International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 7 July 2021, pp: 3413-3417www.ijaem.net ISSN: 2395-5252

Docker Container and Its Use Cases

[1]Nisha Rani, [2]Kiran Ahuja, [3]Rahul Mahawar, [4]Rajshree Panchal
[1,2]*Assistant Professor, Department of CSE, Arya College of Engineering and Research Centre, Jaipur, Rajasthan, India*
[3,4]*B.Tech Student,Department of CSE, Arya College of Engineering and Research Centre, Jaipur, Rajasthan, India*

Submitted: 15-07-2021 Revised: 29-07-2021 Accepted: 31-07-2021

ABSTRACT: For many academic areas, the capacity to duplicate and repeat scientific discoveries has become an increasingly essential problem. Contributions of scientific work in computer science and, more especially, software and web engineering rely on established algorithms, tools and prototypes, quantitative assessments, and other computational studies.Many undocumented assumptions, dependencies, and settings exist in published code and data, making replication difficult to establish. This lesson explains how Docker containers may help with reproducibility of research products in software and web engineering, as well as their practical applications.In today's ERA, many tools and technologies present in the computer world, docker is one of them. It is a lightweight, opensorce, secure platform, that simplifies building, shipping, running applications. It relies on "images" and "containers".

Keywords : Docker; Container; Virtual Machine; Docker Image

I. INTRODUCTION

Years ago, before Docker containers, huge corporations like Walmart, Target, and Chase Bank that relied on technology employed servers and added many of them, resulting in over-allocation, in order to manage the growing number of requests from customers.The disadvantage of over-allocating servers was that it was incredibly expensive, and if the servers did not scale effectively, the firm would die.After a few years, VMware developed the notion of virtualization, which enabled several operating systems to operate on the same host, basically allowing any programme to operate in isolation on the same server and infrastructure, as if it were running a completely distinct computer on the same computer, which was a game-changer for many industries[1].

Although the concept of Virtualizationcame with a lot of benefits, but it was also highly expensive. To begin with, there are numerous kernels for each guest operating system that will operate on the infrastructure; also, resources must be allotted to each guest operating system that is introduced to the infrastructure.[2]Furthermore, virtualization is costly because, despite the absence of actual hardware, virtual hardware consumes resources, including the guest operating system space and RAM allocation necessary for this operating system [3].

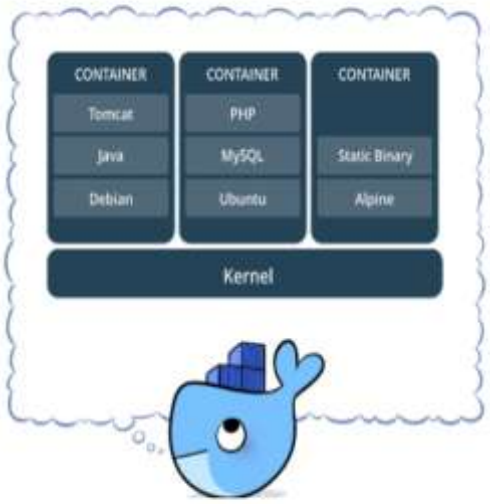

Fig. 1 Docker Containers

II. DOCKER RELATED TECHNOLOGIES

Container technology is becoming more popular among businesses as cloud computing becomes more prevalent. Docker is an LXC-based container engine that employs container technology

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 7 July 2021, pp: 3413-3417www.ijaem.net ISSN: 2395-5252

for software development and deployment, and distributes it in mirror mode to consumers.[4] When the programme is running, the user can obtain the needed software. It provides features like as portability, cross-platform compatibility, and ease of use. Docker contains numerous key mirror [5] container and warehouse components. The fundamental environment for application execution is provided by mirroring, which is the cornerstone of developing a container. Warehouses are collections of mirrors, whereas containers support application instances operating within them. Docker uses the C/S service architecture [6], as indicated in the diagram.

