IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

"Cloud Based Solution Architecture For College Placement Cell Web Based Application"

*Mr. Sunil Yadav[1], Dinesh Temgire[2], Vivek Raut [3], Vedant Thorat [4], Anjali Babar [5]
[1]Assistant Professor, [2,3,4,5] BE Student

Department of Computer Engineering
Dr. D.Y. Patil College of Engineering & Innovation Varale, Talegaon, Pune, Maharashtra, India.

Abstract: The paper shares some really helpful ideas about general cloud security. These can also fit into the bigger picture of cloud security for software. Why? Well, it gives us ways to understand threats better, plus it helps design good security measures and manage cloud safety. It covers important stuff like looking at attack models, different kinds of security solutions, and even basics about three-tier cloud architecture. There's also info on detection of intrusions and how to manage security platforms. This paper talks about a new platform that helps make digital tasks better in factories. Docker lets apps run in easy-to-manage spaces. On the other hand, Cloud Storage is all about keeping files safe and sound. With this plat form, folks can share processes as simple Docker images. Isn't that neat? Plus, there's a smart way to match jobs with the quickest nodes how cool is that? The whole setup is used in a real-life example of making eye lenses, which shows how useful it can be. Additionally the entire Cloud services that we will see and use will be cost efficient and secure which will have some layers of security and the solution architecture will only improve it by utilizing the resources effectively.

Keywords: Cloud Service Provider, Cloud Computing, Solution Architecture, Cloud Security, DevOps, Digital cloud deployment

I. INTRODUCTION

A college placement cell plays a pivotal role in connecting students with potential employers. A web-based application can streamline this process, but to ensure scalability, reliability, and cost effectiveness, a cloud-based architecture is increasingly favored. Cloud Computing (CC), and Edge Computing (EC), have emerged as a pivotal technology in the realm of Intelligent Connected web applications. While cloud offers numerous benefits, it also inherits security vulnerabilities from its constituent technologies [1]. Cloud communication is 1 characterized by its dynamic nature, di verse communication modes, and reliance on wireless technologies. Ensuring the security and privacy of data transmit ted within cloud is paramount. This includes protecting against unauthorized access, data tampering, and denial-of service attacks [2]. Cloud faces various attack threats, including routing attacks, identity at tacks, data attacks, DoS attacks, and malicious attacks. To mitigate these threats, a combination of techniques can be employed. These include cryptographic techniques for data encryption and authentication, intrusion detection systems to identify malicious activities, access control mechanisms to restrict unauthorized access, and anomaly detection algorithms to detect deviations from normal behavior. A secure and reliable cloud security architecture is proposed, comprising three types of clouds: vehicular, edge, and central clouds. This architecture integrates a Vehicle Intrusion Detection and Prevention System and a Vehicular Cloud Security Management Platform to address security concerns. The cloud ensures local security and privacy, the edge cloud provides proximity-based services, and the central cloud manages overall security policies and data [3].

2. Literature Review

Amazon RDS Optimized Reads achieve faster query processing by placing temporary tables generated by MySQL on NVM e based SSD block storage that is physically connected to the host server. Queries that use temporary tables, such as those involving sorts, hash aggregations, high load joins, and Common Table Expressions (CTEs) can execute up to 50Amazon RDS Optimized Writes deliver an improvement of up to 2x in write transaction throughput at no extra charge, and with the same level of provisioned IOPS. Optimized Writes are a great fit for write heavy workloads that generate lots of con current transactions. This includes digital payments, financial trading platforms, and online games [1].

How can you improve network latency issues? You can re duce network latency by optimizing both your network and your application code. The following are a few suggestions. Upgrade network infrastructure you can upgrade network devices by using the latest hardware, software, and network configuration options on the market. Regular network maintenance improves packet processing time and helps to reduce network latency. Monitor network performance Network monitoring and management tools can perform functions such as mock API testing and end-user experience analysis. You can use them to check network latency in real time and troubleshoot network latency issues. Group network endpoints Subnetting is the method of grouping network end points that frequently communicate with each other. A subnet acts as a network inside a network to minimize unnecessary router hops and improve network latency. Use traffic-shaping methods you can improve network latency by prioritizing data packets based on type. For example, you can make your network route high-priority applications like VoIP calls and data center traffic first while delaying other types of traffic. This improves the acceptable latency for critical business processes on an otherwise high-latency network [4].

