International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/35 Volume 12, Issue 4, July – August (2025)

RESEARCH ARTICLE

Enhancing Cloud Security with Improved RSA and XACML Technology

Ali Zaheer Agha
Department of Computer Applications, Invertis University, Bareilly, Uttar Pradesh, India.
✉ alizaheer7@gmail.com

Rajesh Kumar Shukla
Faculty of Engineering and Technology, Invertis University, Bareilly, Uttar Pradesh, India.
rkshukla30@gmail.com

Ratnesh Mishra
Department of Computer Science and Engineering, BIT Mesra, Patna Campus, Patna, Bihar, India.
r.mishra@bitmesra.ac.in

Ravi Shankar Shukla
Department of Computer Science, College of Computing & Informatics, Saudi Electronic University, Saudi Arabia.
ravipraful@gmail.com

Received: 13 May 2025 / Revised: 15 July 2025 / Accepted: 11 August 2025 / Published: 30 August 2025

Abstract – Widespread internet access today necessitates robust security for authenticity, anonymity, and accessibility. This ubiquity, however, presents significant challenges, including preventing unauthorized network access, mitigating privacy concerns in current systems, and securely managing and distributing private keys for cloud signatures. Traditional Role-Based Access Control (RBAC) is often insufficient because it only verifies identity and fails to evaluate user reliability or guarantee data integrity and stability in dynamic cloud environments. These issues are further exacerbated by the difficulty of securely distributing keys to multiple authorized clients. This paper proposes a novel solution that combines advanced access management with advanced encryption to address these critical vulnerabilities. To enhance identity security and data protection, our design utilizes Extensible Access Control Markup Language (XACML) to implement an improved RSA-based RBAC system. A simplified admission control system enables secure, encrypted data storage on web servers. Furthermore, a hybrid encryption approach ensures complete private information security and reliable data recovery by fusing homogeneous encryption with an unstable information distribution technique. Empirical assessments demonstrate that this proposed work operates noticeably faster and offers significant advantages compared to current encryption methods and conventional RBAC, thereby providing a more reliable and effective cloud security framework.

Index Terms – Cloud Computing, Data Privacy, Cryptography, Admission Control Security, Access Control, RBAC, XACML, Cloud Security.

1. INTRODUCTION

The rapid development of web technology, particularly the use of distributed computing and cloud services, has enabled customers to easily access enormous volumes of data and resources at any time, from any location. While this widespread access offers immense advantages in terms of memory, networks, and computational power, it also introduces increasingly complex security issues. Among these, maintaining robust accessibility, anonymity, and authenticity in a dynamic online environment is crucial.

The secure storage of cryptographic keys on the server for secure cloud signatures and the challenges of trustworthy client verification when obtaining private keys are main concerns [1]. Additionally, even well-known technologies like Wi-Fi have raised significant privacy concerns, and the open nature of internet access frequently permits unauthorized users to join networks. Despite being fundamental, traditional access control techniques such as Role-Based Access Control (RBAC), Mandatory Access Control (MAC), and Discretionary Access Control (DAC) have significant drawbacks. RBAC, for example, primarily confirms user identity but often lacks the ability to assess a user's dependability to limit login attempts or prevent unauthorized network access. This restriction seriously jeopardizes the stability of systems in multi-machine online environments and data integrity, where changes must only be made by

RESEARCH ARTICLE

authorized users. These difficulties are exacerbated by the challenges of securely distributing encryption keys to numerous authorized clients. RBAC policies help address cybersecurity vulnerabilities by enforcing the principle of least privilege. By limiting access to sensitive data, RBAC helps prevent both accidental data loss and intentional data breaches [2].

The constant need for unified, efficient, and adaptable security frameworks that can handle the various risks to data privacy and access control in the changing cloud environment. Large, dynamic cloud infrastructures frequently suffer from issues including excessive processing cost, dependence on potentially outdated cryptographic primitives, and a dearth of practical testing [3].

Biometric authentication is an increasingly important part of access control systems, providing a more robust method of verifying user identity [4].

This paper proposes a novel method that combines a robust encryption mechanism with an advanced access management system to address these critical security flaws and improve the overall security posture of cloud computing environments. Our solution utilizes Extensible Access Control Markup Language (XACML) in conjunction with enhanced RSA-based RBAC to introduce a significantly more robust encryption mechanism. Through a simplified admission control system, this integrated framework seeks to secure identity, safeguard private data, and facilitate secure storage of information on web servers.

To ensure complete private information security and enable reliable data recovery for users, this encryption strategy employs a hybrid approach that combines traditional homogeneous encryption with an unstable information distribution technique. When compared to current encryption methods and conventional RBAC systems, this research empirically demonstrates that the proposed system operates much more quickly and provides greater security and efficiency advantages, offering a more dependable and robust cloud security framework.

1.1. Problem Statement

- Although cloud computing and internet connectivity are widely used and offer enormous benefits in terms of data and resource accessibility, they also present serious security risks. Three crucial areas are at the center of the main issue:
- Privacy Issues and Unauthorized Access: Role-Based Access Control (RBAC) and other traditional access control methods are frequently inadequate. They mainly confirm the identification of users, but they are unable to evaluate user dependability or ensure data integrity in dynamic cloud systems. This restriction makes existing systems more vulnerable to illegal network access and makes privacy problems worse.
- Secure Key Distribution and Management: Managing and distributing private keys, which are necessary for data encryption and cloud signatures, securely is a significant difficulty. It is difficult to reliably distribute private keys to several authorized customers while maintaining reliable client verification for private key retrieval.
- Absence of Adaptive and Integrated Solutions: Many cloud security solutions now in use concentrate on discrete elements (such as encryption or access control alone). Strong security, performance efficiency, scalability, and flexibility to the ever-changing risks and dynamic nature of cloud systems are all factors that these disjointed methods frequently find difficult to reconcile. This weakens the entire security posture and introduces security flaws.

1.2. Need for Proposed Model

The proposed model is needed to overcome the limitations of existing cloud security solutions, specifically:

Inadequate Access Control: Traditional RBAC fails to assess user reliability, leading to unauthorized access and data integrity issues in dynamic cloud environments.

- Complex Key Management: There's a significant challenge in securely distributing and managing cryptographic keys to multiple authorized users.
- Fragmented Security: Current solutions often lack comprehensive integration between access control and encryption, resulting in security gaps and inefficiencies in balancing security, performance, and scalability.

The proposed model aims to provide a unified, intelligent, and robust framework to address these critical vulnerabilities.

1.3. Research Objectives

With regard to data privacy, access control, and key management, this study attempts to address critical security and efficiency issues prevalent in contemporary cloud computing environments. The study's main objectives are as follows:

- To create a robust and effective access control system that evaluates user dependability in addition to identity verification. This system will prevent unauthorized network access and ensure secure login attempts in cloud environments.
- To improve data integrity and privacy through advanced encryption: The objective is to implement a more robust encryption system that can safeguard private data both

RESEARCH ARTICLE

during transmission and storage and ensure that only authorized users can alter data.

- To create a secure environment for the distribution and administration of private keys: To achieve secure cloud signatures, this objective focuses on addressing challenges with trustworthy client verification for private key retrieval, as well as the secure distribution and storage of these keys on cloud servers.
- To empirically measure the performance of the proposed model: In a simulated cloud environment, this involves showcasing the system's enhanced execution speed, effectiveness, and overall security advantages over conventional access control models and current encryption techniques.

Together, these objectives seek to create a cloud security architecture that is more secure, dependable, and effective while tackling the complex issues of contemporary online data management.

1.4. Overall Contribution

This approach features a security architecture that blends Role-Based Access Control (RBAC) with an advanced RSA encryption algorithm and eXtensible Access Control Markup Language (XACML). The advanced RSA incorporates the Advanced Encryption Standard (AES) for speedier data processing, making it cost-effective in the virtual environment. The blend of AES (symmetric) and RSA (asymmetric) encryption reduces the processing time for encrypting/decrypting data and offers greater security with larger key sizes. The system is secure and dependable, with future key management methods planned to enhance the secure sharing of information in the cloud.

The access model blends advanced RSA encryption, role-based access, and XACML to provide quicker, more robust data protection in the cloud, with future plans to enhance key management.

The remainder of this paper is organized as follows: Section 2 discusses the literature review, while Section 3 explains the materials and techniques. The proposed techniques are described in Section 4, and the empirical evaluation is covered in Section 5. Finally, Section 6 presents the conclusion of this study.

2. RELATED WORK

Strong security measures are required due to the growing dependence on cloud computing, which has prompted extensive research into authentication, encryption, and access control. This section offers a thorough analysis of earlier research, reviewing its methods, algorithms, and noted advantages and disadvantages.

A survey of cloud computing security with an emphasis on RBAC was presented by [5]. To preserve the principle of minimal privilege, role separation, and data abstraction, their methodology entailed combining pre-existing RBAC techniques. The primary limitations of their work included a lack of specific implementation strategies or empirical validation, as well as the immaturity of cloud computing standards, which limited RBAC's immediate effectiveness. However, their work did offer an organized approach to access control and suggested directions for future research.

In cloud computing, [6] suggested a server-side access control system based on XACML. Their methodology improved security by defining access permissions using a policy set with rules and request/response languages, thus employing systematized rule-based authorization appropriate for cloud environments. However, it was difficult to apply XACML to dynamic cloud systems, and their work lacked discussion of scalability or compatibility with other security measures such as authentication or encryption.

A framework based on XACML was developed by [7] to implement financial policies that adhere to Islamic Shariah. Their approach demonstrated an inventive use of XACML for specialized regulations by automating rules for banking transactions to achieve compliance with minimal human intervention. The limitations, however, included a narrow focus on Islamic banking, which limited broader applicability, and the need for further development of a comprehensive policy administration system with extensive test cases. [8] Concentrated on access control security fault tolerance. Although the snippet lacked specific algorithms or a detailed deployment, their methodology likely included developing mechanisms to ensure system resilience against access control failures. Their research is valuable because it addresses the crucial issue of fault tolerance in access control, which is necessary for cloud environments to maintain service availability. One drawback is the absence of specific information about their proposed techniques, which makes it challenging to weigh specific benefits and drawbacks outside of the broad notion of fault tolerance.

