COGNITIVE ALLOCATOR: AN END-TO-END SECURE AND DYNAMIC VIRTUAL MACHINE ALLOCATION FRAMEWORK FOR OPTIMIZED RESOURCE MANAGEMENT IN CLOUD DATA CENTERS

K SARAVANAN[1] , DR.R.SANTHOSH[2]

[1]Research scholar Department of Computer Science and Engineering
Faculty of Engineering Karpagam Academy of Higher Education Coimbatore, Tamil Nadu, India

[2]Professor and Head Department of Computer Science and Engineering Faculty of Engineering
Karpagam Academy of Higher Education Coimbatore, Tamil Nadu, India

Email:saraengg@gmail.com[1],santhoshrd@gmail.com[2]

ABSTRACT

Cloud Computing (CC) offers a ubiquitous service to the Information Technology (IT) environment with higher resources. However, due to massive usage of cloud resources the availability of services often faced with service outage, power consumption, and security issues. Several state-of-the-art works tends to tradeoff the resource usage optimization and security in cloud Virtual Machines (VMs). To overcome this issue, we design a secure and effective cloud VM allocator named “Cognitive Allocator”. The designed Cognitive Allocator model ensures trade-off among the end-to-end security and resource management in the cloud computing environment by utilizing Deep Learning (DL), Deep Reinforcement Learning (DRL), and Optimization algorithms respectively. The entities involved in the model includes Cloud Users (CUs), Cloud Service Providers (CSPs), Authentication Server (AS), Edge Server (ES), and Cloud Broker. Firstly, the CUs and CSPs are authenticated to the AS for guaranteeing the authenticity for jeopardizing the network traffic attacks. The authenticated CUs are then restricted by the ESs with its hybrid access control (Role and policy-based access control) mechanism using Dual Agent DRL (DA-DRL) algorithm. The DA-DRL algorithm learns the past experience and firmly controls access for the CUs based on their roles, Service Level Agreement (SLA), and their authenticity. Finally, we perform secure and optimized VM allocation in the cloud broker server using DL and optimization algorithm named Attention Classification Network (AC-Net) and Horse Herd Optimization (H2O) respectively. The AC-Net classifies the user task into three classes such as public, confidentiality, and highly sensitive. Based on the classified user task and security parameters, the H2O algorithm effectively allocates the VM for the CSPs. The proposed work is implemented using Cloud Sim of version 3.03 with various validation metrics such as security and privacy analysis, cost execution, resource utilization, power consumption, and allocation time. From the validation results, the proposed cognitive allocator model outpaces than the state-of-the-art models.

Keywords – *Cloud Computing (CC), Virtual Machine (VM), Cloud Service Providers (CSPs), Edge Server (ES), Deep Learning (DL), and Deep Reinforcement Learning (DRL)*

I. INTRODUCTION

Security breaches in cloud data centers pose significant risks to data availability, confidentiality, and integrity. Common threats include unauthorized access, data breaches, malware infections, DDoS attacks, and insider threats [1]. The shared infrastructure and multi-tenancy of cloud environments increase the attack surface, while complex networks and dependencies introduce vulnerabilities. Additionally, the opacity and complexity of cloud resources exacerbate security risks. To mitigate these risks, robust security measures such as encryption, access control, monitoring, and incident response procedures are essential [2]-[4]. Continuous threat intelligence, frequent audits, and adherence to compliance standards are crucial for maintaining the security posture of cloud data centers. Given the critical role of cloud data centers in processing, storing, and managing vast amounts of sensitive data, their security is paramount [5]. Breaches can lead to severe consequences, including financial losses, damage to reputation, and legal liabilities. Organizations must prioritize safeguarding their cloud data centers to protect their valuable assets. Moreover, the concentration of valuable resources and data in cloud data centers makes them attractive targets for cybercriminals. Malicious actors exploit security vulnerabilities to launch sophisticated attacks such as ransomware and DDoS attacks, causing service disruptions, unauthorized access, and data theft.

To effectively avoid, identify, and respond to cyber-attacks, cloud data centers must be secured through the adoption of robust security mechanisms such as encryption, access control, network segmentation, and intrusion detection systems. Sensitive data housed in cloud environments must also be protected, per industry norms and compliance standards. Serious fines and legal repercussions may result from noncompliance. Cloud data center security promotes confidence with partners, customers, and regulatory bodies in addition to guaranteeing compliance with legal requirements. In an increasingly digital and interconnected world, it is essential to maintain data confidentiality, guarantee operational continuity, fulfill regulatory requirements, and protect business reputation. Enhancing cloud data center security requires a strong emphasis on authentication and access control. It is imperative to employ strong authentication methods, such as biometric and multi-factor authentication, to confirm users' identities before allowing them access to cloud services. Organizations can increase the security of their cloud environments and reduce the risk of unauthorized access by utilizing multiple authentication techniques, such as passwords, biometrics, or token-based verification.

Furthermore, it is crucial to establish precise access control policies based on roles, privileges, and the principle of least privilege. This approach guarantees that users are granted access only to the resources and data pertinent to their roles, thus minimizing the risk of unauthorized actions or data breaches. By enforcing stringent access controls, organizations can reduce the attack surface, fortify security measures, and prevent unauthorized individuals from compromising sensitive data or disrupting cloud services. Authentication and access control mechanisms serve as pivotal layers of defense in bolstering the overall security stance of cloud data centers. Ensuring secure virtual machine (VM) allocation in these centers is indispensable for upholding data confidentiality, integrity, and availability while mitigating security vulnerabilities. The VM allocation process should incorporate robust security protocols to thwart unauthorized access, prevent data breaches, and mitigate malicious activities. One effective approach involves deploying encryption techniques to safeguard VM data both at rest and in transit, thereby safeguarding sensitive information against unauthorized access. Additionally, employing secure authentication mechanisms like multi-factor authentication verifies the identity of users seeking VM allocation, ensuring that only authorized individuals can access cloud resources. Moreover, implementing role-based access control (RBAC) assigns specific roles and permissions to users based on their responsibilities, thus restricting access to VMs and resources to only those with a legitimate need. Regular security audits and monitoring tools play a vital role in promptly detecting and responding to potential security threats, further enhancing the security of VM allocation in cloud data centers.

1.1. Problem statement and research questions

Current solutions often fail to simultaneously manage dynamic workload fluctuations, security vulnerabilities, and critical resource optimization factors such as energy efficiency, SLA adherence, and load balancing. Many existing algorithms emphasize cost and performance, while offering limited support for comprehensive security measures, real-time anomaly detection, and adaptive reallocation strategies. Additionally, most VM allocation methods function in isolated layers whether scheduling, migration, or monitoring lead to fragmented management and increased operational overhead. These limitations highlight the need for a unified, end-to-end secure VM allocation framework that can dynamically redistribute resources, counter security threats, and optimize system performance in real time. Following the literature critique and identified gaps, the research is guided by the following core questions:

1. How can an end-to-end secure and dynamic VM allocation framework be designed to optimize resource management while ensuring robust security in large-scale cloud data centers?
2. How does the proposed framework compare with existing state-of-the-art VM allocation methods in terms of performance metrics such as energy efficiency, load balancing, SLA compliance, latency, and security resilience?

1.2. Research contributions

By integrating these robust security measures, organizations can effectively shield their cloud infrastructure and data from a myriad of security threats, bolstering overall protection and ensuring operational resilience. The prime contributions of this research are listed below,

- To the best of my knowledge, the proposed cognitive allocator work performs end-to-end security for VM allocation starting from authentication, access control, and secure VM allocation. The secure framework allegedly reduces the malicious traffic rate.
- The exploiting of Attention Classification Network (AC-Net) for CUs task classification in three levels public, confidentiality, and highly sensitive diminishes the cost of execution, and VM allocation time respectively.
- The power consumption and resource utilization rate can be optimized by Horse Herd Optimization (H2O) algorithm based on VM/PM status, resource type, VM availability, and security issues respectively.

The rest of the paper is organized as follows; section II explains the literature survey with specific research gaps. Section III emphases the research methodology with suitable theoretical and mathematical equations. Section IV implements and compares the proposed model with the existing works using several validation metrics, and section V concludes the proposed model.

II. Literature Survey

Saxena et al. [21] proposed a method for secure resource management system through performing threat prediction and evaluation of workload in industrial cloud. This paper mainly focused on resource allocation corresponding to evaluated workload which method is named as ETP-Workload Estimation. Here, VM risks are analysed by adapting Risk-Score matrix that aids to secure allocation. Zhou et al. [22] introduce energy-efficient model for VM allocation in cloud data centers, achieving by designing allocation algorithm AFED-EF. Besides, this study also performs deployment mechanism for alleviating problems in IoT applications. The proposed AFED-EF algorithm significantly handles load variation while VM placement. Caviglione et al. [23] deep reinforcement learning (DRL) techniques have witnessed a major shift in the research environment concerning multi-objective deployment of virtual machines in cloud data centers. This strategy has the potential to optimize several goals at the same time, but persistent issues with interpretability and scalability highlight the need for more development and application of DRL-based solutions in real-world cloud systems. The evolution of cloud computing has spurred research into mitigating security vulnerabilities, particularly focusing on preventing side-channel attacks through secure virtual machine allocation designed by Rout et al. [24]. This literature review highlights the importance of addressing security concerns in cloud environments and the growing emphasis on proactive measures such as secure VM allocation to safeguard against potential threats. The investigation of safe virtual machine scheduling techniques has resulted from cloud data centers' pursuit of energy efficiency and privacy preservation. Han et al. [25] highlights the need to strike a balance between privacy concerns and energy conservation, highlighting the importance of creative scheduling strategies to maximize resource utilization while protecting sensitive data.

Efforts to secure data transmission during virtual machine migration in cloud environments were addressed by Mangalagowri et al. [26], exploring methods to safeguard sensitive data during migration processes, enhancing security and privacy. Saxena et al. [27] introduced a model for managing high availability in cloud applications by prioritizing virtual machines and estimating resource requirements. By integrating VM ranking and resource estimation, the model aims to enhance the reliability and availability of cloud services. The SA2-MCD architecture proposed by Yadav et al. [28] presents a secure framework for allocating virtual machines in multi-tenant cloud databases, focusing on data isolation and access control to enhance overall security and trustworthiness. Cao et al. [29] introduced a real-time secure VM allocation strategy to defend against co-residence attacks in energy-efficient cloud environments. By leveraging optimization techniques, the strategy aims to mitigate security risks associated with VM co-residency. Dubey et al. [30] extended the intelligent water drop approach to optimize virtual machine allocation within a secure cloud computing framework, aiming to enhance efficiency and security in VM allocation processes, thus improving resource utilization and data protection in cloud environments.

