IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

A Cloud Secure Storage Mechanism Based on Load Balancing

Asst. Prof. Jayanti Parashar[1], Ritika Ledange[2], Prachi Matte[3], Damini Dehare[4]
Assistant Prof, Computer Science and Engineering[1]
Students, Computer Science and Engineering[2-4]
Shri Sai College of Engineering, Chandrapur, India

Abstract: *Imagine one day you wake up and your application is suddenly flooded with unexpected traffic. Sounds exciting, right? But the real question is—is your application prepared to handle this surge? Can it maintain performance and availability under pressure?*
To tackle such challenges, we propose using an Elastic Load Balancer (ELB)—a powerful service offered by Amazon Web Services (AWS). ELB automatically distributes incoming application traffic across multiple EC2 instances, which may be spread across various availability zones. This helps in enhancing both the speed and responsiveness of web-based applications.
Modern load balancers, like ELB, typically use the Round Robin algorithm to efficiently balance traffic. ELB plays a critical role in real-world scenarios such as high-traffic websites, e-commerce platforms, online games, media streaming services, and mobile applications.
By integrating ELB into your system architecture, you ensure improved scalability, availability, and fault tolerance, making your applications more resilient and performance-ready.

Keywords: Elastic Load Balancer (ELB), Amazon Web Services (AWS), EC2 (Elastic Compute Cloud), Round Robin Algorithm, Application Scalability

I. INTRODUCTION

Amazon Web Services (AWS) is a cloud computing platform offered by Amazon that delivers on-demand services and APIs to individuals, organizations, and government entities. These services operate on a pay-as-you-go, metered model, allowing users to access resources as needed without upfront investments.Often, AWS is used alongside autoscaling, which automatically adjusts computing power based on real-time application demand—scaling up during traffic spikes and scaling down during idle periods to optimize performance and reduce costs.AWS offers a wide range of cloud-based services including computing, storage, networking, middleware, and various software tools—all delivered through its global infrastructure of server farms. This enables users to focus on development and innovation while eliminating the burden of managing, scaling, and maintaining physical hardware and operating systems.

Figure 1.1 Amazon web services

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

- **Reliability & Availability:** AWS ensures high availability and redundancy through its globally distributed data centers. This results in minimal downtime and enhanced application reliability, even during peak traffic or system failures.
- **Calability**: Resources can be scaled up or down automatically based on demand.
- **Cost Efficiency:** Pay-as-you-go model helps avoid large upfront costs—pay only for what you use.
- **Global Reach:** Data centers across the globe ensure low latency and faster access for users worldwide.
- **Wide Range of Services:** Offers compute, storage, databases, machine learning, analytics, networking, and more.
- **Innovation & Speed:** Supports rapid development, testing, and deployment of applications.
- **Managed Infrastructure:** AWS handles server maintenance, updates, and security patches.

II. LITERATURE REVIEW

Several researchers have proposed different distributed load balancing algorithms to improve performance, resource utilization, and efficiency in cloud environments. The following is a comparative summary of some key contributions:

Lua et al. [1] – Join-Idle-Queue Algorithm

This distributed load balancing scheme uses dispatchers to maintain queues of idle processors. Incoming jobs are assigned to these idle queues, which helps offload overloaded nodes and reduce response time.

Chen et al. [2] – User-Priority Guided Min-Min Scheduling (1995)

This algorithm calculates the minimum execution time for all tasks and then selects the maximum of those values for assignment. Designed for static environments, it was simulated using CloudSim.

Liu et al. [3] – Lock-Free Multiprocessing Load Balancing (2006)

This scheme aims to minimize the use of shared memory, thereby improving performance in multi-core systems. It offers better concurrency without traditional locking mechanisms.

Nitish C. et al. [4] – HEFT-Based Workflow Scheduling

This algorithm focuses on cost optimization under deadlines in hybrid cloud environments. The work was simulated on workflows and emphasizes minimizing the cost while meeting task deadlines.

Randles et al. [5] – Comparative Classification

They classified distributed load balancing approaches in large-scale cloud systems into three categories:

- Nature-Inspired Algorithms
- Random Sampling of System Domain
- Restructured Systems for Optimized Job Assignment

Chunling C. et al. [6] – Energy-Aware Scheduling Using Vacation Queuing Theory

Proposed for dynamic cloud environments, this model schedules tasks using vacation queuing with exhaustive service. The algorithm analyzes energy consumption based on busy periods and times, and was simulated using MATLAB.

