

ORIGINAL

Resource allocation on periotity based schuduling and improve the security using DSSHA-256

Asignación de recursos basada en la periodicidad y mejora de la seguridad mediante DSSHA-256

K.Prathap Kumar¹ , K. Rohini² 

¹Research Scholar, Vistas, Chennai-48, India.

²Professor, Department of Computer Application, Vistas, Chennai-48, India.

Cite as: Kumar KP, Rohini K. Resource allocation on periotity based schuduling and improve the security using DSSHA-256. Data and Metadata 2024;3:193. <https://doi.org/10.56294/dm2024193>.

Submitted: 28-10-2023

Revised: 06-12-2023

Accepted: 07-02-2024

Published: 08-02-2024

Editor: Prof. Dr. Javier González Argote 

ABSTRACT

Cloud computing has gained popularity with advancements in virtualization technology and the deployment of 5G. However, scheduling workload in a heterogeneous multi-cloud environment is a complicated process. Users of cloud services want to ensure that their data is secure and private, especially sensitive or proprietary information. Several research works have been proposed to solve the challenges associated with cloud computing. The proposed Adaptive Priority based scheduling (PBS) focuses on reducing data access completion time and computation expense for task scheduling in cloud computing. PBS assigns tasks depending on its size and selects the minimum cost path for data access. It contains a task register, scheduler, and task execution components for efficient task execution. The proposed system also executes a double signature mechanism for data privacy and security in data storage. This study correlates the performance of three algorithms,

PBS, (Task Requirement Degree) TRD and (recommended a Risk adaptive Access Control) RADAC, in terms of task execution time and makespan time. The experimental results demonstrate that PBS outperforms TRD and RADAC in both metrics, as the number of tasks increases. PBS has a minimum task execution time and a lower makespan time than the other two algorithms.

Keywords: Cloud Computing; Cryptography; PBS; TRD; RADAC.

RESUMEN

La computación en nube ha ganado popularidad con los avances en la tecnología de virtualización y el despliegue de 5G. Sin embargo, programar la carga de trabajo en un entorno heterogéneo de nubes múltiples es un proceso complicado. Los usuarios de servicios en la nube quieren garantizar la seguridad y privacidad de sus datos, especialmente de la información sensible o sujeta a derechos de propiedad. Se han propuesto varios trabajos de investigación para resolver los retos asociados a la computación en nube. La propuesta de programación adaptativa basada en prioridades (PBS) se centra en reducir el tiempo de finalización del acceso a los datos y el gasto computacional para la programación de tareas en la computación en nube. PBS asigna tareas en función de su tamaño y selecciona la ruta de coste mínimo para el acceso a los datos. Contiene un registro de tareas, un planificador y componentes de ejecución de tareas para una ejecución eficiente de las mismas. El sistema propuesto también ejecuta un mecanismo de doble firma para la privacidad y seguridad en el almacenamiento de datos. Este estudio correlaciona el rendimiento de tres algoritmos, PBS, (Task Requirement Degree) TRD y (recommended a Risk adaptive Access Control) RADAC, en términos de tiempo de ejecución de tareas y tiempo de makespan. Los resultados experimentales demuestran que PBS supera a TRD y RADAC en ambas métricas a medida que aumenta el número de tareas. PBS tiene un tiempo de ejecución de tarea mínimo y un tiempo de makespan menor que los otros dos algoritmos.

Palabras clave: Cloud Computing; Criptografía; PBS; TRD; RADAC.

INTRODUCTION

Cloud computing acquired tremendous popularity in recent years, mainly due to advancements in virtualization technology. This technology has shown a new era to many applications that applies speech and signal processing^(1,2) with 5g,⁽³⁾ which provides higher bandwidth and flexibility. To get pay-per-use services, many applications were forwarded to data centers. Due to resource limitations, on-demand workloads are often moved to other data centers. As a result of the wide range of cloud availability,^(4,5,6,7) the task becomes complicated and challenging for scheduling workload in the heterogeneous multi-cloud environment. Cloud workloads were organized using virtual machines (VM) that are arranged based on the volume of traffic and requests. This methodology allows users to access high-quality services and applications on demand from a shared pool of configurable computing resources without the need to worry about data storage and maintenance on their devices.⁽⁸⁾ However, when handlers with limited computer resources depend on cloud computing, it will be a challenge for them to ensure the security of data integrity. Moreover, the users have to handle cloud storage in the same manner as local storage, without bothering about the reliability of the data. The applications also contain sensitive or proprietary data, so there is a need to execute secure infrastructure.⁽⁸⁾ In order to handle sensitive requests, they are given special treatment and executed on a unique virtual machine (VM). The process of locating the most suitable server or physical machine (PM) to anchor a VM is known as VM placement. It is essential for improving the usage of computing resources, power efficiency, and quality of service (QoS) in cloud computing. However, implementing VM placement is a challenging and complex task because it is difficult to predict how VM requests will arrive and understand their patterns. Additionally, the large data makes it hard to identify the suitable scenario for the required workload.