Considering various factors, Verma et al. [31] presents an enhanced optimization model for secure virtual machine (VM) migration in cloud systems.

The study addresses performance, cost, and security concerns, aiming to improve the security and efficiency of VM migration procedures through a multi-criteria approach. Huang et al. [32] introduces the SSUR technique, which aims to optimize virtual machine allocation strategies in cloud data centers based on user requirements. This study focuses on enhancing resource utilization and user satisfaction in cloud environments by tailoring VM allocation to individual user demands. Naik et al. [33] proposes a support value-based gaming policy to safeguard virtual machine allocation in cloud environments from attacks. By integrating game theory and support value analysis, the research aims to enhance security measures and resilience against malicious assaults, ensuring robust protection of VM resources. The metaheuristic architecture presented by Alsadie et al, [34] offers a dynamic approach to optimizing job scheduling and VM allocation in cloud data centers. By employing metaheuristic methods, the study aims to enhance overall performance and resource allocation efficiency in dynamic cloud environments. Talwani et al. [35] introduce a machine-learning-based method to streamline virtual machine migration and allocation procedures in cloud infrastructures. This approach aims to enhance resource usage, scalability, and efficiency in cloud data centers by adaptively distributing and moving virtual machines using machine learning approaches.

Alsahlani et al. [36] propose a lightweight multi-factor authentication and authorization approach to secure real-time data access in Internet of Things (IoT) cloud environments. By leveraging multiple authentication elements, the study aims to enhance security and efficiency in IoT data access, advancing the development of secure IoT cloud architectures. To bolster cloud security, Mostafa et al. [37] introduce a novel multi-factor, multi-layer authentication system, focusing on cloud user authentication. By integrating multiple authentication layers and elements, the framework strengthens cloud authentication systems against diverse security threats, emphasizing the importance of robust authentication procedures in cloud environments. Keswari et al. [38] investigates the development of fuzzy logic-based trust-based access control models for cloud computing environments. These models aim to provide flexible and adaptable access control methods in cloud settings by dynamically adjusting access control decisions based on changing trust connections. Sucasas et al. [39] presents an attribute-based pseudonymity technique for authentication, addressing privacy concerns in cloud services. By utilizing pseudonyms based on user traits, the approach ensures user privacy while enabling secure authentication in cloud services, highlighting the importance of privacy-enhancing technology in cloud security. Focusing on cloud data security, Saxena et al. [40] introduces a role-based access control system utilizing identity and broadcast-based encryption. This mechanism enhances data security in cloud environments by associating access rights with user roles and employing encryption techniques.

A lightweight authentication protocol invoked at session/request time to authenticate cloud users (CUs) and service endpoints. Use a streamlined message flow to limit cryptographic overhead while preserving mutual authentication and freshness checks. (Design informed by lightweight authentication studies.) A hybrid access-control module that combines rule/policy evaluation with a DRL agent that dynamically adjusts access decisions based on historical behavior, role, and risk metrics. The DRL component learns policy weightings and escalation rules to minimize unauthorized access while preserving throughput. The proposed protocol set out to (a) design an integrated end-to-end pipeline (b) optimizes resource utilization and energy while (c) enforcing runtime security and (d) remaining scalable and robust under varied workloads. However, the use of an existing lightweight authentication is consistent with recent proposals in cloud/IoT domains that trade cryptographic weight for performance while aiming to retain acceptable security properties. The proposed model is well-designed and ambitious — it aligns with modern trends (DRL and hybrid optimization, trace-driven evaluation) and adds valuable novelty by tightly coupling security modules with allocation decisions. Recent research has demonstrated that learning-based and meta-heuristic approaches can substantially improve VM placement and consolidation, reducing energy consumption and SLA violations in simulation studies. Parallel research in cloud security has proposed lightweight authentication and access-control schemes for constrained environments. However, these two research streams performance-oriented VM allocation and security protocols largely remain siloed, producing solutions that do not measure or optimize the interaction between placement decisions and security controls. This study addresses that gap by proposing an end-to-end cognitive VM allocator that tightly couples lightweight authentication, a DRL-assisted hybrid access-control layer, a task-sensitivity classifier, and a hybrid allocation optimizer. Unlike prior work that treats security and allocation independently, we evaluate the compound security–performance trade-offs and demonstrate, via trace-driven experiments and ablation studies, that joint optimization yields robust allocations with measurable security benefits and acceptable performance costs.

3. METHOD

The main objective of this study is to enhance security while VM allocation of Cloud data centres. Additionally, lightweight authentication protocol and access control are conducted to further strengthen network security and mitigation measures. We design a novel cloud security framework named End-to-End Cognitive & Dynamic Multi-Objective VM Allocation (Cognitive Allocator). The proposed research is composed of entities such as Cloud Users (CUs), Cloud Service Providers (CSPs), Authentication Server (AS) and Cognitive Cloud Broker (CCB). Figure 1 illustrates the architectural flow of the proposed Cognitive Allocator framework.

A. Novel Lightweight Authentication

Initially, to ensure CUs and CSPs legitimacy, we have performed authentication. For that, both CUs and CSPs are registered to AS to enable authentication. Furthermore, both CUs and CSPs are registered and authenticated by AS. The procedures include in registration and authentication are defined below,

(i) Entity Registration Phase

Step 1: At first, CUs (Ent_{CU}) is registered to AS by affording their credentials ID, name, passwords, IP address and fingerprint which are collectively denoted as $\{\mathrm{Cre}_{ID}, \mathrm{Cre}_{name}, \mathrm{Cre}_{pass}, \mathrm{Cre}_{IP}, \mathrm{Cre}_{FP}\}$ the registration Ent_{CU} can be articulated as,

$$(\mathrm{Ent}_{CU}) \rightarrow \mathrm{AS{:}Reg\ Ent}_{CU}\left(\mathrm{Cre}_{ID}, \mathrm{Cre}_{name}, \mathrm{Cre}_{pass}, \mathrm{Cre}_{IP}, \mathrm{Cre}_{FP}\right) \quad (1)$$

Where $\mathrm{Reg\ Ent}_{CU}(.)$ defines the registration of Ent_{CU} with following credentials $\mathrm{Cre}_{ID}, \mathrm{Cre}_{name}, \mathrm{Cre}_{pass}, \mathrm{Cre}_{IP}, \mathrm{Cre}_{FP}$.

Step 2: Eventually, the CUs (Ent_{CU}) is registered to AS providing their credentials IP address and SLA that are mentioned as $\{\mathrm{Cre}_{IPA}, \mathrm{Cre}_{SLA}\}$ that defined as,

$$(\mathrm{Ent}_{CSP}) \rightarrow \mathrm{AS{:}Reg\ Ent}_{CSP}\ (\mathrm{Cre}_{IPA}, \mathrm{Cre}_{SLA}) \quad (2)$$

Step 3: Once, the registration of Ent_{CU} and Ent_{CSP} is completed, the AS encrypt credentials using Lightweight Authentication Protocol (LAP) algorithm and provide the secret key $\left(\mathcal{S}ecret_{key}\right)$ for enabling its authenticity which can be formulated as,

$$AS \rightarrow (\mathrm{Ent}_{CU}, \mathrm{Ent}_{CSP}){:}\mathcal{S}ecret_{key} \quad (3)$$

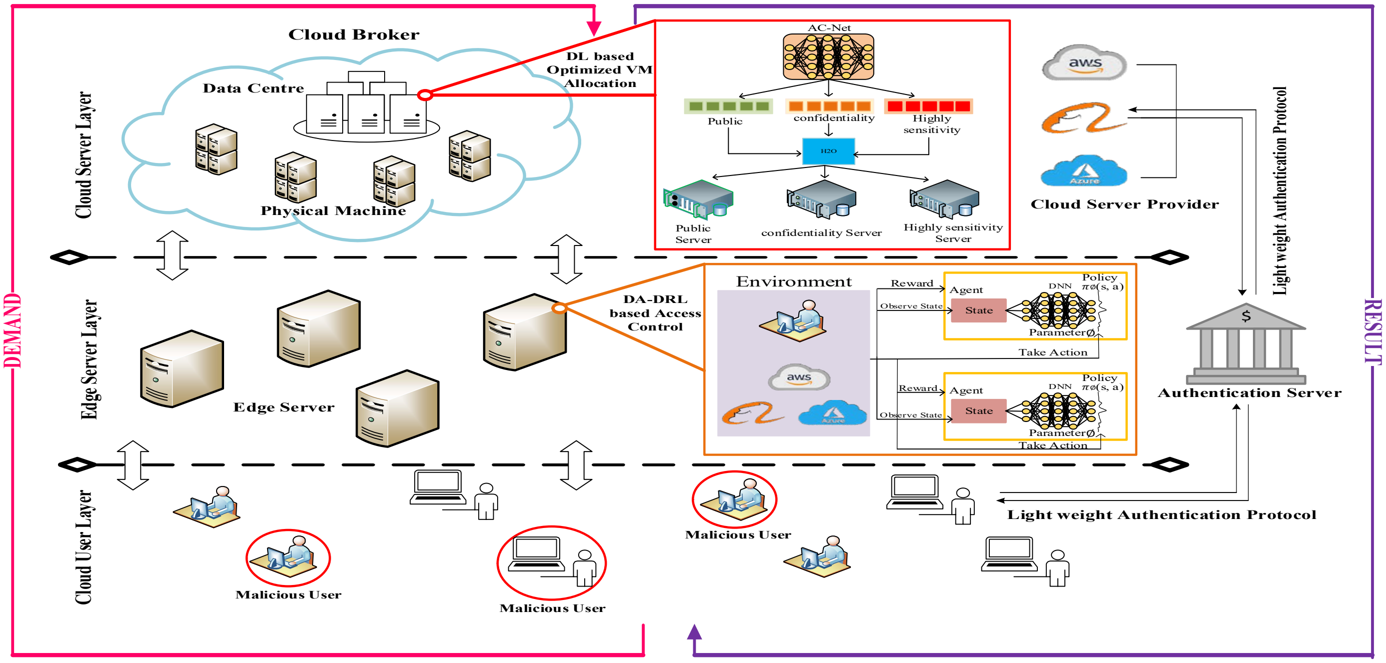

Fig.1. Overall Architecture of the Proposed Cognitive Allocator

(ii) Entity Authentication *Phase*

Step 4: In authentication phase, individual registered CU and CSP sent an request to AS for authentication through confirming their legitimacy which can be defined as,

$$\mathrm{Ent}_{CU}\rightarrow \mathrm{AS{:}Aut}_{req}\left(Ꞩecret_{key},\mathrm{Ent}_{pass}'\right)$$
(4)

Where Aut_{req} refers authentication request and Ent_{pass}' password provided by the Ent_{CU} while authentication. Similarly, aforementioned step is performed for other individual authenticated CU.