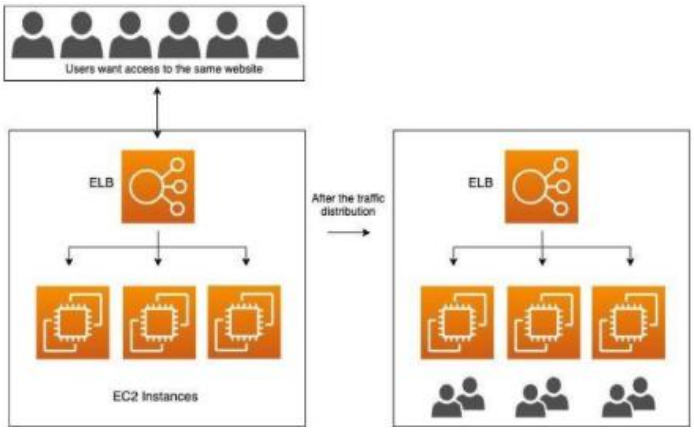

Figure 2.1 Working of ELB

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

In AWS, the **Elastic Load Balancer (ELB)** service—now extended as **Application Load Balancer (ALB)** and **Network Load Balancer (NLB)**—supports a **flow-based routing approach**. Instead of traditional round-robin, this method routes incoming traffic based on parameters such as **source IP**, **destination IP**, and **TCP/UDP ports**.

Although ELB doesn't strictly follow the round-robin method, **round-robin-like behavior** can still be achieved by configuring the **target group** to use routing algorithms such as:

Least Outstanding Requests

Least Connections

These algorithms help balance the traffic by routing requests to targets with the **fewest active connections or pending requests**, promoting fair and efficient load distribution.

When using the **AWS Management Console**, if you configure the ALB to use **Round Robin**, it will distribute incoming requests evenly across healthy targets, leading to improved performance and better resource utilization.

III. PROPOSED SYSTEM

In AWS, the **Elastic Load Balancer (ELB)**—now available as **Application Load Balancer (ALB)** and **Network Load Balancer (NLB)**—typically uses a **flow-based routing mechanism** rather than a traditional round-robin algorithm. This flow-based method routes incoming requests based on parameters such as the **source IP address**, **destination IP**, and **TCP/UDP ports**, ensuring appropriate request handling and session consistency.

While ELB does not implement a strict round-robin technique, **similar load distribution** can be achieved by configuring the target group with algorithms like:

Least Outstanding Requests

Least Connections

These algorithms help ensure balanced traffic by directing new requests to the **target instance with the fewest active requests or connections**.

Through the **AWS Management Console**, users can configure the load balancer to follow this strategy, effectively creating a **round-robin-like behavior**, which results in a **fair and even distribution of incoming requests** among healthy instances.

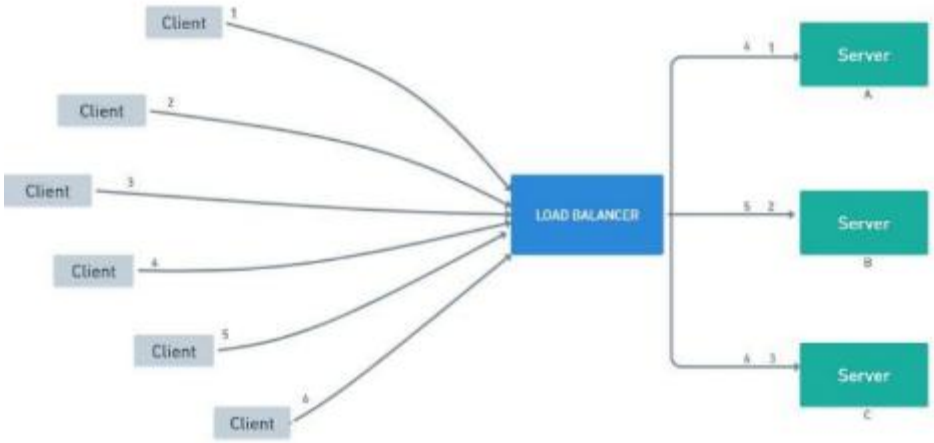

Figure 3.1 Nginx Load Balancer

Nginx is an open-source, high-performance **web server**, **reverse proxy**, and **load balancer**. It efficiently handles HTTP/HTTPS traffic and is known for its **scalability** and **low resource usage**.

