Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH

Open Access

Admission control policy and key agreement based on anonymous identity in cloud computing

D. Paulraj[1], S. Neelakandan[1], M. Prakash[2] and E. Baburaj[3*]

Abstract

Cloud computing has completely revolutionized the concept of computing by providing users with always-accessible resources. In terms of computational, storage, bandwidth, and transmission costs, cloud technology offers its users an entirely new set of advantages and cost savings. Cross-cloud data migration, required whenever a user switches providers, is one of the most common issues the users encounter. Due to smartphones' limited local storage and computational power, it is often difficult for users to back up all data from the original cloud servers to their mobile phones to upload and download the data to the new cloud provider. Additionally, the user must remember numerous tokens and passwords for different applications. In many instances, the anonymity of users who access any or all services provided by this architecture must be ensured. Outsourcing IT resources carries risks, particularly regarding security and privacy, because cloud service providers manage and control all data and resources stored in the cloud. However, cloud users would prefer that cloud service providers not know the services they employ or the frequency of their use. Consequently, developing privacy protections takes a lot of work. We devised a system of binding agreements and anonymous identities to address this problem. Based on a binding contract and admission control policy (ACP), the proposed model facilitates cross-cloud data migration by fostering cloud provider trust. Finally, Multi-Agent Reinforcement Learning Algorithm (MARL) is applied to identify and classify anonymity in the cloud by conducting various pre-processing techniques, feature selection, and dimensionality reduction.

Keywords Cloud computing, Cross-cloud data migration, Key agreement, Multi-agent reinforcement learning algorithm (MARL), Admission control policy (ACP)

*Correspondence:
E. Baburaj
baburajcse@bhu.edu.et
[1] Department of Computer Science and Engineering, R.M.K. Engineering College, Chennai, India
[2] School of Computing Science and Engineering, VIT University, Vellore, India
[3] Department of Electrical and Computer Engineering, Bule Hora University, Bule Hora, Ethiopia

Introduction

As a result of the industries' rapid growth, the widespread adoption of smartphones has occurred. In December 2018, 99.1% of China's 847 million mobile Internet users used smartphones [1]. Large data files (video, audio, and streaming media) are typically stored on a cloud server instead of smartphones due to their limited storage and processing capacity. As a result, research into cloud computing has progressed [2]. We need to increase our ability to provide customers with more convenient data storage. Smartphone manufacturers are developing and organizing cloud computing services. This facilitates contrasts between distributed computing paradigms. For data archiving, mobile device manufacturers offer their own distributed computing facilities to their customers. Numerous individuals depend on their smartphones, tablets, and other portable electronic devices. Multiple high-tech devices can be owned and utilized by a single individual. People frequently reuse their smart devices because new releases from various manufacturers include more enticing built-in functions. Suppose a user switches to a smart device manufactured by a different company. This procedure involves logging into the initial cloud worker, retrieving the required data using intelligent terminal devices, and transferring it to the new cloud worker. This cycle wastes valuable properties. Therefore, it is crucial to promote a reliable and secure method of information transfer among cloud workers [3].

Diverse cloud specialists discuss various client capacities, common skepticism, and data transmission security risks. Information bottlenecks have been the focus of recent studies. Dana Petcu stated that cloud interoperability is the most difficult challenge in distributed computing [4] and proposed an alternate strategy for cloud adaptability. The cloud migration system Binz et al. [5] developed permits the movement of composite applications within and among clouds.

Cloud computing is one of the emerging technologies with the highest growth rate in the IT industry. It integrates a variety of online resources to increase its effectiveness. Remote users can access the cloud through terminals and the Internet and pay for on-demand services on an as-needed basis [6]. Microsoft Azure, Google App Engine, Rackspace, Amazon's S3, and EC2, as well as others, are also widely used [7]. This new computing paradigm increases user agility and reduces start-up and operational costs. As more companies and individuals discover the advantages of cloud computing, they migrate their IT solutions to the cloud. Despite cloud computing's many benefits, its architecture's security is a concern [8] and demands prompt attention.

Cloud Migration is a crucial platform for managing the cloud that encounters serious issues while migrating data from an enterprise server to a server that creates a shadow in various locations. Businesses can access information in a digital environment through the cloud. Because of this, it is smooth functioning largely depends on how prepared and knowledgeable cloud vendors in this field will be. Data governance is another issue of some importance that only needs a specific technology [9]. Some businesses had data governance initiatives directly related to their migration, some had governance efforts that weren't associated with the migration, and some didn't have any data governance.

Reinforcement learning is currently a booming field. Studies have been effective in robots, game playing, and other complicated control problems since the breakthrough of deep learning approaches. Learning approaches that use neural networks as function approximators are the foundation of these findings [10]. Despite these successes, most research is focused on single-agent environments. Applications include distributed system coordination, communication package networking, financial market trading, autonomous vehicle control, and multi-robot control [11]. Each agent in these systems collaborates with the other entities in the shared environment to develop a strategy, which it then modifies to account for the other agents' changing behavior [12].

As a community, multi-agent reinforcement learning (MARL) has recently grown and experienced a boom in interest due to the developments of single-agent deep RL. A tonne of new material has just appeared [13]. The community moved past tabular problems that had previously been researched by using deep learning techniques [14].

The Main Contribution of this paper is as follows:

- Foster a culture of trust among cloud service providers and prepare for cross-cloud data migration.
- We introduce the cross-cloud computing-applicable identity-based and anonymous key agreement protocol.
- The proposed scheme Admission control policy (ACP) and Key agreement-based model enable cross-cloud data migration and boost cloud provider trust.
- Finally, Multi-Agent Reinforcement Learning Algorithm (MARL) is applied to identify and classify anonymity in the cloud by conducting various preprocessing techniques, feature selection, and dimensionality reductions.
- We investigate the types and the total number of operations that relevant security algorithms must perform, as well as the scheme's effectiveness compared to other comparable methods that have been published in the past.

The following is the format for this essay. Section 2 includes a list of relevant works. Section 3 provides, in that order, the proposed scheme and its security analysis. The performance analysis is presented in Section 4. The application scenario in Section 5 is a resolution and potential improvement.

Literature survey

A novel and decentralized access management approach for the safe storage of data on the cloud with supported anonymous authentication has been suggested by Dickson et al. [15]. Without the knowledge of the operator's personality, the authenticity of the series is verified by the cloud before the data is stored. This supports access management, meaning that only legit users can decrypt and retain the information. Replay attacks are pre-empted. The creation, alteration, and reading of the cloud data are supported. It can aid the revocation of the user. Compared with the access management schemes designed for the centralized clouds, this scheme for authenticating and access management is decentralized. The centralized approach performs the communication, computation, and storage overheads.

Considerable authority ciphertext policy that prioritizes privacy Li et al. [16] developed Attribute-based encryption (ABE) to encrypt attribute data with ciphertext and track down the identity of a dishonest user who divulges the decryption key. This system was designed to track a fraudulent user's identity while hiding attribute information in the ciphertext. Through an efficiency evaluation, it has been demonstrated that this particular method works well and that the computing cost of the tracing algorithm is inversely related to the length of the identity. It exemplifies using cloud computing to create a dependable and sophisticated data access control system.

The approach for user identification and key negotiation that Tseng et al. [17] presented was built on an identity-based cryptosystem. They completed this as a part of their employment. They requested that their protocol is safe against temporary secret leak attacks on multiple mobile servers. Their system has the lowest possible communication overhead. Analogous user authentication has been the subject of a significant amount of research.

According to Lu and his colleagues [18], their strategy does not help with identity tracking or resistance shaping. Mobile health networks with an interoperable handshake protocol were also proposed, and an Android app was developed for testing.

According to [19], a dependable real-time group data-sharing system must first overcome several obstacles. Query costs, data confidentiality, and integrity, user privacy and security, and user authentication are all client-side concerns. Access control, resource utilization, performance, and user identification and auditing issues are just a few examples of how service provider issues can manifest. There are two types of problems and concerns faced by service providers and customers.

Hana et al. [20] developed a cloud-based method called the cloud-based scheme for protecting source location privacy (CPSLP) to solve the issue of source location privacy. The authors provide a transmission method that randomly modifies packet destinations. Additionally, numerous sinks are used to develop a variety of routing pathways. The routing path becomes more unpredictable and variable with adding an intermediary node. Make a bogus cloud-based hotspot to send simulated data packets to the WSN. This perplexes the opponent's strategy and establishes a private environment. Each priceless datagram follows a circuitous path that is challenging to decipher. Intercepting datagrams becomes difficult because of this. CPSLP protects users' privacy and prevents adversarial capture, according to the simulation.

By requiring in-the-moment consumer confirmation via a public channel, multi-server authentication would allow instant access to services [21]. Since then, many multi-server authentication protocols have become available to the public. However, the academic research community is looking into more effective and reliable authentication techniques. We present an anonymous system that, at a reasonable processing cost, is resistant to the main security concerns, including insider attacks, impersonation attacks, and password modification attacks.

Furthermore, we created an elliptic, cryptography-based approach to mutual confirmation and binding agreement in distributed cloud computing. Authentication certificates are not required in this system. Zhong et al. [22] also offer an effective data movement model between cloud providers. The suggested plan establishes the groundwork for cross-cloud data migration to become a reality and promotes confidence across various cloud providers.

Panico et al. [23] present a protocol for an authenticated key agreement based on users' unique identities. This protocol was designed specifically for use with this fog setup. Thanks to this protocol, which is compatible with the fog's architecture, end devices and moisture can communicate securely without disclosing their identities to one another. The cloud server is the only place to manage devices and fog IDs. We formally prove that the session keys are secure for long-term private keys, previous session keys, and state-specific data. The Canetti-Krawczyk model assumes that attackers have access to this type of data, which is not the case here. This is because the Canetti-Krawczyk security model is

required for the security model to function correctly. The scheme employs only basic symmetric vital operations and operations on elliptic curves, which contributes to its efficiency.