Step 5: Following that, the AS retrieves the Ent_{CU} encrypted information from blockchain, then it compares Ent_{CU} and Ent_{CU}' to ensure its authenticity as,

$$\mathrm{AS}\rightarrow \mathrm{Ent}_{CU}{:}\mathrm{if}\begin{cases}\mathrm{Ent}_{pass}=\mathrm{Ent}_{pass}' & \text{Authenticated}\\ \mathrm{Ent}_{pass}\neq \mathrm{Ent}_{pass}' & \text{Declined}\end{cases}$$
(5)

Here, if the Ent_{pass}' provided by Ent_{CU} is equal to Ent_{pass} then the entity is authenticated, otherwise that specific entity will blocked. Similarly, aforementioned steps 4 and 5 is performed for for Ent_{CU} authenticity is ensured by Ent_{BIO}.

B. DA-DRL based Hybrid Access Control

The authenticated CUs and CSPs are then subjected to access control to ensure the privacy of data. For access control, we utilize hybrid access control mechanism which is the combination of role and policy-based access control using Dual Agent Deep Reinforcement Learning (DA-DRL) algorithm. Figure 2 illustrates the hybrid access control mechanism using DA-DRL algorithm. Table 1 provides the glimpse of DRL parameters.

Generally, the DRL algorithms are based on the Markov Decision Process (MDP) which is defined by the tuple (St, Ac, ρ,Re,ψ) in which St and Ac denotes the state and action respectively, ρ signifies the transition probability that can be represented as $\rho{:}\mathrm{St}\times \mathrm{Ac}\rightarrow \Delta(\mathrm{St})$ in which Δ(St) signifies the transitional probability for the St'. For the given action $\mathrm{ac}\in\mathrm{Ac};\mathrm{Re}{:}\mathrm{St}\times\mathrm{Ac}\rightarrow\varpi$ in which ϖ defines the reward function that provides the instant reward for the transition actions during (St,Ac) to $\mathrm{St}';\zeta\in[0,1]$. The ζ is the discount factor that compromise the reward functions. Table denotes the state and action representations in the proposed work.

Table.1
Glimpse of DRL Parameters

DRL Parameters	Description
State (St)	The St defines the environment state with set of CUs and CSPs
Action (Ac)	The Ac defines the agent reaction towards environment in terms of role and policy-based access control
Reward (Re)	The Re defines the maximized security over malicious traffic

For every time t, every agent ag_j chose an action Ac_t^j based on the state (St). The action is performed in joint manner that can be represented as $\mathrm{Ac}_t=(\mathrm{ac}_t^1,\ldots,\mathrm{ac}_t^M)$ which makes the environment to transits to St_{t+1}. Furthermore, for every agent ag_j, the reward Re_t^j can be assigned. To be clearer, the major goal of the agent is to determine the policy $\beth^j{:}\mathrm{St}\rightarrow\Delta(\mathrm{Ac}^j)$. The value function $\mathrm{Va}^j{:}\mathrm{St}\rightarrow\varpi$ for the j-th agent to enable the combined policy $\beth{:}\mathrm{St}\rightarrow\Delta(\mathrm{Ac})$ for $\beth(\mathrm{ac}|\mathrm{st})\coloneqq\prod_{j\in M}\beth^j(\mathrm{ac}^j|\mathrm{st})$. The formulation is provided below,

$$\mathrm{Va}^j_{\beth^j,\beth^{-j}}(\mathrm{st})\coloneqq \mathrm{F}_{\mathrm{ac}_t^j\sim\beth^j(.|\mathrm{st})}\left[\sum_{t\geq 0}\zeta^t\mathrm{Re}_t^j\,|\mathrm{St}_0=\mathrm{st}\right]$$
(6)

From the above equation, -j denotes double agents. Based on the policies, the agent action can be provided as follows.

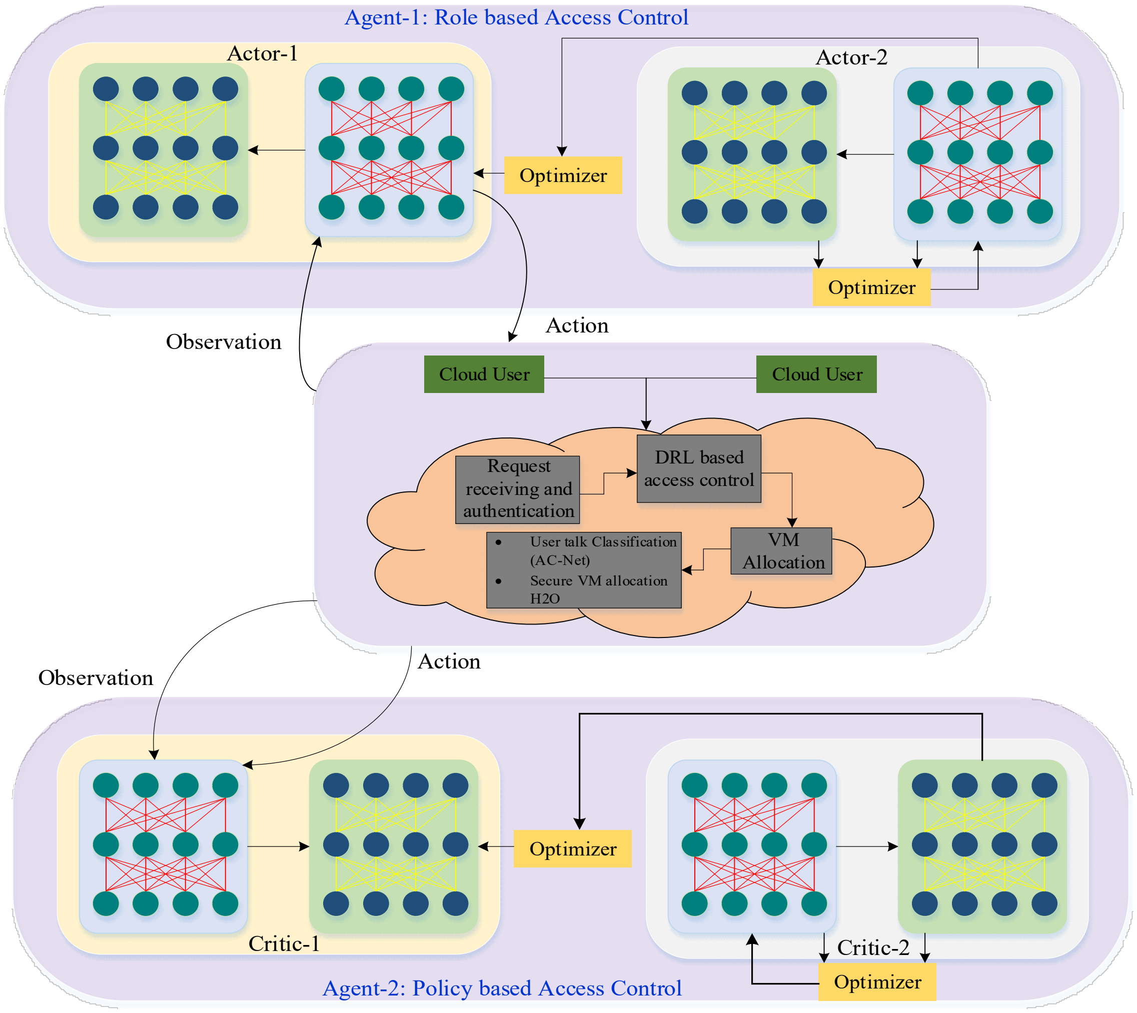

Fig.2. DA-DRL based Hybrid Access Control

(a) Agent 1: Role based Access Control

Entrenched on the CUs roles $Ro = \{Ro_1, Ro_2, \dots, Ro_m\}$ (i.e. Admin, Developer, and Support) in the cloud environment, we perform role-based access control. The roles are assigned grounded on the responsibilities and job functions respectively. Once the roles are identified, we provide permission for every CUs role that can be denoted as $Per = \{per_1, per_2, \dots, per_n\}$. The permission can be represented as $\{Per^{complete\ access}_{admin}, Per^{limited\ acess}_{developer}, Per^{database\ acess}_{support}$. Once the roles and permission are assigned, the corresponding CUs s can be mapped to the specific roles that can be denoted as,

$$\left\{CU_1 \underset{Map}{\Longrightarrow} Per^{complete\ access}_{admin}, CU_2 \underset{Map}{\Longrightarrow} Per^{limited\ acess}_{developer}, CU_3 \underset{Map}{\Longrightarrow} Per^{database\ acess}_{support}\right\} \quad (7)$$

Upon mapping the user roles and permissions, we perform access control decision based on three tuples $\{CU, Res, Action\}$. The CU can be permitted to access the if the $Res \in Action \in \{Ro\}$ otherwise denied to access.

(b) Agent 2: Policy based Access Control

Policy based access control is provided to the CSPs based on their policies provided during SLA agreements. The policies can be defined as $\{\aleph = \aleph_1, \aleph_2, \dots, \aleph_m\}$. Once the policies are framed, the set of conditions be provide based on the access time, attributes, and resources that can be denoted as $Con = \{Con_1, Con_2, \dots, Con_m\}$. Once the conditions are framed, the decision for access control can be granted be examining the conditions and policies that can be denoted as $ACE^{de} = Assess(\aleph, Con)$. Finally, the agent enforce the decision in terms of access blocking to the certain CUs that can be denoted as $\mathrm{ENF} = \mathrm{ENFORCE}\ (\mathrm{ACE}^{\mathrm{de}})$.