As a **reverse proxy**, Nginx forwards client requests to backend servers, helping distribute traffic and improve response times. It supports **load balancing** by evenly distributing requests across multiple servers, which enhances **availability** and **performance** of web applications.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

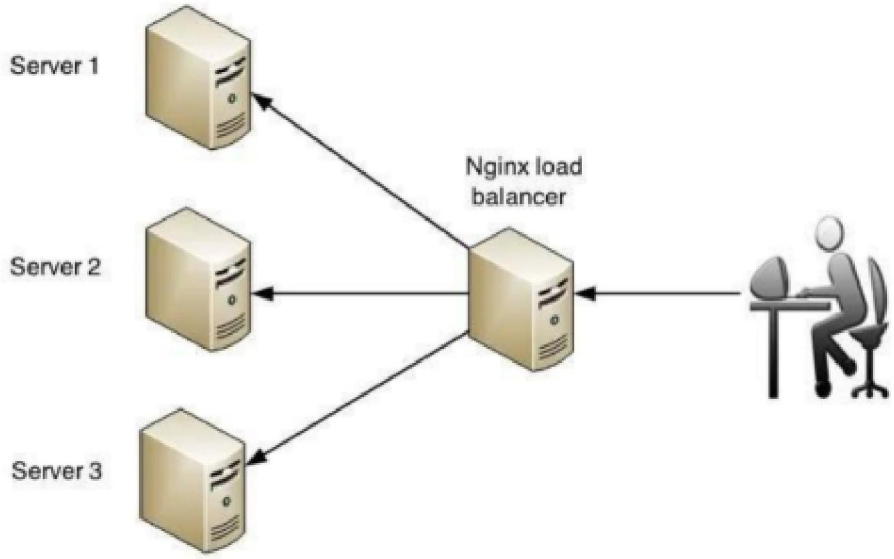

Figure 3.2 Round Robin

In documentation, the proposed approach refers to a structured strategy aimed at solving a specific problem or achieving a defined goal. It typically outlines the steps, methods, tools, and considerations required for successful implementation.

Within the context of AWS, a proposed approach often includes:

Use of relevant AWS services

Best practices and architectural guidelines

Migration strategies and design patterns to support cloud adoption and optimization

Such assessments are usually performed in phases, often aligned with migration waves during the project lifecycle. This phased approach helps manage complexity and ensures a smooth transition to the cloud environment.

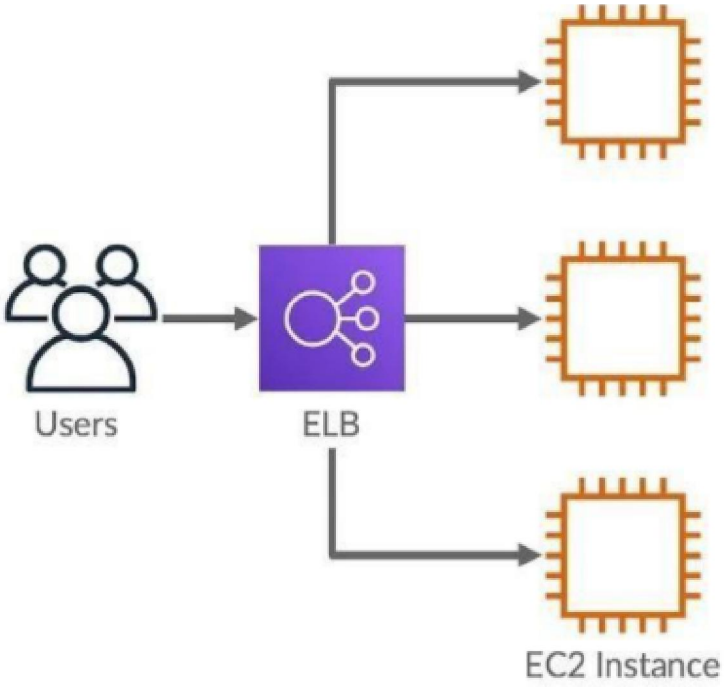

Figure 3.3 Traffic Distribution To Multiple Instance

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

IV. RESULTS

Output on Home Screen:

League 1 will NOT be changed
League 2 will NOT be changed
League 3 will NOT be changed
League 4 will NOT be changed
League 5 will NOT be changed
League 6 will NOT be changed
League 7 will NOT be changed
League 8 will NOT be changed
League 9 will NOT be changed
League 10 will NOT be changed
League 11 will NOT be changed
League 12 will NOT be changed
League 13 will be changed number of VMs:5
League 14 will NOT be changed
League 15 will NOT be changed
League 16 will NOT be changed
League 17 will NOT be changed
League 18 will NOT be changed
League 19 will NOT be changed
Mean fitness for this season: 6032.0977
LCA Threshold for this season: 2121.3716
Season 8
League 0 will be changed number of VMs:0
League 1 will be changed number of VMs:2
League 2 will be changed number of VMs:6
League 3 will be changed number of VMs:9
League 4 will be changed number of VMs:2
League 5 will be changed number of VMs:7
League 6 will be changed number of VMs:5
League 7 will be changed number of VMs:2
League 8 will be changed number of VMs:8
League 9 will be changed number of VMs:4
League 10 will be changed number of VMs:0
League 11 will be changed number of VMs:1
League 12 will be changed number of VMs:8
League 13 will be changed number of VMs:2
League 14 will be changed number of VMs:4
League 15 will be changed number of VMs:9
League 16 will be changed number of VMs:0
League 17 will be changed number of VMs:3
League 18 will be changed number of VMs:3
League 19 will be changed number of VMs:0

DOI: 10.48175/IJARSCT-28211

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

Mean fitness for this season: 13653.094
LCA Threshold for this season: 4801.528
Season 9
League 0 will NOT be changed
League 1 will be changed number of VMs:0
League 2 will be changed number of VMs:3
League 3 will NOT be changed
League 4 will NOT be changed
League 5 will be changed number of VMs:8
League 6 will be changed number of VMs:5
League 7 will be changed number of VMs:3
League 8 will NOT be changed
League 9 will be changed number of VMs:0
League 10 will be changed number of VMs:1
League 11 will be changed number of VMs:8
League 12 will be changed number of VMs:8
League 13 will NOT be changed
League 14 will be changed number of VMs:1
League 15 will NOT be changed
League 16 will NOT be changed
League 17 will be changed number of VMs:0
League 18 will be changed number of VMs:0
League 19 will be changed number of VMs:5
Mean fitness for this season:5998.4424
LCA Threshold for this season:2109.5356
Season 10
League 0 will be changed number of VMs:9
League 1 will be changed number of VMs:2
League 2 will be changed number of VMs:4
League 3 will be changed number of VMs:6
League 4 will be changed number of VMs:4
League 5 will be changed number of VMs:8
League 6 will be changed number of VMs:1
League 7 will be changed number of VMs:6
League 8 will be changed number of VMs:3
League 9 will be changed number of VMs:1
League 10 will be changed number of VMs:3
League 11 will be changed number of VMs:6
League 12 will be changed number of VMs:3
League 13 will be changed number of VMs:1
League 14 will be changed number of VMs:7
League 15 will be changed number of VMs:5
League 16 will be changed number of VMs:9
League 17 will be changed number of VMs:3
League 18 will be changed number of VMs:4
League 19 will be changed number of VMs:5
Mean fitness for this season:143650.95
LCA Threshold for this season:50519.25

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

```
Best solution found at index:9
Initialising...
Starting CloudSim version 3.0
Datacenter_0 is starting...
Broker is starting...
Entities started.
0.0: Broker: Cloud Resource List received with 1 resource(s)
0.0: Broker: Trying to Create VM #5 in Datacenter_0
0.0: Broker: Trying to Create VM #4 in Datacenter_0
0.0: Broker: Trying to Create VM #1 in Datacenter_0
0.0: Broker: Trying to Create VM #3 in Datacenter_0
Task 58, executing on Vm:2
Task 72, executing on Vm:0
Task 77, executing on Vm:1
Task 79, executing on Vm:3
Task 82, executing on Vm:2
Task 90, executing on Vm:2
Task 94, executing on Vm:2
```

V. CONCLUSION

This project successfully demonstrated the implementation of Elastic Load Balancing (ELB) within the Amazon Web Services (AWS) ecosystem, utilizing the round-robin algorithm implemented in the C programming language. By combining cloud infrastructure with a fundamental load balancing approach, the system achieved effective distribution of incoming network traffic across multiple backend servers.

Throughout the course of the project, significant insights were gained in the domains of cloud computing, distributed systems, and network management. The ELB was configured to distribute requests evenly among a pool of EC2 instances, thereby enhancing system availability, performance, and resource efficiency.

The use of the round-robin algorithm provided a clear and efficient method for traffic routing in environments where backend servers are homogeneous and capable of processing requests independently. This approach contributed to fault tolerance and ensured consistent load distribution, supporting the system's scalability and stability.