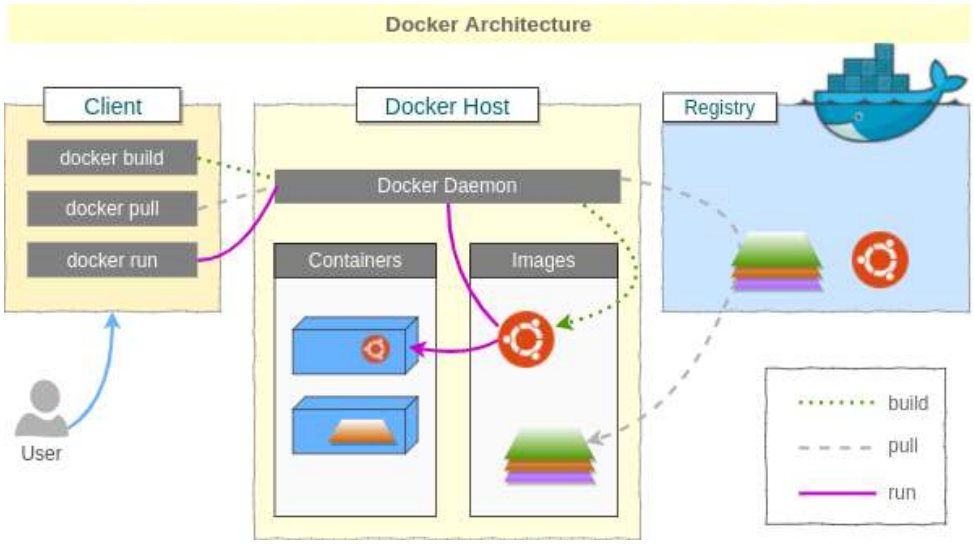

Fig. 2Docker architecture diagram

The Docker client makes a request[8] to the Docker server, which is answered by the Docker Daemon. To make a container, you must first download the mirror and store it to a graph, then use the network driver to establish the container's network environment. Namespaces and groups are used to achieve resource isolation and limitation in order to assure security. Etcd is a key value storage repository that is distributed and highly consistent. It's easy to use, safe, quick, and dependable. It's the cornerstone of distributed container design. Etcd is used to link distributed nodes and programmes within the container, as well as to monitor the cluster as a whole, in the building of distributed container applications.

Throughout the design, hosts in each cluster node execute numerous containers. The monitor agent is similar to the collector in that it gathers container performance data and broadcasts it. Receiving data, processing and storing data, listening to interface information, responding to requests and providing data to the Web interface, and alerting when monitoring node and container exceptions are all responsibilities of the server. The database is primarily in charge of storing and processing data as well as offering query services. Etcd is in charge of keeping information about the cluster's nodes and containers for the server to monitor. Requests and data are transmitted over the monitoring service interface.

III. DOCKER INSIDE

Docker have 4 primary core components Client and Server, Docker Images, Docker Registries, and Docker Containers.The next sections will go through each of these components in further depth.

3.1 Docker Client and Server

Docker may be viewed as a client-server programme. When a docker client sends a script, the docker server handles it accordingly. Docker includes a command-line client application as well as the whole RESTful (Representational state transfer) API.A local docker client can connect to a remote server or daemon on a different system, or a local docker client can connect to a remote server or daemon on a different system. [9].

3.2 Docker Images

Building a picture may be done in two ways. The first is to use a read-only template to create a picture. A basic image is the foundation of any image. Operating system images, such as Ubuntu 14.04 LTS or Fedora 20, are essentially base images. The operating system images combine to form a container that can execute a whole operating system.

It is also possible to construct a base image from scratch.It is possible to modify the

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 7 July 2021, pp: 3413-3417www.ijaem.net ISSN: 2395-5252

basic picture to include desired programmes, but it is necessary to generate a new image. "Committing a change" refers to the act of producing a new image. The second method involves using a Docker file. When you run the "Docker build" command from the bash terminal, it generates an image by following all of the instructions in the docker file. This is a technique for making things.

3.3 Docker Registries

Docker registries are where your Docker images are stored. It works in the same way as source code repositories do, allowing images to be pushed or downloaded from a single location. The two types of registers are public and private registers. Docker Hub is a public registry where anybody may download and publish existing images rather than starting from scratch. Using the docker hub function, images may be delivered to a certain region (public or private).

