International Journal of Modern Computer Science and IT Innovations (IJMCSIT)
eISSN: 3087-4289
Volume03 Issue07 | Page No: - 35-46

Received: 30 May 2026, Revised: 15 June 2026, Accepted: 26 June 2026, Published: 21 July 2026

DOI: https://doi.org/10.55640/ijmcsit-v03i07-04

Survey of Secure Workload Scheduling and Management in Cloud Computing Systems

Vivek Sharma [1]
[1] System Analyst, MITS Deemed University, India

ABSTRACT: Cloud computing provides scalable, on-demand computing resources for a wide range of applications, making efficient workload scheduling and management essential for optimizing resource utilization, reducing execution time, and ensuring Quality of Service (QoS) and Service Level Agreement (SLA) compliance. However, the distributed and multi-tenant nature of cloud environments introduces security challenges such as unauthorized access, data breaches, malicious workloads, and resource misuse. This survey presents a comprehensive review of secure workload scheduling and management techniques in cloud computing systems. It discusses the fundamentals of workload scheduling, including cloud architectures, service models, deployment models, and workload management concepts. The survey reviews secure scheduling approaches, including static and dynamic scheduling, heuristic and metaheuristic algorithms, Artificial Intelligence (AI)- and Machine Learning (ML)-based techniques, and load balancing strategies. It also examines key security mechanisms such as authentication, authorization, encryption, trust management, workload isolation, and intrusion detection for protecting cloud workloads. Furthermore, the survey summarizes recent research contributions, identifies existing limitations, and highlights open challenges, including scalability, multi-cloud security, real-time scheduling, edge–cloud integration, green cloud computing, and Zero-Trust security. Finally, it outlines future research directions for developing intelligent, secure, and efficient workload scheduling frameworks in modern cloud computing environments.

KEYWORDS: Cloud Computing, Secure Workload Scheduling, Management, Task Scheduling, Resource Allocation, Cloud Security, Load Balancing.

1. Introduction

Cloud computing has emerged as a transformative computing paradigm that provides on-demand access to scalable computing resources, including processing power, storage, networking, and software services over the Internet [1]. The rapid adoption of cloud computing across domains such as healthcare, finance, education, e-commerce, and the Internet of Things (IoT) has significantly enhanced volume and diversity of workloads processed by cloud data centers. Efficient workload scheduling and management have become essential for optimizing resource utilization, reducing execution time, minimizing operational costs, and ensuring Quality of Service (QoS) and Service Level Agreement (SLA) compliance [2]. Traditional scheduling approaches primarily focus on performance metrics such as makespan, throughput, and load balancing; however, they often overlook critical security requirements in dynamic and multi-tenant cloud environments [3].

The shared and distributed nature of cloud computing introduces several security challenges, including unauthorized access, data breaches, malicious workloads, insider attacks, virtual machine vulnerabilities, and resource misuse [4]. As cloud infrastructures become increasingly complex, secure workload scheduling has gained significant attention as a research area that integrates security mechanisms into scheduling decisions [5][6]. Modern workload scheduling techniques not only optimize resource allocation but also incorporate authentication, authorization, trust management, encryption, workload isolation, intrusion detection, and Artificial Intelligence (AI)-based threat detection to ensure secure and reliable workload execution. Furthermore, AI, ML, DL, and Reinforcement Learning (RL) have enabled intelligent scheduling frameworks capable of predicting workloads, adapting to dynamic resource conditions, and enhancing both scheduling efficiency and cloud security [7].

This survey presents a comprehensive review of secure workload scheduling and management techniques in cloud computing systems. It discusses the evolution of scheduling approaches, including static and dynamic scheduling, heuristic and metaheuristic optimization algorithms, AI-driven scheduling methods, and load balancing techniques. The main contributions of this survey are summarized as follows:

- This survey presents a detailed overview of the fundamentals of workload scheduling in cloud computing, with cloud architecture, service models, deployment models, and the core concepts of workload scheduling and management.
- The survey classifies and analyzes existing workload scheduling approaches into static and dynamic scheduling, heuristic and metaheuristic algorithms, AI/ML-based scheduling techniques, and load balancing methods, highlighting their characteristics, advantages, and limitations.
- This work reviews the major security mechanisms used in cloud workload management, including authentication and authorization, data protection and encryption, trust management, workload isolation, intrusion detection, and access control, emphasizing their role in securing cloud workloads.
- The survey provides a comparative analysis of recent research on secure workload scheduling and management, summarizing state-of-the-art techniques, performance improvements, and research gaps to offer insights into current developments.
- The survey identifies key challenges and discusses promising future research directions for developing intelligent, secure, scalable, and energy-efficient cloud workload scheduling frameworks.

The remainder of this paper is planned as follows. Section II presents the fundamentals of workload scheduling in cloud computing. Section III reviews secure workload scheduling techniques. Section IV discusses security mechanisms for cloud workload management. Section V highlights the open challenges and future research directions. Section VI presents the literature review of recent studies. Finally, Section VII concludes the paper by summarizing the key findings and outlining future research perspectives.

2. Fundamentals of Workload Scheduling in Cloud Computing

Cloud computing has revolutionized the delivery of computing resources by providing scalable, on-demand, and cost-effective services through distributed data centers. As cloud infrastructures continue to support large numbers of users and diverse applications, efficient workload scheduling and resource management have become essential for maintaining high system performance, resource utilization, and service quality. Though, shared and dynamic nature of cloud environments also introduces significant security limitations, including unauthorized resource access, data leakage, multi-tenancy risks, and malicious workload execution. Therefore, secure workload scheduling and management integrate traditional scheduling objectives with security mechanisms such as authentication, authorization, trust management, access control, and secure resource allocation to ensure that cloud resources are utilized efficiently while preserving confidentiality, integrity, and availability. Based on the surveyed literature, the fundamental concepts of secure workload scheduling and management can be categorized into cloud computing and workload management, workload scheduling concepts, resource allocation and provisioning, quality of service and service level agreements, and challenges in secure workload scheduling.