C. DL based Optimized Secure VM Allocation

Drawing inspiration from the feature pyramid network (FPN) within the object detection community, our proposal introduces an Attention Classification Network (AC-Net). This mechanism aims to augment the features across the terminal FC layers crucial for the ultimate classification task. The majority of CNN-based models currently in use just use scale fea_5, or the final stage's features, as the final classification features (i.e., the network's backend). As a result, these models are all single-scale classification models. On the other hand, networks' front-end features (from fea_1 to fea_4) have more small-ship data. The tiny spacecraft is quantified here at the pixel level rather than its real physical size, which is the standard practice. The AC-Net mechanism derives its framework from the comprehensive structure of the AC-Net. Each stage or level ($fea_1, fea_2, fea_3, fea_4$ and fea_5) bears the responsibility of rendering the final determination regarding CUs task classification. Figure 3 illustrates the DL (AC-Net) based CUs task classification in three levels.

We should underscore that our proposed AC-Net mechanism diverges from the original FPN approach or multi-output terminals. Rather than adopting this, we consolidate five inputs ($In"_1, In"_2, In"_3, In"_4$ *and* $In"_5$) into a single output to ensure more stable training, thereby utilizing a single-output terminal.

$$O_{MLS-CL} = \frac{1}{5} . \sum_{i=1}^{5} In"_i$$
(8)

where FLCi represents the FLC layer of the ith classification scale, In_i represents the input of the ith classification scale, flatten (·) means reshaping into a long column vector (from $\Re^{64\times64\times8}$ *to* $\Re^{32\,768\times1}$ for the fea_1 scale, from $\Re^{32\times32\times16}$ *to* $\Re^{16\,384\times1}$ for the fea_2 scale, from $\Re^{16\times16\times32}$ *to* $\Re^{8192\times1}$ for the fea_3 scale, from $\Re^{8\times8\times64}$ *to* $\Re^{4096\times1}$ for the fea_4 scale, and from $\Re^{4\times4\times128}$ *to* $\Re^{2048\times1}$ for the fea_5 scale).

Self-Attention Module (SAM): Additionally, we propose a Self-Attention Module (SAM) aimed at enhancing the core attributes of these backbone networks, thereby augmenting their capacity for global expression. Specifically, this mechanism emphasizes the importance of significant spatial global information, thereby rendering features from diverse scales more discriminative. "Our proposed Self Attention Module (SAM) enhances the attributes of each scale to deliver a more tailored global response.

$$Ot_i = \frac{1}{Ce(I)} . \sum_{\forall j} \left(fn(I_i I_j) . h(I_j) \right)$$
(9)

The input at the i-th location is represented as I_i , and its corresponding output as Ot_i . The similarity between inputs I_i and I_j is computed using the function $fn(.)$, while the feature representation at the -j th location is described by $h(.)$. The input response is denoted by $Ce(.)$, a normalized coefficient. In this context, the answer at the current location is represented by the information at the ith location, whereas the overall response is conveyed by the information at the j-th location g(·) can be understood as a linear embedding

$$h(I_j) = Wg_h I_j$$
(10)

In order to learn the weight matrix, denoted asWg_h, a 1×1 convolutional layer is utilized during training. Within an embedding space, the similarity, denoted as $fn(.)$, is determined through the utilization of a Gaussian function.

$$fn(I_i I_j) = \mathbb{e}^{\vartheta}(I_i)^y \partial(I_j)$$
(11)

The weight matrices to be learned are denoted as Wg_ϑ and Wg_∂ . These matrices can be acquired by incorporating two additional 1 × 1 convolutional layers. In this setup, $\vartheta(I_i) = Wg_\vartheta(I_i)$ and $\partial(I_j) = Wg_\partial(I_j)$ represent two embeddings. The normalized coefficient Ce(·) is determined by

$$Ce(I) = \sum_{\forall j} fn(I_i I_j)$$
(12)

The Self Attention Module is ultimately implemented as:

$$Ot_i = \frac{\mathbb{e}^{\vartheta}(I_i)^y \partial(I_j) .Wg_h I_j}{\sum_{\forall j} \mathbb{e}^{\vartheta}(I_i)^y \partial(I_j)}$$
(13)

where $\mathbb{e}^{\vartheta}(I_i)^y \partial(I_j) / \sum_{\forall j} \mathbb{e}^{\vartheta}(I_i)^y \partial(I_j)$ can be done through a softmax function/layer. Following the computation of ϑ and ∂ using two 1 × 1 convolutional layer, the similarity function fn is obtained by performing matrix multiplication between the resultant matrices ϑ and ∂^y . The characteristic representation h (·), denoted as h, is defined by a single 1 × 1 convolutional layer. Ultimately, the self-attention output Ot is derived by applying a softmax function/layer with sigmoid activation to the product of h and fn. The self-attention output Ot undergoes further processing through a 1 × 1 convolutional layer (depicted by a dotted box). This convolutional layer aims to align Ot with the dimensions of the initial input I, facilitating element-by-element addition,

$$In = Wg_{\mathbb{O}}Ot + I \qquad (14)$$

In the training process, an additional 1 × 1 convolutional layer can be employed to obtain $Wg_{\mathbb{O}}$, a weight matrix that needs to be learned. Consequently, the enhanced features gained from integrating self-attention data would undergo further processing in subsequent phases.

Stable Fully Connected Module (SFCM): In_1, In_2, In_3, In_4 *and* In_5 denote the outputs of the first five stages ($fea_1, fea_2, fea_3, fea_4$ and fea_5) respectively. The feature dimensions of In_1 are 64×64×8 = 32,768 In_2 are 32 × 32 × 16 = 16,384, In_3 are 16×16×32 = 8,192, In_4 are 8 × 8 × 64 = 4,096, and In_5 are 4 × 4 × 128 = 2,048. With fea_1 having 32,768 features, fea_2 with 16,384 features, fea_3 with 8,192 features, fea_4 with 4,096 features, and fea_5 with 2,048 features, it's evident that the contribution to categorization varies across different scales. Such discrepancies in feature numbers among scales may result in inconsistent learning. To mitigate this issue, we propose the Feature Fusion Module (F2M), which integrates feature maps of various scales into a unified feature dimension to balance the classification contributions across scales. To achieve feature balance in the F2M, we incorporate an FC layer after each scale accordingly.

$$In"_1 = FLC_1\ (In'_1)\ 32\ 768 \rightarrow \mathbb{N}_{FLC} \qquad (15)$$

$$In"_2 = FLC_2\ (In'_2)\ 16\ 384 \rightarrow \mathbb{N}_{FLC} \qquad (16)$$

$$In"_3 = FLC_3\ (In'_3)\ 8192 \rightarrow \mathbb{N}_{FLC} \qquad (17)$$

$$In"_4 = FLC_4\ (In'_4)\ 4096 \rightarrow \mathbb{N}_{FLC} \qquad (18)$$

$$In"_5 = FLC_5\ (In'_5)\ 2048 \rightarrow \mathbb{N}_{FLC} \qquad (19)$$

Setting the total number of neurons in each FLC layer ($FLC_1, FLC_2, FLC_3, FLC_4$ *and* FLC_5 to be consistent across all scales ensures uniformity. By unifying characteristics from multiple scales into a single $\mathbb{N}_{FLC}$ dimension, it signifies that the classification contributions from diverse scales are standardized.

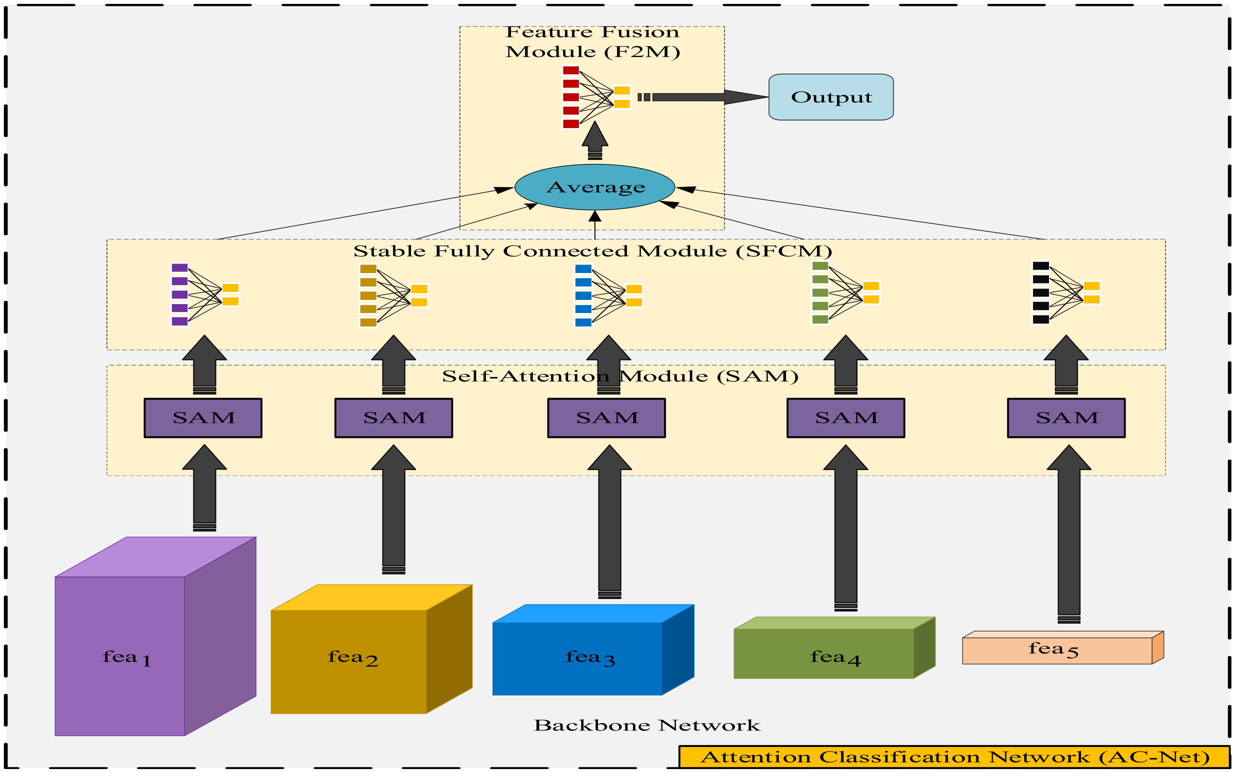
Feature Fusion Module (F2M)
Output
Average
Stable Fully Connected Module (SFCM)
Self-Attention Module (SAM)
SAM
SAM
SAM
SAM
SAM
fea 1
fea 2
fea 3
fea 4
fea 5
Backbone Network
Attention Classification Network (AC-Net)