3.4 Docker Containers

A Docker image is used to generate a docker container. Containers hold all of the components required for a program's execution, allowing it to run independently. Consider the following scenario: there is an image of Ubuntu OS with SQL SERVER; A container is created when this image is launched using the docker run command, and SQL SERVER runs on Ubuntu OS.

IV. KEY DOCKER USECASES

A. Simplifying Configuration -This is a simple use case; the Docker configurations can be reused in a variety of situations. When it comes to running any platform with its configurations, virtual machines have an advantage; Dockers have the same advantage, but without the Virtual Machine overhead, and allow you to put your configurations and environment into the code and deploy it. The application environment and infrastructure requirements are separate. When it comes to the real-world use case, it helps organisations speed up project setup by allowing them to jump right into development without having to go through the tedious process of setting up environments and configuration procedures [10].

B.Code Pipeline Management -The code pipeline management is heavily influenced by the previous use case. A few minor differences can be seen as code written in a developer environment progresses through various stages (each of which uses different platforms/environments) and approaches the production stage; Dockers, on the other hand, provide a consistent environment throughout all phases of development and deployment, allowing for a simple development and deployment pipeline. The Docker image's stability and ease of use can also help with the aforementioned pipeline management.

C. Docker for Development Productivity - Docker makes it easier to achieve two goals in a development environment: the first is to keep a developer near to production; Docker makes this possible since it has no or little overhead while working remotely.The second need is to have an active development environment for interactive use; Docker makes this possible by making application code accessible to the container from the host OS through shared volumes. This provides advantages such as allowing the developer to alter the source code from the platform of his choosing while still keeping track of it.

D. Multi-Tenancy -Docker is utilised in multi-tenant systems since it helps to avoid large application rewrites. The codebases of multi-tenant systems are far more complicated, stiff, and difficult to manage. Redesigning an application takes a significant amount of effort and money.Docker simplifies the process of creating limited environments to execute multiple instances of programmes for each tenant easier and more cost-effective. Docker's easy-to-use API makes it possible to programmatically start containers.

E. Debugging Capabilities -Docker offers a number of tools that work well with the container idea. To mention a few of its features, one is the ability to checkpoint containers and their versions, as well as distinguish between two containers, allowing for quick application fixes.

F. Better disaster recovery -A Docker image, also known as a snapshot, may be saved at a certain moment in time and recovered in the event of an emergency. A file can be duplicated to new, different hardware using Docker. Using a Docker image, you may switch between two distinct versions of the same programme.

G. Improvement in the adoption of DevOps - Docker has been used by the DevOps community to standardise confined deployment. The association between Docker and DevOps is determined through CI/CD. Docker maintains consistency across the testing and production environments. Dockerstandardises the configuration interface and makes machine setup easier. Docker can be employed to introduce improvements in the DevOps of the company.

H. Rapid Deployment- Before Virtual Machines, creating new hardware resources was a multi-day operation that was eventually reduced to a matter of minutes by Virtualization. Docker, on the other hand, decreases the time to seconds by just requiring the establishment of a container for the process rather than booting up an OS.It makes it

easier to create and destroy resources without having to worry about the expense of doing so again. The lower cost of launching a new instance allows for a more dynamic deployment of resources.

V. DOCKER USAGE

A. WHEN TO USE?

i. Discovering the latest technologies: Docker creates a disposable and isolated environment that allows you to get started with a new tool without having to spend a lot of effort on setup and configuration[11]. Several projects save docker images containing apps that have already been installed and configured.

ii.Basic use cases: Docker Hub is a cloud registry service that allows you to get Docker images created by other communities. It's a fantastic way to get images for either Basic or Standard applications.

iii. App Isolation: When hosting many apps on a single server, managing each application component by keeping it in its own container eliminates dependencies.

iv. Developer Teams: For developers working in a separate arrangement, Docker provides a close match between the local development environment and the production environment.