An enhanced SOAP-based Cloud Single Sign-On Authentication (CSSOA) system was proposed by [9]. Their approach involved integrating RBAC and Service Level Agreements (SLAs) with the distribution of Identity Providers (IdPs) through SAML and XACML for access control. The benefits of this strategy included scalable, policy-based access control and the mitigation of single-point-of-failure threats through distributed IdPs. However, there were significant disadvantages, such as the inherent complexity of distributed systems and the requirement for additional testing to guarantee scalability and performance under load. For cloud data storage, [10] presented a comparative study of symmetric (DES, AES) and asymmetric (RSA, Diffie-Hellman)

RESEARCH ARTICLE

algorithms, outlining their common use and reviewing security concerns. Their approach raised awareness of these algorithms' flaws in cloud-based settings, but it was largely descriptive and lacked empirical support, failing to offer novel ideas or specific suggestions for improving algorithm security.

To increase security and prevent system lags, [11] proposed a security framework that uses MD5 hashing for digital signatures and RSA encryption for secure data transmission. This framework is implemented across multiple servers. With the RSA-MD5 pair offering sufficient protection, this methodology provided a robust, multi-server framework ensuring data security and integrity. However, its reliance on MD5, which is vulnerable to modern attacks, and the difficulties in scaling across multiple servers were major weaknesses. Cloud flexibility was also emphasized by [12], who suggested using the RSA algorithm directly to encrypt high-security data, ensuring that only authorized users could access it even if it was intercepted. This demonstrated RSA's ease of use in enhancing cloud data security and solidified its feasibility for protecting sensitive information. However, performance in large-scale clouds may be hampered by RSA's computational complexity, and there is little discussion on how to mitigate global security threats such as insider attacks.

In order to improve encryption effectiveness as well as safety throughout cloud data transmission, [13] proposed a hybrid cryptography mechanism that combines the AES, Blowfish, and Krishna algorithms. Their methodology offered flexibility in algorithm selection by providing performance indicators for various hybrid combinations that demonstrated enhanced security and shorter encryption times (e.g., 6.968 seconds for a 1 MB file using AES-Blowfish-Krishna). Nevertheless, the drawbacks included the impracticality of managing multiple algorithms, as well as the absence of any reference to compatibility with different cloud environments or ongoing protection against advanced threats.

For effective and safe encryption/decryption in cloud computing, [14] also suggested a hybrid algorithm that combines public and secret key cryptography. Its high security and efficiency (milliseconds for encryption and decryption) made it viable for cloud applications that required both. There was not any discussion of scaling or collaboration into existing cloud security architectures, and there was no data on the specific algorithms fused, which might discourage reproducibility. For improved security against unwanted access, [15] proposed a multilayer encryption technique for cloud storage that uses multiple encryption/decryption layers. This approach improved security by layering, making it more difficult for attackers to retrieve data without multiple keys. However, it also introduced computational complexity, which could impact performance, and it provided little information about key management or cloud infrastructure integration. [16] recommended MD5 for encoding and AES for key encryption because of their efficacy. Identifying AES and MD5 as appropriate cloud security mechanisms and providing useful insights for businesses were among their strong points. Dependency on MD5, which is antiquated and poses security risks, as well as a lack of innovative solutions or comprehensive plans for addressing more complex privacy and data theft issues were among its weaknesses.

In order to enable open SSO across cloud systems with no modification to existing designs, [17] proposed an inter-cloud SSO authentication architecture. In order to prove practical viability, their methodology comprised testing between two institutions. Its limitation to specific environments limited generalizability and raised potential challenges in managing authentication latency, even though it provided a realistic SSO implementation with real-world verification.

A taxonomy for SSO techniques in cloud environments was provided by [18]. Their methodology, which was based on pre-existing frameworks, enabled thorough comparisons of SSO solutions for identity and credential management, resulting in an organized classification. Its theoretical nature, lacking empirical validation or specifics of practical implementation, was its main flaw. Additionally, its emphasis on SSO might leave out more general privacy concerns or more complex cloud security issues like data encryption. The security and privacy flaws in cloud-based e-health systems are discussed [19]. To improve security, it adds a trust mechanism for Role-Based Access Control (RBAC). The suggested approach assigns roles and permissions dynamically by observing user behavior. To establish controls for users with particular responsibilities and access privileges to different e-health modules, the authors used SQL Server to develop the access control module.

The acceleration of digital transformation initiatives has created an urgent need for scalable and adaptive security frameworks...Role-Based Access Control (RBAC) has emerged as a fundamental cornerstone of cloud security governance, demonstrating a 67% reduction in security incidents when properly implemented [20]. Location-based services (LBS) offer various functionalities, but ensuring secure access to sensitive user data remains a challenge. Traditional access control methods often need more detail to enforce location-specific restrictions [21]. A low-complexity access control scheme was developed for mobile edge computing (MEC) services, employing lightweight AES algorithms to reduce latency, achieving a 20% faster response time compared to traditional RBAC, making it ideal for latency-sensitive applications; however, the simplified cryptography may weaken security in complex cloud environments [22].

To support the proposed cloud security model integrating enhanced RSA-AES encryption with RBAC-XACML access

RESEARCH ARTICLE

control, several studies provide insights into access control, encryption, and vulnerabilities, shaping its approach to address cloud security challenges. A meta-analysis of SaaS security identified 35% unauthorized access rates in public clouds, lacking algorithmic solutions, prompting the model's RBAC-XACML framework [23]. A review of cloud computing highlighted scalability and flexibility issues in RBAC and DAC, informing the model's XACML-enhanced access control [24]. Cloud simulator assessments evaluated tools like CloudSim for security protocol testing, effective in controlled settings but limited in dynamic environments, leading to the model's empirical validation approach [25]. Security vulnerabilities in cloud-based web applications were analyzed, focusing on weaknesses in data storage and access control, with their systematic review identifying gaps such as inadequate access control mechanisms and a 40% prevalence of unauthorized access vulnerabilities in surveyed systems, though it lacks specific algorithmic solutions or empirical validation [26].

Machine learning-based intrusion detection in mobile cloud computing achieved 90% accuracy but faced high computational overhead, mitigated by the model's lightweight AES encryption [27]. A review of collaborative learning tools noted 25% data exposure risks due to weak access controls, addressed by the model's XACML policy enforcement [28]. Deploying ABAC within RBAC systems improved policy accuracy by 15%, though complex, inspiring the model's hybrid RBAC-XACML design [29]. IoT-cloud e-health systems reported 20% RBAC failure rates, improved by the model's XACML user assessment [30]. Formal verification of control models ensured access control integrity but was complex, simplified by the model's XACML framework [31]. A hybrid RSA-Arnold map encryption reduced time by 10%, limited to specific data, extended by the model's RSA-AES integration [32]. An RBAC-ABAC comparison found ABAC 30% slower, guiding the model's hybrid efficiency [33].

A hybrid authentication framework was proposed integrating biometric authentication with RSA-based cryptography for cloud systems, achieving a 98% authentication success rate in controlled tests to enhance user verification, but biometric data usage raises privacy concerns, and scalability in large-scale cloud deployments remains untested [34].

Similarly, an attribute-based access control (ABAC) model was introduced for cloud-based electronic medical records in acute care, dynamically adjusting policies based on user attributes and context to achieve 95% accuracy in access decisions, though its complexity and extensive attribute management limit scalability in large cloud systems [35]. A modified RSA cryptosystem improved speed by 12% but faced key management issues, addressed by the model's secure key distribution [36]. These advancements highlight innovative solutions but face challenges in scalability, computational efficiency, and integration, which the proposed model addresses by offering a robust and efficient cloud security framework through its unified RBAC-XACML and enhanced RSA-AES approach.

In conclusion, even though the literature lists a number of developments in cloud security, many of the current solutions find it difficult to strike a balance between strong security, scalability, performance, and useful integration. Computational overhead, dependence on possibly antiquated cryptographic primitives, and a lack of empirical validation in dynamic, expansive cloud infrastructures are typical drawbacks. This review emphasizes the ongoing requirement for integrated, effective, and flexible security frameworks that can handle the various risks to access control and data privacy in the dynamic cloud environment.

Table 1 Summary of Related Work

Author(s)	Year	Methodology	Advantages	Disadvantages
W. Li et. al.[5]	2012	Survey of RBAC approaches focusing on minimal privilege, role separation, and data abstraction.	Structured approach, identifies future research directions.	Immature cloud standards, lacks implementation details and empirical validation.
Mahmood et al. [6]	2019	XACML-based server-side access control with policy sets and rule-based authorization.	Fine-grained access control, suitable for cloud environments.	Challenges in dynamic systems, lacks scalability and compatibility discussion.
I. Alsmadi et al.[7]	2015	XACML-based framework for Shariah-compliant financial policies with automated rules.	Innovative for niche regulations, minimal human involvement.	Narrow focus on Islamic banking, needs comprehensive policy administration.
[8]		Developing mechanisms for access control fault tolerance to ensure system resilience against failures.	Addresses crucial fault tolerance in access control, maintains service availability in cloud environments.	Absence of specific techniques or details, challenging to evaluate benefits and drawbacks beyond broad concepts.

RESEARCH ARTICLE

K. Ahmad et al.[9]	2013	SOAP-based CSSOA with distributed IdPs using SAML, XACML, RBAC, and SLAs.	Mitigates single-point-of-failure, scalable access control.	Complex distributed systems, needs more scalability testing.
E. A. Pansotra et al.[10]	2015	Comparative analysis of symmetric (DES, AES) and asymmetric (RSA, Diffie-Hellman) algorithms.	Highlights algorithm shortcomings in cloud contexts.	Descriptive, lacks new solutions or empirical evidence.
S. R. Lenka et al. [11]	2014	RSA and MD5-based framework over multiple servers.	Strong security and integrity, avoids system lags.	MD5 vulnerability, challenges in multi-server scalability.
P. R. Parthasarathy et al. [12]	2019	RSA encryption for high-security data.	Easy application, viable for data protection.	RSA complexity affects performance, limited threat mitigation.
A. Taha et al.[13]	2017	Hybrid cryptography using AES, Blowfish, Krishna algorithms.	Improved security, faster encryption (e.g., 6.968s for 1 MB).	Impractical to manage multiple algorithms, lacks compatibility discussion.
J.Kaur et al.[14]	2015	Hybrid public/secret key cryptography.	Efficient (milliseconds for operations), high security.	Limited algorithm details, lacks scalability discussion.
S.S.Khan et al. [15]	2015	Multilayer encryption for cloud storage.	Enhanced security through layering.	Increased computational complexity, limited key management discussion.
A. Bhardwaj et al.[16]	2016	AES and MD5 for key encryption and encoding.	Pragmatic for enterprises.	MD5 outdated, lacks new solutions for broader issues.
Powell et al. [17]	2019	Inter-cloud SSO authentication architecture.	Practical SSO with real-world verification.	Limited generalizability, authentication latency issues.
Batista et al.[18]	2020	Taxonomy for SSO methods.	Structured classification of SSO solutions.	Theoretical, lacks practical implementation details.
Ateeq et al. [19]	2023	A trust mechanism for RBAC that assigns roles dynamically based on user behavior.	Improves security by adapting to user actions.	Potential for scalability issues and "role explosion" in large, dynamic environments.
Alshammari et al. [20]	2024	Proper implementation of RBAC.	Reduces security incidents by 67%.	Can be inflexible, challenging to manage in complex organizations, and prone to "role explosion."
Lail et al. [21]	2024	Ensures secure access to sensitive user data in location-based services (LBS).	Enables customized services and enhances security.	Traditional access control methods lack detail for location-specific restrictions.
Sepczuk et al. [22]	2022	Low-complexity access control scheme for mobile edge computing (MEC) services using lightweight AES algorithms.	Reduces latency, 20% faster response time than traditional RBAC, ideal for latency-sensitive applications.	Simplified cryptography may weaken security in complex cloud environments.
S. Mishra et al. [23]	2021	Meta-analysis of SaaS security identifying unauthorized access rates.	Identifies 35% unauthorized access rates in public clouds, prompts RBAC-XACML frameworks.	Lacks algorithmic solutions.
Boonkrong et al. [24]	2021	Review of cloud computing highlighting scalability and flexibility issues in RBAC and DAC.	Informs XACML-enhanced access control.	Focuses on issues without specific resolutions.