A. Cloud Computing Architecture

Cloud computing architecture refers to the structural design of cloud systems that enables on-demand delivery of computing resources over the Internet, shows in figure 1. It consists of two main components: the front end, which includes client devices and user interfaces, and the back end, which comprises cloud servers, storage systems, virtualization platforms, databases, and management services. These components communicate through high-speed networks and application programming interfaces (APIs) to provide scalable and reliable cloud services.

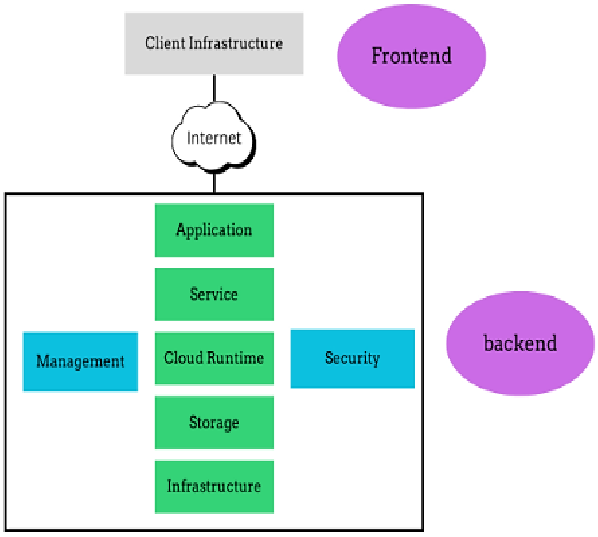

Fig. 1. Cloud Computing Architecture

The architecture also includes essential elements such as virtualization, resource management, scheduling modules, monitoring systems, and security mechanisms. Virtualization enables multiple virtual machines (VMs) or containers to share the same physical hardware, improving resource utilization and flexibility. Workload schedulers operate within this architecture by assigning tasks to suitable computing resources based on workload characteristics, resource availability, and performance requirements [8]. An efficient cloud architecture therefore provides the foundation for secure and optimized workload scheduling.

B. Cloud Computing Models

Table I summarizes the major cloud service models (IaaS, PaaS, SaaS) and deployment models (Public, Private, Hybrid, and Community Cloud) along with their characteristics and relevance to workload scheduling in cloud computing.

TABLE I. CLOUD SERVICE AND DEPLOYMENT MODELS IN CLOUD COMPUTING

Category	Model	Description	Workload Scheduling Focus / Key Features
Service Model	Infrastructure as a Service (IaaS)	Manages the underlying physical infrastructure and provide virtualised computing resources like storage, networking, and virtual machines.	Efficient VM allocation, resource provisioning, and infrastructure optimization.
	Platform as a Service (PaaS)	Gives you everything you need to build and launch apps without having to worry about managing the underlying infrastructure.	Application deployment, platform resource optimization, and scalability.
	Software as a Service (SaaS)	Provides access to pre-installed software programs through the Internet, with the vendor handling all management tasks.	High availability, load balancing, and improved user experience.
Deployment Model	Public Cloud	Shared infrastructure owned by external service providers.	Highly scalable and cost-effective but requires strong multi-tenant security.
	Private Cloud	Infrastructure in the cloud that is exclusively used by one company.	Enhanced privacy, security, compliance, and customized resource management.
	Hybrid Cloud	Facilitates the mobility and adaptability of workloads by integrating public and private clouds.	Secure workload migration, dynamic resource allocation, and business continuity.
	Community Cloud	Shared infrastructure for organizations with common business or regulatory requirements.	Secure collaboration, resource sharing, and regulatory compliance.

C. Workload Scheduling and Management

Workload scheduling and management are fundamental functions in cloud computing that ensure computational tasks are executed efficiently using available cloud resources. A workload refers to a collection of tasks, applications, virtual machines (VMs), containers, or data-processing jobs that require computing resources such as CPU, memory, storage, and network bandwidth. The scheduler determines when, where, and how these workloads should be executed to achieve optimal system performance while satisfying user requirements.

1) Workload scheduling

Workload scheduling is the process of assigning workloads to appropriate cloud resources based on factors such as resource availability, workload priority, execution time, QoS, and SLA requirements. Its primary objectives include minimizing makespan, reducing response time, improving resource utilization, balancing system load, and lowering operational costs.

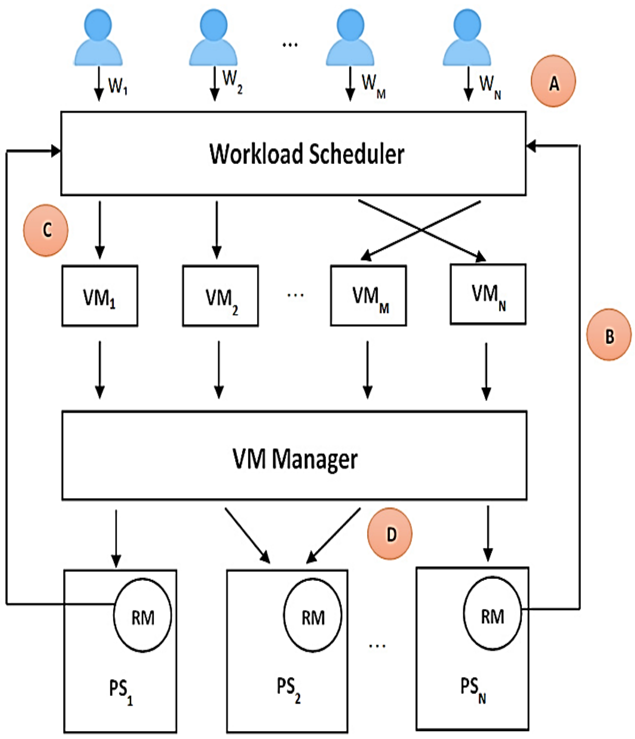

Fig. 2. Workload scheduling via VMs in the CC paradigm [9]

Figure 2 illustrates the workload scheduling process in a cloud computing environment using Virtual Machines (VMs). User workloads are submitted to the workload scheduler, which collects resource availability information from Resource Managers (RMs) monitoring the Physical Servers (PSs). Based on workload requirements and current resource utilization, the scheduler selects the most suitable VM for execution. Finally, the selected VM is allocated to an appropriate physical server, ensuring efficient resource utilization and improved workload performance.