In order to handle sensitive requests, they are given special treatment and executed on a unique virtual machine (VM). The process of locating the most suitable server or physical machine (PM) to anchor a VM is known as VM placement. It is essential for improving the usage of computing resources, power efficiency, and quality of service (QoS) in cloud computing. However, implementing VM placement is a challenging and complex task because it is difficult to predict how VM requests will arrive and understand their patterns. Additionally, the large data makes it hard to identify the suitable scenario for the required workload. Confidentiality, integrity, and data availability ensure the security of the data. Cloud users expect affirmation that their information is secure and protected from attacks, such as intrusion, hacking, data theft, and denial of service attacks. Unfortunately, 57 % of the companies reported security violations while using cloud services.⁽⁹⁾ Data privacy is considered more substantial than data security because cloud service providers have unrestricted access to all user information and observe the user's action, which can compromise users privacy. For instance, a person who is a diabetic might search for what type of medication they have to follow and research more about their symptoms, cloud services observe the activities and store information about their health. Since the cloud service provider has unrestricted access to this information, there is a risk that sensitive data may be compromised or shared with third-party companies like an insurance company or private pharmacy without the user's consent.⁽¹⁰⁾ This being the case, some of the users may be hesitant to store their personal or sensitive data in the cloud due to a lack of trust in the service provider and this causes a lot of legal issues. One possible solution is for the user to install a proxy that encrypts and protects their data before forwarding it to the cloud service provider.⁽¹¹⁾ This approach can help reduce the risk of data breaches or unauthorized access.

Related Works

Shojafar et al. suggested a genetic-based scheduler in their work.⁽¹²⁾ This proposed scheduler aims to mitigate energy consumption and makespan. The scheduler contains two phases: task prioritization and processor assignment. The authors applied three different prioritization methods to find the priority of the tasks, which helped in creating optimized initial chromosomes. The optimized chromosomes were then used to align the tasks to processors based on an energy-aware model. In reference⁽¹³⁾, the authors proposed an algorithm for energy-efficient task scheduling in a heterogeneous environment. The purpose of this algorithm is to prevail over the drawbacks of task consolidation and scheduling. The proposed algorithm considers both the completion time and total resource utilization. To proceed with a scheduling decision, the algorithm follows a normalization procedure. In their work referenced as⁽¹⁴⁾, Yousri et al. recommended a mechanism that takes into account both load balancing and temperature awareness in virtual machine (VM) allocation. Their proposed algorithm aims to ensure the quality of service by choosing a physical machine to host a virtual machine according to user requirements, load on the hosts, and temperature of the hosts. The algorithm assigns VMs to physical hosts with the temperature and CPU utilization of processors to prevent overloading or overheating of the target host. Reference⁽¹⁵⁾ explains the work of authors who formulated an energy efficiency problem to improve resource

utilization and system throughput. Their work involves a multiple-procedure heuristic workflow scheduling and consolidation strategy that aims to maximize resource utilization and mitigate power consumption. To achieve this, various techniques are utilized, such as dynamic voltage and frequency scaling (DVFS) with task module migration for workload balancing and task consolidation for virtual machine (VM) overhead reduction. Zhao et al.⁽¹⁶⁾ presented an energy and deadline-aware task scheduling approach that involves two methods. Firstly, tasks and datasets are tailored as a binary tree using a data correlation clustering algorithm, which aims to mitigate the quantity of global data transmission and ultimately decrease the rate of SLA violations. Secondly, a Tree-to-Tree task scheduling approach with the calculation of the Task Requirement Degree (TRD) is proposed.

