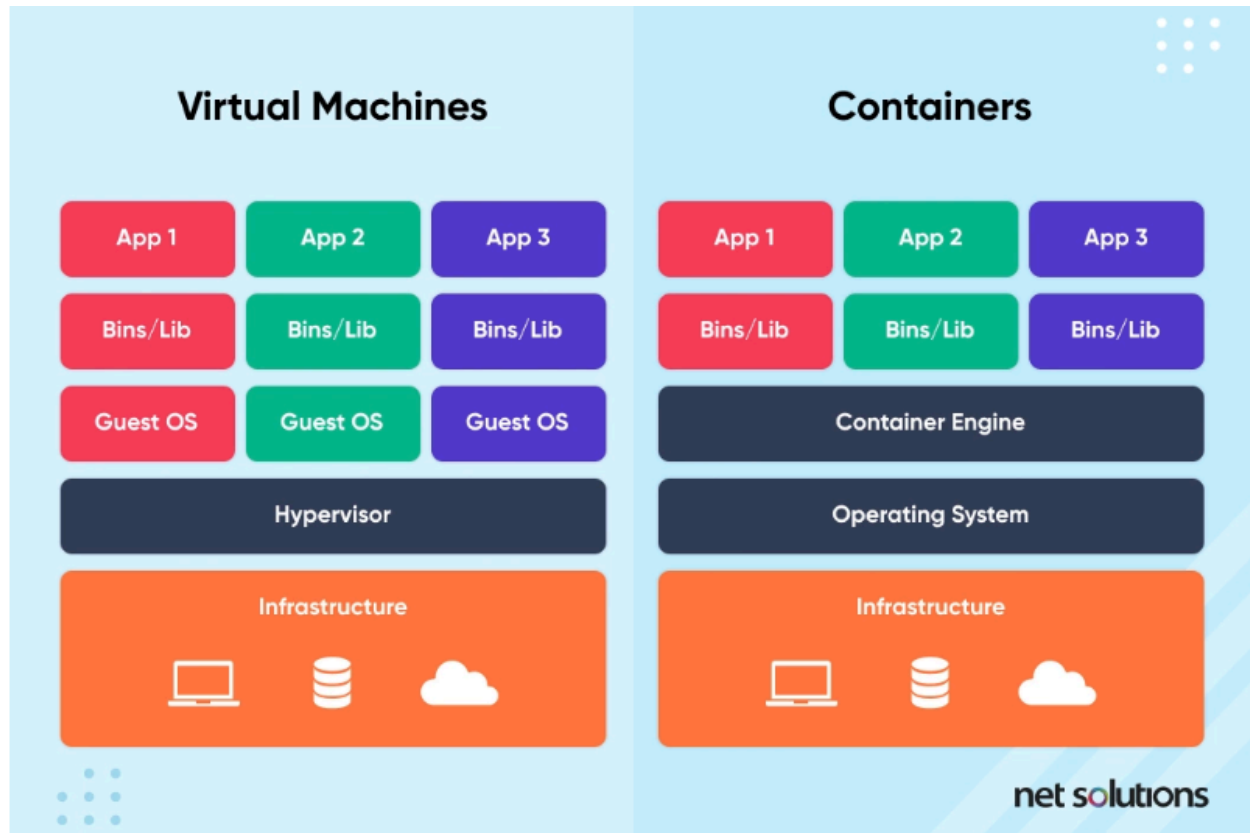


Virtualisation V/s Containerization



- Virtualization and containerization are both methods of running multiple applications on a single physical server, but they differ in how they isolate and share resources. Virtualization creates virtual machines, each with its own operating system and hardware resources, while containerization allows multiple applications to run on the same operating system, sharing the kernel.

Virtualization:

- Creates virtual machines:
- Each VM has its own operating system and hardware resources, providing strong isolation.

Uses a hypervisor:

- A hypervisor manages the virtual machines and allocates resources.

More resource-intensive:

- Requires more resources to run, including separate operating systems for each VM.

Better for legacy applications:

- Suitable for running applications with specific operating system or hardware requirements.

Containerization:

Uses a single operating system kernel:

- Multiple applications share the same kernel, making it more lightweight.
- Packages applications with their dependencies:
- Containers include all the necessary files, libraries, and configurations for an application to run.

Faster and more efficient:

- Containers are lightweight, start up quickly, and use fewer resources than virtual machines.

Ideal for modern, cloud-native applications:

- Well-suited for microservices architectures and applications that need to be deployed quickly and easily.

Key Differences:

- Isolation:
- Virtualization provides stronger isolation with each VM having its own OS and resources, while containerization relies on the host OS kernel for isolation.

Resource overhead:

- Containers are more lightweight and efficient than virtual machines, as they share the host OS kernel and resources.

Portability:

- Containers are highly portable and can be run consistently across different environments due to their lightweight nature and container images. VMs can be less portable, requiring specific hypervisor configurations for migration.

Use cases:

- Virtualization is better for running legacy applications and environments with specific OS requirements, while containerization is ideal for modern, cloud-native applications and microservices.