Ahmad et al. [24] have highlighted the potential issues as the threat of cloud service abuse, BotNet, BotCloud, Shared Technology Vulnerability, and Malicious Insiders. It also details how the challenges were attacked, how they affected things, and why. The study assessed the presently available remedies and suggested mitigating security measures to address the security risks and difficulties created by the threat of cloud service abuse.

Manzoor introduced a marketplace for buying and selling IoT data based on blockchain with his colleagues [25]. employ proxy re-encryption to maintain the security of all information sent between the producer and the consumer. Data is encrypted and uploaded to cloud storage. Real-time intelligent contracts are created between the data collector and sensor to exchange IoT data without a reliable intermediary. Proxy re-encryption is used. Only the owner of the smart contract and the current individual have access to its data. Smart contracts and proxy re-encryption make it possible to securely store, trade, and control sensor data in a timely and cost-effective manner.

Anthi et al. [26] described an IoT infrastructure system that uses a supervised approach known as a 3-layer system for intrusion detection to identify security problems in intelligent home-based IoT platforms. This system consists of three entities: the first entity represents the usual behavior of the IoT device profile and kind, and each device is linked to intelligent home network-based IoT devices. Second, the network system detects malicious transmitted packets on the network. Finally, the network implemented in the system classifies the type of assault. Thus, the performance of the network activity and automated function from natural testbed settings is evaluated using a machine-learning approach. As a result, the IDS detects the attack deployed on the networks. The network, however, does not support automatic detection.

Punithavathi et al. [27] presented a safe, lightweight authentication strategy based on a cancelable biometric system (CBS) in the cloud environment. The first step in biometric image processing is to perform Region of Interest (ROI), thinning, binarization, and histogram localization. The whole procedure of the feature extraction stage is then obtained. Finally, the corresponding templates are recovered and stored in the Cancelable Template Database (CTD). A new cancelable template is then used to construct the further transformation of the authorized key. The evaluation considers real-world situations to authenticate device data with little overhead and high accuracy. This method is also supported in intelligent IoT environments.

Gochhayat et al. [28] proposed a distributed critical management method for IoT security. This method effectively secures IoT devices by offloading the most resource-intensive cryptographic processes to a local organization. This technique has a lower communication overhead and generation time. However, storage costs a lot of money.

Shahid et al. [29] proposed a Proficient Security over Distributed Storage (PSDS) technique to address data security concerns during multi-cloud data transmission. The cloud database divides the data into two categories for this purpose: sensitive and normal. Standard data is encrypted and uploaded by a single cloud, whereas sensitive data is encrypted and distributed by multiple clouds. As a result, the PSDS approach secures the data against different threats. However, the PSDS technique cannot encourage validating the essential agreement between the user and the cloud service by a trusted authority. A summary of the vital methodologies in this area of study is listed in Table 1.

Table 1 Admission control policy and Key Agreement Based on Anonymous Identity

Author	Proposed System	Algorithm	Year
Nirmal Kumar et al. [30]	Deep Reinforcement Learning	multiagent deep RL (MADRL)	2020
Qu et al. [31]	Reinforcement Learning	Policies for multi-agent Networked Systems	2020
Spooner et al. [32]	Through Adversarial Reinforcement, Learning	Strong market-making is possible thanks to adversarial reinforcement learning.	2020
Manzoor et al. [25]	Proxy broadcast repeat encryption	The blockchain-based platform for sharing Internet of Things data is secure and anonymous thanks to proxy re-encryption.	2021
Patonico et al. [23]	Canetti and Krawczyk developed a risk assessment model.	Canetti-Krawczyk-resistant protocol for anonymous and identity-based fog computing	2019
Chomping et al. [33]	Revocable identity-based broadcast proxy re-encryption for data sharing in clouds (RIBBPRE)	In RIB-BPRE, the agent has the re-encryption key and can revoke the authority of a delegation group.	2019

Proposed system

As a potential solution, this study proposes a multi-server anonymous authentication protocol. Even though the proposed protocol necessitates additional processing on the server's end, it is commonly assumed that the server has sufficient resources. Therefore, since the server can support it, the user's computation cost is reduced. In addition to the user (Uu) and the server, the system employs a multi-server architecture. RC facilitates the distribution of server-side resources and facilitates user registration. Here, we use x, the RC system's controller secret key user registration process. The three phases of the new scheme, like its predecessors, are authentication, registration, and password reset. Figure 1 depicts the suggested block diagram; more information is provided below.

Server registration phase

To become a legitimate server, it must follow the steps necessary to create an account.

SR Step1: S_i prefers to introduce himself as IE_i, and it does so over a secure, encrypted connection.

SR Step2: Following the acquisition IE_i, RC of calculations.

After obtaining the numbers IE_i, RC calculates

$$s = h\left(IE_i \| y\right) \tag{1}$$

$$pks_i = sP \text{ and } pk_{RC} \tag{2}$$

Where y is the one who keeps the hidden key.

SR Step3: After that, RC sends s, pks_i, pk_{RC} and S_i and cancels the registration. Table 2 shows commonly used notations.

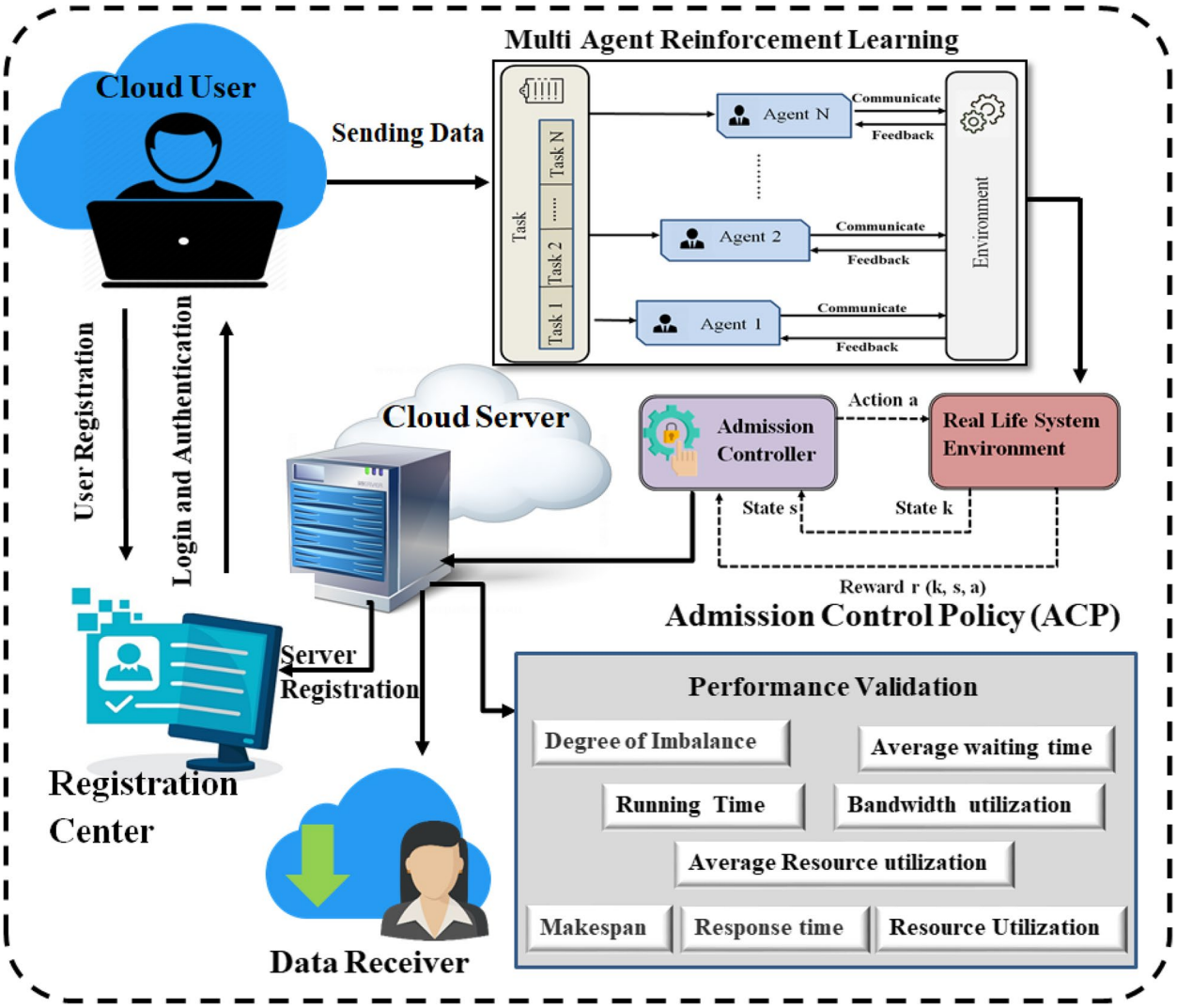


Fig. 1 The proposed method of MARL-ACP

Table 2 Commonly used notations

Common notations	blueElucidations
Uu	uth user of the system
RC	Centralized registration center of the infrastructure
IDu	Specific user's identity
PWu	Specific user's password
Bu	Biometric identity of specific user
PIDu	User's pseudo identity
SCu	Smart card issued to each specific user
Sj	jth service provider of the infrastructure
IDj	Identity of service provider
X	Secret key of RC
pkRC	Public key of RC
pkSj	Public key of Sj
S	Secret key of Sj
P	Base point of the elliptic curve Ep(e,f)
H(.)	Function specified for Bio-hash
h(.)	One-way digest function of hashing

User registration phase

V_v uses to become a legal network user.