Fig.3. Deep Learning based CU Taks Classification

Finally, based on the CUs request and security constraints we perform secure and optimal VM allocation method using Deep Learning (DL) and Optimization algorithms respectively. The DL algorithm utilized named Attention Classification Network (AC-Net) that classifies the user request into three classes public, confidentiality, and highly sensitive respectively. Based on the classified classes, the Horse Herd Optimization (H2O) algorithm optimally selects the optimal servers and allocate the VM. Here, the VMs are allocated by CB based on VM availability, resource type, security measures, VM status and PM availability. At diverse ages, horses (CB) represents several behaviours, horses are divided into four groupings corresponding to their ages 0-5, 5-10, 10-15 and elder than 15, which are denoted by $\vartheta, \sigma, \mu,$ and $\forall$ respectively. Generally, H20 utilize six common horse behaviours at stated ages to pretend their social life, Behaviours are including as “grazing, hierarchy, sociability, imitation, defence mechanism and roaming. The horse activity at individual iteration is expressed as,

$$\mathfrak{X}_m^{Iter,Age} = \vec{\mathbb{V}}_m^{Iter,Age} + \mathfrak{X}_m^{Iter,Age}, Age = \vartheta, \sigma, \mu, \forall \qquad (20)$$

Here, $\mathfrak{X}_m^{Iter,Age}$ denoted as m-th horse position, $\vec{\mathbb{V}}_m^{Iter,Age}$ is m-th vector horse velocity, Age denotes the range of horse age and $Iter$ is the present iteration. To detect vector of velocity, the stimulated steps stated six behaviours which are accomplished mathematically. While individual cycle algorithm, the horse vector of several ages are detailed as,

$$\vec{\mathbb{V}}_m^{Iter,\vartheta} = \vec{G}_m^{Iter,\vartheta} + \vec{D}_m^{Iter,\vartheta} \qquad (21)$$

$$\vec{\mathbb{V}}_m^{Iter,\sigma} = \vec{G}_m^{Iter,\sigma} + \vec{H}_m^{Iter,\sigma} + \vec{S}_m^{Iter,\sigma} + \vec{D}_m^{Iter,\sigma} \qquad (22)$$

$$\vec{\mathbb{V}}_m^{Iter,\mu} = \vec{G}_m^{Iter,\mu} + \vec{H}_m^{Iter,\mu} + \vec{S}_m^{Iter,\mu} + \vec{I}_m^{Iter,\mu} + \vec{D}_m^{Iter,\mu} + \vec{R}_m^{Iter,\mu} \qquad (23)$$

$$\vec{\mathbb{V}}_m^{Iter,\vartheta} = \vec{G}_m^{Iter,\forall} + \vec{I}_m^{Iter,\forall} + \vec{R}_m^{Iter,\forall} \qquad (24)$$

Grazing: Horse is one among grazing animal which graze at every stage of their lives for 16-20 hour per day. Following equations are formulate accomplish this behaviour of HOA.

$$\vec{G}_m^{Iter,Age} = \mathfrak{G}_{Iter}(\tilde{u} + \tilde{p}) + [\mathfrak{X}_m^{(Iter-1)}], \quad Age = \vartheta, \sigma, \mu, \forall \qquad (25)$$

$$\mathfrak{G}_m^{Iter,AGE} = \mathfrak{G}_m^{(Iter-1),AGE} \times \varphi_{\mathfrak{G}} \qquad (26)$$

From aforementioned equation, $\vec{G}_m^{Iter,Age}$ denotes the i-th parameter horse motion denoting its inclination for grazing. With $\varphi_{\mathfrak{G}}$ in individual iteration, this factor minimizes linearity. $\tilde{u}$ and $\tilde{p}$ parameters are upper and lower graze bound.

Hierarchy: Horse among 5 and 15 ages are exposed to adapt hierarchy law which is mathematically defined as,

$$\vec{H}_m^{Iter,Age} = h_m^{Iter,AGE}[\mathfrak{X}_*^{(Iter-1)} - \mathfrak{X}_m^{(Iter-1)}] \quad Age = \vartheta, \sigma, \mu \qquad (27)$$

$$h_m^{Iter,AGE} = h_m^{(Iter-1).Age} \times \varphi_h \qquad (28)$$

Where $\vec{H}_m^{Iter,Age}$ represents the influence of horse leader location at velocity and $\mathfrak{X}_*^{(Iter-1)}$ is horse location.

Sociability: This behaviour in H2O is contemplated the activity towards other horse position in herd and it is mathematically expressed as,

$$\vec{S}_m^{Iter,Age} = s_m^{Iter.Age}\left[\left(\frac{1}{N}\sum_{j=1}^{N} \mathfrak{X}_j^{(Iter-1)}\right) - \mathfrak{X}_m^{(Iter-1)}\right] \quad Age = \vartheta, \sigma \qquad (29)$$

$$s_m^{Iter,AGE} = s_m^{(Iter-1).Age} \times \varphi_s \qquad (30)$$

Where, $\vec{S}_m^{Iter,Age}$ represents the vector motion of i-th horse and $s_m^{Iter.Age}$ is the similar alignment horse to herd of $Iter^{th}$ iteration.

Imitation: Horse determine individual others excellent and disagreeable behaviours and habits through imitating another. This imitation is denoted as,

$$\vec{I}_m^{Iter,Age} = i_m^{Iter.Age}\left[\left(\frac{1}{pN}\sum_{j=1}^{pN}\mathfrak{X}_j^{(Iter-1)}\right) - \mathfrak{X}_m^{(Iter-1)}\right]$$

$$Age = \sigma \quad (31)$$

$$i_m^{Iter,AGE} = i_m^{(Iter-\ ).Age} \times \varphi_i \quad (32)$$

Where, $\vec{I}_m^{Iter,Age}$ represents the vector motion of i-th horse and $s_m^{Iter.Age}$ is the similar alignment horse to herd of $Iter^{th}$ iteration. pN denotes the overall number of horses which have optimal locations.

Defense: Mechanism of horse attention is other behaviour utilized in H2O and described as consecutively absent from horses which show responses of non-optimal. Defense mechanism are defined as,

$$\vec{D}_m^{Iter,Age} = -d_m^{Iter.Age}\left[\left(\frac{1}{qN}\sum_{j=1}^{pq}\mathfrak{X}_j^{(Iter-1)}\right) - \mathfrak{X}_m^{(Iter-\ )}\right]$$

$$Age = \vartheta, \sigma, \mu \quad (33)$$

$$d_m^{Iter,AGE} = d_m^{(Iter-1).Age} \times \varphi_d \quad (34)$$

Here, $\vec{D}_m^{Iter,Age}$ refers i-th horse escape vector from average of few horses with wickedest locations, that are exhibit vector through $\mathfrak{X}$ vector. qN represents the number of horses which in worst locations.

Roaming: This behaviour is contemplated as arbitrary activity of herd horse which can be defined as,

$$\vec{R}_m^{Iter,Age} = r_m^{Iter.Age}\left[\left(\frac{1}{qN}\sum_{j=1}^{pq}\mathfrak{X}_j^{(Iter-1)}\right) - \mathfrak{X}_m^{(Iter-\ )}\right]$$

$$Age = \vartheta, \sigma, \mu \quad (35)$$

$$r_m^{Iter,AGE} = r_m^{(Iter-\ ).Age} \times \varphi_r \quad (36)$$

From above-mentioned, $\vec{R}_m^{Iter,Age}$ denotes the vector of arbitrary velocity in i-th horse for an escape from local minima and local search. Based on those five horse behaviours, the VM allocation to solve complex problem. It entails maximizing VM allocation accuracy thereby increasing security in cloud environment. Achieving the best accuracy with the fewest features is the ideal situation. Furthermore, in order to prevent complexity, it is imperative to keep a minimum amount of features, as the modified H2O metaheuristic algorithm selects the characteristics that are used in the VM allocation process. Using a multi-objective optimization strategy becomes crucial when investigating various objective functions. These techniques present a variety of solutions that highlight the trade-offs between various target functions to engineers and system designers. Following equation can be used to mathematically express a multi-objective optimization issue as a minimization problem.

$$\begin{gathered} Minimize\ f_m(x), \quad m = 1,2,\dots,M \\ Subject\ g_i(x) \geq 0, \qquad j = 1,2,\dots,J \\ h_k(x) = 0, \ \ k = 1,2,\dots,K \\ \mathbb{L}_i \leq x_i \leq U_i, \dots, \quad i = 1,2,\dots,n \end{gathered} \quad (37)$$

Here, M denotes the quantity of objectives, J is the number of constraints of inequality, K is the number of constraints of equality and $[\mathbb{L}_i, U_i]$ represented as boundaries of i-th variables. Therefore, the fitness function can be calculated as,

$$Fitness = \vartheta\sigma_R(D) + \mu\frac{|R|}{|N|} \quad (38)$$

From the above H2O behaviour, the cloud broker firmly allocates the VM server for the CSPs based on the aforementioned metrics.

IV Experimental Results

The propounded Cognitive Allocator is implemented using Cloudsim simulation tool of version 3.03. The major usage of Cloudsim is to enable the virtual machine conception and firmly allocating the servers for real time allocation of virtual machines. More specifically, the designed algorithms in this research are examined based on the Cloud User (CUs) request within the range of 25-150. In addition to that, we have also ensured secure communication among CUs, VMs, and servers using authentication and encryption mechanism. Different from the conventional works, we have evaluated the proposed model with the varied evaluation metrices such as (a) security and privacy analysis, (b) Cost of Execution, (c) Resource Utilization, (d) Power Consumption, and (e) Allocation time. The explanation of those evaluation metrics is briefed as follows,

(a) Security and Privacy in VM: The VM security analysis enables scalability, portability, flexibility, agility, and also sandboxing the servers in the untrusted environment. The proposed work secures the environment by adopting authentication, access control, and classification respectively.

(b) Cost of Execution $(\boldsymbol{Co_{Ex}})$***:*** The $\mathrm{Co_{Ex}}$ in the VM can be determined based on the cost of the overall workload ($\mathrm{Tot_{Wc}}$), overhead due to virtualization ($\mathrm{VR_{ov}}$), overhead due to security breach ($\mathrm{SB_{ov}}$), and managing overhead ($\mathrm{MA_{ov}}$). The formulation of $\mathrm{Co_{Ex}}$ can be formulated as,

$$\mathrm{Co_{Ex}} = \mathrm{Tot_{Wc}} + \mathrm{VR_{ov}} + \mathrm{SB_{ov}} + \mathrm{MA_{ov}} \quad (39)$$

(c) Resource Utilization (RU_VM): The $\mathrm{RU_{VM}}$ in the VM can be determined based on the bandwidth of the network, input/output disk operations, Utilization of memory, and utilization of CPU. The formulation of $\mathrm{RU_{VM}}$ is provided below,

$$\mathrm{RU_{VM}} = \frac{\mathfrak{R}\mathfrak{x}^{\mathrm{VM}}}{\mathrm{Tot^{Res}}} \times 100\% \quad (40)$$

From the above equation, $\mathfrak{R}\mathfrak{x}^{\mathrm{VM}}$ denotes the usage of resource in VM, and $\mathrm{Tot^{Res}}$ denotes the amount resource available in the VM.

