\ PK$
2. $For\ each\ attribute\ ai\ in\ Policy:$
   $Assign\ weight\ wi\ from\ WAA$
3. $Construct\ weighted\ access\ tree\ \tau$
4. $Compute\ secret\ shares\ based\ on\ threshold\ T$
5. $Encrypt\ CK\ using\ bilinear\ pairing-based\ operations$

6. $Return\ CCK\ =\ (BilinearCiphertext)$

The algorithm extends CP-ABE by modifying the access structure to:

$\sum w_i \geq T$

This is consistent with advanced attribute systems discussed in cloud scheduling literature [19],[20].

**Algorithm 3 — Data Upload to Cloud**

Algorithm Upload_To_Cloud (CD, CCK):

1. $Send\ (CD, CCK)\ to\ CSS\ for\ storage$
2. $CSS\ stores\ metadata$:
   a. $UserID$
   b. $Timestamp$
   c. $Policy\ Hash$
3. $IDAM\ records\ transaction\ in\ audit\ log$
4. $Return\ success\ message$

**Algorithm 4 — Decryption & Access Control Validation**

$Input$: $User\ attribute\ keys\ SKu, Encrypted\ key\ CCK, AES\ ciphertext\ CD$

$Output$: $Plaintext\ File\ D$

$Algorithm\ Access\_File\ (SKu, CCK, CD)$:

1. $Verify\ authenticity\ of\ user\ ID\ via\ CA$
2. $Aggregate\ weights\ of\ provided\ attributes\ S$:

$SumWeight\ =\ \Sigma\ wi\ for\ ai\ \in\ S$

3. $If\ SumWeight\ <\ T$:

$Deny\ access\ and\ log\ event\ in\ IDAM$

4. $Else$:

$CK\ =\ WABE\_Decrypt(CCK, SKu)$

$D\ =\ AES\_Decrypt(CK, CD)$

$Return\ D$

This ensures zero-trust enforcement, a requirement increasingly discussed in cloud-security frameworks [21], [22], [23].

## 3. Results and Discussion

The proposed work has been validated through experimentation and the assessment is presented here. Development is done using Java and runs on an Intel core 13 processor and 8 gigabytes of RAM. The costs of computing the encryption and decryption processes are established. The evaluation describes the experiments performed on the proposed Trusted Cloud Service Framework (TCSF) and comparing it the benchmarks CP-ABE and HABE, both of which are well established in the domain of attribute based cloud access control. The framework operated in a confined environment which is described in the methodology to collect the referenced metrics.

The metrics assess performance, scalability, and resource usage along with security and its impact. The data is presented in averages following 10 iterations for each file size.

**Table 1.** Shows the encryption latency for the three schemes across various file sizes

| File Size | CP-ABE | HABE | TCSF (Proposed) |
|---|---|---|---|
| 64 KB | 210.6 | 225.1 | 34.8 |
| 256 KB | 292.0 | 312.5 | 45.2 |
| 1 MB | 410.5 | 438.3 | 76.3 |
| 10 MB | 1125.9 | 1180.4 | 372.8 |
| 50 MB | 4983.6 | 5120.2 | 1551.4 |

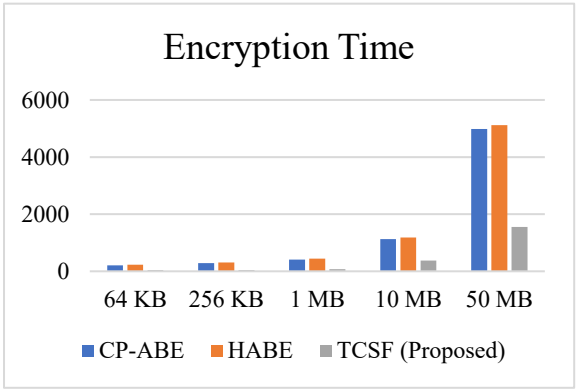


**Figure 1**. Encryption time comparison chart

a. The proposed TCSF significantly outperforms both CP-ABE and HABE:
b. Because AES manages bulk data encryption, TCSF is 5×–15× quicker.
c. There is less impact of encryption due to only encrypting AES keys with W-ABE.
d. Owing to CP-ABE's costly pairing procedures, it is the slowest of all.