UR Step1: The operator selects his identity IE_v, password PW_v, biometric impression T_v and produces an arbitrary nonce a. Ten user controls

$$N = H\left(IE_v \middle\| T_v\right), \tag{3}$$

$$BW = h\left(a \oplus H\left(T_v \middle\| PW_v\right)\right) \tag{4}$$

and sends IE_v, N, BW to RC for implementation of the registration.

UR Step2: After that, RC determines.

$$Y_v h\left(IE_v \middle\| pk_{RC}\right) \tag{5}$$

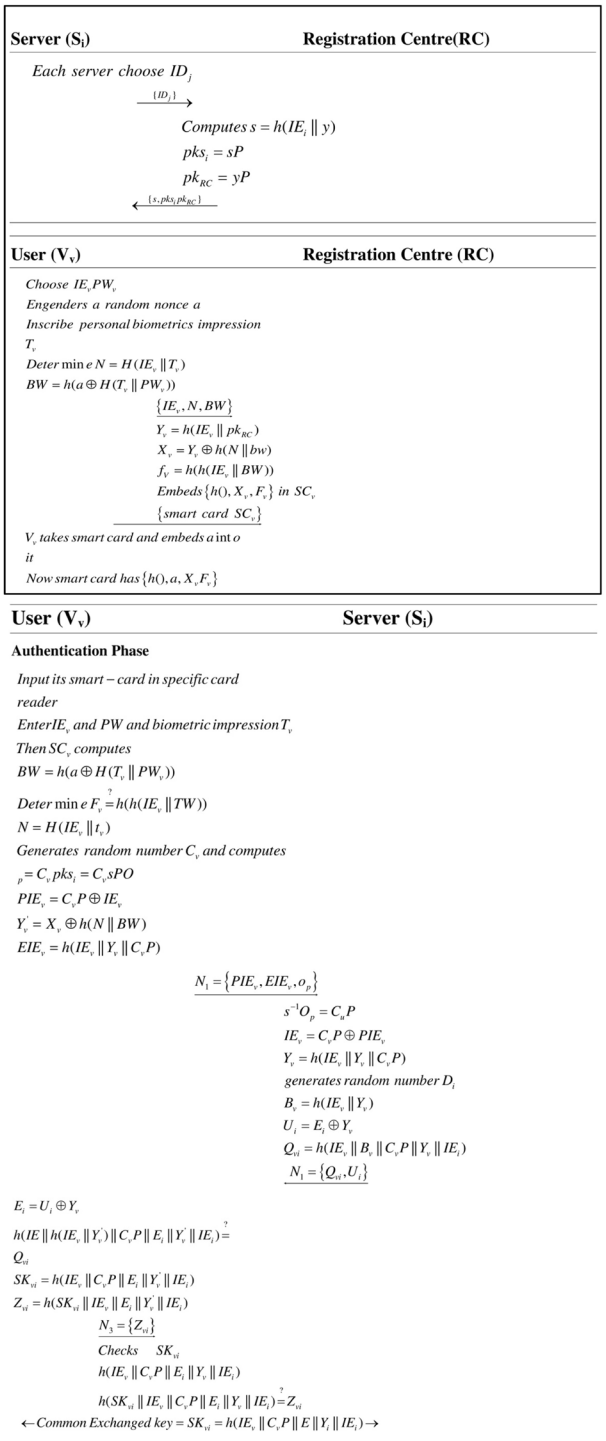
$$X_v = Y_v \oplus h\left(N \middle\| BW\right) \tag{6}$$

and

$$F_v = h\left(hIE_v \middle\| BW\right)\Big) \tag{7}$$

then RC stores $h()$, X_v, F_v in intelligent card and sends (SC_u) towards V_v.

UR Step3: V_v also incorporates a numerical into SC_u. Now smart card has $\{h(), X_v, F_v, a\}$.



Algorithm 1 Proposed Scheme

Login and authentication phase

In this phase, the user V_v is authenticated so that they can proceed to Phases 2 and 3 to show with service providers S_i, V_v, *and* S_i respectively.

LAP Step1: V_v inputs IE_v, password PW_v and scan biometric impressions in the scanner. Ten smart cards determine.

$$BW = h\left(a \oplus H\left(T_v \middle\| PW\right)\right) \tag{8}$$

and checks whether

$$F_v \stackrel{?}{=} h\left(h\left(IE_v \middle\| BW\right)\right). \tag{9}$$

If yes, then determines.

$$N = H\left(IE_v \middle\| T_v\right), \tag{10}$$

Uu creates a random number C_v and computes.

$$O_p = C_v pks_i = C_v sP, \tag{11}$$

$$PIE_u = C_u P \oplus IE_v, Y_v^{'} = X_v \oplus h\left(N \middle\| BW\right) \tag{12}$$

and

$$EIE_v = h\left(IE_v \middle\| X_u^{'} \middle\| C_v P\right). \tag{13}$$

Ten V_v sends $N_1 = PIE_v, EIE_v, O_p to S_i$.

LAP Step2: After receiving $N_1 = PIE_v$, EIE_v, $O_p to\ S_i$ using his secret key s computes.

$$s^{-1} O_p = C_v P, \tag{14}$$

$$IE_v = C_v P \oplus PID_v \tag{15}$$

and

$$Y_v = h\left(IE_v \middle\| pk_{RC}\right)\right). \tag{16}$$

After that S_i checks

$$EIE_v \stackrel{?}{=} h\left(IE_v \middle\| Y_u^{'} \middle\| C_v P\right). \tag{17}$$

RC will generate a random nonce E_i and determine the course of action to be taken if it turns out to be correct.

$$U_i = E_i \oplus Y_v \tag{18}$$

and

$$Q_{vi} = h\left(IE_v \| B_v \| C_v P \| E_i \| Y_v \middle\| IE_i\right). \tag{19}$$

Subsequently, S_i sends a message.

$$N_2 = Q_{vi}, U_i to\ V_v \text{u}. \tag{20}$$

LAP Step3: V_v determines $E_i = U_i \oplus Y_v$ after receiving N_2 and checks.

$$h\left(IE_v \middle\| h\left(IE_v \middle\| Y_v^{'}\right) \middle\| C_v P \| E_i \| Y_v^{'} \middle\| IE_i\right) = Q_{vi} \tag{21}$$

LAP Step4: If the

$$h\left(IE_v \middle\| h\left(IE_v \middle\| Y_v^{'}\right) \middle\| C_v P \| E_i \| Y_v^{'} \middle\| IE_i\right) \stackrel{?}{=} Q_{vi}(IE) \tag{22}$$

holds true, V_v further determines.

$$SK_{vi} = h\left(IE_v \| C_v P \| E_i \middle\| Y_v^{'} \middle\| IE_i\right) \tag{23}$$

and computes

$$Z_{vi} = h\left(SK_{vi} \| IE_v \| E_i \middle\| X_v^{'} \middle\| IE_i\right). \tag{24}$$

V_v sends $N_3 = Z_{vi}\ S_i$ so it can check on E_i 's challenge.

LAP Step5: After getting N_3, the server S_i determines.

$$SK_{vi} = h(IE_v \| C_v P \| E_i \| Y_v \| IE_i). \tag{25}$$

It demonstrates that the equation is correct. i.e., $h\left(SK_{vi} \| IE_v \| C_v P \| E_i \| Y_v \middle\| IE_i\right) \stackrel{?}{=} Z_{vi}$. Finally, if the justification is successful, the server will exchange the session key SK with the user $h(IE_v \| C_v P \| E_i \| Y_v \| IE_i)$. Algorithm.1 describes this protocol.

Task admission control policy algorithm

An algorithm for task admission based on RL is suggested in this paper. The pseudo-code is in Algorithm.2. The RL-based policy procedure has two loops with execution times. The RL-based policy algorithm requires storage space for intermediate variables. As a result, the RL-based policy procedure has both a high time complexity and an increased space complexity [30–32]. Figure 2 depicts how the admission controller learns by interacting with the system's atmosphere. The RL-based policy procedure generates the approximate optimal policy from real-world data or system simulations without having comprehensive system info. This policy can be created using real-world or simulated data.

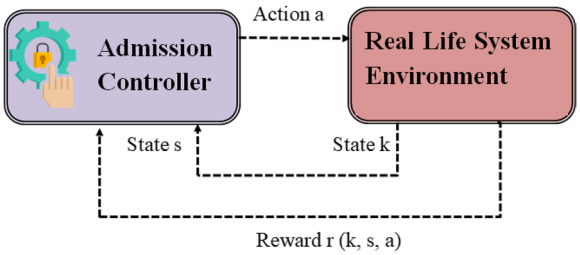


Fig. 2 The RL model is used in the proposed task admission control policy algorithm

As a result, it can be used in more complicated situations. The long-term average, the policy is developed using Q-learning, a value iteration-based RL method [33–35]. After learning an action-value function, the admission organizer can make decisions for each state-action pair. This is possible with Q-learning. The optimal action for a given state is the one with the highest action-value function value.

The QoS constraint must be considered during the learning process, and the system's state must be modified to accommodate the condition. The following description separates the accomplishment value function and the QoS constraint into their respective roles as fundamental building blocks of the RL-based policy procedure.

