# Microsoft Azure Services : Continuous Integration and Deployment with Docker and Kubernetes

**B.Deepthi Reddy,K.Shireesha,D.Sravanthi**

[1]Assistant Professor,[2]Assistant Professor,[3]Assistant Professorr
[1]Department of Information Technology,
[1]Vidya Jyothi Institute of Technology, Hyderabad, India

**Abstract** - Cloud computing applications and platforms are rapidly growing across all industries, serving as the IT infrastructure that drives new digital businesses. These platforms and applications have revolutionized the ways in which businesses function, and have made processes easier. While there are many cloud computing platforms available, two platforms dominate the cloud computing industry. Amazon Web Services (AWS) and Microsoft Azure are the two giants in the world of cloud computing. Azure is a Cloud Computing platform and an online portal that allows to access and manage cloud services and resources provided by Microsoft. These services and resources include storing the data and transforming it, depending on requirements. Azure Pipelines supports continuous integration (CI) and continuous delivery (CD) to continuously test, build, and deploy the code. Container as a Service(CaaS) enables development teams to think at the higher order container level instead of mucking around with lower infrastructure management. This brings development teams better clarity to the end product and allows for more agile development and higher value delivered to the customer.

**IndexTerms** - Azure, Infrastructure, Continuous Integration, Continuous Delivery, Build, Pipeline, Deploy, Container.

## I.INTRODUCTION

Due to COVID-19 crises demand of IT services are increased. Employees' safety and health is the priority of every organization. Information technology is the backbone for all sectors over the globe in the Post COVID period .According to research report of International Data Corporation (IDC) cloud market will raise to $7.4 billion by 2024**.** In fact, more than 77 percent of businesses today have at least some portion of their computing infrastructure in the cloud. These services and resources include storing the data and transforming it, depending on requirements. To get access to these resources and services, all we need to have is an active internet connection and the ability to connect to the Azure portal. Azure Pipelines supports continuous integration (CI) and continuous delivery (CD) to continuously test, build, and deploy the code. Container as a Service(CaaS) enables development teams to think at the higher order container level instead of mucking around with lower infrastructure management. This brings development teams better clarity to the end product and allows for more agile development and higher value delivered to the customer.

## II.SURVEY

Today cloud computing is a promising and highly demanding technology. It is Internet-based technology for sharing data, resources, infrastructure, platform as per demand so called as Pay-As-Go technology. Cloud computing is the visualization technology provide Software as a Service (SaaS), Platform as-a Service and Infrastructure as a service (IaaS). Different cloud providers like Google, Amazon, Microsoft Azure, Oracle, Salesforce provides service to the end users to access applications from thin client or web browser. While using the SaaS service end user not need to worry about operating system, network, server, storage because this is managed by cloud providers [5]. Developers or programs can use PaaS service for application development, configuration, maintenance as well as deployment. Cloud providers support for various frameworks like android, PHP, Java, Python, Ruby, .Net etc. for application development. They provides high computing power, GPU's, CPU's and storage through managed data centers with high scalability, reliability, load balancing, multi-tenant environment support .Configuration of cloud services is based on virtualization requirements. Organizations can subscribe IaaS service for accessing operating system, Virtual Machines, storage, computing power as per their need and requirement without any physical maintenance installation and configuration. Private, Public, Hybrid and Community are the most popular cloud computing deployment models. As per budget, functionality requirement, legal and security compliance organizations can use any model for using cloud services.

**Benefits:**

- Azure supports multiple programming languages, including Java, Node Js, and C#.
- Another benefit of Azure is the number of data centers it has around the world. There are 42 Azure data centers spread around the globe, which is the highest number of data centers for any cloud platform.
- Azure provides more than 200 services, are divided into 18 categories.
- These categories include computing, networking, storage, IoT, migration, mobile, analytics, containers, artificial intelligence, and other machine learning, integration, management tools, developer tools, security, databases, DevOps, media identity, and web services.

**Use of Azure:**

- Application development: You can create any web application in Azure.
- Testing: After developing an application successfully on the platform, you can test it.
- Application hosting: Once the testing is done, Azure can help you host the application.
- Create virtual machines: You can create virtual machines in any configuration you want with the help of Azure.
- Integrate and sync features: Azure lets you integrate and sync virtual devices and directories.
- Collect and store metrics: Azure lets you collect and store metrics, which can help you find what works.

