

# Study of Container Technology with Docker

Sudip Patil

Department of Information Technology

Sir Sitaram and Lady Shantabai Patkar College of Arts and Science, Mumbai, India

**Abstract:** *As the technology is increasing day by day, demand of user has been growing rapidly. Several organizations are improving technology and providing better flexibility for running applications. As Docker is an open source platform means it is used by developers to deploy application in containers and then run them in production. In this paper, the components of Docker are discussed and with that pros and cons of docker are discussed as well. At last, some different containers are compared with docker to see which one is better.*

**Keywords:** Docker, Components of Docker, Virtual Machine, Kubernetes.

## I. INTRODUCTION

The demand for open source platform has been increasing more and something like docker is now an open source platform means it allows to run applications, it helps in sending the packed code to someone else to get the same result. Docker is very helpful when it comes to software delivery because it usually enables to separate the application from an actual infrastructure to deliver it. Docker basically uses containers to wrap or encapsulate an application into it. With the help of these containers it becomes easy for one to send the data in a secure way and the arrangement of data won't get disturbed. These Docker containers can be run on any OS platform such as Linux or Windows which has the Docker runtime installed.

vWell this Docker project was found by Solomon Hykes and started in France as an internal project. It was a startup project within dotCloud which is a platform-as-a-service company. Many of the technologies were introduced at that time and were ruling up to 10 years, and the Hykes released the Docker which was an open source platform in March 2013.

Docker overcame other technologies and stand as a best because it helps in creating and controlling containers. In Docker not only encapsulation is provided but also there are some more benefits added to it, such as portability isolation and control. Usually Docker Containers are small in size take it as megabytes. Docker containers can be start quickly or instantly. They have the capability of versioning and component reuse because of their own built-in mechanism.

Docker can be said to be as a product set of Platform-as-a-service (PaaS). This product helps in delivering the OS-level virtualization software in packages which are named as containers. Docker can run multiple Virtual Machine simultaneously on a single server because these docker containers are lightweight. On the pinnacle of the operating systems kernel, containers are situated which keeps on running but in a separate way. Most of all compare to other container technologies, docker are capable of carrying more virtual situations on same tools.

Docker are more beneficial because of well-organized use of system resources. Some also says that docker container is said to be computer inside your computer or system. Docker will make sure that analyses are replicable. Docker is useful because it allows to quarrel province, beginning from the operating system and ending up to details such as latex and R package versions.

Docker Container is deployed with all resources which are essential for users. It includes its own CPU, memory, network resources and other things also but it is not dependent on other individual operating system and external kernel.

## II. DOCKER

Defining Docker as a tool used to reduce the load and make work easier for creating, deploying and running applications with the help of containers. Docker contains libraries, codes, runtime, dependencies, and system settings.

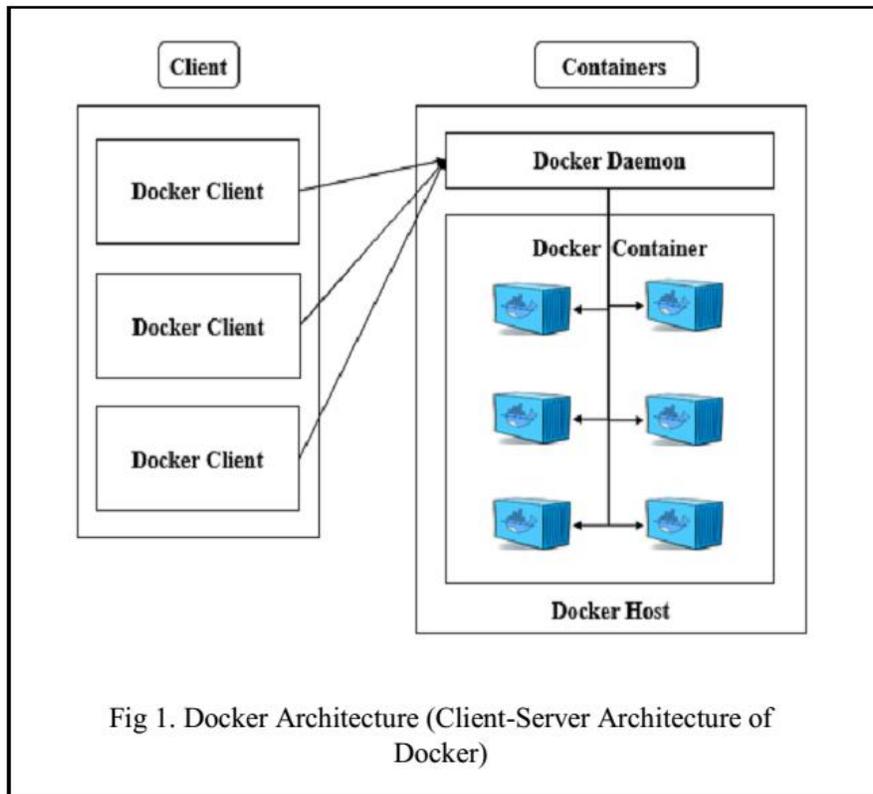
One can run many containers at same time on a designated host with the help of segregation and security. In Docker, one should not be dependent on what is inside or installed on the host, because of the nature of the containers which are lightweight and thus, contains everything needed to run an application. Docker always allows to test the code which is inside the container and deploy it into the fabricating domain as quickly as possible. One of the best use of docker is done in Microservices Architecture. Docker can be stated as a completely evolved the containerization world.