$01: initialize\ n=1, u=0, v=1, Q(\hat{s},a)=0, Q^{+}(\hat{s},a), \rho=0, \rho^{+}=0, r_n=0, \tau_n=0$
$02: repeat$
$03: \quad repeat$
$04: \quad \hat{s} \leftarrow current\ state$
$05: \quad if(\varphi \le P(a_{greedy} \rightarrow a_{random}))then$
$06: \quad\quad a \leftarrow a_{greedy};$
$07: \quad else\ then$
$08: \quad\quad a \leftarrow a_{random};$
$09: \quad endif$
$10: \quad simulates\ the\ next\ state\ k$
$11: \quad r_n \leftarrow r_n + r(\widehat{k,s},a,\omega);$
$12: \quad \tau_n \leftarrow \tau_n + \tau(\widehat{k,s},a);$
$13: \quad updates\ \rho,\alpha,\gamma,Q(\hat{s},a)$
$14: \quad m \leftarrow m+1:$
$15: \quad until\ m=M+1$
$16: if(P^{r}_{\omega_y} - P^{r}_{l} \le \varepsilon_1 \,\&\&\, \rho > \rho^{+})then$
$17: \ \rho^{+} \leftarrow \rho;$
$18: Q^{+}(\hat{s},a) \leftarrow Q(\hat{s},a)$
$19: endif$
$20: updates\ \ \omega$
$21: u \leftarrow u+1$
$22: until\ u \le m$
$23: return\ admission\ control\ poliy\ \pi(\hat{s}) derived\ \ from\ Q^{+}(\hat{s},a)$

Algorithm 2 RL-based Admission control policy

Data migration ensures that data is successfully and securely transferred to another application, storage system, or cloud. Although moving data from one platform to another can be risky and costly, it provides an organization with numerous benefits. For example, in addition to upgrading applications and services, organizations can boost their productivity and reduce storage costs.

The following best practices should be used to protect data during a migration:

- Back up before migrating data. If something goes wrong during migration and the data is lost, it can be restored from the backup.
- Understand what data is being migrated, where it lives, what form it's in, and what it will take at its new destination.
- Extract, transform, and deduplicate data before moving it.
- Implement data migration policies, so data is moved in an orderly manner.
- Test and validate data migration during the planning and design phase to ensure it's accurate.
- Audit and document the entire data migration process.

Feature extraction using multi-agent reinforcement learning

Markov Games are a widespread basis for multi-agent sequential decision-making. Littman proposed the Markov Game to extend MDPs to include multiple agents interacting in a shared environment and possibly with one another. The Markov Game generalizes MDPs. The notation for discrete time is:

Definition 1 *The Markov Game generalizes MDPs. The notation for discrete time is:* $(\aleph, \chi, \{\upsilon^i\}, \rho, \{R^i\}, \gamma)$, *where.*

$$\aleph = \{1, \ldots N\} \tag{26}$$

Represents the group of $N>1$ agents that communicate with one another, and χ is the shared state space for all agents. The term "joint action space" refers to the

$$\upsilon = \upsilon^1 \times \cdots \times \upsilon^N \tag{27}$$

Which is the sum of all agents' action domains $i \in \aleph$. The transitional probability function $\rho: \chi \times \upsilon \rightarrow P(\chi)$. It has some reward functions. $R^i: \chi \times \upsilon \times \chi \rightarrow \mathrm{R}$ As a result, an immediate feedback signal is generated. Finally, $\gamma \in \{0,1)$ defines how the discounting factor works.

Each currently has a representative $i \in \aleph$ creates and implements a policy action plan $\Pi^i: \chi \rightarrow P(\upsilon^i)$. Because of everyone's hard work, the system is no longer in its original configuration ρ considering the likelihood of various states, x_{t+1} to the following condition R^i but each legislator is given (Fig. 3).

Strategic-form game 2 is a stateless MG particular case. "Strategic-form games" are one-shot connections in which all agents perform an action concurrently and receive a reward before the game ends.

The simplified investigation of this stateless setting has advanced MARL. This setting is still being

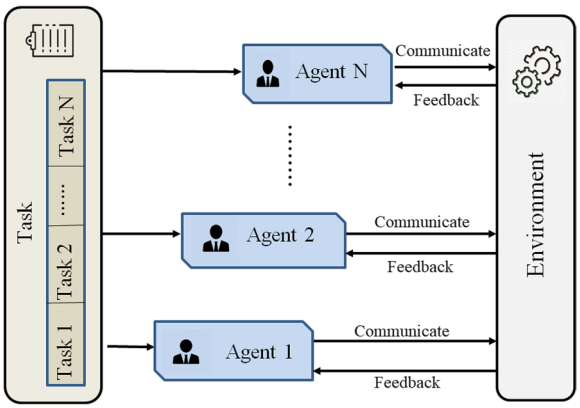


Fig. 3 Multi-Agent Reinforcement Learning Algorithm (MARL)

researched for the treatment of a variety of pathologies. Extensive-form game is a formalism with multiple stages.

The value function differs for each agent.

$$V^i = \chi \to \mathbb{R} \tag{28}$$

The agent i's value function is the expected sum of his and other agents' policies.

$$V^i{}_{\pi^i,\pi^{-i}}(x) = \mathrm{E}_{x_{t+1}\sim\rho,u_t\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t R^i\left(x_t,u_t,x_{t+1}\right)|x_0 = x\right] \tag{29}$$

When individual agents follow the guidelines of the policy π. We represent the collective policy. $\pi:\chi\to P(\upsilon)$ as a set of different rules, $\pi=\{\pi^1,....\pi^N\}$.

$$\pi^{-i} = \left\{\pi^1,\ldots,\pi^{i-1},\pi^{i+1},\ldots\pi^N\right\}. \tag{30}$$

Individual and group strategies determine the optimal policy. Agent π^i_* can maximize its efficacy when the other agents' strategies are fixed by determining the optimal response.

Definition 2 *The best response* $\pi^i_* \in \Pi^i$ *to of the agent to the joint policy* π^{-i} *of other agents is*

$$V^i_{\pi^i_*,\pi^{-i}}(x) \geq V^i_{\pi^i_*,\pi^{-i}}(x) \tag{31}$$

for all states $x\in\chi$ and policies $\pi^i\in\Pi^i$.

If all agents learn simultaneously, the optimal response may not be truly unique. The primary and secondary influential game theory solution concept is the Nash equilibrium, which can be described as the best response.

Definition 3 *Nash's Equilibrium A solution in which the policy of each agent is considered* π^*_i *the most effective way to counteract competing brokers' strategies* π^{-i}_* *To demonstrate the following discrepancy,*

$$V^i_{\pi^i_*,\pi^{-i}}(x) \geq V^i_{\pi^i_*,\pi^{-i}}(x) \tag{32}$$

We know we're in a Nash equilibrium if "holds for all states $x\in\chi$ and all policies $\pi^i\in\Pi^i\,\forall$i. ".

In a Nash equilibrium, no agent can recover their position by deviating individually from the policies of all other agents. An agent needs to improve their situation. However, a Nash equilibrium is not the only possible result. Pareto-optimality might be helpful.

4th Definition of Pareto-optimality a joint policy 𓃹 If and only if a second consensus policy meets the Pareto threshold $V^i_\pi(x) \geq V^i_{\hat{\pi}}(x)\ \ \forall i,\forall \in \chi, V^i_\pi(x) > V^i_{\hat{\pi}}(x)\ \ \exists j,\exists x \in \chi$.

If there is no more excellent equilibrium and the one in question is not Pareto-dominated, it is a Pareto-optimal Nash equilibrium. Canonical MARL literature can be divided into mission and agent resources. This section will present MARL concepts according to Busoniu et al. taxonomy [36, 37]. First, the type of task influences the learned agent's behavior. The incentive structure promotes either competition or cooperation.

(1) Collaborative environment Each agent receives the same reward.

$$R = R^i \cdots = R^N \tag{33}$$

Change state Agents are motivated to work together in an equal-rewards environment because they want to avoid individual failure to maximize team performance. When actors are encouraged to work together but not equally rewarded, we call that a collaborative setting.

(2) Competition-based This is a zero-sum Markov Game, meaning the total rewards remain the same regardless of the chosen state. Agents should maximize their rewards and minimize their peers'. "Competitive games" refers to situations in which players are incentivized to perform well against other players, but the total number of rewards is not zero.

(3) Ambient The mixed location is neither cooperative nor competitive, so the agent's goals are unrestricted. General-sum games are similar.

In addition to other taxonomies, the reward structure can be used to differentiate agent information.

Result and discussion

Performance analysis

In this unit, we compare the efficacy of our protocol to the Degree of Imbalance, Makespan, Average Resource

utilization, Average waiting time task, Response time, Resource Utilization, and Bandwidth utilization. It will be necessary to consider both the communication cost and the security properties. Contrasting the proposed method with Proxy broadcast repeat encryption (PBRE), secure data sharing cloud (SeDaSC), Certificateless public key cryptography (CL-PKC), peer-to-peer cloud authentication, and key agreement (PCAKA).

Using the JPBC, we implemented four schemes on both the client and server. Android Studio bundle version 2.2.0.0 simulates the client on an Honor phone running EMUI 4.0.1 with a 1.5 GHz octa-core CPU and 2 GB RAM. The simulation was run on Intel Core-i 3–3110 servers with dual 2.40 GHz cores and 4 GB of RAM using Java. Table 3 presents the analysis of our schemes with related schemes [16–18, 20, 22]. As per the analysis, we can conclude that our protocol is more secure than [16–18, 20, 22]. All these protocols depend upon hash-based symmetric cryptography and are similar.

Degree of imbalance

Figure 4 and Table 4 show a degree of imbalance examination of the MARL-ACP approach compared to other existing methods. The graph shows that using cloud computing has resulted in higher performance with a lower degree of imbalance. For example, with task 100, the MARL-ACP model has a degree of imbalance of 25.65, whereas the PBRE, SeDaSC, CL-PKC, and PCAKA models have slightly higher degrees of imbalance of 38.21, 35.52, 33.87, and 29.64, respectively. The MARL-ACP model, on the other hand, has demonstrated maximum performance for various tasks with a low degree of imbalance values. Similarly, the degree of imbalance value of MARL-ACP under 300 tasks is 25.88, 38.43, 36.72, 34.62, and 32.98 for PBRE, SeDaSC, CL-PKC, and PCAKA models, respectively.