2) Workload management

Workload management extends beyond scheduling by overseeing the entire lifecycle of cloud workloads, including resource provisioning, monitoring, load balancing, workload migration, fault tolerance, and performance optimization. It continuously monitors resource utilization and workload demands to dynamically allocate or reallocate resources, ensuring efficient execution and preventing resource overloading. Modern workload management systems also integrate security mechanisms e.g. authentication, access control, encryption, trust management, and intrusion detection to protect workloads from cyber threats while maintaining confidentiality, integrity, and availability [10].

Figure 3 shows that in order to handle the end users' workload, each cloud provider uses a network of data centers located in different parts of the world. The load balancer receives requests from end users and distributes workloads around data centers. Whether cloud providers aim to reduce costs, increase redundancy and reliability, or lessen their impact on the environment determines the workload distribution strategy. Cloud providers are informed about workload and energy signals by data centers so that they can decide the most effective workload dispatch. In order to accomplish the operational goals, the cloud service provider decides how to dispatch workloads depending on signals received from the data centers.

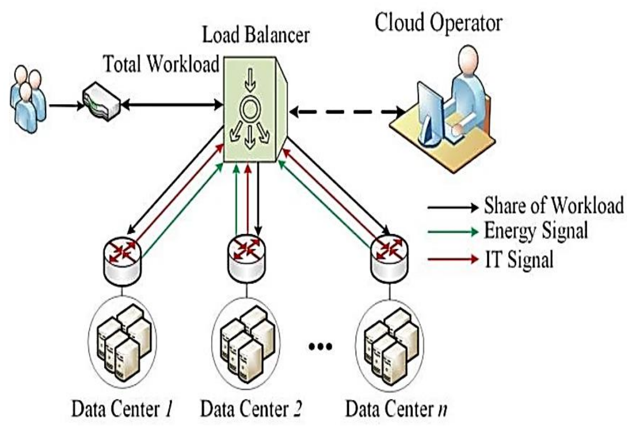

Fig. 3. Workload management among data centers [11].

3. Secure Workload Scheduling Techniques in Cloud

Secure workload scheduling techniques are responsible for assigning computational workloads to appropriate cloud resources while simultaneously optimizing performance, resource utilization, Quality of Service (QoS), and security. Traditional scheduling approaches primarily focus on execution efficiency and resource optimization, whereas modern scheduling techniques incorporate security-aware decision-making, trust management, workload isolation, intelligent resource allocation, and adaptive scheduling policies [12]. Based on the surveyed literature, workload scheduling techniques can be broadly classified into static and dynamic scheduling, heuristic scheduling algorithms, metaheuristic scheduling algorithms, AI-driven and machine learning-based scheduling, and load balancing and workload distribution.

A. Static and Dynamic Scheduling

Workload scheduling in cloud computing is generally classified into static scheduling and dynamic scheduling based on how scheduling decisions are performed. Static scheduling assigns tasks to computing resources before execution using predefined information such as task execution time, resource availability, workload characteristics, and dependency relationships [13]. Since scheduling decisions are made in advance, static scheduling introduces minimal runtime overhead and is suitable for stable and predictable cloud environments where workload patterns rarely change. Common static scheduling algorithms include First Come First Serve (FCFS), Shortest Job First (SJF), and Priority Scheduling. However, static scheduling lacks flexibility because it cannot respond to unexpected workload variations, resource failures, or changing user demands.

In contrast, dynamic scheduling continuously monitors cloud resources and workload conditions during execution, allowing scheduling decisions to be updated in real time. It considers factors such as CPU utilization, memory availability, network bandwidth, and workload priority before allocating tasks to available resources [14]. Dynamic scheduling improves scalability, resource utilization, and QoS by adapting to fluctuating workloads and changing cloud conditions. In secure cloud environments, dynamic scheduling can also incorporate authentication, trust management, access control, and secure resource allocation to ensure that workloads are executed efficiently while maintaining data security and system reliability [15].

B. Heuristic Scheduling Algorithms

Heuristic algorithms generate scheduling decisions using predefined rules with low computational complexity. Common techniques include FCFS, SJF, Round Robin, Priority Scheduling, Min-Min, Max-Min, and Earliest Deadline First (EDF) [16]. These methods provide fast scheduling but often produce locally optimal solutions in dynamic cloud environments.

C. Metaheuristic Scheduling Algorithms

Metaheuristic algorithms employ optimization strategies to obtain near-optimal scheduling solutions. Popular approaches include Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Simulated Annealing (SA), Grey Wolf Optimizer (GWO), Whale Optimization Algorithm (WOA), Firefly Algorithm (FA), and Artificial Bee Colony (ABC) [17]. These methods improve resource utilization, reduce makespan, execution cost, and workload imbalance.

D. AI-Driven and Machine Learning-Based Scheduling

These have become important technologies for intelligent workload scheduling in cloud computing. Unlike traditional scheduling algorithms that rely on predefined rules, AI-based scheduling techniques analyze historical workload data, resource utilization, and execution patterns to make adaptive scheduling decisions [18][19]. Machine learning models can predict future workload demands, estimate task execution times, and identify suitable computing resources for workload execution. Deep Learning, Reinforcement Learning, and Neural Networks enhance workload prediction, resource allocation, QoS, SLA compliance, and energy efficiency while also improving security through anomaly detection and trust-aware scheduling These capabilities enable cloud systems to improve resource utilization, reduce execution delays, and optimize scheduling performance under dynamic cloud environments.