Docker, which helps with containerization, along with cloud storage, are super important in making everything work better in this new setup. Docker's Role in Cloud Manufacturing Let's talk about Docker. Its containers pack up applications with everything they need into one neat unit. This it easy to run them anywhere. You can get manufacturing up and running quick! Plus, these containers keep things separate. That means fewer problems between apps and better security. This is really important since many people might use the same infrastructure at once. Docker is also great for something called micro services! This means instead of having one big system, you can break it down into smaller parts that work on their own. It's easier to manage this way! And when it comes to DevOps– that's the teamwork between development operations– Docker helps speed up how fast you create and test your applications [2].

Cloud Storage for Manufacturing Data Now onto cloud storage. It offers a nearly endless amount of space and keeps your data safe. So no worries there! You're usually only paying for what you use, which is a big plus compared to buying hardware upfront. Plus, cloud storage plays nice with data analytics tools! That means you can get useful information about how your manufacturing process is doing. Collabo ration also becomes easier when teams can share data quickly from a common storage [5].

Optimization of the cloud can be done in various ways like there is Relational database optimization and these kinds of optimization can be done using concept optimized reds-writes These principles, especially those related to security, data integrity, and resource optimization, are transferable to general cloud computing environments beyond networks. The same cloud architecture concepts, encryption protocols, and security measures can be applied in general software-based cloud services. These principles, especially those related to security, data integrity, and resource optimization, are transferable to general cloud computing environments beyond vehicular networks. The same cloud architecture concepts, encryption protocols, and security measures can be applied in general software-based cloud services [3].

• Routing Attacks: Malicious nodes manipulate routing protocols to disrupt communication or intercept data.

• Data Attacks: Targeting the integrity, confidentiality, and availability of data transmitted.

• DoS Attacks: Overwhelming cloud resources with excessive traffic to disrupt normal operations.

• Identity Attacks: Compromising the authentication and authorization of entities within.

• Malicious At tacks: Employing various techniques to gain unauthorized access, steal data, or control.[3]

Specific examples of attacks include: Black Hole, Gray Hole, and Wormhole at tacks (routing)

• Data interception, tampering, leakage, replay, injection, and eavesdropping (data)

• DDOS, jamming, and flooding attacks (DoS)

• Authentication attacks, identity masquerading, impersonation, Sybil attacks, session hi jacking, timing attacks, and MITM at tacks (identity)

• Malware attacks, spamming attacks, botnet attacks, and location spoofing attacks (malicious)

Addressing these threats requires a comprehensive security approach, including: Encryption,

Authentication, Intrusion detection, Se cure boot, Patch management, Network segmentation, Access control, Security awareness, Regulatory compliance, Continuous monitoring [3].

3. Motivation and Objective

3.1 Motivation

A cloud-based system can really boost how a college placement cell works. It brings many benefits for student's employers alike. First, by putting all student profiles, job listings, applications in the cloud, the placement cell can handle in formation really well. The best part? It can grow easily as more students and employers join in. So, there's no worry about running out of space! With a portal, student's employers can access everything from anywhere—any time they want. It's super easy to set up profiles, apply for jobs, and keep in touch because the interface is user friendly. Also, built-in tools like messaging and chat help students talk directly with employers. This makes ap plying for jobs smoother. Plus, with shared calendars document sharing, everyone—students, employers, staff—can work together better. Using a cloud-based solution gives great insights too! You get to see student profiles, job trends, placement successes through analytics. This way, the placement cell can make smart choices on how to improve services. Not to mention, cloud solutions can save money compared to old-school systems. There's no need to spend a lot on hardware software upfront! Plus, pricing can be flexible. Cloud providers often have strong security measures to keep data safe from bad access and come with plans for disaster recovery if things go wrong. By tapping into a cloud-based system, a college placement cell can build an easier, more open way to connect students with jobs.

Key motivations behind this project are:

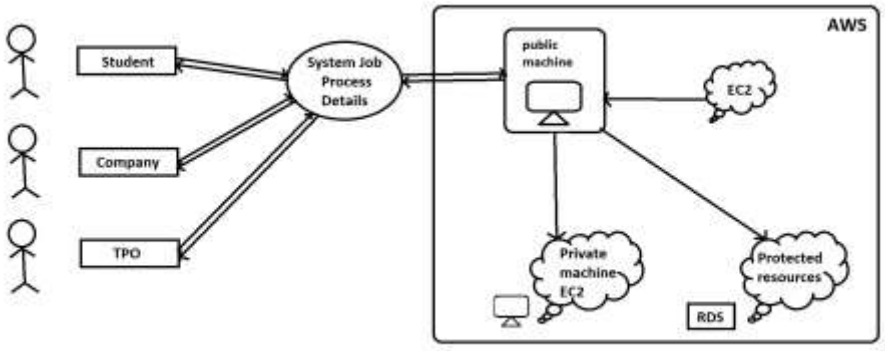
1. Combating Misinformation
2. Improve Efficiency and Scalability
3. Optimizing the cost and the Resources
4. Improve Reliability and Availability
5. Reliable system design and maintenance