This aims to improve energy efficiency by reducing the number of active machines, mitigating global time consumption on data transmission, and optimizing the utilization of computing resources and network bandwidth. Fall et al.⁽¹⁷⁾ recommended a Risk adaptive Access Control (RAdAC) for real-time privacy preservation of sensitive information. RAdAC combines Policy-Based Access Control, Attribute-Based Access Control, and Machine learning to calculate risk, establish acceptable risk levels, and tailor data to not exceed them. It gives flexibility and adaptability compared to traditional access control by checking past access control decisions, quantifying privileged, subject and object, and discovering failures. Yu et al.⁽¹⁸⁾ suggested a framework that models user interactions to achieve privacy goals, allowing reasoning about non-functional requirements by modeling relationships between users. The framework adapts to different alternatives and could be enhanced to study the relationship between trust and privacy, as the existence of pre-established relationships can probably impact privacy. Kobasa et al.⁽¹⁹⁾ presented security requirements for privacy preservation and user anonymity in personalized cloud systems, implementing an architecture that guarantees the security and privacy of the cloud systems. The authors in⁽¹⁸⁾ made a method to enable users to remain anonymous throughout data collection. Kobasa et al. utilized the concept of Pseudo Anonymity to halt N entities from revealing user identity and included a trusted entity in every $N+1$ component to improve anonymity. Coppolino et al.⁽¹⁹⁾ suggested Homomorphic Encryption (HE) preserve user privacy while identifying anomalous intrusion and detection in network traffic. The HE scheme encrypted data from third-party monitoring services, providing ad hoc intrusion and detection while maintaining privacy. Although the HE scheme was able to monitor generated code injection attacks,⁽²⁰⁾ there were limitations in terms of overhead and performance, and the IDS did not have access to the private key.

METHODS

Adaptive Work Size Based Queuing Process: This section explains the proposed Adaptive Priority based scheduling (PBS) for task scheduling in virtual machines (VMs) to attain cost-efficiency. We aimed to enhance previous scheduling processes that considered overall execution time and total monetary costs but were unsuccessful due to collaborative factors. Execution time refers to the time between task assignment and completion in VMs. As per our inspection, we found varying time execution, and there is an increased resource cost for computation, communication, storage, data transfer, and more. The focus of PBS is mainly on the finishing time of the data access and computation expense. VMs require data from data centers located in different locations to execute tasks, and also scheduling tasks derived from VM processing is pivotal. User requests are initially gathered and processed in a FIFO queue. In PBS, task size is considered, and a higher priority is given to smaller tasks as they can be implemented faster.

The service time of data access is essential for achieving productivity, and the minimum or maximum data access time depends on choosing suitable data paths. The network input/output requests in the physical machines or VMs are applied as a parameter for finding access time. As PBS allocates tasks depending upon their size and the path with the lowest cost. If a virtual machine fails to respond, the task is redirected to the nearest alternative virtual machine to prevent long loading times. When a task wants a large amount of data, it needs to be transferred through a larger data path, which can result in lower priority. Consequently, the task may still perform, but using fewer available resources. Moving on to the AWSQ Architecture, it involves various components that work together to perform its services.

Task register: It is responsible for receiving user requests and forming a queue for further processing. It undergoes the FIFO(First-In-First-Out) principle, which means that the first task to enter the queue will be the first to be processed. The task register prioritizes the queue according to their task size, giving higher priority to smaller tasks and lower priority to larger tasks

Scheduler: It assigns available virtual machines to perform tasks. It determines the task size and processing time, taking into account data that may be located in various data centers across different locations. If a task is overloaded or if the time to access data increases, the scheduler redirects the task to the next available virtual machine. The scheduler chooses the most cost-effective data path for optimal execution speed.