- Virtual hard drives: These are extensions of the virtual machines; they provide a huge amount of data storage.

**Azure Services:**

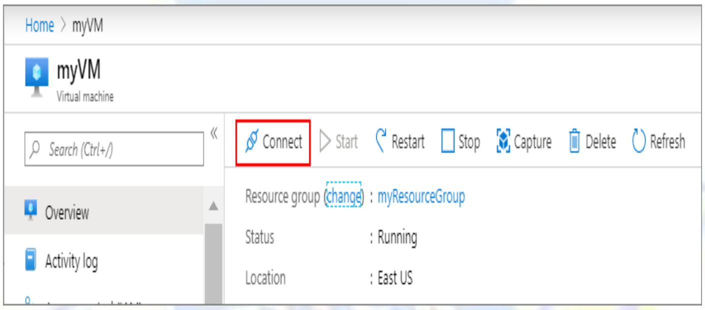
Some most popular Microsoft Azure services, specifically the top 10 azure services and the way we can use them across the entire architecture:

- Azure DevOps
- Azure Blob Storage
- Azure Virtual Machines
- Azure Backup
- Azure Cosmos DB
- Azure Logic Apps
- Azure Active Directory
- API management
- Azure Content Delivery Network
- Azure Site Recovery
- Azure Bots

## III.EXISTING SYSTEM

Azure virtual machines (VMs) can be created through the Azure portal. This method provides a browser-based user interface to create VMs and their associated resources. To see VM in action, download the RDP to the VM and install the IIS web server.

Create a remote desktop connection to the virtual machine.



Resource group (change) : myResourceGroup

Status : Running

Location : East US

**Figure:1 Remote Desktop Connection to VM**

## IV.PROPOSED SYSTEM

- Docker is a container-based platform for building applications. It has quickly become an alternative to Virtual Machines (VMs). Docker containers are lightweight, simple to configure and work consistently in diverse IT environments.
- Kubernetes is a portable and open-source platform for managing containers (emphasis on the "managing").
- Kubenetes and Docker cannot replace each other. Instead, when we combine the two, we are able to boost the efficiency of application development and deployment.

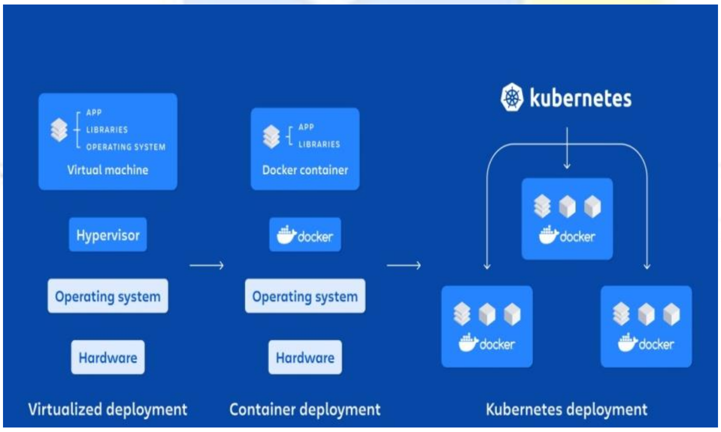


**Figure:2 Azure Services Architecture with Kubernetes**

## V.AZURE DEVOPS

It is one of the first Azure cloud services launched into the market. The **Azure DevOps** services are ideal for smarter planning and better collaboration to ensure faster delivery. Users could find Azure DevOps services as ideal choices for building, testing, and deploying with CI/CD. Based on your business needs, one can choose any DevOps service.

The services are:

- **Azure Repos**

This provides Git repositories or Team Foundation Version Control (TFVC).

- **Azure Pipelines**

This provides build and release services to support continuous integration and delivery of the application.

- **Azure Boards**

Azure Boards delivers a set of tools to support planning and tracking work, code defects, and issues using Kanban and Scrum methods.

- **Azure Test Plans**

This provides tools to test the applications, including both manual testing and continuous testing.