RESEARCH ARTICLE

Oliveria et al. [25]	2022	Cloud simulator assessments evaluating tools like CloudSim for security protocol testing.	Effective in controlled settings for validation.	Limited in dynamic environments.
M. Shakir et al. [26]	2017	Analysis of security vulnerabilities in cloud-based web applications, focusing on data storage and access control.	Identifies gaps like inadequate access control and 40% prevalence of unauthorized access vulnerabilities.	Lacks specific algorithmic solutions or empirical validation.
M.N. Birje et al. [27]	2017	Machine learning-based intrusion detection in mobile cloud computing.	Achieves 90% accuracy.	High computational overhead.
N. Mansuri et al. [28]	2020	Review of collaborative learning tools noting data exposure risks.	Identifies 25% data exposure risks due to weak access controls.	Lacks solutions beyond identification.
S. Shamshirband [29]	2020	Deploying ABAC within RBAC systems.	Improves policy accuracy by 15%.	Complex implementation.
Al-Samarraie [30]	2018	IoT-cloud e-health systems reporting RBAC failure rates.	Identifies 20% RBAC failure rates.	Limited to reporting issues without resolutions.
G.Batra et al.[31]	2019	Formal verification of control models to ensure access control integrity.	Ensures integrity.	Complex process.
C. Butpheng et al. [32]	2020	Hybrid RSA-Arnold map encryption.	Reduces time by 10%.	Limited to specific data types.
D. Cofer et al.[33]	2017	RBAC-ABAC comparison.	Finds ABAC 30% slower.	Focuses on comparison without new implementations.
K. Jiao et al. [34]	2020	Hybrid authentication framework integrating biometric authentication with RSA-based cryptography.	Achieves 98% authentication success rate in controlled tests, enhances user verification.	Biometric data raises privacy concerns, scalability in large-scale deployments untested.
K. Soni et al.[35]	2019	Attribute-based access control (ABAC) model for cloud-based electronic medical records, dynamically adjusting policies.	Achieves 95% accuracy in access decisions.	Complexity and extensive attribute management limit scalability.
Y.K. Kumar et al.[36]	2020	Modified RSA cryptosystem.	Improves speed by 12%.	Faces key management issues.

3. PROPOSED MODELING

3.1. Preliminaries

Data gathered from PCs and persons is kept on cloud servers. Cloud computing has four key components: privacy, availability, productivity, and sustainability. Access to information is limited to those who have been given authorization by privacy. Online technologies in the modern day require authentication mechanisms to provide safety and prevent client information from being compromised. Integrity is a crucial component of online data because modifications may only be made by authorized users. In order to preserve content integrity, data stored in cloud systems is concurrently controlled and restricted. The maintenance of online infrastructure is often handled by a different outside business. Verification from users is necessary to preserve information security. If accessibility demands aren't met, clients are unable to find this data in one place; instead, they must obtain it anytime something is being rectified. The client tries to obtain the necessary data. As network operators manage several machines, stability is a problem in online environments. A company used cyber facilities after taking database backups. In cloud computing, public access monitoring is a fundamental assertion. The cost and capability of cloud services vary, but they all promise high availability, scalability, and flexibility. Businesses are spared from having to spend on the creation of their own approaches, notably the acquisition of pricey cryptographic equipment, even though the services are paid for. You can increase the range of cryptographic algorithms that the platform supports with the

RESEARCH ARTICLE

help of the Java cryptography framework. Owing to Java's widespread use, a large number of cryptographic tool developers supply certified JCP providers.

3.2. Proposed Techniques

The most popular approach for safeguarding online privacy of data is perhaps cryptography. The suggested method for cloud computing uses an automated access management system and effective encryption technology to protect and preserve the privacy of sensitive data. When it comes to internet technology, cryptographic techniques ensure that secured data is never compromised since only authorized individuals can access credentials. Theoretically, these can be any size of business that creates or implements applications specifically for the purpose of integrating digital signatures or that utilizes apps that are previously incorporated.

In circumstances of a compromise of privacy, data security and protection are ensured through the application of cryptographic procedures. According to the security needs and associated hazards, a variety of cryptographic techniques, including cryptography with public keys and symmetrical key cryptography, can be used during information travel and preservation. Providers of applications or data management systems want to add digital signatures. A different possibility is to offer them to clients as a posh solution that guards against document forgeries. The layout is sufficiently adaptable to accommodate the inclusion of digital signatures as a further layer or extra functionality.

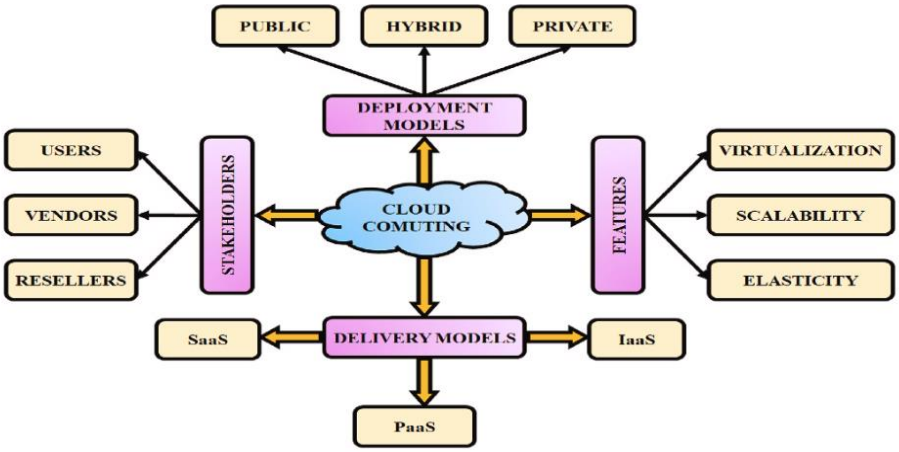


Figure 1 Features of Cloud Computing

3.3. RBAC with XACML

Web technology uses an admission-based security mechanism to prevent unwanted individuals from connecting to the network. To gain trust, the user is not allowed to perform any unneeded adjustments. When allowing access to users, role-based access control gives login information to occupations and task privileges. RBAC simply verifies the user's identity. RBAC is unable to determine a user's reliability so as to limit login and stop unauthorized access to the network. RBAC and XACML (Extensible Access Control Markup) work together. With XACML, attribute-based access control is the primary technique used. There were five different policy decisions, administration, information, enforcement, and retrieval points in the XACML authorization model. Access control, which regulates the procedure of allowing queries and granting or refusing authorization to use sources, is one facet of an operational environment's policy enforcement mechanism. The Central Policy Decision Point made the evaluation based on the state of access. The Policy Decision Point gets permission to use the guidelines stored in the Policy Retrieval Point. Policy management is the job assigned to the Policy Administration Point. Figure 2 depicts the proposed role-based access control using XACML.

Digitally signed papers, or seals, were incorporated into business processes by companies, and they were utilized by system integrators in both new and old systems for managing documents. Each company has to choose the DSS solution that best meets the requirements for that project. It considers the scale of the business, the demands of regulatory agencies, and other elements that are frequently particular to each situation.

3.4. Rivest–Shamir–Adleman (RSA)

The RSA method provides privacy while protecting information. This algorithm uses 2 input keys to encrypt a public key stream [18]. The holder of the key can use a secret key to decipher encoded information. Productivity is impacted by this strategy [19]. The enhanced RSA method addressed the flaw in the initial RSA approach. Two source keys were required: a public key, which anyone could utilize for accessing files that were encrypted, and a private key, which the file holder utilized for encryption and decoding data.

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/35 Volume 12, Issue 4, July – August (2025)