3.2 Objective

1. Secure and Efficient Message Distribution.
2. Privacy and Security from various cyberattacks.
3. Scalability and Performance disaster recovery and backups with proper maintenance.
4. Trustworthy and reliable system for college to carry out proper and correct placement activities

4 Methodology and Architecture

4.1 System Architecture



4.1.1 System architecture

4.2 Security Practices

Least Privilege:

The least privilege principle states that users should have only the minimum set of permissions they need to carry out their tasks. This approach limits the risk of harm that could stem from unauthorized access or misuse of privileges. By restricting access to just vital functions, the likelihood of security incidents can be significantly diminished.

Separation of Duties:

Separation of duties entails distributing essential tasks among various individuals to ensure that no single person has full control over a process. This practice reduces the risk of mistakes, fraud, and unauthorized access. For example, one individual may handle the creation of user accounts, while another manages the assignment of permissions.

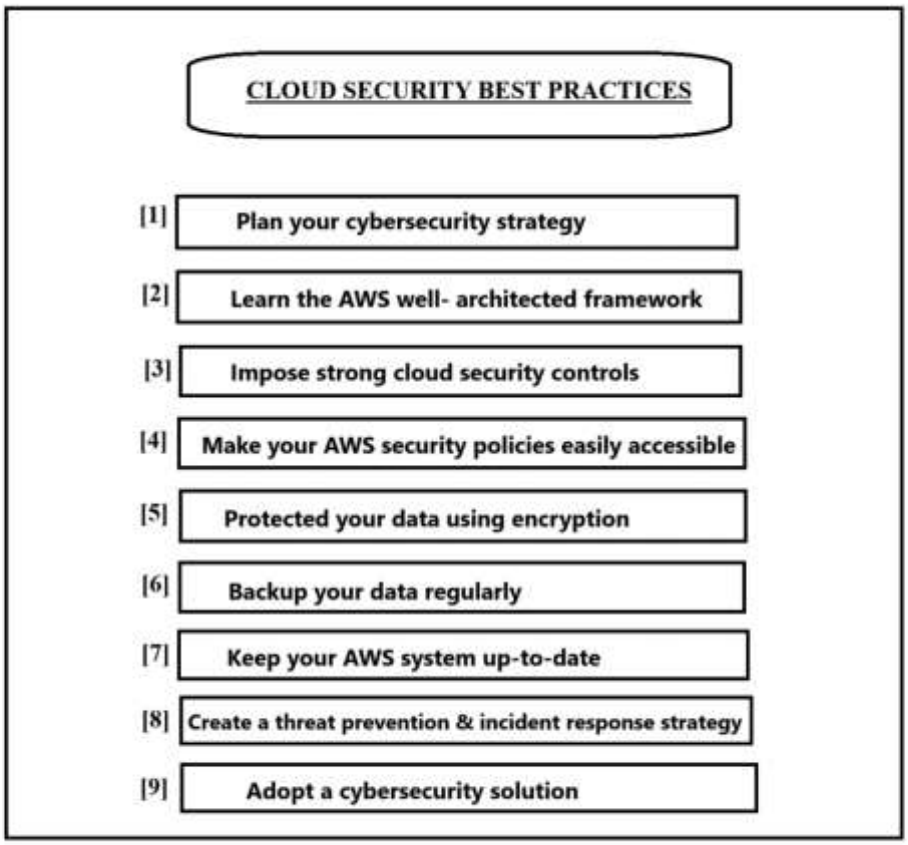
Defense in Depth:

Defense in depth, or layered security, refers to the implementation of several security measures to safeguard your systems against various threats. This strategy acknowledges that no solitary security approach is in fallible, and combining different methods can strengthen overall protection. Examples include firewalls, intrusion detection systems, data encryption, and access management.

Continuous Monitoring:

Continuous monitoring is the ongoing practice of watching your systems for any signs of suspicious behavior or security breaches. This may involve scrutinizing network traffic, log files, and system operations. By identifying potential threats at an early stage, you can take preventative measures to address them and avert further complications.