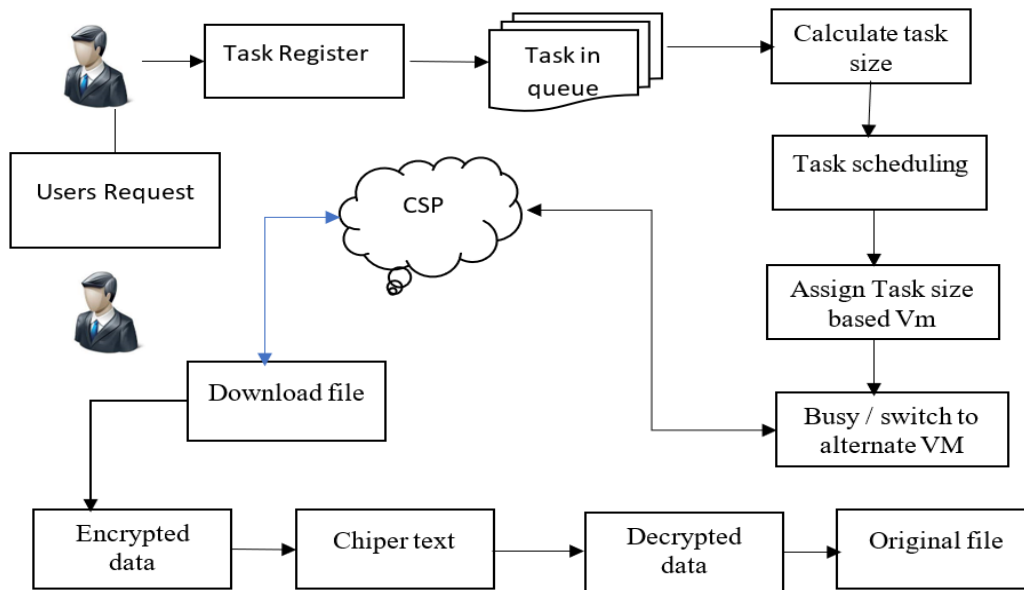


Figure 1. Proposed structure diagram

Performance analysis

It calculates the data access completion time for each executed task. The computation and communication costs are also estimated to determine the total cost of each data path. High priority tasks have minimum access times compared to lower priority tasks. In addition to this, the CPU utilization and bandwidth of each executed task are also formulated to obtain a more comprehensive analysis.

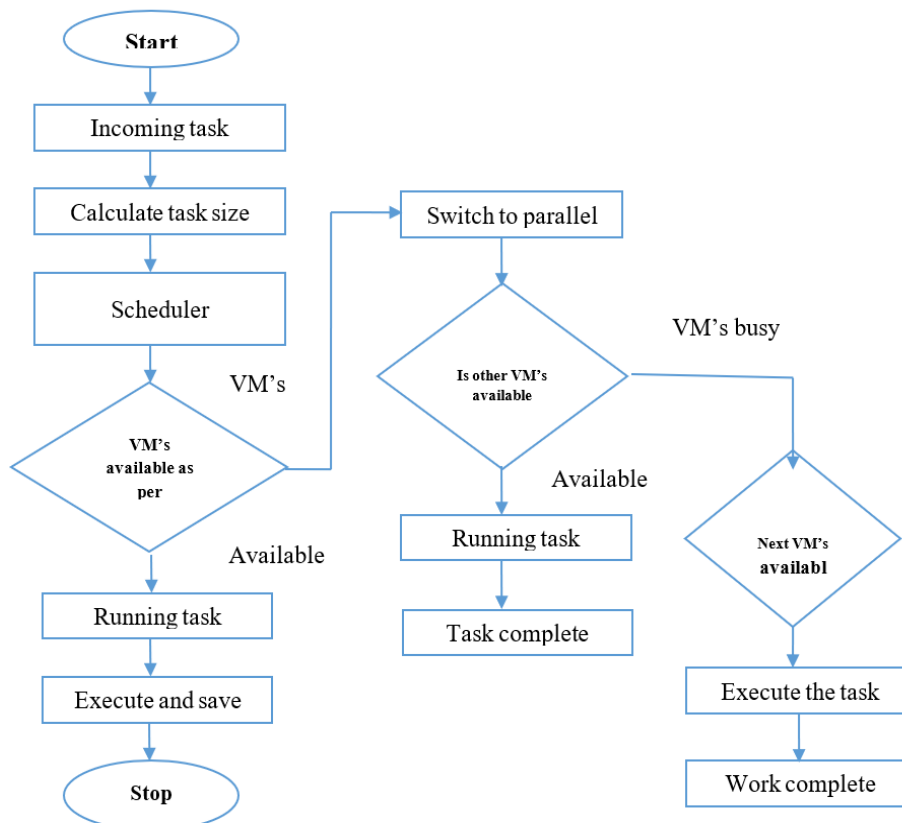


Figure 2. illustrates the functioning of the PBS system

Users direct task requests to the task register, then queues the tasks depending upon their size and assigns priority based. The task scheduler uses the proposed PBS mechanism to allot the available path and a select a cost-effective path for efficient processing. If a virtual machine is busy or the loading time for data access

expands, the task is terminated from the processing VM and allocated to the next available VM. Similarly, if a VM is busy, the task will be allotted to multiple VMs to finish processing. The PBS system accurately calculates the required bandwidth to finish each task within the deadline.