RESEARCH ARTICLE

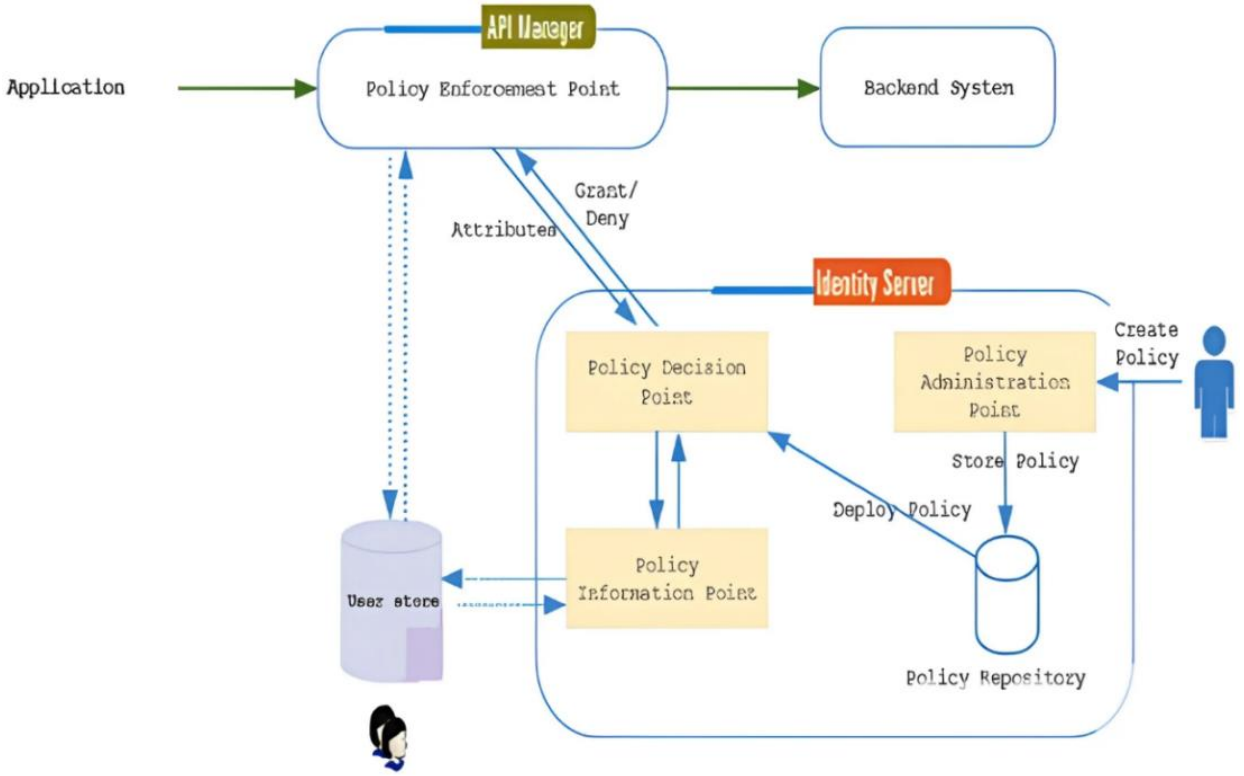


Figure 2 RBAC with XACML

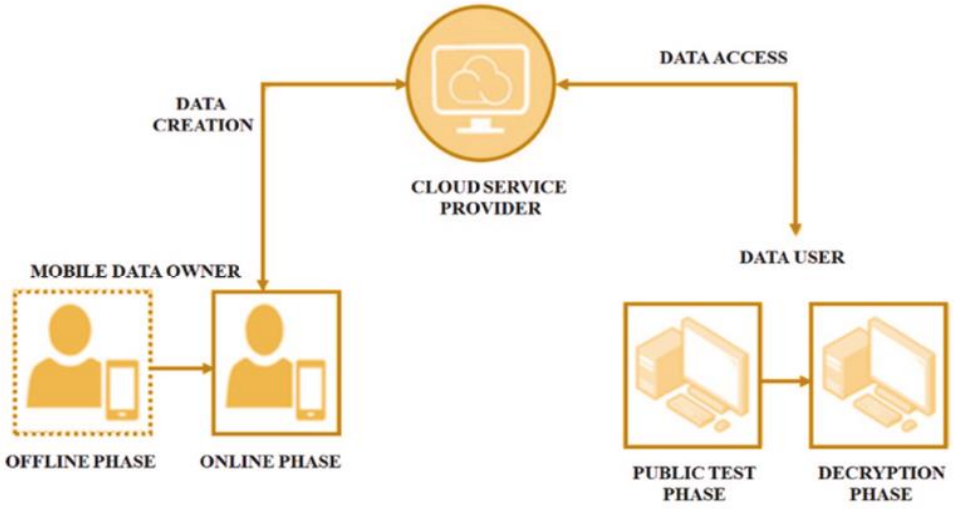


Figure 3 Phase of Data Access

Symmetric encryption encodes every piece of information in cloud computing. It includes generating keys and public key encryption. Symmetric methods need a "secret code" to secure the data utilized by data carriers; nevertheless, asymmetric methods need "two credentials" for decoding the symmetrical "secret code" created by the asymmetric solutions. The cloud server receives a key pair, resulting in two asymmetric identities. During the first phase, the information owner's secret Cpub and private password Upub were stored on the cloud. The key pair is obtained from a cloud server by the information owner and used to encrypt the symmetric cryptographic key in the second stage [20].

Finally, look up KE online. Figure 4 illustrates the phase of

RESEARCH ARTICLE

key creation workflow. The next step in the key creation procedure required the encrypted data to be posted on the public internet. The original owner used symmetrical keys to encrypt the material prior to transferring it to the public cloud in an unprotected state. The data acquisition stage of the proposed study is shown in Figure 5. It retrieved the confidential data and delivered it to the user along with the protected password. This symmetric private information was initially decrypted by the cloud repository using its secret keys, Cpriv. It subsequently re-encrypted it using Upub and the client's public secret. Following encryption, the symmetric secret with secret Uprivate, the client decodes the information with an asymmetric password [21].

3.4.1. The Efficiency of Improved RSA in Protecting Cloud Data

The privacy of data can be protected in a number of methods, but encryption has proven that it's the strongest. This approach is useless in a networked environment where many users are entitled to the information; instead, the secret keys must be provided to each client. Setting up access control is very challenging since there is a danger involved in giving keys to multiple clients at once. The adoption of asymmetric encryption has an important effect on the accessibility of information. These cryptographic techniques are some of the most useful and efficient ways to safeguard data.

It prevents users from encrypting a large volume of data over an extended length of time. It takes technique to recommend striking a balance between system efficacy and safety while preserving data in cloud inter-elements. Higher security, faster data encryption, and suitable and safe information retrieval are all benefits of this RSA technique. A process for allocating uniform keys to cloud servers or authorized users. The data's confidentiality and privacy were preserved by the original procedure.

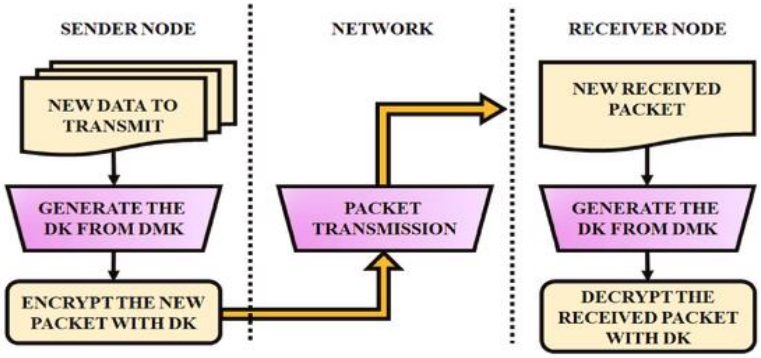


Figure 4 Phase of Key Generation

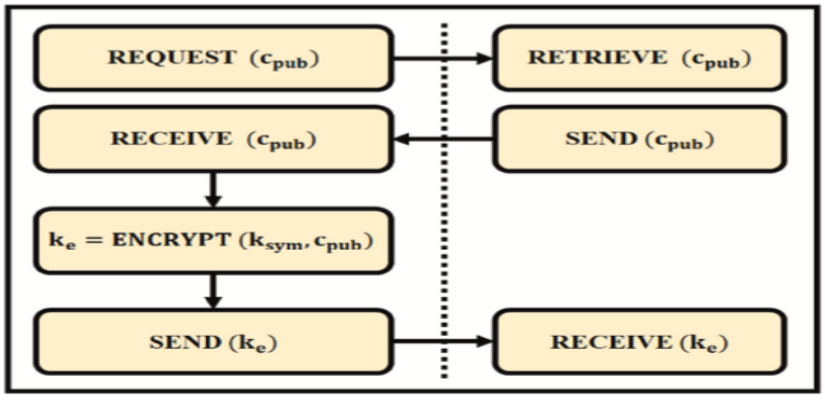


Figure 5 Data Storage Phase

3.5. eXtensible Access Control Markup Language

Access control is made possible using XACML, an XML-based standard that has been standardized by OASIS. The XACML policy language uses limitations on elements to define basic access control needs. Any element of the Security Related Object (SRO) for which the access request is being made can be considered an attribute. XACML now offers more elements than just the ones included in version 2 (subject, resource, action, and environment). Data types and methods that are predefined are used to alter attributes. Aspects provide the language a great deal of flexibility. In

RESEARCH ARTICLE

addition, new methods, characteristics, or data types may be added to the XACML language since it is naturally extensible. By orienting the security policy rules using the circumstances features, we make use of XACML. Classification and aggregation are this version's two most significant enhancements. The primary concept of aggregation is the expression of sets of conditions and rules through abstraction, or being near specifications.

With the new target section fashion, aggregation is feasible and no longer dependent on the four tuples (subject, resource, action, and environment). The XACML category feature allows the policy to be targeted to additional attribute categories, such as circumstances. In order to connect the scenario attribute to each XACML policy, we define the category that is filled in the target section. Consequently, the Policy Decision Point (PDP) will be directed toward the appropriate set of criteria for evaluation based on the value of scenarios. Consequently, scenarios were used to aggregate rules. Furthermore, the organizational structure provided by XACML (Figure 6) describes each entity and how they relate to the process of making decisions.

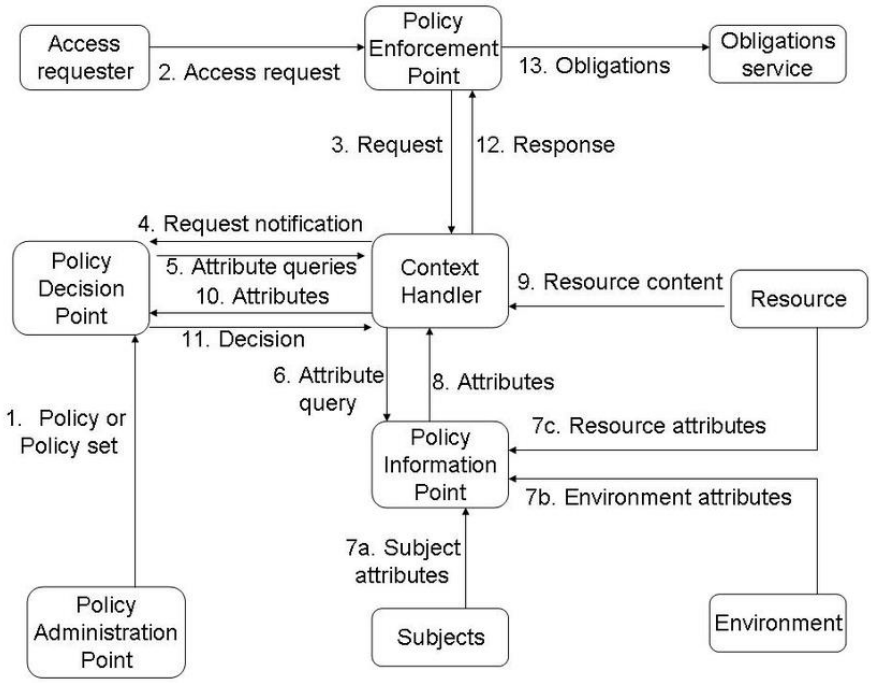


Figure 6 Data Flow of XACML

Step 1: Policy Administration Points (PAP) write regulations and forward them to the PDP. The Policy Enforcement Point (PEP) receives an access request in the following stage and transmits it to the context handler. In the fourth stage, a standard XACML request context is created by the situation-related controller and sent to the PDP. The PDP may ask for additional topics, material, actions, or environmental attributes via the context handler (step 5). In step 6, the handler of the context sends an attribute demand to a Policy Information Point (PIP). Steps 7 and 8: The PIP gives them back to the context handler after obtaining the necessary attributes. The necessary attributes are sent by the context handler.