Patch Management: Patch management involves regularly applying software updates and security patches to your systems to fix vulnerabilities and bolster security. Consistent patching is crucial for safeguarding your systems against known threats and ensuring compliance with security regulations.



4.2.1 Best Practices for Cloud Security

Docker is a client-server framework that enables the execution of multiple applications in isolated environments known as containers on a single host. Unlike traditional virtual machines, containers do not require a hypervisor; instead, they utilize the host operating system's kernel. The platform provides comprehensive lifecycle management, allowing for actions like creating, starting, stopping, and destroying containers, as well as features for gathering metrics [2].

Each container operates based on a Docker image, which serves as a template created from a set of instructions contained within a Docker file. These instructions form layers that are added to the image file system. A unique fingerprint for each image can be identified through a SHA-256 hash, which accounts for all of its layers. Docker images are commonly stored in a registry, functioning as a repository where users can push or pull images based on their access rights and maintain multiple versions of a single image identified by different tags [7].

Cloud Storage ranks among the leading cloud technologies available, provided by major companies such as Amazon and Microsoft through their S3 service and Azure Blob Storage platform, respectively. Essentially, it is a data storage service that allows users to save their files in the Cloud [8].

4.3 Resource provisioning in cloud computing

Resource provisioning in cloud computing refers to the distribution of resources and services from a cloud provider to a customer, and it is also known as cloud provisioning. This involves the, deployment, and management of software such as load balancers and database management systems) and hardware resources (like CPUs, storage, and networks) to ensure optimal application performance [8].

To make the most of these resources while adhering to service level agreements (SLAs) and meeting quality of service standards, it is essential to implement either Static or Dynamic Provisioning and Static or Dynamic Resource Allocation, tailored to the specific needs of the application. It is crucial to avoid both over-provisioning and under-provisioning of resources. Additionally, power consumption presents another key challenge that needs to be managed carefully. Strategies should be in place to minimize power use, heat dissipation, and virtual machine placement, thus preventing any unnecessary energy consumption. Ultimately, a cloud user's primary goal is to acquire resources at the lowest feasible cost, whereas a cloud service provider aims to maximize profit through efficient resource distribution [8].

Static Provisioning or Advance Provisioning: Static provisioning is effective for applications that have known and generally constant demands or workloads. In this scenario, the cloud provider offers a fixed amount of resources to the client, who can then use these resources as needed. It is the client's responsibility to ensure that they do not exceed resource limits. This approach works well for applications with steady and predictable requirements. For example, a client may need a database server with a specific amount of CPU, RAM, and storage. When a consumer signs up for services with a provider, the provider prepares everything necessary before the service can commence. The customer may incur either a one-time charge or a recurring monthly fee. Cloud service providers allocate resources in advance to customers. This means that users must decide how much capacity they require in a fixed manner before utilizing these resources. Static provisioning can lead to issues with either over-provisioning or under-provisioning [9].

Dynamic Provisioning or On-demand Provisioning: Dynamic provisioning allows providers to add resources as needed and remove them once they are no longer required. This operates on a pay-per-use basis, meaning clients are billed solely for the resources they actually consume. Each time clients utilize the services allocated to them by the cloud provider, they are charged appropriately. This is also referred to as a pay-as-you-go model. Dynamic provisioning techniques enable the movement of virtual machines (VMs) to different computing nodes within the cloud as demand varies. This method is ideal for applications with unpredictable and fluctuating workloads. For instance, a client might request a web server with an adjustable amount of CPU, memory, and storage. In this case, the client can use the resources as needed and is charged only for

what is actually consumed. The responsibility lies with the client to ensure that resources are not oversubscribed, as this could lead to increased costs [9].

Self-service Provisioning or User Self-provisioning: User self-provisioning, also known as cloud self-service, allows customers to request resources through a web form. They can set up an account and complete payment via credit card. Shortly afterward, the resources become available for the client's use [9].

5. Project Feasibility and Scope

A cloud-based setup for a college placement cell web app is really helpful. It brings many advantages like better efficiency, more, and easier access for everyone. However, you to check if the project doable outline what it cover before jumping n. When we talk about technical feasibility, it means looking at if the cloud platforms are good enough, what kind of equipment you'll need, which technologies to use. You should think about costs, how well it can grow with your needs, security, what features each cloud platform offers. Plus, can it handle the amount of work expected? How well does it fit with systems already in place?