Makespan

Figure 5 and Table 5 show a Makespan comparison of the MARL-ACP approach with other existing methods. The graph shows that the cloud computing strategy yielded higher performance with a lower Makespan value. For 100 tasks, the Makespan value for MARL-ACP is 98.78 sec, while the PBRE, SeDaSC, CL-PKC, and PCAKA models have slightly higher Make span values of 120.65 sec, 133.98 sec, 122.13 sec, and 142.98 sec, respectively. The MARL-ACP model, on the other hand, has demonstrated maximum performance for various data sizes with low Makespan values. Similarly, for 300 tasks, the Makespan value of MARL-ACP is 118.22 sec, while for PBRE, SeDaSC, CL-PKC, and PCAKA models, it is 132.98 sec, 137.43 sec, 132.67 sec, and 149.21 sec, respectively.

Average resource utilization

Figure 6 and Table 6 show a comparison of the average resource utilization of the MARL-ACP approach with other existing methods. According to the graph, the cloud computing strategy has resulted in higher performance with average resource utilization. For example, with task 100, the MARL-ACP model has an average resource utilization of 93.78%, whereas the PBRE, SeDaSC, CL-PKC, and PCAKA models have average resource utilization of 72.78%, 76.46%, 79.56%, and 84.21%, respectively. The MARL-ACP model, on the other hand, has demonstrated maximum performance with various tasks. Similarly, for 300 tasks, the average resource utilization value of MARL-ACP is 94.61%, while for PBRE, SeDaSC, CL-PKC, and

Table 3 Comparison of security parameters

Scheme	MARL-ACP	Li et al. [16]	Tseng et al. [17]	Lu et al. [18]	Hana et al. [20]	Zhong et al. [22]
Immune to smart card stolen attack	Yes	Yes	Yes	Yes	No	No
Efficient password modification	Yes	Yes	Yes	No	No	No
Ensuring anonymity	Yes	Yes	Yes	Yes	Yes	Yes
Immune to insider attack	Yes		Yes	Yes	Yes	Yes
Immune to trace attack	Yes	Yes	Yes	Yes	Yes	No
Immune to impersonation attack	Yes	No	No	No	No	No
Support mutual authentication	Yes	No	No	No	No	Yes
Repair ability	Yes	Yes	No	No	Yes	Yes
Supports session key security	Yes	Yes	Yes	Yes	Yes	No
Immune to ofine password guess- ing attack	Yes	Yes	Yes	No	No	Yes
Immune to KCI attack	Yes	Yes	Yes	Yes	Yes	No

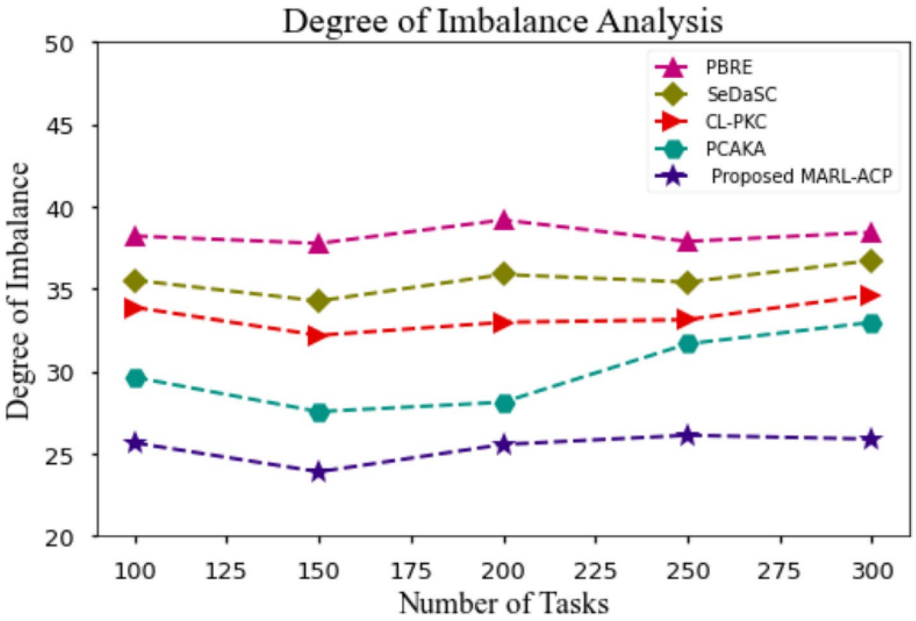


Fig. 4 Degree of Imbalance Analysis for MARL-ACP method with the existing system

Table 4 Degree of Imbalance Analysis for MARL-ACP method with the existing system

No of tasks	PBRE	SeDaSC	CL-PKC	PCAKA	MARL-ACP
100	38.21	35.52	33.87	29.64	25.65
150	37.76	34.26	32.19	27.55	24.89
200	39.19	35.87	32.98	28.12	25.55
250	37.88	35.41	33.13	31.67	26.11
300	38.43	36.72	34.62	32.98	25.88

PCAKA models, it is 75.87%, 78.21%, 82.53%, and 85.34%, respectively.

Average waiting time task

The average waiting time analysis of the MARL-ACP technique using existing methods is described in Table 7 and Fig. 7. The graph clearly shows that the MARL-ACP process has outperformed the other techniques in all aspects. For example, with 100 tasks, the MARL-ACP method took only 1.654 seconds as average waiting time, while other existing techniques such as PBRE, SeDaSC, CL-PKC, and PCAKA took 7.193 sec, 6.124 sec, 5.198 sec, and 2.675 sec, respectively. Similarly, for 300 tasks, the MARL-ACP method has an average waiting time of 2.132 sec, while other existing techniques such as PBRE, SeDaSC, CL-PKC, and PCAKA have average waiting times of 9.197 sec, 7.665 sec, 6.876 sec, and 4.764 sec, respectively.

Response time

The response time analysis of the MARL-ACP technique using existing methods is described in Table 8 and Fig. 8. The graph clearly shows that the MARL-ACP process has outperformed the other techniques in all aspects. For example, with 100 tasks, the MARL-ACP method took only 22.77 seconds to respond. In contrast, existing approaches, such as PBRE, SeDaSC, CL-PKC, and PCAKA, took 32.19 seconds, 35.98 seconds, 29.12 seconds, and 26.12 seconds, respectively. Similarly, for 300 tasks, the MARL-ACP method has a response time of 23.66 sec, while other existing techniques like PBRE, SeDaSC, CL-PKC, and PCAKA have response times of 35.88 sec, 41.55 sec, 31.88 sec, and 27.99 sec, respectively.

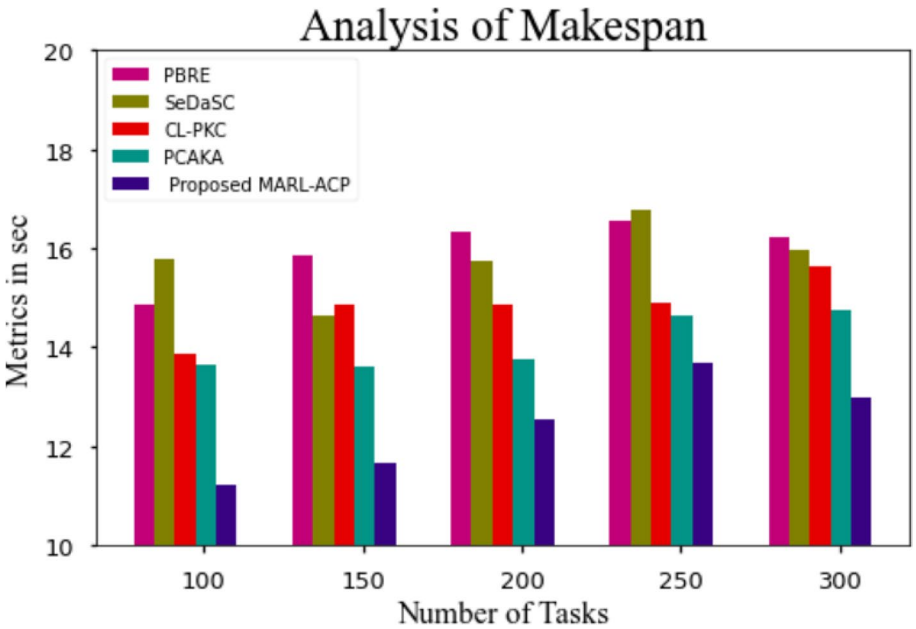


Fig. 5 Makespan Analysis for MARL-ACP method with the existing system

Table 5 Makespan Analysis for MARL-ACP method with the existing system

No of tasks	PBRE	SeDaSC	CL-PKC	PCAKA	MARL-ACP
100	120.65	133.98	122.13	142.98	98.78
150	123.98	134.92	145.55	135.98	99.76
200	125.77	135.87	134.12	146.23	103.33
250	130.67	136.22	139.67	148.65	113.17
300	132.98	137.43	132.67	149.21	118.22