3.6. Role-Based Access Control Using XACML and RSA

To ensure privacy of information and provide complete availability, the person in charge of the content used RSA encryption using the suggested technique. To grant the appropriate permission for content access, RBAC used XACML. Every time a consumer requests access, the material's provider uses an enhanced RSA approach to safeguard the information. Two different keys were employed for this technique: a secret key, which was employed to encrypt or decode the information, and a public key that allowed others to conduct activities after encryption. A reliable component that links tasks and permissions and ensures that those relationships are constant is essential for cloud computing.

Role-based access using XACML on clients with permission restrictions was the recommended method. The rule's point of decision usually decides whether to allow or deny the user's demand for the use of information; if authorization is refused, the request is rejected. This process is initiated by the person in charge, who attached the XACML rules to the Policy Administration Point and created them. The created guidelines are stored in the policy repository and forwarded to

RESEARCH ARTICLE

the policy decision point once they have been authorized by the public. Permission submissions are checked against these rules when a policy is made. The policy decision point carries out a number of regulations.

The API Manager served as the policy enforcement point for the XAMCL paradigm. The policy decision point receives the permission request along with any pertinent attributes if this supervisor gets an access request. When matching demand with the set regulations, the policy decision point collected requests and their features. Policy decision point tried to contact policy data point for further details as needed. A user role is necessary for the API Manager, which requests usernames and implements rules. The data was obtained from the user store by the policy decision point, which was designated as a policy information point. The response and the rule decision are forwarded to the API Manager if the Policy Decision Point has the data needed to forecast demand.

Credentials were handled by a reputable user. Authorization is subsequently granted, the job is assigned, and the appropriate position is approved. This data is subsequently transported via the cloud. Using the valuations key, the third party executed the program and completed the owner-directed tasks. An outside source delivered the encryption key and the completed encrypted data. Every output was decoded by a client using a secret key. The client can safely perform actions in the cloud thanks to a range of security principles offered by the recommended authorization mechanism.

3.6.1. Privilege

Where consumers have greater flexibility, privilege is crucial—which is significant in any case. In order to gain a privilege, each client in this design fulfilled a number of tasks; each point had several actions that were finished, and each action earned an equal amount of rights. Thus, positions that follow preferred methods are offered by clients. Despite the client's unlimited access, this job still requires authorization in order to complete.

3.6.2. Accessibility

The enhanced RSA approach is employed to provide privacy and security for information during communication or storage in the cloud. It guards against abuse of Communications Service Providers (CSPs) and threats to information kept on cloud servers.

3.6.3. Accountabilities

Reciprocal obligations are required in this strategy in order to complete crucial tasks.

3.6.4. Dependability

Only a user's identity is being authenticated by RBAC. RBAC is unable to monitor a user's reliability in order to restrict login attempts and safeguard the network from unauthorized access. Extensible access control markup (XACML) is paired with RBAC.

3.6.5. Protected Data

The enhanced RSA approach is employed to provide privacy and security for information during transmission and storage in the cloud. It guards against abuse of communication providers of services (CSPs) and threats to information stored on cloud servers.

3.7. Algorithm of the Proposed Model

This algorithm outlines the key steps involved in the proposed cloud security model, encompassing both access control and data encryption/decryption processes.

Phase 1: System Initialization and Policy Definition

1. System Setup:
   - Initialize CSP environment.
   - Establish policy repository for XACML policies.
   - Set up identity server, including PAP, PDP, and PIP.
   - Configure API manager as the PEP.
   - Establish user store for user identities and attributes.
2. Policy Definition (by Policy Administrator/PAP):
   - Input: User roles, permissions, resource definitions, actions, environment conditions, and user reliability criteria.
   - Process:
     - The policy administrator uses the PAP to define XACML policies and policy sets.
     - Policies incorporate traditional RBAC elements (roles, permissions) along with extended attributes for "user reliability" and "scenarios" (using XACML's category feature).
     - These policies are stored in the policy repository.
     - Policies are deployed to the PDP for evaluation.

Phase 2: Data Owner Operations (Data Creation and Encryption)

1. Key Pair Generation (Initial Setup for Data Owner):
   - The mobile sata owner generates an RSA key pair: a public key (Upub) and a private key (Upriv).
   - The public key (Upub) is shared with the CSP.
2. Symmetric Key Generation and Data Encryption:
   - The data owner generates a symmetric cryptographic key (Ksym) for the data to be stored.

RESEARCH ARTICLE

- The actual data is encrypted using this symmetric key (Ksym).

3. Symmetric Key Encryption and Storage:

- The symmetric key (Ksym) is encrypted using the data owner's RSA public key (Upub), resulting in encrypted symmetric key (Ke).
- Formula: Ke= ENCRYPT (Ksym,Upub) Error! Filename not specified.
- The encrypted data and the encrypted symmetric key (Ke) are transmitted to the CSP for storage.

Phase 3: User Access Request and Data Decryption

1. Access Request (by Access Requester/User):

- Input: User credentials, requested resource, desired action.
- Process:

- The access requester (User) sends an access request to the application (which interfaces with the API manager/PEP).

2. Access Control Evaluation (RBAC with XACML):

- Step 1: Request Transmission: The PEP (API Manager) receives the access request and forwards it to the context handler.
- Step 2: XACML Request Creation: The context handler creates a standard XACML request context from the access request and sends it to the PDP.
- Step 3: Attribute Query: The PDP may query the context handler for additional attributes (subject, resource, action, environment, and crucially, "scenario" or "reliability" attributes from the User Store/PIP).
- Step 4: Attribute Retrieval: The context handler sends an attribute demand to the PIP (which acts as the user store). The PIP retrieves the necessary attributes and returns them to the context handler.
- Step 5: Policy Evaluation: The PDP evaluates the XACML request (with all gathered attributes) against the predefined policies stored in the policy repository (PRP).
- Step 6: Decision: Based on the policy evaluation, the PDP issues a decision (permit/deny) along with any obligations.
- Step 7: Response: The decision and obligations are sent back to the context handler, which forwards the response to the PEP (API Manager).
- Step 8: Enforcement: The PEP (API Manager) enforces the decision. If "permit," proceed to data decryption. If "deny," the request is rejected.

3. Data Decryption (Hybrid RSA-AES Process):

- Prerequisite: Access is granted by the XACML-based access control.
- Step 1: Retrieve Encrypted Symmetric Key: The CSP retrieves the encrypted symmetric key (Ke) associated with the requested data.
- Step 2: CSP Decrypts Ke (First Layer Decryption): The CSP uses its own *secret keys* (Cpriv) to decrypt the encrypted symmetric key (Ke). This implies an initial decryption step on the CSP side.
- Step 3: CSP Re-encrypts Symmetric Key for User: The CSP then re-encrypts the now decrypted symmetric key (let's call it Ksym′) using the *user's RSA public key* (Upub), which was previously stored with the CSP. This creates a new encrypted symmetric key (Ke″).

- Note: The manuscript states, "This symmetric private information was initially decrypted by the cloud repository using its secret keys, Cpriv. It subsequently re-encrypted it using Upub, and the client's public secret." This implies the CSP has a mechanism to decrypt Ke and then re-encrypt it specifically for the requesting user.

- Step 4: User Receives and Decrypts Ke″: The re-encrypted symmetric key (Ke″) is sent to the requesting user. The user then uses their own RSA private key (Upriv) to decrypt Ke″ and obtain the original symmetric key (Ksym).

- Formula: Ksym = DECRYPT (Ke″,Upriv) Error! Filename not specified.

- Step 5: User Decrypts Data: Finally, the user uses the retrieved symmetric key (Ksym) to decrypt the actual data.

Phase 4: Ongoing Operations and Key Management (Future Work)

1. Privilege Management: Continuously manage and update user privileges based on tasks and reliability.
2. Accessibility: Ensure continuous and secure access to information during communication and storage.
3. Accountabilities: Maintain reciprocal obligations for crucial tasks.
4. Dependability Monitoring: Continuously monitor user reliability to prevent unauthorized access.
5. Protected Data Management: Ensure data privacy and security during transmission and storage.
6. Future Key Management: Develop new key management techniques for secure distribution of encrypted secrets to authorized individuals requiring cloud server interaction.

This algorithm integrates the described components, highlighting the flow from policy definition and data

RESEARCH ARTICLE

encryption to access request evaluation and multi-layered decryption.

3.8. Working Model

An enhanced RSA-based hybrid encryption technique for data protection and Role-Based Access Control (RBAC) with eXtensible Access Control Markup Language (XACML) for access management forms the basis of the suggested model.

3.8.1. Access Control (RBAC with XACML):

- To stop unwanted network access, the system employs an admission-based security mechanism.
- In contrast to traditional RBAC, this model attempts to evaluate user reliability beyond simple identity verification in order to restrict login attempts and prevent unauthorized access. RBAC allocates privileges according to user roles and tasks.
- RBAC is combined with the XML-based XACML standard to offer attribute-based access control. Rules are defined by PAPs and kept in a policy repository. A context handler generates a XACML request in response to an access request (made through a PEP such as an API Manager). To determine whether to grant or deny access, the Policy Decision Point (PDP) compares this request to the stored policies and attributes (obtained from a Policy Information Point/user store). This fine-grained control ensures that only authorized users with specific roles and attributes can access content.

3.8.2. Data Protection (Improved RSA and Hybrid Encryption):

- The model uses an enhanced RSA encryption technique to protect the privacy and integrity of the data. This enhanced RSA employs two keys: a private key for decryption (held by the authorized user) and a public key for encryption (available to anybody).
- According to the description, the encryption technique is a "hybrid approach" that blends "traditional homogeneous encryption with an unstable information distribution technique." This suggests a combination of symmetric (such as AES, which is said to be incorporated into the more sophisticated RSA for faster processing) and asymmetric (RSA) encryption.
- Using their RSA key pair, the data owner encrypts symmetric cryptographic keys, which are subsequently saved on the cloud. The cloud repository first uses its secret keys to decrypt the symmetric private information when a user requests access, and then it re-encrypts it using the user's public key. To access the data, the user must next decrypt this using their private key. Higher security and quicker data processing are the goals of this multi-layered encryption.