(d) Power Consumption $(\boldsymbol{PC_{VM}})$***:*** The $\mathrm{PC_{VM}}$ of the VM is based on the various power ingesting VM factors such as idle VM power consumption ($\mathrm{Idle_{VM}}$), network and disk power consumption ($\mathrm{N/D_{VM}}$), memory power consumption ($\mathrm{MP_{VM}}$), and CPU power consumption ($\mathrm{CPU_{VM}}$). The formulation of $\mathrm{PC_{VM}}$ is provided as follows,

$$\mathrm{PC_{VM}} = \sum \mathrm{Idle_{VM}} + \mathrm{N/D_{VM}} + \mathrm{MP_{VM}} + \mathrm{CPU_{VM}} \quad (41)$$

(e) Allocation Time (AT_VM): The $\mathrm{AT_{VM}}$ is computed based on the overall time taken to create a newer VM occurrence. The $\mathrm{AT_{VM}}$ can also be determine based on the various factors such as extra configuration time ($\mathrm{ETC_{VM}}$), and provisioning time of VM ($\mathrm{PT_{VM}}$). The formulation of $\mathrm{AT_{VM}}$ can be computed as follows,

$$\mathrm{AT_{VM}} = \sum \mathrm{ETC_{VM}} + \mathrm{PT_{VM}} \quad (42)$$

A. Implementation Analysis

As we mentioned above, the proposed work is implemented using Cloudsim 3.03 version. Furthermore, the security levels, optimization levels, and execution time levels can be verified using the simulation tool. The security and privacy can be examined in triple levels which includes authentication, authorization, and encryption. The resource optimization, VM allocation, and execution time can be controlled using user task classification and optimized VM allocation. More technically, the algorithms and methods implemented at the varied usual of virtual machines, tasks number, and virtual machine parameters security and privacy analysis, $\mathrm{Co_{Ex}}$, $\mathrm{RU_{VM}}$, $\mathrm{PC_{VM}}$, and $\mathrm{AT_{VM}}$ respectively. Table 2 below shows the simulation parameters.

Table.2
Simulation Parameter Description

Simulation Parameter	Value	Unit
# of Data Center	2	-
# of VMs	75-150	-
# of CUs	100	-
# of Hosts	5	-
# of CSPs	10	-
# of Edge Server	5	-
Cost Execution ($\mathrm{Co_{Ex}}$)	65.84-416.49	
Resource Utilization ($\mathrm{RU_{VM}}$)	92.86-97.6	%
Power Consumption ($\mathrm{PC_{VM}}$)	43-72	Joules
Allocation Time ($\mathrm{AT_{VM}}$)	30-49	ms
VM Allocation Start Time	0.5	ms
VM Allocation Finish Time	2.2-3.8	ms

B. Results Analysis

In this section, we analyse the performance of the proposed cognitive allocator with the state-of-the-art works with various metrics on VM. The detailed analysis of the results section is provided below,

(i) Analysis of Security and Privacy

In this sub-section, we have analysed the performance of the proposed security and privacy measures based on number of users Vs Malicious Traffic. The fig 4 shows the comparison of proposed and existing works in terms of user's number and malicious traffic respectively. From the figure, it is shown that when the user count increases the malicious traffic rate also increases. From the graphical inference it is shown that the proposed Cognitive allocator gains security and privacy performance than the existing works. The reason for such better security and privacy performance is that, we design novel authentication protocol for tampering various attacks such as stolen verifier, outdated, and insider attacks respectively. Furthermore, we have adopted DRL based access control mechanism based on policy and roles. The DRL assisted access control effectively sense the current environment based on specific user behaviour and network traffic policies respectively. On the other hand, the existing works LMA2S-IoT [36], MLA-CUA [37], TAC-FLC [38], and ABP-CS [39] lacks with capturing various network attacks and absence of cognitive ability in access control.

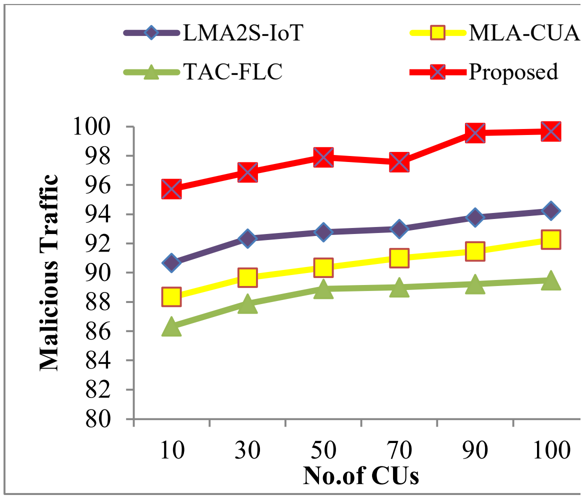

Fig.4. Analysis of Malicious Traffic

Table.3
Comparative Analysis on Malicious Traffic

Number of CUs	LMA2S-IoT	MLA-CUA	TAC-FLC	Proposed Cognitive Allocator	Difference
Min 10 CUs	90.67%	88.34%	86.34%	95.73%	5.06% - 9.39%
Max 100 CUs	94.23%	92.26%	89.49%	99.67%	5.44% - 10.18%-

The quantitative analysis shows in the table 3 provided that, the proposed work objects 95.73% of malicious traffic for 10 number of CUs whereas the existing works LMA2S-IoT, MLA-CUA, TAC-FLC, objective malicious traffic percentage of 90.67%, 88.34%, and 86.34% respectively for 10 CUs. For the 100 CUs, the proposed work achieves 99.76% and the existing works LMA2S-IoT, MLA-CUA, TAC-FLC, and ABP-CS gains 94.23%, 92.26%, and 89.49% respectively. From the comparative analysis, the proposed work achieves better performance than the state-of-the-art works.

(ii) Analysis of Co_{Ex}

The analysis of cost execution for allocation VM is detailed in this sub-section. Fig 5 shows the comparison of $\mathrm{Co}_{\mathrm{Ex}}$ based on number of tasks in X-axis Vs $\mathrm{Co}_{\mathrm{Ex}}$ in Y-axis for the proposed and existing works respectively. From the comparative graph, it is shown that when the number of task increases, the cost execution also increases. By analysing the graphical illustration, the proposed cognitive allocator outperforms than the existing works with lesser cost execution.

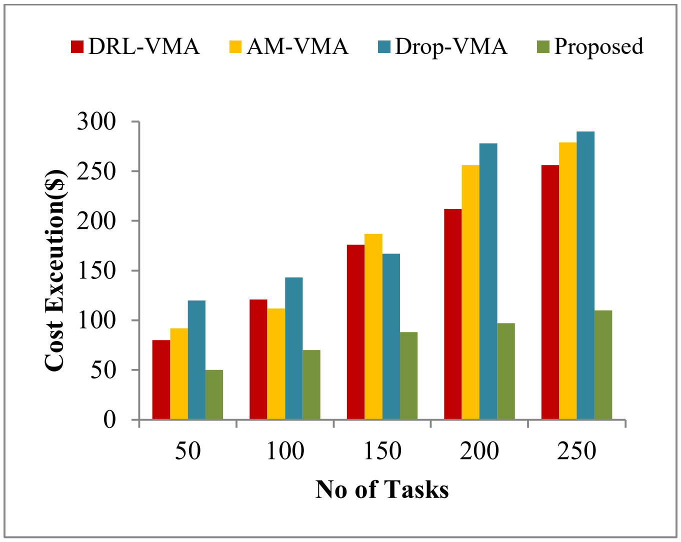

Fig.5. Analysis of Cost Execution

The foremost reason for such lesser cost execution is that, we perform user task classification during VM allocation stage. More clearly, the user task request is directed towards the cloud broker whom performs user task classification based on the task preferences, task type, and security levels respectively using Attention Classification Network (AC-Net). The utilized AC-Net extracts imperative features and provide importance to the specific features for effective user task classification. By classifying the user task, the allocation rate gets optimized thereby optimizing the cost for VM allocation. In contrast, the existing works DRL-VMA [23], RM-VMA [27], and Drop-VMA [30] limelight lesser focus on cost execution leading to higher cost for VM execution. To be more specific, the aforementioned works utilized DRL approach, novel management technique, and optimization approach for VM allocation. Even though they utilized optimized approach, the lack of providing deep analysis towards user task affects the cost execution to higher phase.

Table.4
Comparative Analysis on Cost Execution

Number of tasks	DRL-VMA	RM-VMA	Drop-VMA	Proposed Cognitive Allocator	Difference
Min 50 tasks	80$	92$	120 $	50$	30-70
Max 250 tasks	256 $	279 $	290 $	110$	146-180

From the quantitative results provided in table 4, it is clearly examined that the proposed cognitive allocator work achieves lesser $\mathrm{Co}_{\mathrm{Ex}}$ of 50$ for 50 number tasks than the existing DRL-VMA, RM-VMA, and Drop-VMA of 80$, 92$, and 120$ respectively. Whereas for the 250 number tasks, the proposed work gains execution cost of 110$ and the existing works achieves 256$, 279$, and 290$ respectively. Note that, the difference of cost rate among proposed and existing works also highly differs.

(iii) Analysis of RU_{VM}

The resource utilization examination for the proposed and existing works are explicitly provided in this sub-section. The fig 6 shows the graphical analysis of $\mathrm{RU}_{\mathrm{VM}}$ with respect to number of tasks Vs resource utilization rate. From the inference, it is seen that the tasks increase the resource utilization resource rate increases gradually. The proposed $\mathrm{RU}_{\mathrm{VM}}$ only sucks limited resources even though the task rate gets increased to maximum.