E. Load Balancing and Workload Distribution

Load balancing distributes workloads evenly across cloud resources to prevent overloading and improve resource utilization. It enhances throughput, scalability, response time, and QoS by continuously monitoring virtual machines, containers, and network resources while ensuring reliable and secure workload execution [20].

Table II summarizes the major secure workload scheduling techniques used in cloud computing. It highlights the classification of scheduling approaches, their representative algorithms, and their key advantages for improving resource utilization, security, and overall scheduling performance.

TABLE II. SECURE WORKLOAD SCHEDULING TECHNIQUES IN CLOUD COMPUTING

Scheduling Technique	Description	Representative Algorithms/Methods	Key Advantages
Static Scheduling	Tasks are assigned before execution using predefined resource information.	FCFS, SJF, Priority Scheduling	Low overhead, simple implementation
Dynamic Scheduling	Tasks are allocated during execution based on real-time resource availability.	Dynamic Load Scheduling, Adaptive Scheduling	High scalability, better resource utilization
Heuristic Scheduling	Uses rule-based approaches for fast scheduling decisions.	FCFS, RR, Min-Min, Max-Min, EDF	Low computational complexity, fast execution
Metaheuristic Scheduling	Applies optimization algorithms to obtain near-optimal scheduling solutions.	GA, PSO, ACO, SA, GWO, WOA, FA, ABC	Optimized makespan, cost, and load balancing
AI/ML-Based Scheduling	Uses AI models to predict workloads and optimize scheduling decisions.	ML, DL, Reinforcement Learning, Neural Networks	Intelligent resource allocation, QoS improvement, enhanced security
Load Balancing and Workload Distribution	Distributes workloads evenly across cloud resources to improve performance.	Dynamic Load Balancing, VM Allocation, Resource Distribution	Improved throughput, scalability, and resource utilization

4. Security Mechanisms for Cloud Workload Management

Cloud workload management involves allocating, scheduling, monitoring, and executing workloads across shared cloud resources. Since cloud infrastructures operate in distributed and multi-tenant environments, they are exposed to various security threats such as unauthorized access, data leakage, malicious workloads, insider attacks, and resource misuse [21]. Therefore, effective workload management must integrate robust security mechanisms to protect cloud resources and maintain secure service delivery. Security mechanisms ensure that workloads are executed only by authorized users, sensitive data remains protected during storage and transmission, and cloud resources are allocated according to established security policies [22]. Based on the surveyed literature, the major security mechanisms include security requirements in workload scheduling, authentication and authorization, data protection and encryption, trust management and access control, multi-tenancy security and workload isolation, and intrusion detection with continuous security monitoring.

A. Authentication and Authorization

Authentication verifies the identity of users, applications, or devices before granting access to cloud resources using mechanisms [23], such as passwords, Multi-Factor Authentication (MFA), digital certificates, and Single Sign-On (SSO). Authorization then determines user permissions through models such as Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC), and Policy-Based Access Control (PBAC). Together, these mechanisms prevent unauthorized workload execution and strengthen cloud security [24].

B. Data Protection and Encryption

Cloud workloads often process sensitive data, making data protection essential. Encryption techniques, including symmetric, asymmetric, and hybrid encryption, secure data at rest and in transit. Additional mechanisms such as secure key management, digital signatures, hash functions, and Transport Layer Security (TLS) preserve data confidentiality and integrity while minimizing the risk of data leakage and cyberattacks [25].

C. Trust Management and Access Control

Trust management evaluates the reliability of users, virtual machines, and cloud resources before scheduling workloads. Factors such as security reputation, previous interactions, and compliance are considered during trust evaluation. Combined with access control models like RBAC, ABAC, and PBAC, trust management enhances secure resource allocation and enforces organizational security policies [26].

D. Multi-Tenancy Security and Workload Isolation

Cloud computing supports multiple tenants on shared infrastructure, creating risks such as data leakage, side-channel attacks, and resource interference. Workload isolation techniques—including virtual machine isolation, container isolation, sandboxing, and hypervisor security—provide logical separation between tenants, ensuring confidentiality, integrity, and secure workload migration across cloud resources [27].

E. Intrusion Detection and Security Monitoring

Intrusion Detection Systems (IDS) and continuous security monitoring help identify cyber threats, unauthorized access, malware, and denial-of-service attacks [28]. Signature-based, anomaly-based, and AI-enabled IDS analyze cloud logs and network traffic in real time to detect malicious activities. These mechanisms improve threat detection, reduce false alarms, and enhance the security and reliability of cloud workload management [29][30].

Table III summarizes the major security mechanisms employed in cloud workload management. It highlights their roles in ensuring secure workload scheduling, protecting sensitive data, controlling resource access, detecting security threats, and maintaining confidentiality, integrity, and availability in cloud environments.

TABLE III. SECURITY MECHANISMS FOR CLOUD WORKLOAD MANAGEMENT

Security Mechanism	Purpose	Common Techniques	Benefits	Challenges
Security-Aware Workload Scheduling	Securely schedule workloads while meeting QoS and SLA requirements	Risk-aware scheduling, trust-based scheduling, security policies	Secure resource allocation, improved reliability	Increased scheduling complexity
Authentication and Authorization	Verify user identity and control access to cloud resources	MFA, SSO, Digital Certificates, RBAC, ABAC, PBAC	Prevents unauthorized access and insider threats	Key management and policy complexity
Data Protection and Encryption	Protect data confidentiality and integrity during storage and transmission	AES, RSA, TLS/SSL, Digital Signatures, Hashing	Prevents data leakage and tampering	Encryption overhead and key management
Trust Management	Evaluate the trustworthiness of users, VMs, and cloud resources	Reputation models, Trust scores, Behavioral analysis	Improves secure workload placement	Dynamic trust evaluation
Access Control	Restrict resource access according to security policies	RBAC, ABAC, PBAC, Least Privilege	Fine-grained access control and policy enforcement	Policy conflicts and scalability
Multi-Tenancy Security	Protect workloads in shared cloud environments	VM isolation, Container isolation, Hypervisor security	Prevents tenant interference and data leakage	Side-channel attacks

