**Major differences between Docker and Virtual machines**

**OS Support**

* The basic architecture of Docker containers and Virtual machines differ in their OS support. Containers are hosted in a single physical server with a host OS, which is shared among them.
* Virtual machines, on the other hand, have a host OS and individual guest OS inside each VM. Irrespective of the host OS, the guest OS can be anything – either Linux or Windows.
* Docker containers are suited for situations where you want to run multiple applications over a single OS kernel. But if you have applications or services that need to run on different OS flavors, VMs are required.
* Sharing the host OS between the containers make them very light and helps them to boot up in just a few seconds. Hence, the overhead to manage the container system is very low compared to VMs.

**Security**

* In Docker, since the host kernel is shared among the containers, the container technology has access to the kernel subsystems. As a result, a single vulnerable application can hack the entire host server.
* Providing root access to applications and running them with superuserprivileges, is therefore not recommended in Docker containers because of the security reasons.
* On the other hand, VMs are unique instances with their own kernel and security features. They can, therefore, run applications that need more privilege and security.
* At Bobcares, we implement additional security measures such as SELinux and AppArmor in Docker to secure the containers and to prevent them from cracking each other or the host kernel.

**Portability**

* Docker containers are self-contained packages that can run the required application. Since they do not have a separate guest OS, they can be easily ported across different platforms.
* The containers can be started and stopped in a matter of a few seconds, compared to VMs, due to their light-weight architecture. This makes it easier to deploy Docker containers quickly in servers.
* VMs, on the other hand, are isolated server instances with their own OS. They cannot be ported across multiple platforms without incurring compatibility issues.
* For development purposes where the applications have to be developed and tested in different platforms, Docker containers are thus, the ideal choice.

**Performance**

* Docker and Virtual machines are intended for different purposes, so it’s not fair to measure their performance equally. But their light-weight architecture makes Docker containers less resource-intensive than the virtual machines.
* As a result, containers can start up very fast compared to VM, as the latter has to load an entire OS to start. Resource usage also varies between the two.
* In containers, the resource usage such as CPU, memory, I/O, etc. varies with the load or traffic in it. Unlike the case of VMs, there is no need to allocate resources permanently to containers.
* Scaling up and duplicating the containers is also an easy task compared to that of VMs, as there is no need to install an operating system in them.