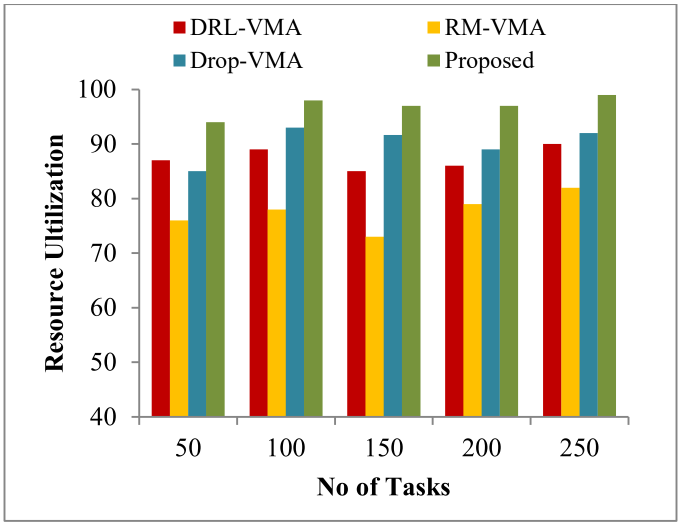

Fig.6. Analysis of Resource Utilization

The reason for such optimized resource utilization rate is that, we adopt optimization based VM allocation using Horse Herd Optimization (H2O) algorithm. The H2O algorithms incorporate several metrics in VM allocation which includes VM availability, resource type, and VM/PM status. With utilizing HHO based VM allocation, the rate of resource utilization gets increased than the existing works DRL-VMA [23], RM-VMA [27], and Drop-VMA [30]. The existing works either utilized DRL algorithms or lacks with considering important metrics for resource VM allocation leading to poor resource allocation rate.

Table.5
Comparative Analysis on Resource Utilization

Number of tasks	DRL-VMA	RM-VMA	Drop-VMA	Proposed Cognitive Allocator	Difference
Min 50 tasks	87 %	76 %	85%	94%	7%-9%
Max 250 tasks	90 %	82 %	92%	99%	7%-9%

From the quantitative results provided in table 5, it is clearly examined that the proposed cognitive allocator work achieves lesser $\mathrm{RU}_{\mathrm{VM}}$ of 94% for 50 number tasks than the existing DRL-VMA, RM-VMA, and Drop-VMA of 87%, 76%, and 85% respectively. Whereas for the 250 number tasks, the proposed work gains $\mathrm{RU}_{\mathrm{VM}}$ of 99% and the existing works achieves 90%, 82%, and 92% respectively.

(iv) Analysis of PC_{VM}

The examination of power consumption for the proposed cognitive allocator Vs existing works are shown in fig 7. From the figure, we can analyse that that that VM count increases with increase in power consumption. The graphical inference shows that the proposed cognitive allocator achieves lesser PC_{VM} by controlling the malicious traffic and isolating the affected VM by utilizing the security parameter during VM allocation. More clearly, we perform DA-DRL based access control and secure VM allocation using H2O algorithm restricts the malicious traffic thereby resulting in lesser power consumption for the proposed work. In contrast, the existing works DRL-VMA [23], RM-VMA [27], and Drop-VMA [30] lacks with poor power consumption as they lacked security parameters for VM allocation leading higher malicious traffic resulting in higher power consumption than proposed work. The table shows the comparison of quantitative analysis of the proposed Vs existing works on power consumption.

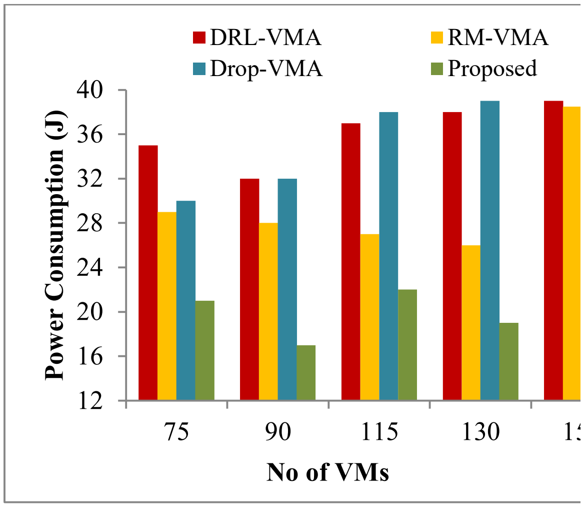

Fig.7. Analysis of Power Consumption

Table.6
Comparative Analysis on Power Consumption

Number of VMs	DRL-VMA	RM-VMA	Drop-VMA	Proposed Cognitive Allocator	Difference
Min 75 VM	35J	29J	30J	21J	9J-14J
Max 150 VM	39J	38.5J	37J	24J	13J-15J

From the quantitative results provided in table 6, it is clearly examined that the proposed cognitive allocator work achieves lesser PC_{VM} of 21J for 75 number of VMs than the existing DRL-VMA, RM-VMA, and Drop-VMA of 35J, 29J, and 30J respectively. Whereas for the 150 number of VMs, the proposed work gains PC_{VM} of 24J and the existing works achieves 39J, 38.5J, and 24J respectively. Overall, the proposed work consumes lesser power than the existing works.

(v) Analysis of AT_{VM}

The comparison of allocation time with respect to number of tasks is provided in fig 8. The number of tasks increases with the increase in VM allocation time. From the analysis, it is seen that the proposed VM allocation time decreases than the existing works. The reason for such lesser VM allocation time is that, we adopt DL based optimized VM allocation method. More clearly, the user task gets classified into three types includes public, confidentiality, and highly sensitive respectively. Based on the classified types, the H20 optimization algorithm firmly allocates the VM based on the availability, security parameter, and VM type respectively. So that, we obtain lesser AT_{VM} than the existing works. On the other hand, the existing works DRL-VMA [23], RM-VMA [27], and Drop-VMA [30] lacks with higher resource allocation as they mainly utilized DRL for VM allocation which only considers VM availability. As a result, the existing works achieves higher VM allocation time.

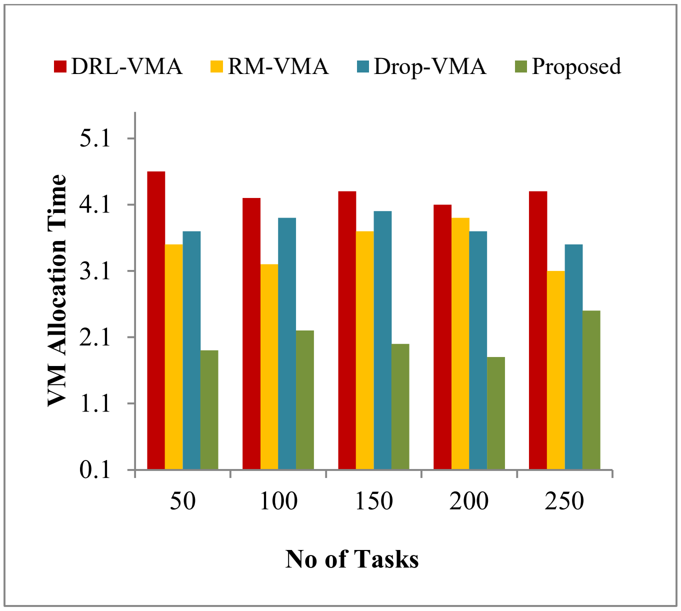

Fig.8. Analysis of VM Allocation

Table.7
Comparative Analysis on VM Allocation Time

Number of tasks	DRL-VMA	RM-VMA	Drop-VMA	Proposed Cognitive Allocator	Difference
Min 50 tasks	4.6 ms	3.5 ms	3.7 ms	1.9ms	1.8ms-2.7ms
Max 250 tasks	4.3 ms	3.1 ms	3.5 ms	2.5ms	1ms-1.8ms

From the quantitative results provided in table 7, it is clearly examined that the proposed cognitive allocator work achieves lesser AT_{VM} of 1,9ms for 50 user tasks than the existing DRL-VMA, RM-VMA, and Drop-VMA of 4.6ms, 3.5ms, and 3.7ms respectively. Whereas for the 250 user tasks, the proposed work gains AT_{VM} of 2.5ms and the existing works achieves 4.3ms, 3.1ms, and 3.5ms respectively. Overall, the proposed work consumes lesser power than the existing works.

C. Discussion

The experimental evaluation demonstrates that the proposed framework achieves superior resource utilization and reduced VM allocation time compared to existing baseline methods. The integration of dynamic allocation policies with real-time monitoring allows the system to adapt to workload fluctuations with minimal performance degradation. These findings validate that the framework successfully meets research objective by offering an integrated approach that enhances responsiveness, scalability, and operational efficiency. The end-to-end design ensures cohesive interactions between scheduling, security enforcement, and resource monitoring layers—an improvement over fragmented strategies documented in the literature. The system's security module demonstrated effective threat detection capabilities, with high accuracy in identifying malicious VM behaviors and preventing unauthorized resource access. Migration events remained protected through lightweight encryption and anomaly-aware decision mechanisms. The results confirm that the proposed framework addresses key security deficits in existing solutions. By embedding intrusion detection, isolation policies, and secure migration protocols, the framework successfully strengthens VM protection throughout allocation and runtime phases, thereby achieving the research objective.

D. Research limitations

Despite the promising performance of the proposed cognitive VM allocation framework, several challenges and open research issues remain unresolved. These gaps highlight opportunities for future exploration and the need for more scalable, secure, and autonomous cloud resource management systems.

While the proposed model adapts to fluctuating workloads, extreme traffic surges or unpredictable demand patterns may still lead to allocation delays or suboptimal resource utilization. Further research is needed to design ultra-responsive allocation mechanisms capable of operating under high-variance, real-time conditions.

Although the framework includes LAP-based authentication and hybrid access control, it does not fully address broader threats such as side-channel attacks, VM escape, or distributed denial-of-service within multi-tenant cloud environments. Incorporating comprehensive threat modeling and zero-trust cloud architectures remains an open issue.