Workload Isolation	Isolate workloads to prevent malicious interactions	Sandboxing, Namespaces, Virtualization, Containers	Enhances confidentiality and system stability	Resource overhead
Intrusion Detection and Monitoring	Detect and respond to cyberattacks in real time	Signature-based IDS, Anomaly-based IDS, AI/ML IDS	Early threat detection and incident response	False positives and computational cost
Secure Resource Allocation	Allocate resources based on security requirements	Trust-aware allocation, Risk-aware provisioning	Secure workload execution	Resource optimization trade-offs
Secure VM and Container Migration	Protect workloads during migration	Encrypted migration, Secure migration protocols	Ensures workload continuity and confidentiality	Migration latency
Privacy Preservation	Protect sensitive user and organizational data	Differential Privacy, Homomorphic Encryption, Data Anonymization	Enhanced privacy and regulatory compliance	High computational overhead
Blockchain-Based Security	Provide decentralized trust and immutable workload records	Smart Contracts, Distributed Ledger	Transparency, integrity, and tamper resistance	Scalability and transaction latency
Zero Trust Security	Continuously verify every access request	Continuous authentication, Least Privilege, Micro-segmentation	Reduces insider and external threats	Complex implementation
Security Monitoring and Auditing	Continuously monitor cloud activities and ensure compliance	SIEM, Log Analysis, Security Auditing	Improves threat visibility and compliance	Large-scale log analysis

5. Open Challenges and Future Direction of Workload Scheduling in Cloud

Despite significant advancements in secure workload scheduling and management, several research challenges remain in cloud computing. Table IV summarizes the major open challenges in secure workload scheduling and highlights potential future research directions for developing intelligent, secure, scalable, and energy-efficient cloud computing systems.

TABLE IV. OPEN CHALLENGES AND FUTURE DIRECTIONS OF WORKLOAD SCHEDULING IN CLOUD COMPUTING

Open Challenge	Description	Future Research Direction
Scalability	Efficiently scheduling large-scale workloads across distributed cloud data centers while maintaining performance and security.	Develop scalable, distributed, and AI-driven scheduling frameworks for large cloud infrastructures.
Multi-Cloud Security	Ensuring secure workload scheduling across heterogeneous cloud providers with different security policies.	Design unified trust management, secure interoperability, and cross-cloud authentication mechanisms.
Real-Time Scheduling	Meeting strict deadline and latency requirements for time-sensitive applications under dynamic workloads.	Develop intelligent real-time scheduling algorithms with adaptive QoS and security guarantees.
Edge–Cloud Integration	Coordinating workload execution between edge devices and cloud resources while minimizing latency and ensuring security.	Design hybrid edge–cloud scheduling frameworks with lightweight security and resource optimization.
Green Cloud Computing	Reducing energy consumption and carbon emissions without degrading workload performance.	Develop energy-aware and carbon-efficient scheduling algorithms using AI and renewable energy models.
Zero-Trust Cloud Security	Continuously verifying users, devices, and workloads before resource allocation in cloud environments.	Integrate Zero-Trust Architecture (ZTA) with workload scheduling and dynamic access control mechanisms.
AI-Driven Secure Scheduling	Balancing intelligent scheduling decisions with robustness against adversarial attacks and biased models.	Develop explainable, trustworthy, and secure AI/ML-based scheduling techniques.
Privacy-Preserving Scheduling	Protecting sensitive user and workload information during scheduling and resource allocation.	Investigate federated learning, homomorphic encryption, and differential privacy for secure scheduling.

Container and Kubernetes Security	Securing workloads deployed in containerized and microservice-based cloud environments.	Develop secure container orchestration, runtime protection, and vulnerability-aware scheduling strategies.
Fault Tolerance and Reliability	Ensuring uninterrupted workload execution despite resource failures and cyberattacks.	Design self-healing, resilient, and predictive scheduling mechanisms for highly available cloud systems.

6. Literature Review

This section reviews existing studies on secure workload scheduling and management techniques in cloud computing systems. The review also explores scheduling algorithms, resource allocation strategies, load balancing techniques, security-aware scheduling models, and AI-driven methods used to enhance cloud performance and protect workloads from security threats. P. K. Boyapati and S. T. R. Kandula (2026) present an intelligent and scalable system for adaptive workload scheduling. The 2019 cluster workload dataset from Google is used in experiments with a preprocessing pipeline in SMOTEENN for data inspection, feature engineering, normalisation, and class imbalance correction. The dataset represents actual resource usage, task patterns, and failures. The MLP and CatBoost models outperform MobileNetV2 (70.4%), Support Vector Regression (70%), and TSSAP clustering (53.8%) when it comes to predicting task failures and scheduling readiness, respectively, with accuracies of 99.15% and 87.62% [31].

Additionally, K. Rajammal et al. (2025) the proposed work incorporates reinforcement learning to optimise SNN decisions based on feedback from the TGNN's state predictions. The findings demonstrate higher predictive accuracy, scalability, and reliability, making the framework applicable to high-performance computing, cloud data centers, and resource-intelligent distributed learning systems. Results from experimental tests using a variety of workload circumstances show that the suggested paradigm significantly improves throughput, energy efficiency, manufacture span, and response time. The suggested model's capacity to manage cloud computing loads is further demonstrated by comparisons with existing optimisation strategies. The suggested approach outperforms the current methods in terms of throughput (20%), makespan (35%), response time (40%) and energy usage (30-40%), according to the results [32].