- **Azure Artifacts**

This allows sharing packages from public sources, private sources and integrating package sharing into the pipelines.

## VI.AZURE PIPELINES

Azure Pipelines using Kubernetes CI/CD tools

**Jenkins**

- Developed by Sun Microsystems, Jenkins is the most stable and widely adopted CI/CD platform globally. Jenkins uses a centralized automation server that takes care of all the CI/CD processes.pports continuous integration (CI) and continuous delivery (CD) to continuously test, build, and deploy your code.The latest way to build pipelines is with the YAML pipeline editor.

**Automate tests, builds, and delivery**

| Continuous integration (CI) | Continuous delivery (CD) |
|---|---|
| - Increase code coverage<br>- Build faster by splitting test and build runs<br>- Automatically ensure you don't ship broken code<br>- Run tests continually | - Automatically deploy code to production<br>- Ensure deployment targets have latest code<br>- Use tested code from CI process |

**Table:1 CI/CD Pipelines**

**pipelines using YAML syntax**

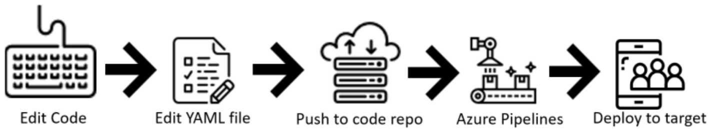


**Figure:3 Pipelines Using YAML syntax**

- The pipeline is versioned with your code. It follows the same branching structure. You get validation of your changes through code reviews in pull requests and branch build policies.
- Every branch you use can modify the pipeline by modifying the `azure-pipelines.yml` file. Learn more about branch consideration for YAML pipelines.
- A change to the build process might cause a break or result in an unexpected outcome. Because the change is in version control with the rest of your code base, you can more easily identify the issue.

**Security in CI/CD pipeline:**

- Security and quality must be embedded into the code and should not be left to quality gates when dealing with automation in the CI/CD pipeline. Developers need integrated security tools in their integrated developer environment in order to properly lint the code.
- Linting is a process performed by tools that identify coding style deviations and unsafe practices. More sophisticated Static Application Security Testing (SAST) tools can find buffer overflows, SQL injection flaws, and other issues.

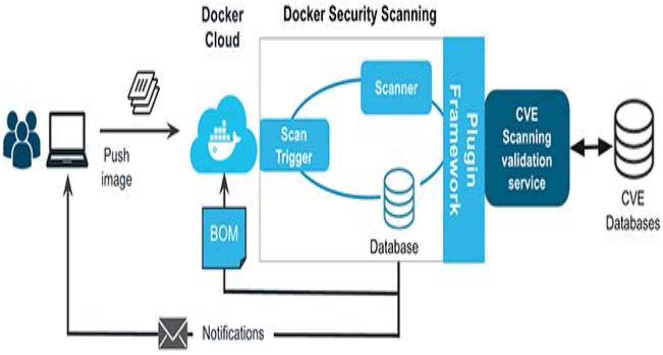


Figure:4 Linting Process

Devops teams should "never put vulnerable software into production".

It's important to recognize that CI/CD doesn't just deliver code. It's also an opportunity to adopt shift-left testing and evolve a continuous testing strategy. Teams that adopt testing as a core principle can then look for opportunities to validate security before triggering CI/CD pipelines to deploy releases to any environment. In addition to integrating SAST security testing, teams should focus on:

- Triggering penetration tests to find back doors and other entry-point vulnerabilities
- Validating security controls and testing authorizations
- Data masking is a critical part of security automation during CI/CD. Other techniques include using synthetic data and service virtualization. A synthetic data set to mimic the real thing will further strengthen security as threat actors gain nothing if that database is breached.