4. CONCLUSION AND FUTURE WORK

Network traffic attacks, privacy concerns, high execution costs, low resource utilization, and increased allocation time remain significant challenges in cloud-based VM allocation. These limitations can be addressed through the adoption of a cognitive allocation framework. The framework begins with authentication using a Lightweight Authentication Protocol (LAP) that evaluates multiple security metrics. Once authenticated, cloud users (CUs) are governed by a hybrid access-control mechanism implemented through the DA-DRL algorithm which takes into account the roles and policies of both CUs and cloud service providers (CSPs). Following access control, virtual machines are assigned to the appropriate CSPs and user tasks through a deep learning–based optimized allocation approach. The DL model, AC-Net, categorizes user tasks into three levels: public, confidential, and highly sensitive. Based on this classification, the H2O algorithm allocates VMs according to request parameters such as VM availability, resource type, security requirements, and the operational status of both virtual and physical machines. The proposed cognitive allocator model is implemented using CloudSim 3.03, and its performance is evaluated against existing methods using multiple metrics. The results demonstrate that the proposed solution significantly improves the efficiency and effectiveness of VM allocation. As part of future work, we intend to incorporate additional security enhancements by integrating blockchain technology to enable privacy-preserving VM allocation. Highly unpredictable workloads—such as those in edge-cloud or IoT-driven environments—require more resilient allocation strategies. Handling extreme volatility with near-zero latency is still an unresolved challenge. Most energy-optimization techniques ignore the additional energy cost of real-time security operations. Integrating energy-aware security policies that maintain protection while reducing consumption is an open research direction.

REFERENCE

[1] Saxena, D., Gupta, I., Kumar, J., Singh, A. K., & Wen, X. (2021). A secure and multiobjective virtual machine placement framework for cloud data center. IEEE Systems Journal, 16(2), 3163-3174.

[2] Parast, F. K., Sindhav, C., Nikam, S., Yekta, H. I., Kent, K. B., & Hakak, S. (2022). Cloud computing security: A survey of service-based models. Computers & Security, 114, 102580.

[3] Siva Kumar, A., Godfrey Winster, S., & Ramesh, R. (2021). Efficient sensitivity orient blockchain encryption for improved data security in cloud. Concurrent Engineering, 29(3), 249-257.

[4] Mohammed, S. J., & Taha, D. B. (2021). From cloud computing security towards homomorphic encryption: A comprehensive review. TELKOMNIKA (Telecommunication Computing Electronics and Control), 19(4), 1152-1161.

[5] Anuradha, M., Jayasankar, T., Prakash, N. B., Sikkandar, M. Y., Hemalakshmi, G. R., Bharatiraja, C., & Britto, A. S. F. (2021). IoT enabled cancer prediction system to enhance the authentication and security using cloud computing. Microprocessors and Microsystems, 80, 103301.

[6] Dinh, P. T., & Park, M. (2021, January). BDF-SDN: A big data framework for DDoS attack detection in large-scale SDN-based cloud. In 2021 IEEE Conference on Dependable and Secure Computing (DSC) (pp. 1-8). IEEE.

[7] Ying, Z., Jiang, W., Liu, X., Xu, S., & Deng, R. H. (2021). Reliable policy updating under efficient policy hidden fine-grained access control framework for cloud data sharing. IEEE Transactions on Services Computing, 15(6), 3485-3498.

[8] Nagaraju, S., Jayakumar, S. K. V., & Priya, C. S. (2021, February). An effective mutual authentication scheme for provisioning reliable cloud computing services. In 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS) (pp. 314-321). IEEE.

[9] Susilo, W., Jiang, P., Lai, J., Guo, F., Yang, G., & Deng, R. H. (2021). Sanitizable access control system for secure cloud storage against malicious data publishers. IEEE Transactions on Dependable and Secure Computing, 19(3), 2138-2148.

[10] Han, D., Zhu, Y., Li, D., Liang, W., Souri, A., & Li, K. C. (2021). A blockchain-based auditable access control system for private data in service-centric IoT environments. IEEE Transactions on Industrial Informatics, 18(5), 3530-3540.

[11] Shafiq, D. A., Jhanjhi, N. Z., Abdullah, A., & Alzain, M. A. (2021). A load balancing algorithm for the data centres to optimize cloud computing applications. IEEE Access, 9, 41731-41744.

[12] Sain, M., Normurodov, O., Hong, C., & Hui, K. L. (2021, February). A survey on the security in cyber physical system with multi-factor authentication. In 2021 23rd international conference on advanced communication technology (ICACT) (pp. 1-8). IEEE.

[13] Shyla, S. I., & Sujatha, S. S. (2022). Efficient secure data retrieval on cloud using multi-stage authentication and optimized blowfish algorithm. Journal of Ambient Intelligence and Humanized Computing, 1-13.

[14] Thangasamy, A., Sundan, B., & Govindaraj, L. (2021, December). Dynamic phad/ahad analysis for network intrusion detection and prevention system for cloud environment. In 2021 4th International Conference on Computing and Communications Technologies (ICCCT) (pp. 273-279). IEEE.

[15] Manjunatha, S., & Suresh, L. (2021, December). Optimal Min-Communication and Migration Cost Algorithm based Approach for Efficient Task Migration in Cloud Computing. In 2021 International Conference on Circuits, Controls and Communications (CCUBE) (pp. 1-6). IEEE.

[16] Gangadevi, K., & Devi, R. R. (2021, March). A survey on data integrity verification schemes using blockchain technology in Cloud Computing Environment. In IOP Conference Series: Materials Science and Engineering (Vol. 1110, No. 1, p. 012011). IOP Publishing.

[17] Alouffi, B., Hasnain, M., Alharbi, A., Alosaimi, W., Alyami, H., & Ayaz, M. (2021). A systematic literature review on cloud

computing security: threats and mitigation strategies. IEEE Access, 9, 57792-57807.

[18] Vengala, D. V. K., Kavitha, D., & Kumar, A. S. (2023). Three factor authentication system with modified ECC based secured data transfer: untrusted cloud environment. Complex & Intelligent Systems, 9(3), 2915-2928.

[19] Saxena, U. R., & Alam, T. (2022). Role based access control using identity and broadcast based encryption for securing cloud data. Journal of Computer Virology and Hacking Techniques, 18(3), 171-182.

[20] Achar, S. (2022). Cloud Computing Security for Multi-Cloud Service Providers: Controls and Techniques in our Modern Threat Landscape. International Journal of Computer and Systems Engineering, 16(9), 379-384.

[21] Saxena, D., Gupta, R., Singh, A.K., & Vasilakos, A.V. (2023). Emerging VM Threat Prediction and Dynamic Workload Estimation for Secure Resource Management in Industrial Clouds. IEEE Transactions on Automation Science and Engineering.

[22] Zhou, Z., Shojafar, M., Alazab, M., Abawajy, J.H., & Li, F. (2021). AFED-EF: An Energy-Efficient VM Allocation Algorithm for IoT Applications in a Cloud Data Center. IEEE Transactions on Green Communications and Networking, 5, 658-669.

[23] Caviglione, L., Gaggero, M., Paolucci, M., & Ronco, R. (2020). Deep reinforcement learning for multi-objective placement of virtual machines in cloud datacenters. Soft Computing, 25, 12569 - 12588.

[24] Rout, C., Sethi, S., Badajena, J. C., & Sahoo, R. K. (2022, July). Secure virtual machine allocation for prevention of side channel attacks in cloud computing. In 2022 International Conference on Intelligent Controller and Computing for Smart Power (ICICCSP) (pp. 1-6). IEEE.

[25] Han, X., Mu, C., Zhu, J., & Jia, H. (2023). A safe virtual machine scheduling strategy for energy conservation and privacy protection of server clusters in cloud data centers. IEEE Transactions on Sustainable Computing.

[26] Mangalagowri, R., & Venkataraman, R. (2023). Ensure secured data transmission during virtual machine migration over cloud computing environment. International Journal of System Assurance Engineering and Management, 1-12

[27] Saxena, D., & Singh, A.K. (2022). A High Availability Management Model Based on VM Significance Ranking and Resource Estimation for Cloud Applications. IEEE Transactions on Services Computing, 16, 1604-1615.

[28] Yadav, A.K., Bharti, R.K., & Raw, R.S. (2021). SA2-MCD: Secured Architecture for Allocation of Virtual Machine in Multitenant Cloud Databases. Big Data Res., 24, 100187.

[29] Cao, L., Li, R., Ruan, X., & Liu, Y. (2022). Defending Against Co-Residence Attack in Energy-Efficient Cloud: An Optimization Based Real-Time Secure VM Allocation Strategy. IEEE Access, 10, 98549-98561.

[30] Dubey, K., & Sharma, S. (2020). An extended intelligent water drop approach for efficient VM allocation in secure cloud computing framework. J. King Saud Univ. Comput. Inf. Sci., 34, 3948-3958.

[31] Verma, G. (2022). Secure VM Migration in Cloud: Multi-Criteria Perspective with Improved Optimization Model. Wireless Personal Communications, 124, 75 - 102.

[32] Huang, Y., Xu, H., Gao, H., Ma, X., & Hussain, W. (2021). SSUR: An Approach to Optimizing Virtual Machine Allocation Strategy Based on User Requirements for Cloud Data Center. IEEE Transactions on Green Communications and Networking, 5, 670-681.

[33] Naik, B.B., Singh, D., & Samaddar, A.B. (2019). Secure virtual machine allocation against attacks using support value-based game policy. International Journal of Communication Systems, 34.

[34] Alsadie, D. (2021). A Metaheuristic Framework for Dynamic Virtual Machine Allocation With Optimized Task Scheduling in Cloud Data Centers. IEEE Access, 9, 74218-74233.

[35] Cano, J., Talwani, S., Singla, J., Mathur, G., Malik, N., Jhanjhi, N.Z., Masud, M., & Aljahdali, S.H. (2022). Machine-Learning-Based Approach for Virtual Machine Allocation and Migration. Electronics.

[36] Alsahlani, A.Y., & Popa, A. (2021). LMAAS-IoT: Lightweight multi-factor authentication and authorization scheme for real-time data access in IoT cloud-based environment. J. Netw. Comput. Appl., 192, 103177.

[37] Mostafa, A.M., Ezz, M.M., Elbashir, M.K., Alruily, M., Hamouda, E., Alsarhani, M., & Said, W. (2023). Strengthening Cloud Security: An Innovative Multi-Factor Multi-Layer Authentication Framework for Cloud User Authentication. Applied Sciences.

[38] Kesarwani, A., & Khilar, P.M. (2019). Development of trust-based access control

models using fuzzy logic in cloud computing. ArXiv, abs/1912.01709.

[39] Sucasas, V., Mantas, G., Papaioannou, M., & Rodriguez, J. (2021). Attribute-Based Pseudonymity for Privacy-Preserving Authentication in Cloud Services. IEEE Transactions on Cloud Computing, 11, 168-184.

[40] Saxena, U.R., & Alam, T. (2021). Role based access control using identity and broadcast-based encryption for securing cloud data. Journal of Computer Virology and Hacking Techniques, 18, 171 - 182