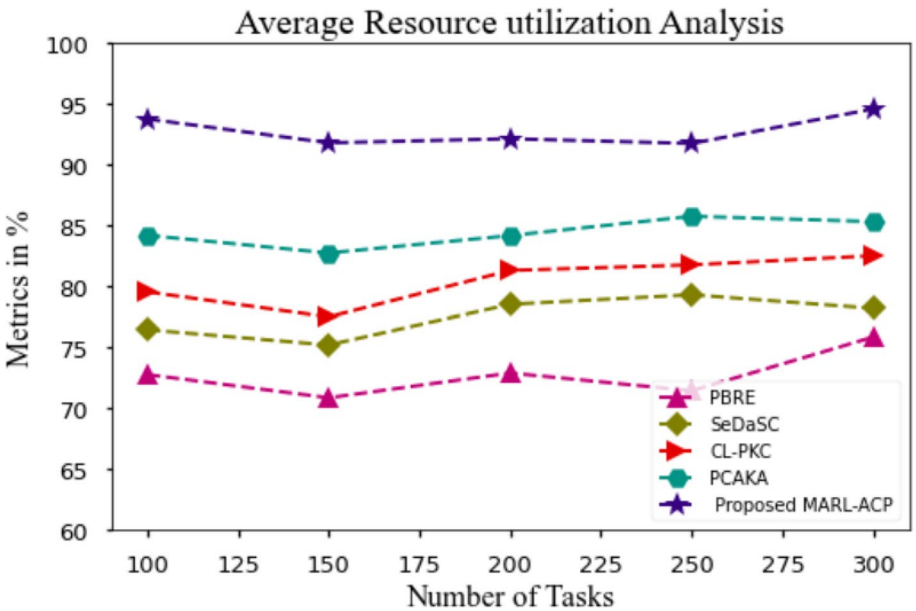


Fig. 6 Average Resource utilization Analysis for MARL-ACP method with the existing system

Table 6 Average Resource utilization Analysis for MARL-ACP method with the existing system

No of tasks	PBRE	SeDaSC	CL-PKC	PCAKA	MARL-ACP
100	72.78	76.46	79.56	84.21	93.78
150	70.87	75.21	77.51	82.76	91.82
200	72.91	78.54	81.32	84.19	92.16
250	71.45	79.32	81.78	85.78	91.77
300	75.87	78.21	82.53	85.34	94.61

Table 7 Average Waiting time Analysis for MARL-ACP method with the existing system

No of tasks	PBRE	SeDaSC	CL-PKC	PCAKA	MARL-ACP
100	7.193	6.124	5.198	2.675	1.654
150	8.423	6.675	5.342	3.896	1.768
200	8.193	6.198	5.197	3.786	1.211
250	9.453	7.234	5.221	4.192	2.554
300	9.197	7.665	6.876	4.764	2.132

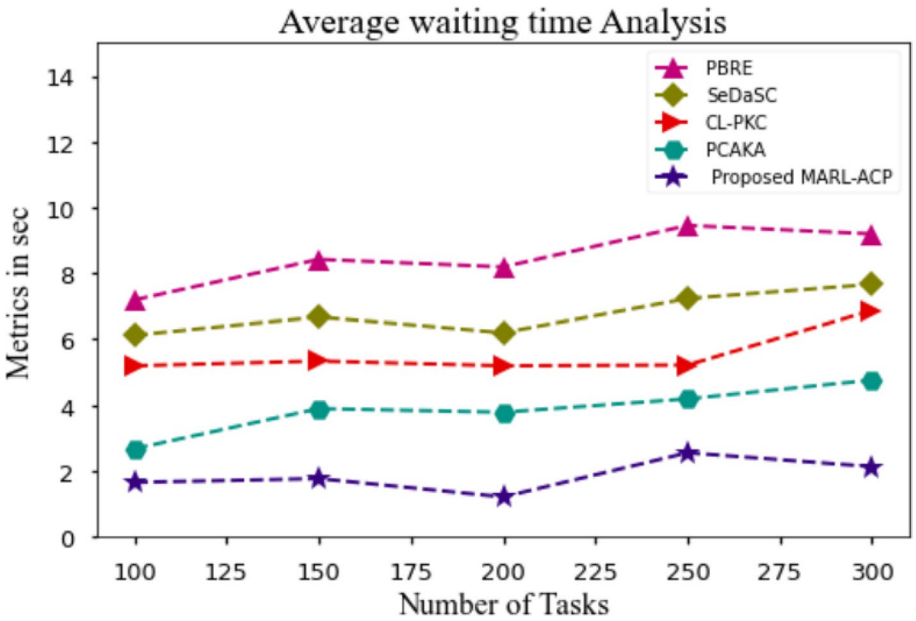


Fig. 7 Average Waiting time Analysis for MARL-ACP method with the existing system

Table 8 Response time Analysis for MARL-ACP method with the existing system

No of tasks	PBRE	SeDaSC	CL-PKC	PCAKA	MARL-ACP
100	32.19	35.98	29.12	26.12	22.77
150	31.98	36.14	29.55	25.87	20.87
200	34.18	39.32	29.32	27.11	22.45
250	35.76	38.22	29.43	25.76	21.17
300	35.88	41.55	31.88	27.99	23.66

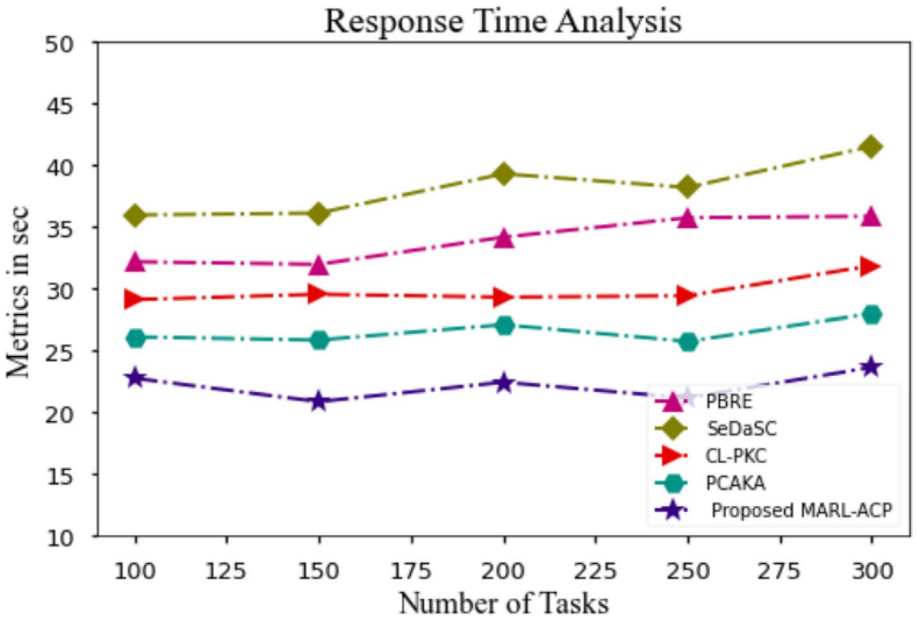


Fig. 8 Response time Analysis for the MARL-ACP method with the existing system

Resource utilization

Figure 9 and Table 9 depict a resource utilization comparison of the MARL-ACP approach with other existing methods. The graph shows that the cloud computing strategy has improved performance and resource utilization. For example, with 100 tasks, the MARL-ACP model has a resource utilization value of 91.43%, whereas the PBRE, SeDaSC, CL-PKC, and PCAKA models have resource utilization values of 81.87%, 85.18%, 84.18%, and 88.21%, respectively. The MARL-ACP model, on the other hand, has demonstrated maximum performance with various tasks. Similarly, under 300 tasks, MARL-ACP has a resource utilization value of 94.22%, while the PBRE, SeDaSC,

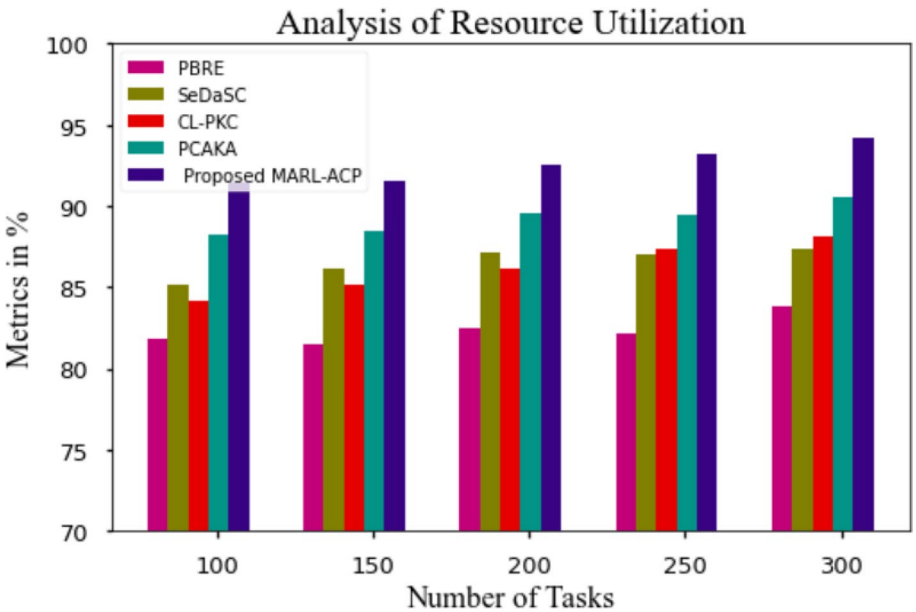


Fig. 9 Resource Utilization Analysis for MARL-ACP method with the existing system

Table 9 Resource Utilization Analysis for MARL-ACP method with the existing system

No of tasks	PBRE	SeDaSC	CL-PKC	PCAKA	MARL-ACP
100	81.87	85.18	84.18	88.21	91.43
150	81.45	86.14	85.17	88.45	91.55
200	82.53	87.14	86.12	89.55	92.54
250	82.18	86.99	87.33	89.43	93.21
300	83.77	87.32	88.16	90.54	94.22

CL-PKC, and PCAKA models have discounts of 83.77%, 87.32%, 88.16%, and 90.54%, respectively.