## VII. DIFFERENCES BETWEEN VIRTUAL MACHINE, DOCKER AND KUBERNETES

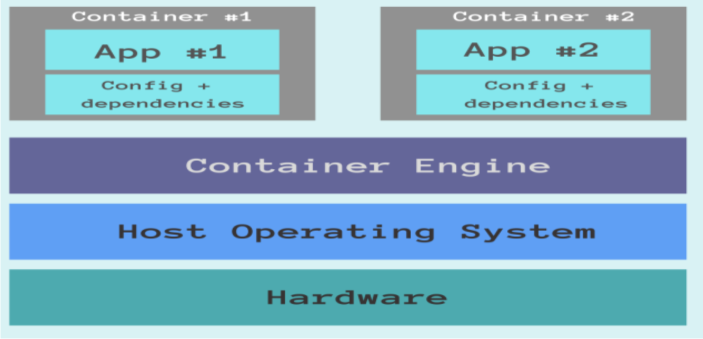


Figure:5 VM Layers

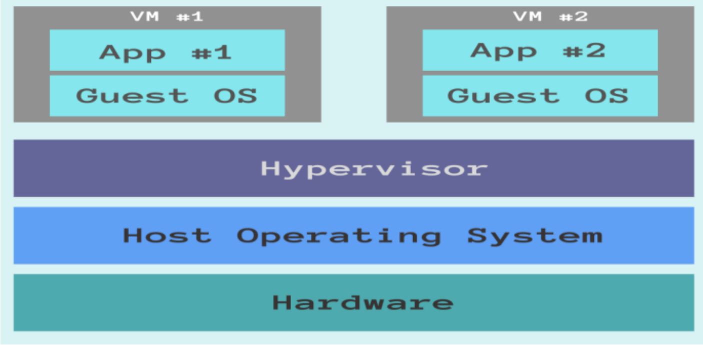


Figure:6 Docker Layers

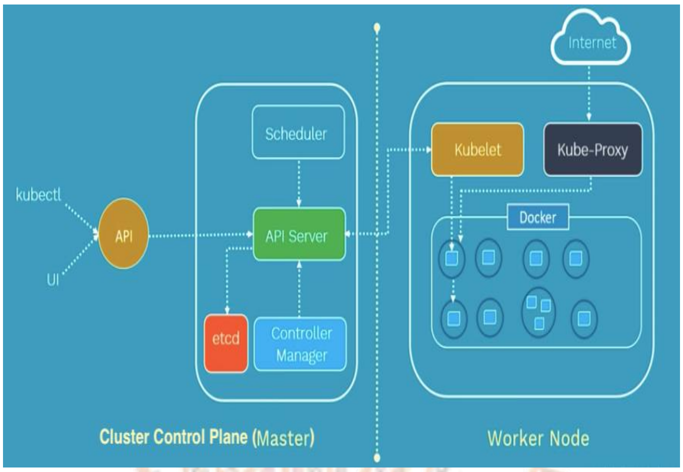


**Figure:7 Kubernetes Architecture**

| | **Container** | **Virtual Machine** |
|---|---|---|
| **Boot-Time** | Boots in a few seconds. | It takes a few minutes for VMs to boot. |
| **Runs on** | Dockers make use of the execution engine. | VMs make use of the hypervisor. |
| **Memory Efficiency** | No space is needed to virtualize, hence less memory. | Requires entire OS to be loaded before starting the surface, so less efficient. |
| **Isolation** | Prone to adversities as no provisions for isolation systems. | Interference possibility is minimum because of the efficient isolation mechanism. |
| **Deployment** | Deploying is easy as only a single image, containerized can be used across all platforms. | Deployment is comparatively lengthy as separate instances are responsible for execution. |
| **Performance** | Limited performance | Native performance |