However, while round-robin offers simplicity and effectiveness, the selection of a load balancing algorithm should always be aligned with the specific performance needs, workload characteristics, and resilience goals of the application.

This project establishes a strong foundation for further research and enhancement. Future work could focus on implementing more adaptive load balancing algorithms, integrating real-time monitoring and analytics, and extending the architecture by incorporating other AWS services such as Auto Scaling, CloudWatch, or Elastic Container Service (ECS). As cloud technologies continue to evolve rapidly, embracing continuous development and innovation is essential. This project represents a meaningful step toward building robust, scalable, and intelligent cloud-based systems.

VI. ACKNOWLEDGMENT

I would like to express my sincere gratitude to the faculty of the department for their guidance and support throughout the project. I also thank my peers for their cooperation and valuable suggestions. Lastly, I am grateful to my family and friends for their continuous encouragement and motivation.

REFERENCES

[1]. Lowlesh Yadav and Asha Ambhaikar, "IOHT based Tele-Healthcare Support System for Feasibility and perfor-mance analysis," Journal of Electrical Systems, vol. 20, no. 3s, pp. 844–850, Apr. 2024, doi: 10.52783/jes.1382.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

[2]. L. Yadav and A. Ambhaikar, "Feasibility and Deployment Challenges of Data Analysis in Tele-Healthcare System," 2023 International Conference on Artificial Intelligence for Innovations in Healthcare Industries (ICAIIHI), Raipur, India, 2023, pp. 1-5, doi: 10.1109/ICAIIHI57871.2023.10489389.

[3]. L. Yadav and A. Ambhaikar, "Approach Towards Development of Portable Multi-Model Tele-Healthcare System," 2023 International Conference on Artificial Intelligence for Innovations in Healthcare Industries (ICAIIHI), Raipur, India, 2023, pp. 1-6, doi: 10.1109/ICAIIHI57871.2023.10489468.

[4]. Lowlesh Yadav and Asha Ambhaikar, Exploring Portable Multi-Modal Telehealth Solutions: A Development Approach. International Journal on Recent and Innovation Trends in Computing and Communication (IJRITCC), vol. 11, no. 10, pp. 873–879, Mar. 2024.11(10), 873–879.

[5]. Lowlesh Yadav, Predictive Acknowledgement using TRE System to reduce cost and Bandwidth, March 2019. International Journal of Research in Electronics and Computer Engineering (IJRECE), VOL. 7 ISSUE 1 (JANUARY- MARCH 2019) ISSN: 2393-9028 (PRINT) | ISSN: 2348-2281 (ONLINE).

[6]. Sandhya.S. Bachar, Neehal.B. Jiwane, Ashish.B.Deharkar "Sentiment analysis of social media" DOI: 10.17148/IJARCCE.2022.111234 International Journal of Advanced Research in Computer and Communication Engineering ISO 3297:2007 Certified□□Impact Factor 7.918□□Vol. 11, Issue 12, December 2022.

[7]. Akshay A. Zade,Lowlesh N. Yadav, Neehal B. Jiwane. "A Review on Voice Browser" DOI: 10.17148/IJARCCE.2022.111238 International Journal of Advanced Research in Computer and Communication Engineering ISO 3297:2007 Certified□□Impact Factor 7.918□□Vol. 11, Issue 12, December 2022.

[8]. Omkar K. Khadke, Lowlesh N. Yadav, Neehal B. Jiwane. "Review On Challenges and Issues in Data Mining" DOI: 10.17148/IJARCCE.2022.111149 International Journal of Advanced Research in Computer and Communication Engineering ISO 3297:2007 Certified□□Impact Factor 7.918□□Vol. 11, Issue 11, November 2022.

[9]. Miss. Vaishali Vaidya, Mr. Vijay Rakhade, Mr. Neehal B. Jiwane. "VOICE CONTROLLED ROBOTIC CAR BY USING ARDUINO KIT" DOI: 10.17148/IJARCCE.2022.111232 International Journal of Advanced Research in Computer and Communication Engineering ISO 3297:2007 Certified□□Impact Factor 7.918□□Vol. 11, Issue 12, December 2022.

[10]. Atharv Arun Yenurkar, Asst Prof. Neehal B. Jiwane, Asst. Prof. Ashish B. Deharkar. "Effective Validation for Pervasive Computing and Mobile Computing Using MAC Algorithm". International Journal of Research Publication and Reviews, Vol 3, no 12, pp 470-473 December 2022.