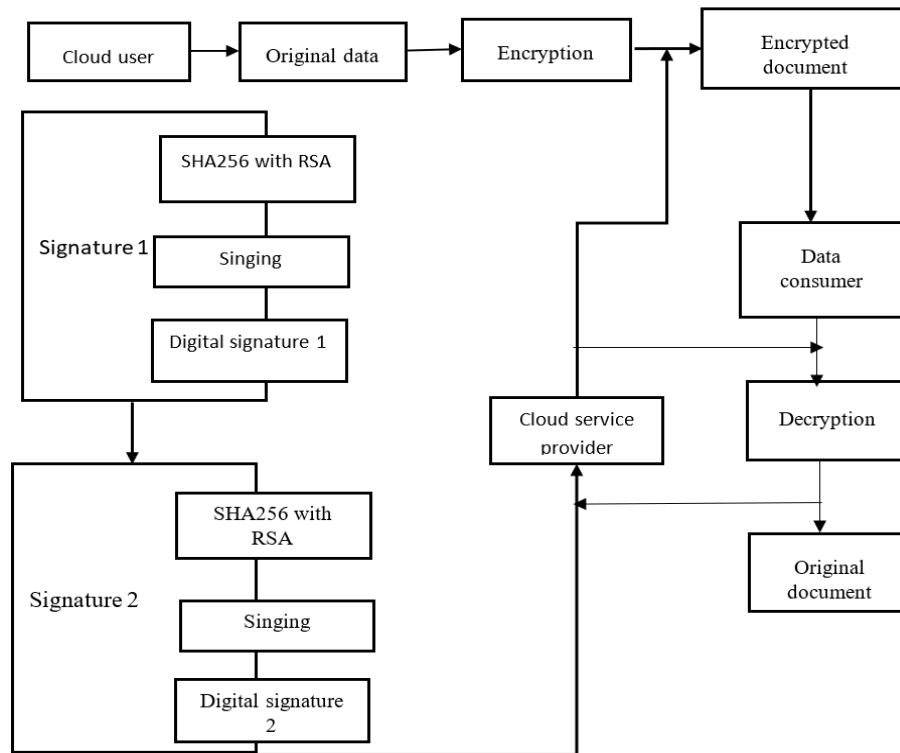


Figure 3. Encryption and decryption architecture

Figure 3 shows how a user can register with a Cloud Service Provider (CSP) to use their services with our proposed methodology. The user can execute their operations in a cloud environment after a successful registration. The focus of our study is on the security of data storage, which involves encrypting various types of original data, including text, images, audio, and video, before placing them in the cloud. To assure data privacy, we are introducing a double signature mechanism. The user first encrypts the data by applying SHA256 with RSA(Rivest-Shamir-Adleman), a symmetric algorithm with public and private keys. The encrypted data is then forwarded to the CSP with digital signature 1, after which CSP performs the second encryption with digital signature 2 before allocating it in the cloud. To retrieve the original data, the user has to request it from the data consumer, who checks and confirms the digital signature of the request. If it's valid, the data can be accessed by providing the private key received during encryption, which is used for decryption to attain the original data.

Experimental result

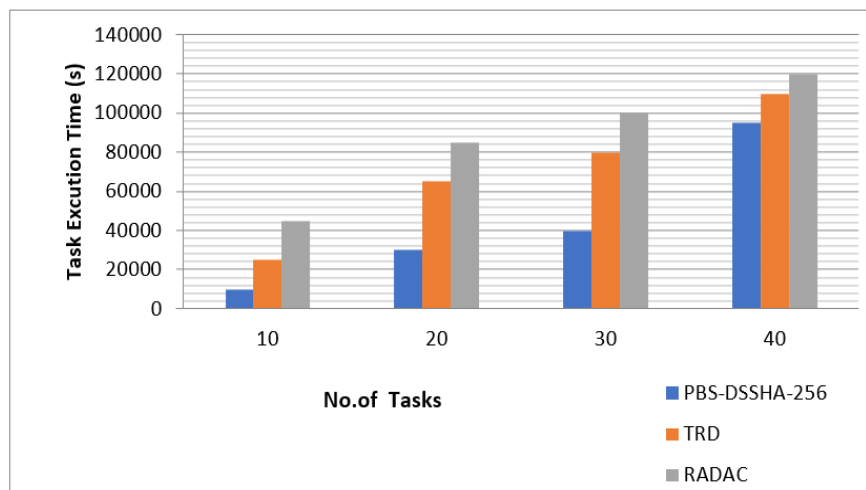


Figure 4. Number of tasks Vs Task Execution time(s)

Figure 4 shows the experimental results of three algorithms, namely PBS-DSSHA-256, TRD, and RADAC, in terms of task execution time in seconds with varying numbers of tasks. The X-axis denotes the number of tasks, with increments of 25 units from 0 to 100. The Y-axis denotes task execution time, with increments of 20 000 units from 0 to 140 000. The blue bars represent the proposed PBS algorithm, the orange bars represent the TRD algorithm, and the gray bars represent the RADAC algorithm. For example, when there are 25 tasks, PBS finishes the tasks in 10 000 seconds, while TRD and RADAC take 24 000s and 44 000s, respectively. As the number of tasks increases, PBS exceeds the other two algorithms in terms of task execution time. Comparing the three algorithms, PBS finishes the tasks faster than TRD and RADAC.