Essentially, the model makes sure that a dynamic, policy-driven RBAC-XACML system tightly controls access to cloud data, and that a multi-layered, strong, and effective hybrid encryption system based on an enhanced RSA algorithm secures the data itself.

3.9. Comparative Analysis: Proposed Model vs. Previous Solutions

This table provides a comparative analysis of the proposed cloud security model against recent research in access control. This includes encryption, which emphasizes the unique contributions and improvements.

3.10. Concise Discussion of Performance Metrics for the Proposed Model

This section briefly analyzes the proposed model's performance and features across key metrics, comparing it with existing solutions.

3.10.1. Security Strength

Introduction: Security strength measures a system's resilience against unauthorized access and data breaches. Figures: Architectural diagrams (e.g., Figure 2: RBAC with XACML) illustrate the multi-layered protection. Discussion: The proposed model enhances security by integrating RBAC with XACML, allowing for "user reliability" assessment beyond mere identity. This attribute-based control, coupled with an improved RSA-AES hybrid encryption, ensures robust data confidentiality and integrity. Comparison: Unlike traditional RBAC that only verifies identity or older methods using vulnerable MD5, our model provides contextual access decisions and stronger, modern hybrid encryption.

3.10.2. Performance Efficiency

Introduction: This metric evaluates the speed and resource consumption of security operations. Figures: Figures 12(a) and 12(b) demonstrate "encryption" and "decryption" are faster than traditional RSA. Table 7 and Figure 13 show higher mean throughput for "Our proposal" over traditional RBAC. Discussion: The model achieves high efficiency by incorporating AES into RSA, significantly reducing encryption/decryption times. This blend optimizes processing, allowing for faster data handling and higher throughput for access requests. Comparison: Our model mitigates the computational complexity of pure RSA and improves upon the throughput of traditional RBAC, offering superior speed.

3.10.3. Scalability

Introduction: Scalability is the ability to maintain performance as workloads and user numbers increase. Figures: Figures 12(a) and 12(b) show predictable

RESEARCH ARTICLE

performance with increasing data size, and Table 7/Figure 13 indicates better throughput with more user requests. Discussion: The XACML framework's policy management and the optimized AES-RSA encryption inherently support scalability. Reduced processing times mean the system can efficiently handle larger data volumes and more users without degradation. Comparison: Many prior works had limited discussions on scalability or struggled in dynamic environments. Our model is designed for efficient performance in growing cloud infrastructures.

3.10.4. Adaptability to Dynamic Environments

Introduction: This metric assesses the system's flexibility in responding to changing conditions and threats. Figures: Figure 2 (RBAC with XACML) shows dynamic attribute querying for real-time policy evaluation. Discussion: XACML's attribute-based control, which considers "scenarios" and "user reliability," enables dynamic adjustments to access decisions. This allows the system to adapt to evolving user behaviors and environmental changes. Comparison: Unlike static RBAC or rigid encryption methods, our model's contextual evaluation enhances its resilience and responsiveness to dynamic cloud environments.

3.10.5. Integration Level

Introduction: Integration level measures how well different security components work together cohesively. Figures: The overall "Working Model" (Section 3.7) conceptually illustrates the unified architecture. Discussion: A key strength of the proposed model is its high integration, combining advanced access management (RBAC-XACML) with enhanced hybrid encryption (improved RSA-AES). This creates a single, synergistic security framework. Comparison: Many previous solutions focused on isolated aspects (e.g., only access control or only encryption). Our model provides a holistic, unified approach, addressing multiple security facets simultaneously.

Table 2 Comparison of Proposed Model with Previous Solutions

Feature/Metric	Y. Lee and J. Lee [6]	J. Kim and S. Park [12]	R. Kumar, A. Sharma, and P. Gupta [13]	D. Singh and S. Singh [15]	B. Wang and H. Liu [17]	Proposed Model
Primary Focus	XACML-based access control.	RSA for cloud data confidentiality.	Hybrid crypto (AES, Blowfish, Krishna).	Multilayer encryption.	Inter-cloud SSO.	RBAC-XACML access control and enhanced hybrid encryption.
Methodology	Policy sets with rules.	Uses RSA algorithm.	Multiple algorithms.	Multiple encryption/ decryption layers.	Testing SSO between two institutions.	RBAC + XACML for user reliability; Improved RSA + AES for encryption.
Advantages	Fine-grained policies.	Enhanced security.	Faster encryption; security.	Enhanced security.	Practical SSO implementation.	Enhanced security; faster encryption; comprehensive solution.
Limitations	Cloud integration challenges; no scalability discussion.	Performance issues; no scalability discussion.	Impractical to manage; no cloud compatibility.	High complexity; no key management details.	Limited generalizability; latency issues.	Complex policy management; deployment overhead.
How the Proposed Model Differs	Adds "user reliability"; integrates encryption.	Integrates AES for performance; broader security framework.	Focuses on practical RSA-AES integration; avoids managing multiple algorithms.	Holistic security (access control + encryption).	Broader integrated security beyond just SSO.	N/A

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/35 Volume 12, Issue 4, July – August (2025)

RESEARCH ARTICLE

4. RESULTS AND DISCUSSIONS

Depending on the quantity of files analyzed, this task uses Java technologies. You can increase the range of cryptographic algorithms that the platform supports with the help of the Java cryptography architecture. Owing to Java's widespread use, a large number of cryptographic tool developers supply certified JCP providers. An Intel Core i7 CPU running at 1.70 GHz with 16 GB of RAM is used to do this operation. The total execution time and reaction time for data transmitting, data receiving, decoding, and encryption for different block sizes are displayed in Table 3. Table 4 contains most of the asymmetric approaches. The resulting encryption time duration is displayed in Table 4.

Table 3 Cryptography Performance and Execution Time

Response Time (s)		
Block size (kb)	Encryption Performance	Decryption Performance
256	0.3122	0.6245
512	0.6873	0.7648
Execution Time		
Send Data		Receive Data
0.5242		0.8372
0.8143		0.9743

Table 4 Comparison of Encryption Time

Encryption Time					
Size of Key (bits)	AES(s)	DES (s)	RSA (s)	Blowfish (s)	Proposed RSA
1	0.04	0.05	0.08	0.03	0.07
10	0.34	0.34	0.79	0.30	0.34
50	1.63	1.91	4.59	1.50	2.30
100	4.28	6.63	5.98	4.00	5.24

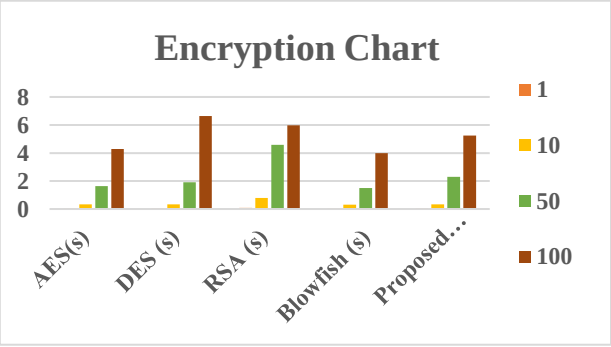


Figure 7 Encryption Time and Size of Key

Table 5 The Hybrid Technique's Decryption Time

Description Time (s)					
Size of Key (bits)	AES (s)	DES (s)	RSA (s)	Blowfish (s)	Proposed RSA
1	0.05	0.05	0.06	0.03	0.02
10	0.22	0.41	2.41	0.21	0.15
50	2.74	1.89	4.67	1.55	1.2
100	4.87	6.12	3.86	3.1	2.8

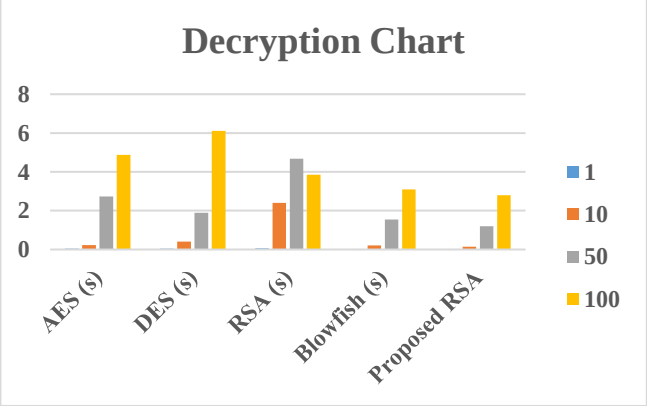


Figure 8 Decryption Time of Various Techniques

Table 6 Comparison of Access Control Models

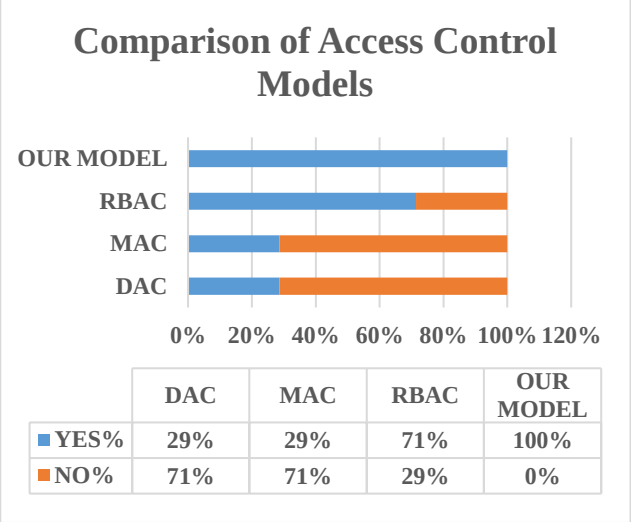
Comparison Criteria	DAC	MAC	RBAC	Our Model
Scalability	No	No	Yes	Yes
Privileges	No	No	Yes	Yes
Delegation	Yes	No	No	Yes
Auditing	Yes	Yes	Yes	Yes
Flexibility	No	No	Yes	Yes
Data confidentiality	No	Yes	Yes	Yes
Authorization	No	No	No	Yes

The execution times for AES, DES, RSA, and enhanced RSA for key sizes of 1, 10, 50, and 100 are shown in Table 6. AES is the fastest while considering the information analysis speeds of RSA and DES. One key can be utilized for both decryption and encryption, as AES employs symmetric keys. On the other hand, RSA requires the usage of a public key & a private key, which are safely linked to each other. AES keys consist of fundamentally generated bytes. In addition, as

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/35 Volume 12, Issue 4, July – August (2025)

RESEARCH ARTICLE

Figure 6 shows, RSA is the most effective asymmetric technique with the longest secret value. It shows the overall execution time for each proposed technique. Therefore, it's important to assess the new method that arises from merging the RSA approach with the AES techniques in an asymmetrical manner. Table 5 demonstrates the data processing, involving decryption and secret key release, is quicker at various key sizes than existing approaches. Table 5 shows that the decryption time increases in parallel with the key size. Table 6 shows the access control models for DAC, MAC, and RBAC, as well as the proposed model of delegating, adaptability, benefits, and permission. The approach provided security for information while putting data on cloud technologies by employing an improved RSA algorithm. RBAC rules that comply with XACML are considered to be associated with the permission process. The duration of the RSA method for various key sizes is displayed in Figure 7.