Thus S. Durga et al. (2025) introduce PConvM-Net, a novel model for workload prediction and resource provisioning. To start, we'll take a look at resource provisioning managers that use Multi-Access Edge Computing (MEC) to supply resources. A Gated Recurrent Unit (GRU) is used to perform workload prediction in the prediction module. The procedure to scale up the threshold is carried out in the decision module. Additionally, a Parallel Convolutional MobileNet (PConvM-Net) is employed to select the quantity of resources for the scale-down and scale-up procedures. Other factors that are taken into account when making the decision include bandwidth, CPU, memory use, energy, and execution time. This is where MobileNet and PCNN come together to form PConvM-Net. Minimum execution time, energy consumption, CPU utilisation, Task Response Time, SLA Violation, and availability were determined by the PConvM-Net simulation results to be 8.616 seconds, 39.876 joules, 83.887 percent, 7.644 seconds, 2.887 percent, and 91.876 percent, respectively [33].

Thus L. Lei and X. Gao (2025) this research presents a proactive approach to scheduling resources in the cloud. Adaptive two-stage multi-neural networks based on long short-term memory (LSTM) are the first approach we suggest for predicting workloads. An optimal number of virtual machines (VMs) to avoid resource adjustment delays and maximise service profit can be found using a proposed algorithm that is based on the network workload prediction algorithm and stochastic queueing theory. The experimental results show that the proactive adaptive elastic resource scheduling framework can optimise the allocation of cloud resources and increase the accuracy of workload predictions (MAPE: 0.0276, RMSE: 3.7085) [34].

H. Chaudhary et al. (2025), present a novel AI-powered framework for dynamic scheduling that integrates deep workload prediction with reinforcement learning. The framework's hybrid decision engine combines traditional queueing metrics with learnt policies, and it uses a two-layer neural network design. Impressive improvements are demonstrated by experimental findings conducted in a simulated cloud environment. Peak resource utilisation is 91%, average waiting time is 30% lower, and queue length is 25% optimised. These results are compared to standard methodologies. Throughput rates are 20-35% greater in the AI-enhanced model across a range of workload intensities, and the reinforcement learning scheduler keeps up its performance even when faced with heavy workloads. The method's efficacy is confirmed by statistical confidence levels that surpass 95%. Findings from this study can help cloud service providers improve service quality while decreasing operating costs through the use of adaptive resource management systems [35].

P. Nehra et al. (2024) suggest a Multiplicative Long

Short-Term Memory (mLSTM)–based approach to workload prediction that takes long-term dependencies into account and permits input-dependent transitions. We put the suggested approach into practice and compare it to other LSTM versions used for workload prediction in the literature. With respect to prediction accuracy, the suggested work surpasses current LSTM variations [36].

A technique that prioritises makespan, average utilisation, and cost is suggested in the paper by V Pillareddy et al. (2023) provide MONWS, a method that calculates the dynamic threshold value for scheduling jobs on virtual machines, to dynamically prioritise tasks in workflow scheduling. This approach employs the min-max technique to minimise finish time and maximise resource utilisation. As compared to preexisting algorithms, MONWS improved makespan by 35%, maximum average cloud utilisation by 8%, and cost by 4% in experiments [37].

Table V provides an overview of current approaches to managing and scheduling workloads in cloud computing environments. The report provides a concise overview of previous research, including its methodologies, datasets, or environments, as well as its main contributions, performance results, and limitations

TABLE V. COMPARATIVE REVIEW OF WORKLOAD SCHEDULING AND MANAGEMENT TECHNIQUES IN CLOUD COMPUTING SYSTEMS

Ref.	Author(s) & Year	Proposed Method	Dataset/Environment	Key Contributions	Performance/Results	Limitations
[31]	P. K. Boyapati and S. T. R. Kandula (2026)	Adaptive workload scheduling using MLP and CatBoost with SMOTEENN preprocessing	Google Cluster Trace 2019	Intelligent workload scheduling, failure prediction, feature engineering	MLP accuracy 99.15%, CatBoost 87.62%; outperformed MobileNetV2, SVR, and TSSAP	Focuses mainly on prediction accuracy; limited security-aware scheduling mechanisms
[32]	K. Rajammal et al. (2025)	Reinforcement Learning with Temporal Graph Neural Network (TGNN) and Spiking Neural Network (SNN)	Diverse cloud workload scenarios	Adaptive resource optimization and workload scheduling	Throughput ↑20%, Makespan ↓35%, Response Time ↓40%, Energy ↓30–40%	High computational complexity and training overhead
[33]	S. Durga et al. (2025)	Parallel Convolutional MobileNet (PConvM-Net) with GRU	Multi-Access Edge Computing (MEC)	Resource provisioning, workload prediction, auto-scaling	Execution Time: 8.616 s, Energy: 39.876 J, CPU Utilization: 83.877%, SLA Violation: 2.877%	Evaluated mainly in MEC environment; security aspects not emphasized
[34]	L. Lei and X. Gao (2025)	Adaptive Two-Stage LSTM with stochastic queueing theory	Cloud computing environment	Proactive VM scheduling and workload prediction	MAPE 0.0276, RMSE 3.7085, R^2 0.9522	Limited discussion on security and attack resilience
[35]	H. Chaudhary et al. (2025)	AI-based workload prediction with Reinforcement Learning scheduler	Simulated cloud platform	Dynamic scheduling and adaptive resource management	Waiting Time ↓30%, Queue Length ↓25%, Resource Utilization 91%, Throughput ↑20–35%	Experimental validation limited to simulation environment
[36]	P. Nehra et al. (2024)	Multiplicative Long Short-Term Memory (mLSTM)	Cloud workload traces	Long-term workload prediction for cloud scheduling	Higher prediction accuracy than conventional LSTM models	Addresses workload prediction only; scheduling optimization not considered
[37]	V. Pillareddy et al. (2023)	MONWS (Min-Max Optimized Network Workflow Scheduling)	Cloud workflow scheduling	Dynamic task priority and workflow scheduling	Makespan ↓35%, Resource Utilization ↑8%, Cost ↓4%	Does not incorporate AI-driven security mechanisms

7. Conclusion

Cloud computing has become an indispensable platform for delivering scalable and on-demand computing services, making efficient and secure workload scheduling and management essential for ensuring high performance, resource utilization, and service reliability. This survey presented a comprehensive review of the fundamentals of workload scheduling, secure scheduling techniques, and security mechanisms employed in cloud workload management. It examined traditional and intelligent scheduling approaches, including heuristic, metaheuristic, and AI/ML-based techniques, along with security mechanisms such as authentication, encryption, trust management, workload isolation, and intrusion detection that enhance the confidentiality, integrity, and availability of cloud workloads. The survey also reviewed recent research contributions, identified current trends and research gaps, and discussed major open challenges. As cloud environments continue to evolve, future research should focus on developing intelligent, adaptive, and security-aware scheduling frameworks that integrate AI, blockchain, edge computing, and Zero-Trust principles to improve workload management. This survey provides researchers and practitioners with a comprehensive understanding of existing techniques, emerging challenges, and future directions for designing secure, scalable, and efficient cloud workload scheduling and management systems.