Now, let's consider economic feasibility too. This means checking out all costs related to cloud setup, development, maintenance, keeping things running day-to day. It's key to look at the possible return on investment (ROI) by weighing in benefits like better efficiency, lower costs, and improved results for students. Don't forget about legal and regulatory feasibility! You've got to make sure everything follows rules about data privacy protects the rights of any creative work. This includes keeping in mind laws like GDPR CCPA along with industry standards. When mapping out the project scope, include key features. Such features could be user profiles, job postings, application management tools, a matching algorithm for jobs, ways for users to communicate. You might also think about extra features—like a resume builder or interview scheduling—as optional choices based on what matters most to your organization. It's really important to focus on must-have features during the first development stage. This way, the app can quickly meet the immediate needs of student's employers. Designing the architecture with future growth in mind is smart too! You should also think about how this new setup fits with existing college systems. By taking a close look at feasibility clearly outlining project de tails from the start, the college placement cell can make sure this cloud-based solution works well with your goals resources. This leads to a better experience overall for everyone involved!

1) Cloud Communication
2) Social Communication
3) DevOps
4) Disaster Recovery
5) Backups
6) Data Security
7) Load Balancing and sharing

6. Conclusion

Solution architecture is the process of de signing and planning a comprehensive solution to address a specific business problem or opportunity. It involves identifying the requirements, selecting appropriate technologies and methodologies, and defining the overall structure of the solution. In this we will be constructing a solution for a web based application of college placement cell where all the resources will be designed in such a way that they can handle lode much more of their capacity because of their proper arrangement. All resource such computing resources and database resources will be secured by various method such that no attack can breach our system. Black hole, Gray Hole, and Wormhole at tacks (routing) Data interception, tampering, leakage, and location spoofing attacks (malicious) etc. these are some of the attacks that can cause much impact on any system. Protection of the system is main motive and giving the quality product, of course we cannot forget about the cost. The cloud services are charged on pay-as you go based model where we only pay for what we use and for how much time we use.

REFERENCES

[1] A. D. a. G. M. H. Abdullah Alelyani, "Optimizing cloud performance: A microservice scheduling strategy for enhanced fault-tolerance, reduced network traffic, and lower latency.," IEEE Access, 2024.

[2] A. M. M. a. M. P. F. Gaetano Volpe, "An architecture combining blockchain, docker and cloud storage for improving digital processes in cloud manufacturing," IEEE Access, 2022.

[3] M. S. A. G. a. R. B. Md Whaiduzzaman, " A survey on vehicular cloud computing. Journal of Network and Computer applications," IEEE Access, 2023.

[4] Y. G. a. C. Delimitrou, "The architectural implications of cloud microservices.," IEEE Computer Architecture Letters, 2018.

[5] L. Y. Y. W. C. G. L. W. G. X. L. Z. K. Y. a. C. X. M. Xu, "Practice of Alibaba cloud on elastic resource provisioning for large-scale microservices cluster," IEEE Access, 2024.

[6] P. R. Krishnan and P. A. R. Kumar, "Detection and mitigation of smart blackhole and gray hole attacks in VANET using dynamic time warping," Wireless Pers. Commun., vol. 124, 2022.

[7] P. C. a. Z. Z. G. Yu, "Microscaler: Cost-effective scaling for microservice applications in the cloud with an online learning approach," IEEE Access, 2022.

[8] Y. H. S. J. G. S. G. D. T. R. G. a. M. A. S. N. Singh, "'Load balancing and service discovery using Docker swarm for microservice based big data applications," J. Cloud Comput vol 12, 20203.

[9] X. G. T. W. G. B. a. B.-Y. C. X. Wan, "Application deployment using microservice and Docker containers: Framework and optimization," J. Netw. Comput. Appl., vol. 119, 2018.

[10] H. Y. Q. Y. L. G. J. Z. a. M. C. B. Bao, "Resource allocation with edge-cloud collaborative traffic prediction in integrated radio and optical networks," IEEE Access, 2023.

[11] D. K. M. A. A. A. Sunil Kumar Yadav, "A Review on the Identification & Sorting of Fruit using Deep Learning," International Journal of Research and Analytical Reviews, 2023.

[12] Z. Y. W. D. a. M. A. D. Liu, "A survey on secure data analytics in edge computing," IEEE Publisher, 2019.

[13] R. Doshi and V. Kute, "A review paper on security concerns in cloud computing and proposed security models," Trends Inf. Technol., 2020.

[14] C. Y. a. W. W. H. D. Yixing, "'Large-scale user password security authentication algorithm under cloud computing technology," Comput. Simul, 2022.

[15] Y. N.Wu, "Design of cloud security management platform based on cloud computing technology," nf. Secur. Commun. Privacy,, 2020.