[11]. Pooja Raju Katore, Asst. Prof. Ashish B. Deharkar, Asst. Prof. Neehal B. Jiwane. "Cloud Computing and Cloud Computing Technologies: A-Review". International Journal of Research Publication and Reviews, Vol 3, no 12, pp 538-540 December 2022

[12]. Combining Vedic & Traditional Mathematic Practices for Enhancing Computational Speed in Day-To-Day Scenarios, Speed in Day-To-Day Scenarios, Conference: Industrial Engineering Journal ISSN: 0970-2555 Website: www.ivyscientific.org, At: Industrial Engineering Journal ISSN: 0970-2555, Website: www.ivyscientific.org. **(UGC JOURNAL)**

[13]. python.net, December 2022, DOI:10.17148/IJARCCE.2022.111237, Conference: International Journal of Advanced Research in Computer and Communication Engineering.

[14]. A Survey for Credit Card Fraud Detection Using Machine Learning, December 2022, DOI:10.17148/IJARCCE.2022.111221, Conference: International Journal of Advanced Research in Computer and Communication Engineering

[15]. GRB 210217A: a short or a long GRB? December2022, DOI: 10.1007/s12036-022-09822, Journal of Astrophysics and Astronomy, Published by Online ISSN: 0973-7758, Print ISSN: 0250-6335.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, June 2025
Impact Factor: 7.67

[16]. Pronunciation Problems of English Language Learners in India, November 2022, DOI: 10.17148/IJARCCE.2022.111151, Conference: International Journal of Advanced Research in Computer and Communication Engineering.

[17]. Photometric and spectroscopic analysis of the Type II SN 2020jfo with a short plateau, November 2022, DOI:10.48550/arXiv.2211.02823, License CC BY 4.0.

[18]. Artificial Neural Network, May 2022, DOI: 10.17148/IJARCCE.2022.115196, Conference: International Journal of Advanced Research in Computer and Communication Engineering.

[19]. Cloud Storage Security Based on Dynamic key Generation Technique, May 2022 DOI: 10.17148/IJARCCE.2022.115189, Conference: International Journal of Advanced Research in Computer and Communication Engineering

[20]. Research on Techniques for Resolving Big Data Issues,May 2022,DOI: 10.17148/IJARCCE.2022.115192, Conference: International Journal of Advanced Research in Computer and Communication Engineering

[21]. STUDY on INTERNET of THINGS BASED APPLICATION, May2022, DOI: 10.17148/IJARCCE.2022.115179, Conference: International Journal of Advanced Research in Computer and Communication Engineering.

[22]. Research on Data Mining, May 2022, DOI: 10.17148/IJARCCE.2022.115176, Conference: International Journal of Advanced Research in Computer and Communication Engineering.

[23]. Security Solution of The Atm and Banking System, May 2022, DOI: 10.17148/IJARCCE.2022.115165, Conference: International Journal of Advanced Research in Computer and Communication Engineering.

[24]. Study on Positive and Negative Effects of Social Media on Society, May 2022, DOI: 10.17148/IJARCCE.2022.115161, Conference: International Journal of Advanced Research in Computer and Communication Engineering.

[25]. Research on Association Rule Mining Algorithms, May 2022, DOI: 10.17148/IJARCCE.2022.115152, Conference: International Journal of Advanced Research in Computer and Communication Engineering.

[26]. Block chain Technology, May 2022, DOI: 10.17148/IJARCCE.2022.115154 Conference: International Journal of Advanced Research in Computer and Communication Engineering.

[27]. INTERNET of THINGS RESEARCH CHALLANGES and FUTURE SCOPE, May 2022 DOI: 10.17148/IJARCCE.2022.115150, Conference: International Journal of Advanced Research in Computer and Communication Engineering.

[28]. Data Collection and Analysis in a Smart Home Automation System, May 2022 DOI: 10.17148/IJARCCE.2022.115148, Conference: International Journal of Advanced Research in Computer and Communication Engineering.

[29]. Using Encryption Algorithms in Cloud Computing for Data Security and Privacy, May 2022, DOI:10.17148/IJARCCE.2022.115149, Conference: International Journal of Advanced Research in Computer and Communication Engineering.

[30]. An Efficient Way to Detect the Duplicate Data in Cloud by using TRE Mechanism,May 2022, DOI:10.17148/IJARCCE.2022.115139, Conference: International Journal of Advanced Research in Computer and Communication Engineering, Volume: 11.