Bandwidth utilization

Figure 10 and Table 10 show a bandwidth utilization comparison of the MARL-ACP approach with other existing methods. The graph shows that the cloud computing strategy has resulted in an improved performance in bandwidth utilization. For example, with 100 tasks, the MARL-ACP model has a bandwidth utilization of 61.98%, whereas the PBRE, SeDaSC, CL-PKC, and PCAKA models have bandwidth utilization of 92.56%, 86.67%, 75.98%, and 72.67%, respectively. The MARL-ACP model, on the other hand, has demonstrated maximum performance with various tasks. Similarly, under 300 tasks, MARL-ACP has a bandwidth utilization of 68.21%, while PBRE, SeDaSC, CL-PKC, and PCAKA have 95.21%, 90.55%, 83.18%, and 71.88%, respectively.

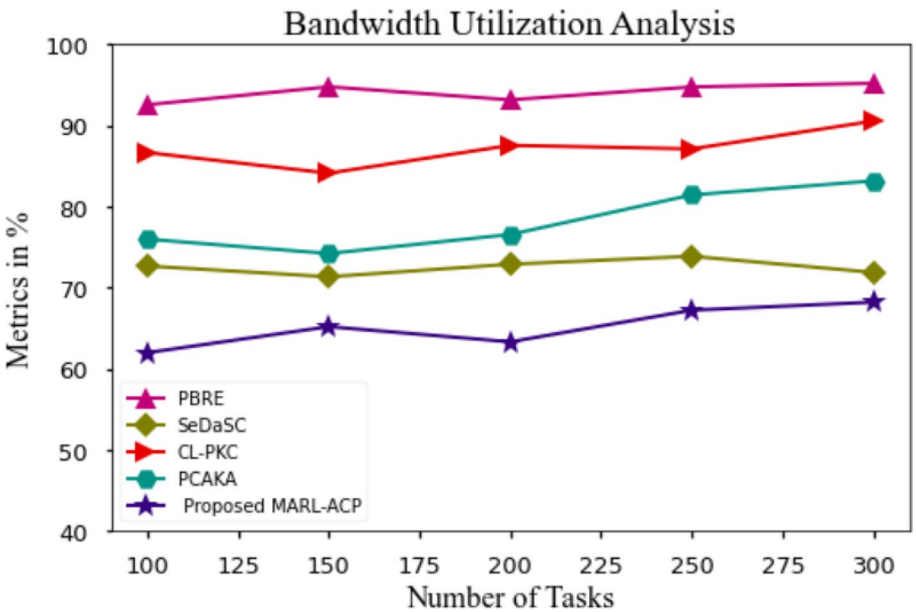


Fig. 10 Bandwidth Utilization Analysis for MARL-ACP method with the existing system

Table 10 Bandwidth Utilization Analysis for MARL-ACP method with the existing system

No of task	PBRE	SeDaSC	CL-PKC	PCAKA	MARL-ACP
100	92.56	86.67	75.98	72.67	61.98
150	94.78	84.12	74.21	71.34	65.19
200	93.17	87.56	76.56	72.89	63.32
250	94.76	87.12	81.44	73.87	67.22
300	95.21	90.55	83.18	71.88	68.21

Accuracy

Figure 11 and Table 11 compare the MARL-ACP approach's accuracy with other existing methods. The graph depicts how the cloud computing technique has improved performance and accuracy. For example, with 100 tasks, the MARL-ACP model achieves an accuracy of 95.92%, while the PBRE, SeDaSC, CL-PKC, and PCAKA models achieve 92.42%, 88.25%, 86.92%, and 81.72%, respectively. However, the MARL-ACP model performed well with diverse tasks. Similarly, the accuracy value of MARL-ACP under 300 tasks is 97.55%, 94.63%, 91.62%, 88.01%, and 84.62% for the PBRE, SeDaSC, CL-PKC, and PCAKA models, respectively.

Running time

The running time analysis of the MARL-ACP technique with existing methods is shown in Table 12 and Fig. 12. The task demonstrates that the MARL-ACP method outperforms the other techniques in every way. For example, with 100 tasks, the MARL-ACP process took 0.134 ms to run, while other existing methods such as PBRE, SeDaSC, CL-PKC, and PCAKA took 0.631 ms, 0.467 ms, 0.321 ms, and 0.217 ms, respectively. Similarly, for 300 tasks, the MARL-ACP method took 0.178 ms, while PBRE, SeDaSC, CL-PKC, and PCAKA took 0.845 ms, 0.631 ms, 0.387 ms, and 0.277 ms, respectively.

Conclusion

A novel scheme for transferring user data between different cloud servers based on a key agreement protocol is proposed. The benefits of our system are demonstrated in three ways mathematical analysis and comparative evaluation. In this paper, we proposed an efficient Anonymous Identity model and a fundamental agreement scheme-based model. The proposed scheme Admission control policy (ACP) and essential

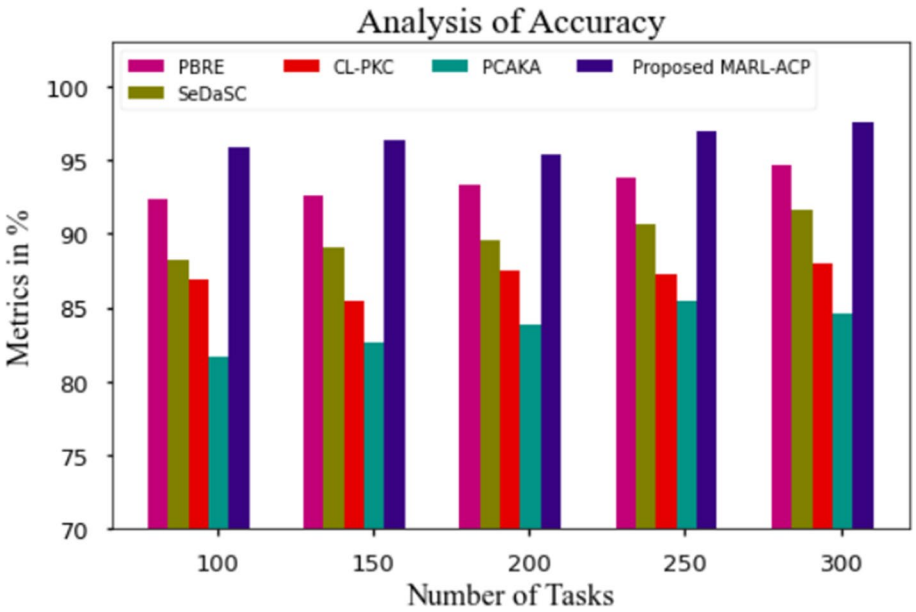


Fig. 11 Accuracy Analysis for MARL-ACP method with the existing system

Table 11 Accuracy Analysis for MARL-ACP method with the existing system

No of tasks	PBRE	SeDaSC	CL-PKC	PCAKA	MARL-ACP
100	92.42	88.25	86.92	81.72	95.92
150	92.63	89.04	85.43	82.61	96.37
200	93.32	89.57	87.53	83.92	95.44
250	93.86	90.62	87.26	85.43	97.03
300	94.63	91.62	88.01	84.62	97.55

Table 12 Running Time Analysis for MARL-ACP method with the existing system

No of tasks	PBRE	SeDaSC	CL-PKC	PCAKA	MARL-ACP
100	0.631	0.467	0.321	0.217	0.134
150	0.743	0.578	0.345	0.278	0.156
200	0.722	0.616	0.378	0.267	0.145
250	0.821	0.587	0.321	0.264	0.112
300	0.845	0.631	0.387	0.277	0.178

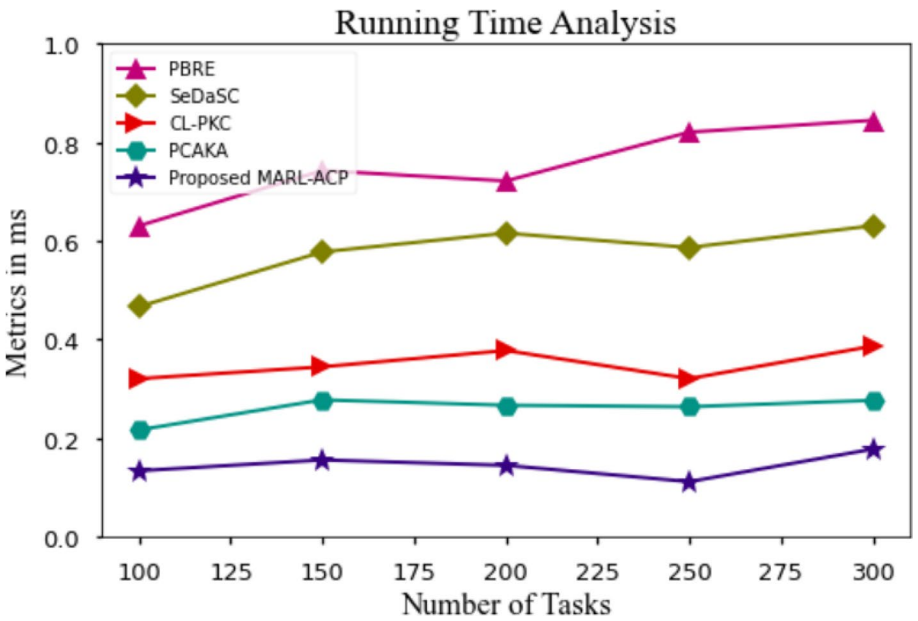


Fig. 12 Running Time Analysis for MARL-ACP method with the existing system

agreement-based model aid in developing trust between different cloud providers and lay the groundwork for cross-cloud data migration. Finally, the Multi-Agent Reinforcement Learning Algorithm (MARL) is used to identify and classify anonymity in the cloud using various pre-processing techniques, feature selection, and dimensionality reduction. This method discovered that existing models such as Proxy broadcast repeat encryption (PBRE), secure data sharing cloud (SeDaSC), Certificateless public key cryptography (CL-PKC), peer-to-peer cloud authentication and key agreement (PCAKA) have little impact on predictive performance, with the proposed model achieving the best result with an overall accuracy of 97.55% in predicting whether a user will belong to a particular group.