B. WHEN NOT TO USE?

Dockercannot be the best solution always; few cases are as described below [12] –

i. **Complicated Applications**: Unlike simple apps, utilising a pre-created docker file or getting images from Docker Hub will not suffice for complex apps since modifying, constructing, and managing requests and answers across several containers on numerous servers takes a long time.

ii. **Performance-critical Applications:** When it comes to performance, Docker outperforms VMs since Containers share the host kernel and imitate the entire operating system. Dockers can be avoided in order to get the greatest performance out of the server since processes operating on a native OS are quicker than those operating within a container.

iii. **Upgrade hassles:**Docker is still a developing technology, which need periodic upgrades in order to take advantage of new capabilities.

iv. **Security is a critical factor:**When it comes to more complex applications, docker containerization's methodology introduces a number of vulnerabilities, including kernel-level threats, inconsistency in docker container updates and patching, unverified docker images, unprotected communication, and unrestrained network traffic.

v. **Multiple Operating System:**Because docker containers share the host computer's operating system, it would be necessary to utilise virtual machines to test or execute the same apps on multiple operating systems.

VI. CONCLUSION

This paper examines a variety of Docker container use cases, including Docker as Edge Computing, developing an IaaS platform for interactive social media apps, and integrating Docker containers with IoT and resource management to improve energy efficiency. This paper also provides an overview of the Docker container's evolution, as well as its use cases and application areas.

REFERENCES

[1] A Survey on Docker and its Use Cases: Department of CSE, RVCE, Bengaluru, Associate Professor, Department of CSE, RVCE, Bengaluru, 2020

[2] Dong-Ki Kang, Gyu-Beom Choi, Seong-Hwan Kim, II-Sun Hwang and Chan-Hyun Youn, “Workload-aware Resource Management for Energy Efficient Heterogeneous Docker Containers”., School of Electrical Engineering.

[3] Sharma, Shachi, Krishna Kumar Sharma and Himanshu Arora, "A Natural Human-Machine Interaction via an Efficient Speech Recognition System" in International Journal of Applied Information System (IJAIS) - ISSN, New York, USA, vol. 4, no. 9, pp. 2249-0868, December 2012

[4] Dong Bo, Wang Xue, Sophie, et al, “Research on virtualization technology based on Docker,” Journal of Liaoning University, Natural Science Edition, pp. 327-330,2016.

[5] Zhao Lele, Huang Gang, Ma Yue, “Research on Hadoop platform architecture based on Docker,” Computer Technology and Development, pp. 99-103, 2016.

[6] Lu Shenglin, Ni Ming, Zhang Hanbo, “Optimization of scheduling policies based on Docker Swarm cluster,” Information Technology, pp.147-152,2016.

[7] Gaurav Kumar Soni, Himanshu Arora and Bhavesh Jain, "A Novel Image Encryption Technique Using Arnold Transform and Asymmetric RSA Algorithm", In. Springer International Conference on Artificial Intelligence: Advances and Applications

2019,Algorithm for Intelligence System, pp-89-90, 2020.

[8] Liu Minxian, Design and implementation of serviceinvocation topology analysis and performance monitoring system based on Docker, Zhejiang: Zhejiang University, 2016.

[9] Turnbull, J. (2014). The Docker Book: Containerization is the new virtualization.

[10] Deploy Docker Open Source, or Enterprise for High Performing Systems, https://www.flux7.com/tech/%20container-technology/docker

[11] An Introduction to Docker and Analysis of its Performance: Asia Pacific University of Technology and Innovation Technology Park Malaysia, Kuala Lumpur, Malaysia.

[12] Research and implementation of Docker performance service in distributed Platform: College of Information Engineering Nanjing Normal University Taizhou College Dongfeng South Road No. 518, Hailing District, Taizhou, Jiangsu, China.

[13] Dr. Himanshu Arora, Mr. Manish Kumar and Mr. Sanjay Tiwari, "Improve Image Security in Combination Method of LSB Stenography and RSA Encryption Algorithm", International Journal of Advanced Science and Technology,Vol-29, No-8, 6167-6177, 2020.

[14] Banga S., Arora H., Sankhla S., Sharma G., Jain B. (2021) Performance Analysis of Hello Flood Attack in WSN. Springer Proceedings of International Conference on Communication and Computational Technologies. Algorithms for Intelligent Systems. Springer, Singapore.