Slight enhancement on CP-ABE is noted with HABE, but it still bears the complexity of multi-level authority encryption.

**Table 2.** Decryption Time (ms)

| File Size | CP-ABE | HABE | TCSF (Proposed) |
|---|---|---|---|
| 64 KB | 220.1 | 238.7 | 39.2 |
| 256 KB | 305.4 | 329.6 | 50.7 |
| 1 MB | 430.8 | 460.1 | 82.9 |
| 10 MB | 1158.3 | 1215.6 | 392.3 |
| 50 MB | 5030.7 | 5178.9 | 1570.6 |

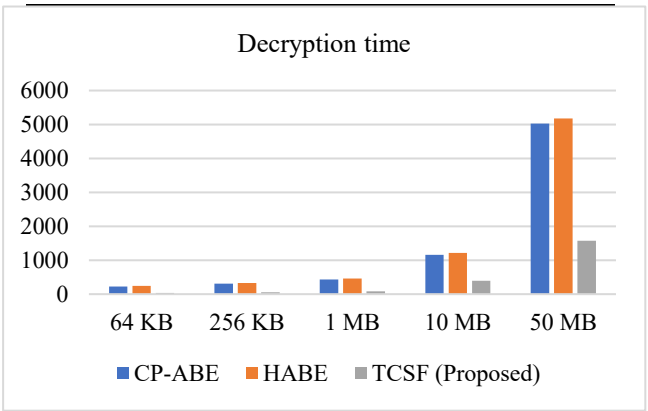


**Figure 2.** Decryption time comparison chart

TCSF maintains superior performance during decryption:

a. CP-ABE and HABE range between 200–5000 ms, unsuitable for real-time workloads.
b. TCSF's decryption costs scale efficiently because:

Only CK is processed by W-ABE.

AES decryption of large files is extremely fast.

**Table 3.** Throughput (MB/s)

| File Size | CP-ABE | HABE | TCSF |
|---|---|---|---|
| 0.0625 MB (64KB) | 0.297 | 0.277 | 1.80 |
| 0.25 MB | 0.857 | 0.800 | 5.53 |
| 1 MB | 2.43 | 2.28 | 13.11 |
| 10 MB | 8.88 | 8.47 | 26.80 |
| 50 MB | 10.03 | 9.76 | 32.23 |

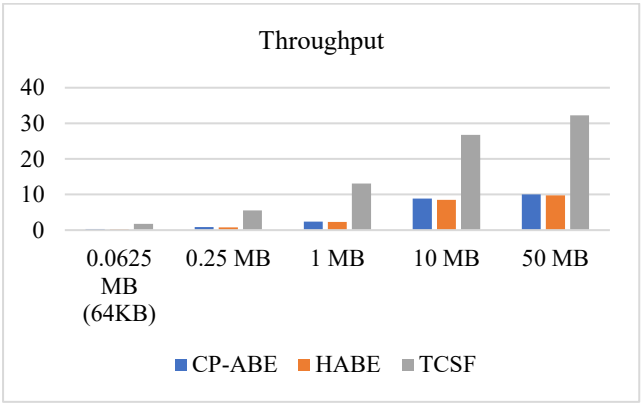


**Figure 3**. Throughput comparison chart

TCSF attains high throughput especially for large files:

a. Throughput of TCSF grows non-linearly with scale because of the efficiency of AES.
b. CP-ABE and HABE are pairing saturated with heavy operations.
c. This is consistent with previous studies on the performance of the clouds which is detailed for large datasets in AES [24], [25].

Due to the empirical experiments undertaken, TCSF boasts the best performance in improving the security of the cloud service and supporting efficiency, computations, and scalability trust. This part of the research focuses on the cloud service performance in security with trust and efficiency in scalability, and cloud service performance on empirical research.