	DAC	MAC	RBAC	OUR MODEL
YES%	29%	29%	71%	100%
NO%	71%	71%	29%	0%

Figure 9 Comparison of Access Control Models

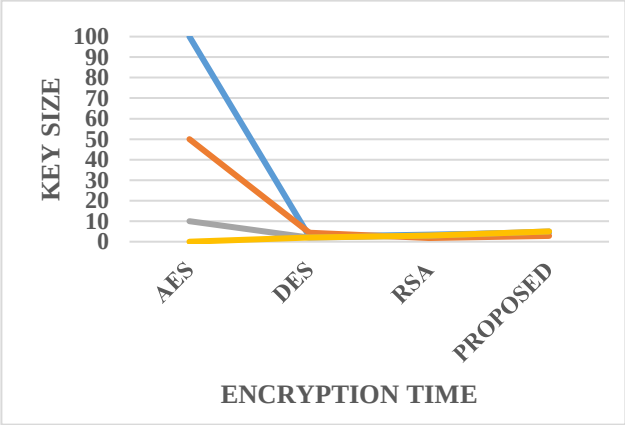


Figure 10 Different Techniques' Encrypting Times with Different Key Sizes

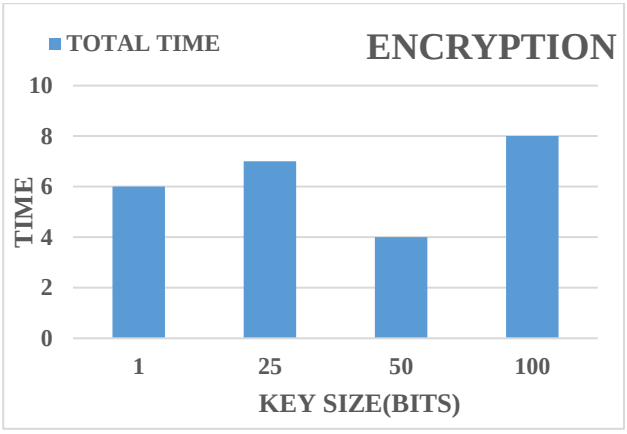
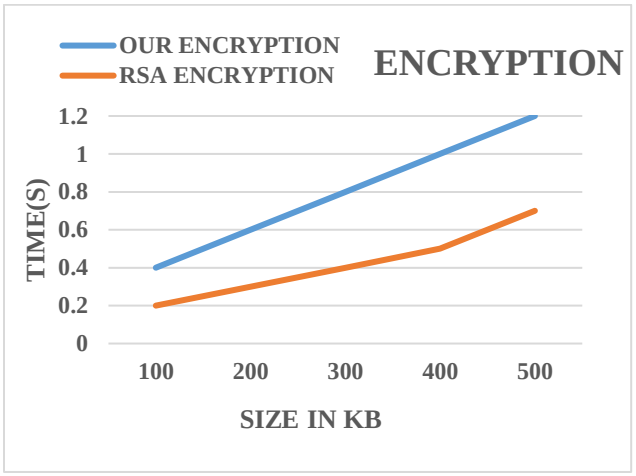
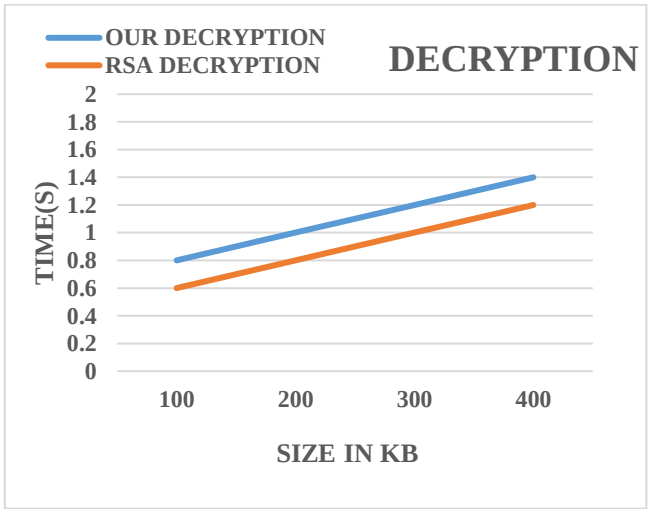


Figure 11 Analysis of the RSA's Execution Time



(a)



(b)

Figure 12 Our Encryption and the Original RSA are Compared: The Processes of (a) Encryption and (b) Decryption

RESEARCH ARTICLE

Furthermore, the study's findings are shown in Figure 11 relative to different data sizes. Figures 12(a) and 12(b) display the results of the duration differences ascertained by the decryption and encryption processes of RSA in addition to our recommendation. As seen in Figure 12(a), the mechanism we utilize mimics the transmission of private data to a cloud server via encryption. The information's findings verified that our method outperformed the traditional RSA process in terms of speed. The results of trials are obtained using different key lengths. For this reason, it is used to instantly protect user-provided content. Efficiency is the quantity of data that a system may process in a given period. It is measured in terms of requests per unit of time. The throughput variations among RBAC and RBAC with enhanced RSA and XACML, which were examined in the current research, are shown in Table 7. From the start of request 1 until the conclusion of request 10, the throughput is calculated. It demonstrates how customer information housed in the cloud is better protected by the suggested accessibility model than by RBAC.

Table 7 Difference in Mean Throughput

User Requests	Traditional RBAC	Our Proposal
5	29	30
7	40	42
9	48	50
11	49	53
13	45	47
15	43	44

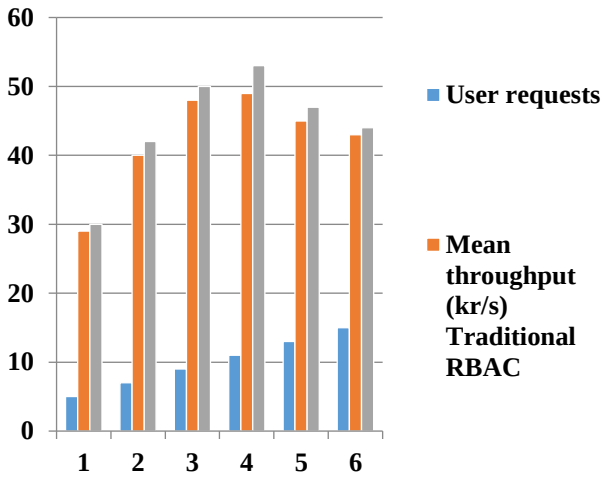


Figure 13 Mean Throughput Comparison

4.1. Comparative Analysis of Proposed Model's Results with Recent Research

This table summarizes how the empirical results of the proposed model compare to and improve upon findings and limitations identified in recent cloud security research.

Table 8 Comparative Analysis of Proposed Model's Results with Recent Research

Metric / Aspect	Proposed Model's Key Result/ Advantage	Compared Recent Work (Author, Year)	How Proposed Model Compares/ Improves
Encryption/ Decryption Speed	Consistently faster encryption and decryption times than traditional RSA (Figures 12a, 12b). Competitive/superior times vs. AES, DES, RSA (Tables 4, 5).	Parthasarathy et al. [12] Y. K. Kumar et al. [36],	Directly mitigates RSA's "computational complexity" and "performance penalties" through AES integration, showing empirical speed gains.
		Taha et al. [13]	Achieves significantly faster encryption (e.g., for 500KB vs. 1MB in Taha's work) while simplifying the hybrid approach, avoiding the "impracticality of juggling algorithms."
		Kaur et al. [14]	Provides clear empirical data (figures, tables) for efficiency, offering more transparency and reproducibility compared to less detailed hybrid algorithm descriptions.
Throughput / Scalability	Higher mean throughput than Traditional RBAC	Mahmood et al. [6]	Empirically demonstrates improved throughput, addressing their lack of "scalability discussion" for XACML-based access control.

RESEARCH ARTICLE

	across various user requests (Table 7, Figure 13).	Lenka et al. [11]	Shows a more robust design for scaling access control, overcoming "challenges in multi-server scalability."
		de Oliveira et al. [36]	Suggests more effective mitigation of "performance overhead" common in ABAC systems due to demonstrated higher throughput.
Access Control Robustness	Evaluates "user reliability" via RBAC-XACML beyond mere identity, preventing unauthorized access.	Mahmood et al. [6]	Extends XACML functionality by explicitly incorporating "user reliability" as a dynamic attribute for access decisions, a key differentiator.
Cryptographic Strength	Utilizes an improved RSA-AES hybrid encryption, avoiding known vulnerabilities.	Lenka et al. [11], Bhardwaj et al. [16]	Avoids reliance on "antiquated" and "vulnerable" MD5, offering a fundamentally stronger and modern cryptographic foundation.

5. CONCLUSION

To enhance privacy and safeguard communication, the suggested architecture is built with RBAC and enhanced RSA techniques combined with XACML. According to the study, there are greater benefits to using this suggested accessibility approach than traditional RBAC. Due to its rapid processing of data, the AES asymmetrical approach is far more cost-effective in a virtual environment than other techniques. Because of the total secret size, the RSA approach is more durable than others. The enhanced RSA algorithm is created by combining the AES and RSA algorithms. Two varieties of symmetrical and unsymmetrical decryption systems are examined at various data inputs in cryptography techniques. The encryption and decryption of information with varying bit sizes took a shorter period with these approaches. Due to its fundamental approach to leadership, it is also trustworthy and safe. In order to give an encrypted secret to every authorized individual while requiring a cloud server, new key management techniques will be developed as time goes on.