REFERENCES

1. J. B. Mehta, "Autonomous Workload Right-Sizing for Multi-Cloud Cost Optimization," in 2026 International Conference on Artificial Intelligence, Systems, and Emerging Technologies (ICAISET), Cairo, Egypt: IEEE, Apr. 2026, pp. 1–11. doi: 10.1109/ICAISET66439.2026.11541899.

2. S. K. Madishetty and G. N. M. V. Kumar, "Intelligent Surgical Tables: Real-Time Patient Monitoring and Autonomous Adjustment Using IoT, Cloud Computing, and Cybersecurity," J. Comput. Anal. Appl., vol. 31, no. 02, pp. 803–809, Feb, Feb. 2023, doi: 10.48047/jocaaa.2023.31.02.50.

3. T. Shah, "Cloud-Based Data Warehousing for Marketing Agility: Lessons from FinTech Migrations to Snowflake and AWS," Int. J. Adv. Res. Sci. Commun. Technol., vol. 4, no. 4, March, pp. 642–652, Mar. 2024, doi: 10.48175/IJARSCT-16000B.

4. Mohit Mittal, "The Great Migration: Understanding the Cloud Revolution in IT," Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol., vol. 10, no. 6, pp. 2222–2228, Dec. 2024, doi: 10.32628/CSEIT2410612423.

5. S. Jain and D. Jain, "Artifact Comparison Analyzer: Evaluating Microservice Build Metrics for Performance and Efficiency Improvements," in 2026 IEEE International Conference on AI Engineering and Innovations (AIEI), IEEE, Mar. 2026, pp. 1–6. doi: 10.1109/AIEI69164.2026.11497468.

6. R. Vasikarla, S. Kakkar, B. Makkena, and A. Kakkar, "An Enhanced Artificial Intelligence Framework for Cloud Resource Usage Anomaly Detection in Site Reliability Engineering," in 2026 2nd International Conference on Cognitive Computing in Engineering, Communications, Sciences and Biomedical Health Informatics (IC3ECSBHI), IEEE, Feb. 2026, pp. 1418–1424. doi: 10.1109/IC3ECSBHI67834.2026.11468993.

7. S. K. Malaraju and S. K. Madishetty, "AI-Augmented Compiler Optimization for Energyefficient Software Execution on Embedded Systems," in 2025 14th International Conference on System Modeling & Advancement in Research Trends (SMART), Moradabad, Uttar Pradesh, India: IEEE, 2025, pp. 1–6, November. doi: 10.1109/SMART66937.2025.11389313.

8. S. K. Anumula, N. Talluru, R. Gangavarapu, S. Gupta, and K. Tatavarthy, "Quantum-Inspired Computing: Classical Approaches to Machine Learning," Quantum-Inspired Neural Networks.

9. D. Kanellopoulos and V. Sharma, "Dynamic Load Balancing Techniques in the IoT: A Review," Symmetry (Basel)., vol. 14, no. 12, p. 2554, Dec. 2022, doi: 10.3390/sym14122554.

10. R. K. Kanneganti, "Security and Compliance Issues in Cloud-Based Deployments of Content and Workflow Management Systems," Eastasouth J. Inf. Syst. Comput. Sci., vol. 2, no. 01, pp. 131–138, Aug. 2024, doi: 10.58812/esiscs.v2i01.1122.

11. A. Vafamehr, "Energy-aware cloud computing," Electr. J., vol. 31, 2018, doi: 10.1016/j.tej.2018.01.009.

12. N. Adik, "The Rise of Open and Free Networks: A Community-Driven Paradigm for Decentralized Connectivity," in 2025 IEEE First International Conference on Innovations in Engineering and Next-Generation Technologies for Sustainability (ICINVENTS), IEEE, Nov. 2025, pp. 1–9. doi: 10.1109/ICINVENTS64613.2025.11402224.

13. A. Warrier, "Hybrid Cloud iPaaS for Healthcare Digital Transformation: Bridging On-Premises and Cloud-Based Health Information Systems," Int. Sci. J. Eng. Manag., vol. 2, no. 1, p. 7, 2023.

14. V. Sharma, "Cloud-Native 5G Deployments: Kubernetes and Microservices in Telco Networks," Int. J. Innov. Res. Eng. Multidiscip. Phys. Sci., vol. 10, no. 3, pp. 1–8, May 2022, doi: 10.37082/IJIRMPS.v10.i3.232706.

15. C. Chen, Y. Li, G. Cao, and J. Zhang, "Research on Dynamic Scheduling Model of Plant Protection UAV Based on Levy Simulated Annealing Algorithm," Sustainability, vol. 15, no. 3, p. 1772, Jan. 2023, doi: 10.3390/su15031772.

16. M. Kari, "A Comprehensive Performance Study of Hashing Algorithms: MD5, SHA-256, and SHA-512 using Java," Int. J. Curr. Eng. Technol., vol. 13, no. 6, pp. 1–8, 2023.

17. S. D. Veeravalli, S. B. Venkata, A. Ashirova, R. MS, S. A. D. Poovaiah, and D. Turaev, "Breaking the Swarm Paradigm: Single Candidate Optimization for Cloud Task Allocation," in 2025 International Conference on Innovations in Intelligent Systems: Advancements in Computing, Communication, and Cybersecurity (ISAC3), Bhubaneswar, India: IEEE, 2025, pp. 1–7, July.