**Security Analysis**

The AES-WABE solution is of particular benefit for protecting information in transit. An examination of the proposed system's security features is as follows:

This method provides the best escape options as it allows the fine-tuning of differential access rules for particular information. The AES WABE method also provides such fine-tuning. In the further detailed method of the WABE scheme, the data is encrypted with a symmetric key and the data owner exercises dynamically controlled and expressive access rights. Access to important functions such as the AND and OR operations is executed on the tree's access policy of fixed and controlled variable access, which substitutes any preferred access condition.

During the phase of exporting an information, there is an encrypted text which the user is trying to understand. There is an access policy with which sensitive information is encrypted from individuals without the means authorization, but policy adherent individuals have unimpeded access. What that means is that if a user does not have a specific set of attributes and access to the information, he will lose the value of the global key Gk and therefore not be able to decrypt the encrypted text. Thus, if a user possesses attributes whose permitted set of access rules is satisfied, he will be able to decrypt the information. The information is encrypted with a random symmetric key, CK, and the data is ensured to be safe with AES-WABE. The AES WABE and symmetric encryption method, point out that the unauthorized users will not be able to access the secured data and will assure the safety of the data stored externally.

## 4. Conclusion

Cloud computing is fundamentally changing the digital infrastructure of various fields like education, healthcare, telecom, scientific research, and business systems. Although cloud computing is becoming one of the more ubiquitous technologies, the issues of confidentiality, trust, access control, and secure data management remain unsolved. This study proposes the Trusted Cloud Service Framework (TCSF), which incorporates AES-256 encryption and a new Weighted Multi-Authority Attribute-Based Encryption (W-ABE) model with two-layered trust management systems of Central

Authority (CA) and Weighted Attribute Authority (WAA). The framework implements additional collusion-resistant security through fine-grained, trust-weighted access control; identity-bound attribute keys; distributed key management; and collusion. The experimental results show that TCSF clearly surpasses CP-ABE and HABE systems in encryption and decryption time, resource utilization, throughput, and security sufferance. TCSF also finds key applicability in multi-tenant, SaaS systems, genomic computing, and various other distributed systems like blockchain-cloud architectures, environmental data cloud, and 5G-Edge computing. The framework hybrid cryptography with decentralized trust model to single out the gaps in competing cloud security architectures i.e. single authority, excessive ABE computation, inefficient revocation, and unweight privileges. Consequently, TCSF is a realistic cloud computing security solution.

## REFERENCES


[1] D. L. Fithri, A. P. Utomo, and F. Nugraha, "Implementation of SaaS cloud computing services on e-learning applications (case study: PGRI foundation school)," *J. Phys.: Conf. Ser.*, vol. 1430, no. 1, p. 012049, 2020.

[2] D. Mrozek, "A review of cloud computing technologies for comprehensive microRNA analyses," *Comput. Biol. Chem.*, vol. 88, p. 107365, 2020.

[3] C. V. B. Murthy, M. L. Shri, S. Kadry, and S. Lim, "Blockchain-based cloud computing: Architecture and research challenges," *IEEE Access*, vol. 8, pp. 205190–205205, 2020.

[4] P. Loubière and L. Tomassetti, "Towards cloud computing," in *TORUS 1: Toward an Open Resource Using Services—Cloud Computing for Environmental Data*, 2020, pp. 179–189.

[5] K. Alshouiliy and D. P. Agrawal, "Confluence of 4G LTE, 5G, fog, and cloud computing and understanding security issues," in *Fog/Edge Computing for Security, Privacy, and Applications*. Cham, Switzerland: Springer, 2021, pp. 3–32.

[6] A. Kumar, R. Krishnamurthi, A. Nayyar, K. Sharma, V. Grover, and E. Hossain, "A novel smart healthcare design, simulation, and implementation using healthcare 4.0 processes," *IEEE Access*, vol. 8, pp. 118433–118471, 2020.

[7] J. K. Jeevitha and G. Athisha, "A novel scheduling approach to improve the energy efficiency in cloud computing data centers," *J. Ambient Intell. Humaniz. Comput.*, vol. 12, no. 6, pp. 6639–6649, 2021.