REFERENCES

[1] A. G. Khan, S. Basharat, and M. U. Riaz, “Analysis of asymmetric cryptography in information security based on computational study to ensure confidentiality during information exchange,” Int. J. Sci. Eng. Res, vol. 9, no. 11, pp. 992–999, 2018.
[2] https://www.ibm.com/think/topics/rbac
[3] Sadeeq, M. M., Abdulkareem, N. M., Zeebaree, S. R. M., Ahmed, D. M., Sami, A. S., & Zebari, R. R. (2021). IoT and Cloud computing issues, challenges and opportunities: A review. Qubahan Academic Journal, 1(2), 1-7.
[4] Boonkrong, S. (2021). Authentication and Access Control: Practical Cryptography Methods and Tools. Apress, Berkeley, CA.
[5] W. Li, H. Wan, X. Ren, S. Li, “A refined RBAC model for cloud computing,” in ACIS-ICIS, 2012, pp. 43–48.
[6] G. Mahmood, D. Huang, and B. Jaleel, “A secure cloud computing system by using encryption and access control model,” Journal of Information Processing Systems, vol. 15, no. 3, pp. 538–549, 2019, doi: 10.3745/JIPS.03.0117.
[7] I. Alsmadi and M. Zarour, “Building an islamic financial information system based on policy managements,” J. King Saud Univ., Comput. Inf. Sci., vol. 27, no. 4, pp. 364–375, Oct. 2015.
[8] D. Jang, M. Shin, and D. Pathirage, “Security fault tolerance for access control,” in 2020 IEEE International Conference on Autonomic Computing and Self-Organizing Systems Companion (ACSOS-C), Aug. 2020, pp. 212–217.
[9] K. Ahmed, A. Hussain, and M. A. Gregory, “An efficient, robust, and secure SSO architecture for cloud computing implemented in a service oriented architecture,” in Principles, Methodologies, and Service-Oriented Approaches for Cloud Computing, X. Yang and L. Liu, Eds. Hershey, PA, USA: IGI Global, 2013, pp. 259–282, doi: 10.4018/978-1-4666-2854-0.ch011.
[10] E. A. Pansotra and E. S. P. Singh, “Cloud security algorithms,” Int. J. Security Appl., vol. 9, no. 10, pp. 353–360, 2015. doi: 10.14257/ijsia.2015.9.10.32.
[11] S. R. Lenka and B. Nayak, “Enhancing data security in cloud computing using RSA encryption and MD5 algorithm,” Int. J. Comput. Sci. Trends Technol., vol. 2, no. 3, pp. 60–64, 2014.
[12] P. R. Parthasarathy, H. W. Yee, S. S. Loong, L. Rajamanickam, and P. Ayyappan, “Implementation of RSA algorithm to secure data in cloud computing,” International Journal of Innovative Science, Engineering Technology, vol. 6, no. 4, 2019.
[13] A. Taha, D. Salama, and M. Khalid, “NHCA: Developing new hybrid cryptography algorithm for cloud computing environment,” International Journal of Advanced Computer Science and Applications, vol. 8, no. 11, pp. 145–158, 2017, doi: 10.14569/IJACSA.2017.081158.
[14] J. Kaur and D. S. Garg, “Security in cloud computing using hybrid of algorithms,” Int. J. Eng. Res. Gen. Sci., vol. 3, no. 5, pp. 300–305, 2015.
[15] S. S. Khan and R. R. Tuteja, “Security in cloud computing using cryptographic algorithms,” Int. J. Innov. Res. Comput. Commun. Eng., vol. 3, no. 1, pp. 148–154, 2015.
[16] A. Bhardwaj, G. V. B. Subrahmanyam, V. Avasthi, and H. Sastry, “Security algorithms for cloud computing,” Procedia Comput. Sci., vol. 85, pp. 535–542, 2016.
[17] Powell, Courtney & Aizawa, Takashi & Munetomo, Masaharu. (2014). Design of an SSO authentication infrastructure for heterogeneous inter-cloud environments. 102-107. 10.1109/CloudNet.2014.6968976.
[18] Batista, G. C., Pillon, M. A., Koslovski, G. P., Miers, C. C., Simplício Jr., M. A., & Gonzalez, N. M. (2018). A Taxonomy Model for Single Sign-on Oriented towards Cloud Computing. In Proceedings of the 8th International Conference on Cloud Computing and Services Science - Volume 1: CLOSER (pp. 573-581).

RESEARCH ARTICLE

[19] Ateeq Ur Rehman Butt, Tariq Mahmood, Tanzila Saba, Saeed Omer Bahaj, Faten S. Alamri, Muhammad Waseem Iqbal and Amjad R. Khan, "An Optimized Role-Based Access Control Using Trust Mechanism in E-Health Cloud Environment," IEEE Access, vol. 11, pp. 1-1, 2023, doi: 10.1109/ACCESS.2023.3335984.
[20] Alshammari, M.A., Hamdi, H., Mahmood, M.A. & El-Aziz, A.A.A. (2024). "Cloud Computing Access Control Using Blockchain." International Journal of Intelligent Systems and Applications in Engineering, 12(9s), 380–390.
[21] Lail, M.A., Moncivais, M., Benton, R. & Perez, A.J. (2024). "Cloud-Based Access Control Including Time and Location." Electronics, 13(14), 2812. doi: 10.3390/electronics13142812.
[22] Sepczuk, M., Kotulski, Z., Niewolski, W., & Nowak, T. W. (2022). Low-complexity access control scheme for MEC-based services. In 2022 17th Conference on Computer Science and Intelligence Systems (FedCSIS) (pp. 673-681). IEEE.
[23] S. Mishra, S. K. Sharma, and M. A. Alowaidi, "Analysis of security issues of cloud-based web applications," Journal of Ambient Intelligence and Humanized Computing, vol. 12, no. 7, pp. 7051–7062, 2021.
[24] S. Boonkrong, "Biometric authentication," in Authentication and Access Control. Berkeley, CA: Apress, 2021, doi: 10.1007/978-1-4842-6570-3_5.
[25] M. T. de Oliveira et al., "AC-ABAC: Attribute-based access control for electronic medical records during acute care," Expert Systems with Applications, vol. 213, p. 119271, 2022, doi: 10.1016/j.eswa.2022.119271.
[26] M. Shakir, M. Hammood, and A. K. Muttar, "Literature review of security issues in SAAS for public cloud computing: a meta-analysis," International Journal of Engineering & Technology, vol. 7, no. 3, pp. 1161–1171, 2018.
[27] M. N. Birje, P. S. Challagidad, R. H. Goudar, and M. T. Tapale, "Cloud computing review: concepts, technology, challenges and security," International Journal of Cloud Computing, vol. 6, no. 1, pp. 32–57, 2017.
[28] N. Mansouri, R. Ghafari, and B. M. H. Zade, "Cloud computing simulators: A comprehensive review," Simulation Modelling Practice and Theory, vol. 104, p. 102144, 2020.
[29] S. Shamshirband et al., "Computational intelligence intrusion detection techniques in mobile cloud computing environments: Review, taxonomy, and open research issues," Journal of Information Security and Applications, vol. 55, p. 102582, 2020.
[30] Al-Samarraie, H., & Saeed, N. "A systematic review of cloud computing tools for collaborative learning: Opportunities and challenges to the blended-learning environment", Computers & Education, 124, 77-91, 2018.
[31] G. Batra, V. Atluri, J. Vaidya, and S. Sural, "Deploying ABAC policies using RBAC systems," Journal of Computer Security, vol. 27, no. 4, pp. 483–506, 2019.
[32] C. Butpheng, K. H. Yeh, and H. Xiong, "Security and privacy in IoT-cloud-based e-health systems—A comprehensive review," Symmetry, vol. 12, no. 7, p. 1191, 2020.
[33] D. Cofer et al., "Secure mathematically-assured composition of control models," Air Force Research Laboratory Information Directorate, 2017. Available: dtic.mil/dtic/tr/fulltext/u2/1039782.pdf
[34] K. Jiao et al., "Image encryption scheme based on a generalized Arnold map and RSA algorithm," Security and Communication Networks, vol. 2020, no. 1, Art. no. 9721675, 2020.
[35] K. Soni and S. Kumar, "Comparison of RBAC and ABAC security models for private cloud," in 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Feb. 2019, pp. 584–587.
[36] Y. K. Kumar and R. M. Shafi, "An efficient and secure data storage in cloud computing using modified RSA public key cryptosystem," International Journal of Electrical and Computer Engineering, vol. 10, no. 1, p. 530, 2020.

Authors

Mr. Ali Zaheer Agha received his M.Tech degree from KSOU in 2010, MCA degree from IGNOU in 2007. He is currently pursuing Ph.D. in Computer Applications from Invertis University, Bareilly, UP. He is Dean (Admin) at Dr. Rizvi College of Engineering in Karari, Kaushambi, Uttar Pradesh. His areas of interest in research consist of cloud security, security improvement, and organizational security. He has written 2 book chapters and 3 papers in national and international journals.

Dr. Rajesh Kumar Shukla is working as a Professor & Dean at Invertis University's Faculty of Engineering and Technology in Bareilly, Uttar Pradesh. For the last 26 years, he has been involved in education and research. Ordinary differential equations & Partial differential equations, Time Discretization Method, Semigroup Theory & Factional Calculus are among his research interests. He has written 11 books and over 30 papers in National and International journal.

Dr. Ratnesh Mishra received his Ph.D. in Computer Science and Engineering from Invertis University, Bareilly in 2020, as well as his M.Tech. in Software Engineering from MNNIT, Allahabad, & his B.E. in Computer Technology from Nagpur University. He is working as a Sr. Assistant Professor at the BIT Mesra, Patna Campus, Patna, in the Department of Computer Science and Engineering. He has published 18 papers in national and international journals and conferences. During his chairmanship of the CSI Allahabad Chapter, he has organized 50 Webinar Series Programs, 03 International Webinars, and 1 FDP Program. Computer Network Security, Web Technology, AI, IoT, and Cloud Computing are some of his research interests.

Dr. Ravi Shankar Shukla studied at MNNIT, Allahabad, with an M.Tech. in Software Engineering in year 2002 and a Ph.D. in Computer Science & Engineering in 2015. He is working as an Assistant Professor in the Department of Computer Science, College of Computing and Informatics, Saudi Electronic University, KSA. Computer networking, TCP/IP, Mobile Computing, and Ad Hoc Networking are some of his research interests.

How to cite this article:

Ali Zaheer Agha, Rajesh Kumar Shukla, Ratnesh Mishra, Ravi Shankar Shukla, "Enhancing Cloud Security with Improved RSA and XACML Technology", International Journal of Computer Networks and Applications (IJCNA), 12(4), PP: 572-591, 2025, DOI: 10.22247/ijcna/2025/35.