18. R. U. L. A. Yeruva, D. Singh, S. Suddala, and N. Bhatt, "Augmented Data Management: Enhancing Cache Performance and Cybersecurity through AI and Mobile Integration," J. Comput. Mech. Manag., 2026.

19. Y. Muni, "Understanding the Evolution of Internet Protocols: An In-Depth Review of IPV4 and IPV6: A Comparative Review of Transition Challenges and Solutions," Int. J. Adv. Res. Comput. Sci., vol. 16, no. 4, pp. 109–117, Aug. 2025, doi: 10.26483/ijarcs.v16i4.7307.

20. K. R. Kiran, "API Integration Barriers in Case-Based Process Management: A Review of Interoperability and Real-Time Response Constraints Outline," Qual. Res., vol. 25, no. 3, pp. 21–48, 2025, doi: https://doi.org/10.5281/zenodo.16782616.

21. A. Joon, B. K. R. Janumpally, A. Gogineni, and P. Chatterjee, "Efficient Large-Scale Intrusion Identification and Prevention in Distributed Cloud Networks Using Artificial Intelligence," in 2025 5th International Conference on Intelligent Technologies (CONIT), HUBBALI, India: IEEE, Jun. 2025, pp. 1–8. doi: 10.1109/CONIT65521.2025.11167760.

22. H. P. Cyril, N. D. Bhandarwar, S. Kumara, and S. Mathur, "Securing Containerised Applications Using Kubernetes-Based Orchestration in Software Development," in 2026 International Conference on Intelligent Computing, Networks, and Security (IC-ICNS), IEEE, Mar. 2026, pp. 1–6. doi: 10.1109/IC-ICNS68863.2026.11537887.

23. V. Chaturvedi, "Modern Software Development with Java , Spring Boot , and Python : A Survey of Frameworks and Best Practices," vol. 3, no. 4, pp. 188–197, 2023, doi: 10.56472/25832646/JETA-V3I8P121.

24. P. C. Pradnya H Desai, Parikshit Mahalle, Intelligent Access Control Schemes for the Internet of Everything: A Survey of Techniques, Challenges, and Future Directions. Springer Nature Switzerland, 2026.

25. J. B. Mehta, "An Autonomous Dependency And Failure-Propagation Modeling System For Security-Critical Distributed Enterprise Cloud Computing Environments," 202641001713, 2026

26. R. Azmeera, "The Effects of Increased Software Errors on Software Product Functionality," University of the Cumberlands ProQuest, 2025.

27. V. Gudur, C. Upadhyay, S. Sudarsan, R. Dandigam, A. Patel, and S. Tiwari, "Trust Deviation Metrics (TDM) for AI-Driven Cloud Orchestration," in Proceedings of the 2026 9th International Conference on Information and Computer Technologies, New York, NY, USA: ACM, Mar. 2026, pp. 539–544. doi: 10.1145/3803291.3803293.

28. M. Mittal, "Federated Learning: An Intrusion Detection Privacy-Preserving Approach to Decentralized AI Model Training for IOT Security," Int. J. Adv. Res. Electr. Electron. Instrum. Eng., vol. 07, no. 01, Jan. 2018, doi: 10.15662/IJAREEIE.2018.0701001.

29. S. Tyagi and A. Tyagi, "Evaluating the Performance with Deep Learning Techniques in Cybersecurity for Effective Threat Detection," 2026, pp. 437–452. doi: 10.1007/978-981-96-8687-2_31.

30. S. C. -, "Risk Management in Advanced Persistent Threats (APTs) for Critical Infrastructure in the Utility Industry," Int. J. Multidiscip. Res., vol. 3, no. 4, Aug. 2021, doi: 10.36948/ijfmr.2021.v03i04.34396.

31. P. K. Boyapati and S. T. R. Kandula, "High-Performance Distributed Deep Learning Using Adaptive Parallelism and Dynamic Workload Scheduling," in 2026 14th International Symposium on Digital Forensics and Security (ISDFS), Boston, MA, USA: IEEE, Mar. 2026, pp. 01–06. doi: 10.1109/ISDFS69419.2026.11459016.

32. K. Rajammal and M. Chinnadurai, "Dynamic load balancing in cloud computing using predictive graph networks and adaptive neural scheduling," Sci. Rep., vol. 15, no. 1, p. 22181, Jul. 2025, doi:

10.1038/s41598-025-97494-2.

33. D. S, E. Daniel, D. S, and R. V.K, “Deep learning-based workload prediction and resource provisioning for mobile edge-cloud computing in healthcare applications,” Sustain. Comput. Informatics Syst., vol. 47, p. 101176, Sep. 2025, doi: 10.1016/j.suscom.2025.101176.

34. L. Li and X. Gao, “Profit-Efficient Elastic Allocation of Cloud Resources Using Two-Stage Adaptive Workload Prediction,” Appl. Sci., vol. 15, no. 5, p. 2347, Feb. 2025, doi: 10.3390/app15052347.

35. H. Chaudhary, G. Sharma, D. K. Nishad, and S. Khalid, “AI-enhanced modelling of queueing and scheduling systems in cloud computing,” Discov. Appl. Sci., vol. 7, no. 4, p. 276, Mar. 2025, doi: 10.1007/s42452-025-06755-2.

36. P. Nehra and N. Kesswani, “A workload prediction model for reducing service level agreement violations in cloud data centers,” Decis. Anal. J., vol. 11, p. 100463, Jun. 2024, doi: 10.1016/j.dajour.2024.100463.

37. V. R. Pillareddy and G. R. Karri, “MONWS: Multi-Objective Normalization Workflow Scheduling for Cloud Computing,” Appl. Sci., vol. 13, no. 2, p. 1101, Jan. 2023, doi: 10.3390/app13021101.