### III. ARCHITECTURE OF DOCKER

Docker architecture talks about the relationship between client and server and so it is named also named as ClientServer Architecture. This architecture has four main components of docker which are: Docker Client & Server, Docker Containers, Docker Images, and Docker Registries. Some detailed explanation of this component is given below.

#### 3.1 Docker Client & Server

As shown in the Figure 1, using Client and Server application, docker can be explained in a simple way. When the request from docker client arrives at docker server end, it will take action correspondingly. Docker will then transfer the complete RESTful API, over UNIX sockets or a network interface and also command line client binary. Now to run client and server application it will require a same machine or remote machine. So Docker Client and Docker Server or Daemon can be run on a particular given machine or can be run on another machine by connecting local docker client to the remote server. Another Docker Client is Docker Compose, it will help in communicating with applications consisting of a set of containers. Talking about Docker Daemon or Server, it will check for requests and manages the other Docker objects. This Docker objects includes networks, containers, images, and volume. A Server can connect or communicate with other servers too for managing the Docker services. Well for Docker Client also, it can communicate with one or more than one servers or daemon.



### **3.2 Docker Images**

In Docker Images, images are created using two techniques. The first technique allows an image to be built by using multiple layers of read-only mechanism. Every image in Docker starts with a base image and these base images are nothing but the operating system images like Fedora 20 or Ubuntu 14.04 LTS. Now completely running operating system capability is used by the container which is created by using the images of operating system. One thing to be noted is that, base images can be made up from the scratch. Using the base image one can add the applications whichever they need to add by making changes in base image. But every time it is mandatory to create new image. And this whole process of creating or building a new image is known as “Committing a Change”. Now second method to create an image is to create a docker file. These docker files are nothing but a set of pre-written commands in a text file. When this docker file is run on a bash terminal following the command “Docker Build” then it will follow all the instructions which are inside the docker file and then it will form an image. The second technique is a way where images are built automatically.

### **3.3 Docker Registry**

In Docker Registry, all the docker images are stored in docker registries. Images are pushed or dragged from a single source from the source code repositories which works correspondingly. In Docker, there are mainly two types of registries, which are, public and private. When the images which are available user can pull those available images and can push their own images which they need without even forming an image from start means no base image is needed. This process takes place in public registry and is named as Docker Hub. So by using characteristics of docker hub, images are given out to a specific area. Another public registry is Docker Cloud. If one doesn't want these public registry then they can even run their own private registry. Docker by default takes images from Docker Hub. But when it comes to security and optimization of building the images, one must go through Artificial Docker Registry.

### **D. Docker Container**

Docker Containers are created by Docker images, it is a remote application built from one or more images. Docker Containers are runnable instances of docker images. Operation such as start, stop, move, delete or create can be performed on containers using Docker API or CLI. Containers are very helpful because it contains whole set or kit of package system which are required for an application to run. Some of the things included in containers are libraries, CPU, memory, network resources and dependencies. Docker Container gives full guarantee that the software can run in any condition or environment because it comes without operating system, so it uses the Host OS for functioning. Thus it is more efficient, portable and lightweight system.

## **IV. PROS AND CONS OF DOCKER CONTAINER**

### **4.1 Pros(Advantages) of Docker Container**

Docker container has provided many benefits and thus has become more popular in less time for virtualization in Linux containers which are in demand and enhanced. There are many advantages of Docker some of them are highlighted here: Portability, Scalability, Speed, Density, and Rapid delivery.

- **Portability:** Containers provide portable environment for the applications created inside the docker. Bundles of applications are moved around as a single unit and it won't affect the performance of the container.
- **Scalability:** Docker can be run on all the system having Linux. Docker has the ability to install it on several physical servers, cloud platforms, data centers, etc. Containers can be simply transferred back and forth from cloud environment to desktop and from there again to cloud with less than a minimal time (rapidly). Scaling is easy and can be handled by the user as per the user's need. User can scale up or down and adjust the scale according to its need. User can scale from one to many or many to one as per its need.
- **Speed:** Small containers require less time to build and process becomes more faster. Speed can be considered as a trump advantage of Containers. Processes such as testing, deployment and development are rapidly

covered as the size of container is small. If the container is built then it is pushed to the testing phase and from there to the production phase.