**Table:2 Differences of VM and Containers**

## VIII. Test Reports

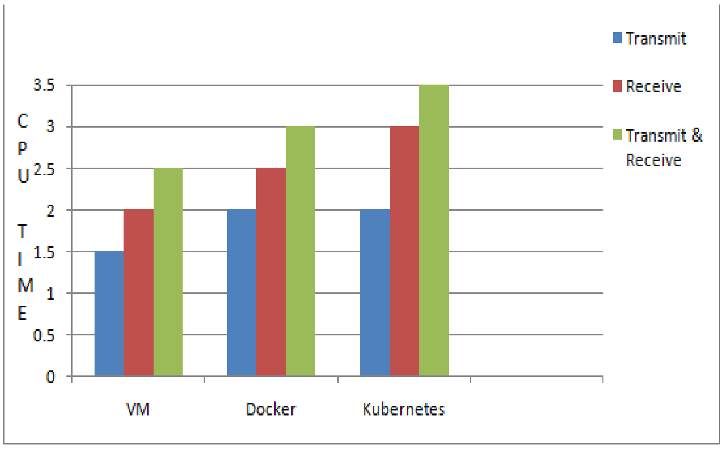


**Figure:8 CPU Time in Bytes**

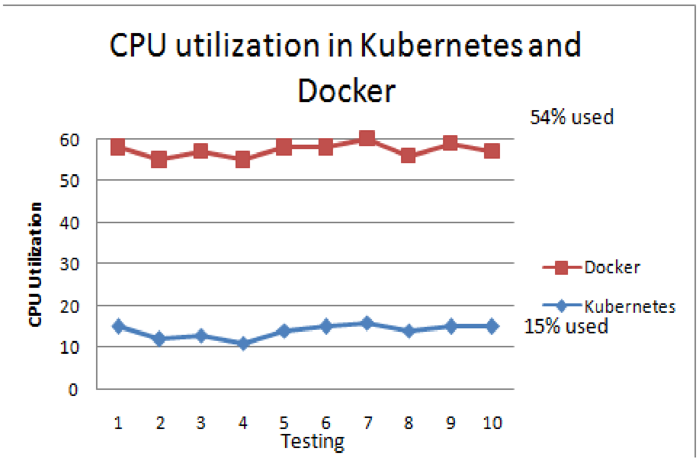


**Figure:9 CPU Utilization**

## IX.CONCLUSION

- Builds and testing can be easily performed manually.It can improve the consistency and quality of code.Improves flexibility and has the ability to ship new functionalities.CI/CD pipeline can streamline communication.It can automate the process of software delivery.CI/CD pipeline helps to increase our product visibility.
- From VMs to Docker to Kubernetes, each has unique features that enable you to combine them, meeting your needs in a more efficient way.
- All three are powerful tools for developers and infra administrators to efficiently manage their infrastructure and ensure reliability and scalability.
- Implementing a CI/CD pipeline with Kubernetes automation capabilities has many benefits. Developers can easily patch updates, solve outage issues in case of an unforeseen traffic spike, and improve resource efficiency.

## X.REFERENCES

[1]. D. Ardagna, G. Casale, M. Ciavotta, J. F. Perez, W. Wang, “Quality-of-Service in cloud computing: modelling techniques and their applications”, Journal of Internet Services and Applications, Volume: 5, Issue: 11,pp.1- 13,2014.

[2]. M. Salam and A. Shawish, “A QoS-oriented inter-cloud federation framework”, IEEE Systems Journal, p. 642-643, 2015.

[3]. Albert, E., de Boer, F.S., Hähnle, R., Johnsen, E.B., Schlatte, R., Tapia Tarifa, S.L., Wong, P.Y.H.: Formal modeling of resource management for cloud architectures: An industrial case study using Real-Time ABS. Journal of Service-Oriented Computing and Applications 8(4), 323–339 (Dec 2014). https://doi.org/10.1007/s11761-013-0148-0

[4].Leonardoleite,Carlarocha,Fabiokon,Dejanmilojicic,Paulomeirelles.In"A Survey of DevOps Concepts and Challenges

[5]. Dirk Merkel. 2014. Docker: lightweight Linux containers for consistent development and deployment. Linux Journal. 2014, 239, Article 2 (March 2014)

[6].Kubeflow Project, [online] Available: https://www.kubeflow.org/.

[7]. Rodriguez, M.A, Buyya, R: Container-based cluster orchestration systems: a taxonomy and future directions. Software: Prac. Exper. 49(5), 698–719 (2019)

[8]. Cote, M.: Why Large Organizations Trust Kubernetes. Web page. Accessed 09-29-2022

[9]. Kumar, M., Sharma, S. C., Goel, A., Singh, S. P.: A comprehensive survey for scheduling techniques in cloud computing.

[10]. The Docker authors: Empowering app development for developer; Docker. Web page. Accessed 28-09-2022, https://www.docker.com/ (2022)