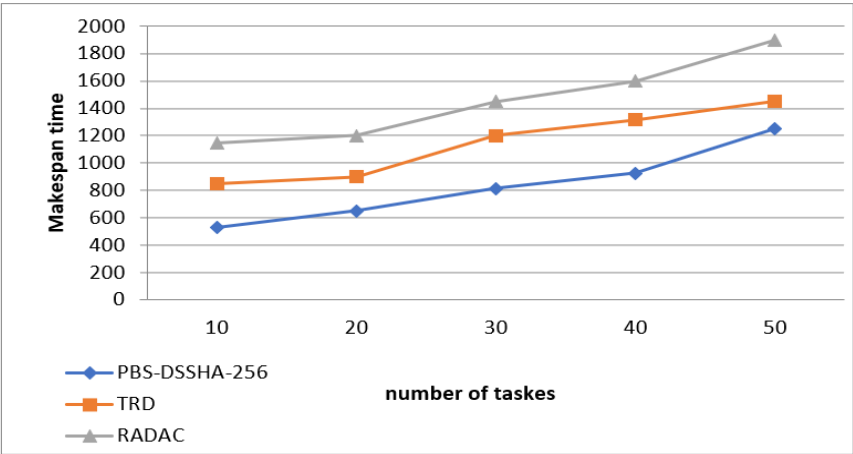


Figure 5. Number of tasks vs Make span time

Figure 5 illustrates the makespan time of PBS-DSSHA-256, TRD, and RADAC with varying numbers of tasks. The X-axis shows the number of tasks, while the Y-axis shows the makespan time. The blue diamond shape corresponds to PBS, the orange square shape corresponds to TRD, and the gray triangular shape corresponds to RADAC. For instance, when there are 50 tasks, PBS has a makespan time of around 1100 seconds, while TRD and RADAC have makespan times of around 1300s and 1900s, respectively. In terms of makespan time PBS outperforms the other two algorithms.

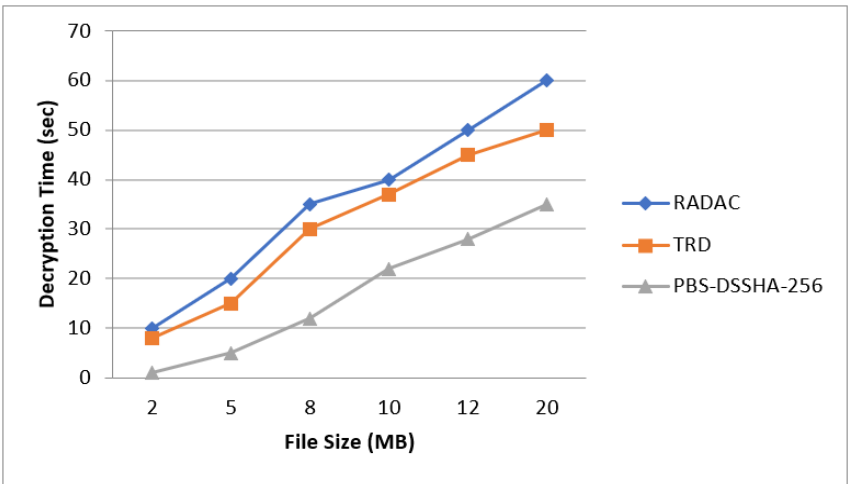


Figure 6. Decryption time versus file size

Figure 6 illustrates the decryption time of PBS-DSSHA-256, TRD, and RADAC with file size. The X-axis shows the file size(MB), while the Y-axis shows the decryption time in seconds. The blue diamond shape corresponds to PBS, the orange square shape corresponds to TRD, and the gray triangular shape corresponds to RADAC. For instance, when the file size is 20 MB, PBS has a decryption time of around 60 seconds, while TRD and RADAC have decryption times of around 50s and 35s, respectively. In terms of decryption time PBS takes short time than the other two algorithms.