- **Density:** In Docker Containers resource consumption is less and it uses the available resources effectively as it does not use hypervisor. Compared to virtual machines, containers can be run on a single host. Docker gives high performance because of higher density and no wastage of resources to overhead.
- **Rapid Delivery:** A standardized container format of Docker helps programmers or software teams to be free of worry about other tasks, no stress need to be taken. Administrator and programmer have separate responsibility of deploy and maintain the server of containers, and take care of the applications which are located inside the containers respectively. All the applications inside the containers which are tested can work in any environment because they have all the required dependencies embedded in it.

Other advantages of docker containers are: It is Open Source and free, Consistent, Isolated, and Security is well maintained.

#### **4.2 Cons (Disadvantages) of Docker Container**

Docker container also have some drawbacks so before using it one must consider what to do and what not to.

1. Preserving the data is complicated: If all of the sudden container in docker gets shutdown then the data inside it gets completely lost forever, means data cannot be restored. But before shutting it down if data is saved somewhere else then data remains and can be gained. Usually Docker has a way to preserve the data in Docker Data Volumes, but to use it is a hard task, one has to face many challenges while storing in it.
2. Platform Dependent: Docker can be Platform Independent but to do so it requires an additional layer between Host OS and Docker. At initial stage when Docker was brought into picture it was designed to only on Linux System but since technology is evolving, docker now also supports other OS such as Windows, Mac OS X, etc. by applying additional layer.
3. Bare Metal Speed is not achieved: Docker consumes most of the resources which are available and uses it efficiently but along with it, it also uses overlay networking which reduce the performance of the containers which is not up to the mark.
4. Compatibility Issue: When it comes to local machines docker faces the compatibility issue because Docker is compatible with local machine having 64-bit, but it does not support older machine less than 32-bit.

Other disadvantages of docker containers are which also should to known to the users are:

- Will the Docker be adopted by the scientific researcher and teaching community is the major concern.
- Another major issue arises is about the security of the data, the security issues should be handled properly.
- Docker depends on Linux Kernel, provided by the local host, and docker cannot provide complete virtualization because of it.

### **V. COMPARISON OF DOCKER**

There are many containerization technology which can be compared with the Docker. Some similarities and differences can be seen in these technology.

#### **5.1 Docker vs CoreOS Rkt**

The first challenger to Docker superiority was CoreOS Rkt with respect to container technology. Rkt is better than docker because it resolves many flaws which docker inherent. Alternative to docker one can use Rkt Container technology for more secure building up container. CoreOS Rkt is an interoperable, open and secure. All the previous versions of docker run as root, so the exposure which is residing inside a containers can give opportunity for an attacker to have a super-user privileges. Next comparison talks about the open operability in which Rkt uses an open source container format called appc, whereas talking about docker it uses its own exclusive image formatting. Comparing some the features and checking for similarities and differences.

- 1. Capability Set:** Both have equal dominance when it comes to containers and both of the vendors have offered strong core container technology by building a complete ecosystem of container technology. Docker offers Docker's Datacenter as a solution for enterprise container orchestration, enterprise-grade security, and application management. Whereas CoreOS locates Rkt and more concerned and focused on security of container solutions. Container Linux by CoreOS is based on Linux Kernel which is an open-source lightweight operating system.
- 2. 3rd Party Integrations:** Docker provides more 3rd party integrations than CoreOS. Docker Hub is a service hosted by company which offers around 100,000 free apps, private and public registries, including the official repositories from leading 3rd part vendors which are Nginx and Ubuntu to MongoDB and Redis. Whereas CoreOS Rkt projects are all available on GitHub.
- 3. API and Elasticity:** CoreOS offers a RESTful API by using gRPC which is high performance, open-source universal RPC framework. Whereas Docker provides full set of REST APIs and SDKs which allows developers to take control of every characteristic of the container stack from custom applications. Docker and CoreOS source code are fully available on GitHub, and so they are open-source projects.
- 4. Security rating:** Security ratings of Docker differs from CoreOS. Docker have security rating of 760 whereas of CoreOS the security rating is of 675. But they both have some issues, docker resilience position is ruined because of disabled DNSSEC and other security issues and CoreOS is facing some problems with gaps. And yet they still have such good security ratings.
- 5. Support:** As of now it's been known that Docker and CoreOS Rkt are free and open-source offerings. They have some paid for enterprise offerings which consists of primary support and value-added services.