In the future, we intend to investigate and develop a protocol that allows multiple users to share data across different cloud servers to improve the efficiency of data sharing among multiple users.

Abbreviations

ACP	Admission control policy
MARL	Multi-Agent Reinforcement Learning Algorithm
MADRL	Multiagent deep RL
RC	Registration Center
PBRE	Proxy broadcast repeat encryption
SeDaSC	Secure Data Sharing Cloud
CL-PKC	Certificateless public key cryptography
PCAKA	Peer-to-peer Cloud Authentication, and Key Agreement

Acknowledgements
Not applicable.

Authors' contributions
Dr.D.PAULRAJ have drafted the abstract. Dr.S.NEELAKANDAN and Dr.M.PRAKASH have drafted the main subject with vast literature survey. Dr.E.BABURAJ has devised the discussion section along with numerical analysis. The author(s) read and approved the final manuscript.

Funding
The author(s) received no financial support for the research and publication of this article.

Availability of data and materials
All data generated or analysed during this study are included in this published article.

Declarations

Ethics approval and consent to participate
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Received: 18 December 2022 Accepted: 18 April 2023
Published online: 04 May 2023

References

1. C. I. network information center, "The 44th China statistical report on internet development," 2019. [Online]. Available: http://www.cnnic.net.cn/hlwfzyj/hlwxzbg/hlwtjbg/201908/P020190830356787490958.pdf. Accesed on 12 Sept 2022
2. Cui J, Zhou H, Zhong H, Xu Y (2018) AKSER: attribute-based keyword search with efficient revocation in cloud computing. Inf Sci 423:343–352
3. Cui J, Zhong H, Luo W, Zhang J (2017) Area-based mobile multicast group key management scheme for secure mobile cooperative sensing. Sci China Inf Sci 60(9):Art. no. 098104
4. Cui J, Zhou H, Xu Y, Zhong H (2019) OOABKS: online/offline attribute-based encryption for keyword search in mobile cloud. Inf Sci 489:63–77
5. Binz T, Leymann F, Schumm D (2011) Cmotion: a framework for migration of applications into and between clouds. 2011 IEEE international conference on service oriented computing and applications (SOCA). Irvine, pp 1–4. https://doi.org/10.1109/SOCA.2011.6166250
6. Sermakani AM (2020) Effective data storage and dynamic data auditing scheme for providing distributed services in federated cloud. J Circuits, Syst Comput 29(16):205–259. https://doi.org/10.1142/S021812662050259X
7. Praveen DS, Raj DP (2021) RETRACTED ARTICLE: smart traffic management system in metropolitan cities. J Ambient Intell Human Comput 12:7529–7541. https://doi.org/10.1007/s12652-020-02453-6
8. Jouini M, Rabai L (2019) A security framework for secure cloud computing environments. In: Cloud security: concepts, methodologies, tools, and applications, pp 249–263
9. Devi K, Paulraj D (2017) Multi level fault tolerance in cloud environment. In: 2017 international conference on intelligent computing and control systems (ICICCS), pp 824–828. https://doi.org/10.1109/ICCONS.2017.8250578
10. Arulkumaran K, Deisenroth MP, Brundage M, Bharath AA (2017) Deep reinforcement learning: a brief survey. IEEE Signal Process Mag 34(6):26–38
11. Devi K (2021) Multilevel fault-tolerance aware scheduling technique in cloud environment. J Internet Technol 22(1):109–119
12. Hernandez-Leal P, Kartal B, Taylor ME (2019) A survey and critique of multiagent deep reinforcement learning. Auton Agent Multi-Agent Syst 33(6):750–797. https://doi.org/10.1007/s10458-019-09421-1
13. Nguyen TT, Nguyen ND, Nahavandi S (2020) Deep reinforcement learning for multiagent systems: a review of challenges, solutions, and applications. IEEE Trans Cybern 50(9):3826–3839
14. Baker B, Kanitscheider I, Markov T, Wu Y, Powell G, McGrew B, Mordatch I (2020) Emergent tool use from multi-agent autocurricula. In: International conference on learning representations https://openreview.net/forum?id=SkxpxJBKwS
15. Dickson D, Rose A, Anu KT, Poulose D (2017) Cloud security with anonymous authentication of data stored in cloud. Int J Eng Sci 1(1):70–75
16. Li J, Chen X, Chow SS, Huang Q, Wong DS, Liu Z (2018) Multiauthority fine-grained access control with accountability and its application in cloud. J Netw Comput Appl 112:89–96
17. Tseng YM, Huang SS, You ML (2017) Strongly secure IDbased authenticated key agreement protocol for mobile multiserver environments. Int J Commun Syst 30(11):e3251–e3n/a. https://doi.org/10.1002/dac.3251.E3251IJCS-16-0586.R1
18. He D, Kumar N, Wang H, Wang L, Choo K-KR, Vinel A (2018) A provably-secure cross-domain handshake scheme with symptoms matching for mobile healthcare social network. IEEE Trans Dependable Secure Comput 15(4):633–645
19. Hariharan B (2019) A hybrid framework for job scheduling on cloud using firefly and BAT algorithm. Int J Business Intell Data Mining 15(4):388–407
20. Han G, Miao X, Wang H, Guizani M, Zhang W (2019) CPSLP: a cloud-based scheme for protecting source location privacy in wireless sensor networks using multi-sinks. IEEE Trans Veh Technol 68(3):2739–2750
21. Akram MA, Ghaffar Z, Mahmood K, Kumari S, Agarwal K, Chen CM (2020) An anonymous authenticated key-agreement scheme for multi-server infrastructure. Human-centric Comput Inf Sci 10(1):1–18
22. Zhong H, Zhang C, Xu Y, Liu L (2020) Authentication and key agreement based on anonymous identity for peer-to-peer cloud. In: IEEE transactions on cloud computing
23. Patonico S, Braeken A, Steenhaut K (2019) Identity-based and anonymous key agreement protocol for fog computing resistant in the Canetti–Krawczyk security model. Wirel Netw 29:1017–1029
24. Ahmad I, Bakht H (2019) Security challenges from abuse of cloud service threat. Int J Comput Digital Syst 8(01):19–31
25. Hariharan (2019) WBAT job scheduler: a multi-objective approach for job scheduling problem on cloud computing. J Circuits, Syst Comput 29. https://doi.org/10.1142/S0218126620500899
26. Anthi E, Williams L, Słowińska M, Theodorakopoulos G, Burnap P (2019) A supervised intrusion detection system for smart home IoT devices. IEEE Internet Things J 6(5):9042–9053
27. Punithavathi P, Geetha S, Karuppiah M, Islam SH, Hassan MM, Choo KKR (2019) A lightweight machine learning-based authentication framework for smart IoT devices. Inf Sci 484:255–268
28. Gochhayat SP, Lal C, Sharma L, Sharma DP, Gupta D, Saucedo JA, Kose U (2020) Reliable and secure data transfer in IoT networks. Wirel Netw 26(8):5689–5702
29. Shahid F, Ashraf H, Ghani A, Ghayyur SA, Shamshirband S, Salwana E (2020) PSDS–proficient security over distributed storage: a method for data transmission in cloud. IEEE Access 8:118285–118298
30. Kumar AN, Jegadeesan R, Ravi CN, Greeda J (2019) A secure transaction authentication scheme using Blockchain based on IOT. Int J Sci Technol Res 8(10):2217–2221
31. Qu G, Wierman A, Li N (2020) Scalable reinforcement learning of localized policies for multi-agent networked systems. PMLR, Cloud, Proc Mach Learn Res 120:256–266
32. Spooner, T. and Savani, R., 2020. Robust market making via adversarial reinforcement learning. arXiv preprint arXiv:2003.01820
33. Chomping G, Liu Z, Xia J, Liming F (2019) Revocable identity-based broadcast proxy reencryption for data sharing in clouds. In: IEEE transactions on dependable and secure computing
34. Parthiban S, Harshavardhan A, Prashanthi V, Alolo A-RAA, Velmurugan S (2022) Chaotic Salp swarm optimization-based energy-aware VMP technique for cloud data centers. In: Computational intelligence and neuroscience. https://doi.org/10.1155/2022/4343476
35. Mardani A, Mohan P, Mishra AR, Ezhumalai P (2023) A fuzzy logic and DEEC protocol-based clustering routing method for wireless sensor networks. AIMS Mathematics 8(4):8310–8331. https://doi.org/10.3934/math.2023419
36. Zhao Z, Li X, Luan B, Jiang W, Gao W (2023) Secure internet of things (IoT) using a novel brooks Iyengar quantum byzantine agreement-centered blockchain networking (BIQBA-BCN) model in smart healthcare. Inf Sci. https://doi.org/10.1016/j.ins.2023.01.020
37. Gangathimmappa M, Sambath V, Ramanujam RAM, Sammeta M (2022) Deep learning enabled cross-lingual search with metaheuristic web-based query optimization model for multi-document summarization. Concurr Comput Pract Exp:e7476. https://doi.org/10.1002/cpe.7476

Publisher's Note