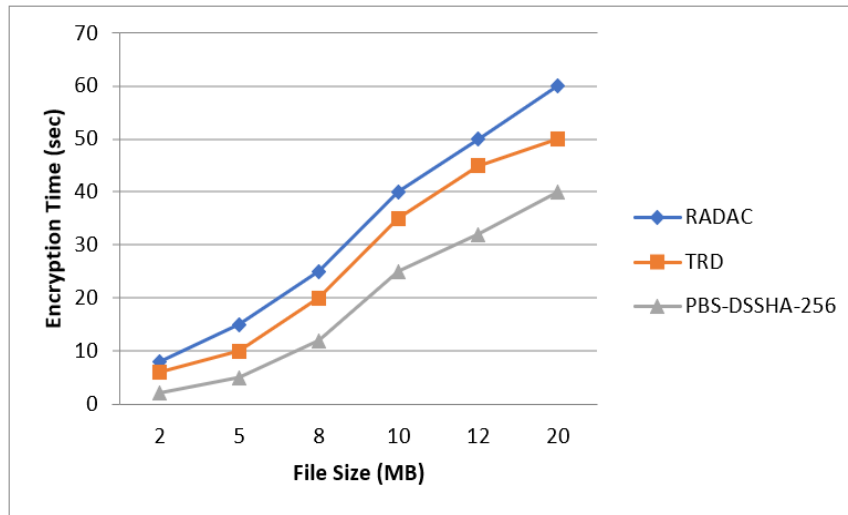


Figure 7. Encryption time versus file size

Figure 7 illustrates the encryption time of PBS, TRD, and RADAC with file size. The X-axis shows the file size(MB), while the Y-axis shows the encryption time in seconds. The blue diamond shape corresponds to PBS, the orange square shape corresponds to TRD, and the gray triangular shape corresponds to RADAC. For instance, when the file size is 20 MB, PBS has a decryption time of around 60 seconds, while TRD and RADAC have decryption times of around 50s and 40s, respectively. In terms of decryption time PBS takes longer time than the other two algorithms.

CONCLUSIONS

In conclusion, cloud computing provides flexible and cost-effective access to a range of computing resources. However, it poses significant challenges in terms of workload scheduling and ensuring data privacy and security. The proposed Adaptive Priority based scheduling (PBS) system focuses to improve cost-efficiency in task scheduling for virtual machines. By prioritizing smaller tasks and selecting the most cost-effective data path, PBS can optimize the execution time and minimize resource expenses.. The proposed double signature mechanism safeguards data privacy and security in the cloud environment. The experimental results show that the PBS algorithm performance exceeds the TRD and RADAC algorithms in terms of both task execution time and makespan time. As the number of tasks increases, PBS sustains its superiority over the other two algorithms. These findings suggest that PBS can be a more efficient choice for scheduling the task. Overall, cloud computing has immense strength for transforming the IT industry and enabling new possibilities for businesses and individuals alike.

REFERENCES

1. S. Debnath, B. Soni, U. Baruah, and D. K. Sah, "Text-dependent speaker verification system: a review," in Proceedings of the 2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO), pp. 1-7, IEEE, Coimbatore, India, January 2015
2. D. K. Sah, C. Shivalingagowda, and D. P. Kumar, "Optimization problems in wireless sensors networks," Soft computing in wireless sensor networks, vol. 2018, pp. 41-62, 2018.
3. D. N. K. Jayakody, K. Srinivasan, and V. Sharma, 5G Enabled Secure Wireless Networks, Springer International Publishing, Cham, 2019.
4. H. Hu, Z. Li, H. Hu et al., "Multi-objective scheduling for scientific workflow in multicloud environment," Journal of Network and Computer Applications, vol. 114, pp. 108-122, 2018.
5. C. Georgios, F. Evangelia, M. Christos, and N. Maria, "Exploring cost-efficient bundling in a multi-cloud environment," Simulation Modelling Practice and Teory, vol. 111, Article ID 102338, 2021.
6. Y. Kang, L. Pan, and S. Liu, "Job scheduling for big data analytical applications in clouds: a taxonomy study," Future Generation Computer Systems, vol. 135, pp. 129-145, 2022.