## 5.2 Docker vs Virtual Machine

Docker is a technology which provides containerize environment which are simply user space of the operating system. Containers are separated from the rest of the system and they are just a set of processes, when at low level, they will run from a distinct image. All the files which are essential to support those processes are provided by the distinct image. Whereas for Virtual Machine it is different, because it is not based on container technology. Well VMs are made up of not only by user space but also includes kernel space of an operating system. For virtualizing the server hardware one can use virtual machines. Each and every virtual machine has an operating system (OS) and applications. In Virtual Machine, hardware resources are shared from Host.

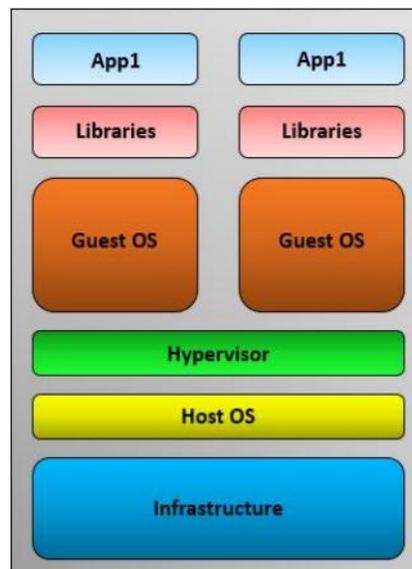
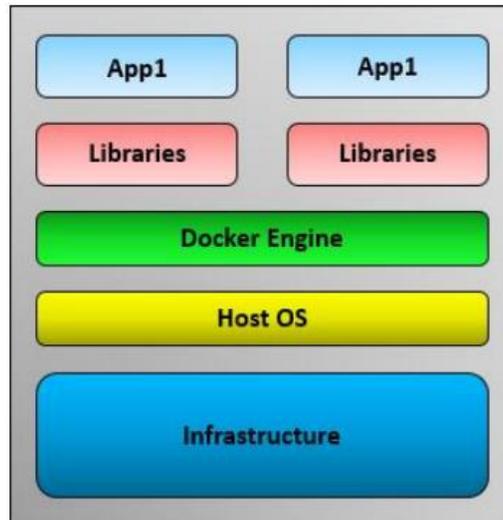


Figure 2: Virtual Machine Architecture

Virtualization is a concept used in cloud computing for system constitution, multi-tenancy, and resource provisioning. Virtualization have two terms, heavy weight virtualization and light weight virtualization. Resources which are virtualized uses cloud computing techniques to solve problems. In between, Guest OS and Host OS systems Hypervisor is present. In a server, hypervisor is a virtual platform that have capability to handle more than one operating system.



**Figure 3:** Docker Container Architecture

Docker tools are used for managing Linux Containers. It is a method of Operating System level virtualization. In Docker Container Architecture which is shown in below figure, shows that there are many Linux Containers in a single control host which are isolated. Containers include resources such as CPU, Memory, Network, and Block I/O and are located by Linux kernel. It can also deal with the cgroups without starting VM.

## VI. CONCLUSION

Docker Containers are efficient, portable and lightweight systems. Docker components such as Client & Server, Images, Registries, and Containers are core part of Docker. There are many containerization technology few of them were discussed in this paper such as CoreOS Rkt and Virtual Machine. Some Comparison between these containers with docker leads to the point where docker can be said to be a best container technology.

## REFERENCES

- [1] Bhatia, Gaurav & Choudhary, Arjun & Gupta, Vipin. (2017). The Road To Docker: A Survey. International Journal of Advanced Research in Computer Science. 8. 83-87. 10.26483/ijarcs.v8i8.4618.
- [2] Vase, Tuomas Advantages of Docker Jyväskylä: University of Jyväskylä, 2015.
- [3] Bashari Rad, Babak & Bhatti, Harrison & Ahmadi, Mohammad. (2017). An Introduction to Docker and Analysis of its Performance. IJCSNS International Journal of Computer Science and Network Security. 173. 8.
- [4] Guillermo Velez: Kubernetes vs. Docker: A Prime, Container Journal, January 14, 2019.
- [5] Serdar Yegulalp, Why you should use Docker and Containers, infoworld October 10, 2018, available on: <https://www.infoworld.com/article/3310941/why-you-should-usedocker-andcontainers.html#:~:text=Because%20Docker%20containers%20encapsulate%20everything,can%20run%20a%20Docker%20container.>
- [6] Docker vs. Virtual Machine: Where are the differences?, DevOpsCon available on: <https://devopscon.io/blog/docker/docker-vs-virtualmachine-where-are-the-differences/#:~:text=In%20Docker%2C%20the%20container%20running,space%20of%20an%20operating%20system.&text=But%20with%20a%20container%20environment,can%20run%20with%201%20S.>