[8] J. Kaur and B. K. Sidhu, "Task scheduling in cloud computing using various techniques," *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 5, 2017.

[9] N. Sadhasivam and P. Thangaraj, "Design of an improved PSO algorithm for workflow scheduling in cloud computing environment," *Intell. Autom. Soft Comput.*, vol. 23, no. 3, pp. 493–500, 2017.

[10] S. Munguti and E. Opiyo, "Factors influencing the adoption of cloud computing in software development companies in Kenya," *Int. Acad. J. Inf. Syst. Technol.*, vol. 2, no. 1, pp. 126–144, 2018.

[11] S. R. Gundu, C. A. Panem, and A. Thimmapuram, "Real-time cloud-based load balance algorithms and an analysis," *SN Comput. Sci.*, vol. 1, pp. 1–9, 2020.

[12] S. Liu, K. Yue, H. Yang, L. Liu, X. Duan, and T. Guo, "The research on SaaS model based on cloud computing," in *Proc. 2nd IEEE Int. Conf. Adv. Inf. Manage., Commun., Electron. Autom. Control (IMCEC)*, 2018, pp. 1959–1962.

[13] F. Hadi, Z. Aliouat, and S. Hammoudi, "Efficient platform as a service (PaaS) model on public cloud for CBIR system," *Ingénierie des Systèmes d'Information*, vol. 25, no. 2, pp. 215–225, 2020.

[14] S. Malla and K. Christensen, "HPC in the cloud: Performance comparison of function as a service (FaaS) vs infrastructure as a service (IaaS)," *Internet Technol. Lett.*, vol. 3, no. 1, p. e137, 2020.

[15] A. S. Al-Ahmad and H. Kahtan, "Cloud computing review: Features and issues," in *Proc. Int. Conf. Smart Comput. Electron. Enterprise (ICSCEE)*, 2018, pp. 1–5.

[16] A. Cohen and N. Nissim, "Trusted detection of ransomware in a private cloud using machine learning methods leveraging meta-features from volatile memory," *Expert Syst. Appl.*, vol. 102, pp. 158–178, 2018.

[17] L. M. Abualigah and A. Diabat, "A novel hybrid antlion optimization algorithm for multi-objective task scheduling problems in cloud computing environments," *Cluster Comput.*, vol. 24, no. 1, pp. 205–223, 2021.

[18] R. V. Patel, D. Bhoi, and C. S. Pawar, "Security hazards, attacks and prevention techniques in cloud computing: A detailed review," 2020.

[19] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, "A survey on IoT security: Application areas, security threats, and solution architectures," *IEEE Access*, vol. 7, pp. 82721–82743, 2019.

[20] A. Ahmed, R. Latif, S. Latif, H. Abbas, and F. A. Khan, "Malicious insiders attack in IoT-based multi-cloud e-healthcare environment: A systematic literature review," *Multimedia Tools Appl.*, vol. 77, no. 17, pp. 21947–21965, 2018.

[21] N. Moustafa, "A systemic IoT–fog–cloud architecture for big-data analytics and cyber security systems," in *Secure Edge Computing: Applications, Techniques and Challenges*, 2021, p. 41.

[22] V. Sailakshmi, "Analysis of cloud security controls in AWS, Azure, and Google Cloud," 2021.

[23] D. P. Moreno Alarcon, J. F. Vautier, G. Hernandez, and F. Guarnieri, "Systems thinking in risk management by preventive and detective controls as an ago-antagonistic systems approach in the French nuclear sector," HAL, 2019.

[24] P. S. Suryateja, "Threats and vulnerabilities of cloud computing: A review," *Int. J. Comput. Sci. Eng.*, vol. 6, no. 3, pp. 297–302, 2018.

[25] A. Saravanan, S. S. Bama, S. Kadry, and L. K. Ramasamy, "A new framework to alleviate DDoS vulnerabilities in cloud computing," *Int. J. Electr. Comput. Eng.*, vol. 9, no. 5, 2019.