7. S. A. Zakaryia, S. A. Ahmed, and M. K. Hussein, "Evolutionary offloading in an edge environment," *Egyptian Informatics Journal*, vol. 22, no. 3, pp. 257-267, 2021.
8. S. K. Panda and P. K. Jana, "Efficient task scheduling algorithms for heterogeneous multi-cloud environment," *9e Journal of Supercomputing*, vol. 71, no. 4, pp. 1505-1533, 2015
9. T. Alam, "Cloud computing and its role in the information technology," *IAIC Transactions on Sustainable Digital Innovation (ITSDI)*, vol. 1, no. 2, pp. 108-115, 2020
10. Castillo-Gonzalez W, Lepez CO, Bonardi MC. Augmented reality and environmental education: strategy for greater awareness. *Gamification and Augmented Reality 2023*;1:10-10. <https://doi.org/10.56294/gr202310>.
11. Aveiro-Róbaló TR, Pérez-Del-Vallín V. Gamification for well-being: applications for health and fitness. *Gamification and Augmented Reality 2023*;1:16-16. <https://doi.org/10.56294/gr202316>.
12. Z. Sanaei, S. Abolfazli, A. Gani, and R. Buyya, "Heterogeneity in mobile cloud computing: taxonomy and open challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 369-392, 2014
13. R. Branch, H. Tjeerdsma, C. Wilson, R. Hurley, and S. McConnell, "Cloud computing and big data: a review of current service models and hardware perspectives," *Journal of Software Engineering and Applications*, vol. 07, no. 08, pp. 686-693, 2014.
14. Shojafar, M.; Kardgar, M.; Hosseinabadi, A.A.R.; Shamshirband, S.; Abraham, A. TETS: A Genetic-Based Scheduler in Cloud Computing to Decrease Energy and Makespan. In *Proceedings of the 15th International Conference HIS 2015 on Hybrid Intelligent Systems*, Seoul, South Korea, 16-18 November 2015; pp. 103-115.
15. Panda, S.K.; Jana, P.K. An energy-ecient task scheduling algorithm for heterogeneous cloud computing systems. *Clust. Comput.* 2018.
16. Mhedheb, Y.; Streit, A. Energy-ecient Task Scheduling in Data Centers. In *Proceedings of the 6th International Conference on Cloud Computing and Services Science*, Rome, Italy, 23-25 April 2016; Volume 1, pp. 273-282.
17. Khaleel, M.; Zhu, M.M. Energy-ecient Task Scheduling and Consolidation Algorithm for Workflow Jobs in Cloud. *Int. J. Comput. Sci. Eng.* 2016, 13, 268-284.
18. Auza-Santiváñez JC, Díaz JAC, Cruz OAV, Robles-Nina SM, Escalante CS, Huanca BA. Bibliometric Analysis of the Worldwide Scholarly Output on Artificial Intelligence in Scopus. *Gamification and Augmented Reality 2023*;1:11-11. <https://doi.org/10.56294/gr202311>.
19. Castillo JIR. Aumented reality im surgery: improving precision and reducing ridk. *Gamification and Augmented Reality 2023*;1:15-15. <https://doi.org/10.56294/gr202315>.
20. Zhao, Q.; Xiong, C.; Yu, C.; Zhang, C.; Zhao, X. Anew energy-aware task scheduling method for data-intensive applications in the cloud. *J. Netw. Comput. Appl.* 2016, 59, 14-27.
21. Fall, D.; Blanc, G.; Okuda, T.; Kadobayashi, Y.; Yamaguchi, S. Toward quantified risk-adaptive access control for multi-tenant cloud computing. In *Proceedings of the 6th Joint Workshop on Information Security*, Tokyo, Japan, 8-10 November 2011; pp. 1-14.
22. Yu, E.; Cysneiros, L. Designing for privacy and other competing requirements. In *Proceedings of the 2nd Symposium on Requirements Engineering for Information Security (SREIS'02)*, Raleigh, NC, USA, 16-18 October 2002; pp. 15-16.
23. Kobsa, A.; Schreck, J. Privacy through pseudonymity in user-adaptive systems. *ACM Trans. Internet Technol.* 2003, 3, 149-183.
24. Sgaglione, L.; Coppolino, L.; D'Antonio, S.; Mazzeo, G.; Romano, L.; Cotroneo, D.; Scognamiglio, A. Privacy Preserving Intrusion Detection Via Homomorphic Encryption. In *Proceedings of the 2019 IEEE 28th*

International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Napoli, Italy, 12-14 June 2019; pp. 321-326.

FINANCING

The authors did not receive financing for the development of this research".

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest".

AUTHORSHIP CONTRIBUTION

Conceptualization: K. Prathap Kumar and K. Rohini.

Data curation: K. Prathap Kumar and K. Rohini.

Formal analysis: K. Prathap Kumar and K. Rohini.

Acquisition of funds: K. Prathap Kumar and K. Rohini.

Research: K. Prathap Kumar and K. Rohini.

Methodology: K. Prathap Kumar and K. Rohini.

Project management: K. Prathap Kumar and K. Rohini.

Resources: K. Prathap Kumar and K. Rohini.

Software: K. Prathap Kumar and K. Rohini.

Supervision: K. Prathap Kumar and K. Rohini.

Validation: K. Prathap Kumar and K. Rohini.

Display: K. Prathap Kumar and K. Rohini.

Drafting - original draft: K. Prathap Kumar and K. Rohini.

Writing - proofreading and editing: K. Prathap Kumar and K. Rohini.