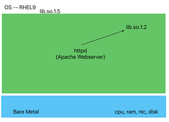
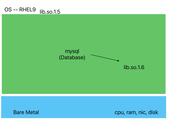
Task: Deploy a Webserver

1. Dedicated Server

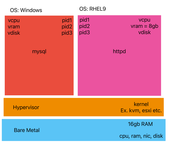


Problems

1. RHEL gets an update lib.so.1.2 --> lib.so.1.5
2. MySQL --> lib.so.1.6
3. Migration
4. Resource Utilisation
5. Default Programs
6. Each Application requires a new Dedicated Server



2. Virtualization

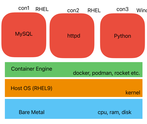


Problems

1. RHEL gets an update lib.so.1.2 --> lib.so.1.5 — Not Solved
2. MySQL --> lib.so.1.6 — Solved
3. Migration — Not Solved
4. Resource Utilisation — better
5. Default Programs — Not Solved
6. Each Application requires a new Dedicated Server — Solved

3. Cloud

4. Containerization



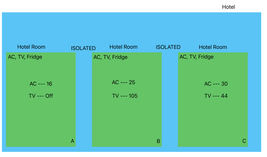
Problems

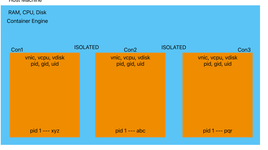
1. RHEL gets an update lib.so.1.2 --> lib.so.1.5 — 1. Solved
2. MySQL --> lib.so.1.6 — 2. Solved
3. Migration — 3. Very Easy
4. Resource Utilisation — 4. Solved
5. Default Programs — 5. Solved
6. Each Application requires a new Dedicated Server — 6. Solved

Containers

A Container is a set of one or more processes that are isolated from each other and the system as well
It is a standard unit of software that is bundled with an application and its dependencies
They are lightweight, standalone.
Benefits:
-- Low Hardware Footprint
-- Environmental Isolation
-- Quick Deployment
-- Reusable
-- Secure
-- Standard

Namespace





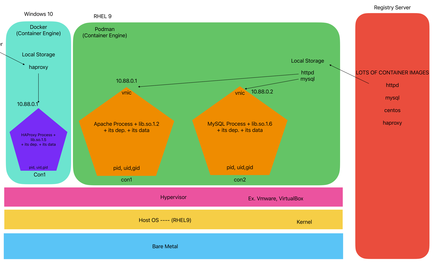
AC
A --- 16
B --- 25

TV
C --- 44
B --- 105

ISO Image (1Gb) : BaseOS (150mb) + Many Programs + Their Data + Their Libraries + Their Dependencies + Kernel ----- Run this Image ----- **OS**

Container Image (400mb) : BaseOS (150mb) + One Program + Its Data + Its Libraries + Its Dependencies ----- Run this Image ----- **Container**

Architecture



Registry Server

A Collection of Container Images
We can pull or push images as we want

Types:
Private Registries
Public Registries

Popular Registry Servers:

1. docker.io (hub.docker.com)
2. quay.io
3. catalog.redhat.com

OCI:
Open Containers Initiative

docker <==> podman

Naming Convention

registry_server_name/user_name/image_name:version

Ex.

docker.io/clearlinux/httpd:2.4

PODMAN

podman run

-d : Detached Mode, Run in Background
--name : To give name to your container
-e : To specify any Environment Variable

```
podman -v
podman search httpd
podman info
podman search nginx
podman pull docker.io/clearlinux/httpd:2.4
podman image ls
podman pull docker.io/clearlinux/httpd
podman image ls
podman pull docker.io/httpd
podman pull docker.io/httpd:2.4-alpine3.17
podman image ls
podman inspect docker.io/library/httpd:latest
podman image ls
podman images
podman pull nginx
podman image ls
podman run nginx:latest
podman run -d nginx:latest
podman container ls
podman run -d --name my_container httpd:latest
podman container ls
podman run -d --name nginx-container -e TRAINING=RedHat nginx:latest
podman container ls
podman stop kind_shtern
podman rm kind_shtern
podman container ls
podman container ls -a
podman kill my_container
podman container ls -a
podman rm my_container
podman container ls -a
podman rmi httpd:2.4-alpine3.17
podman image ls
podman logs
podman run -d --name new_con nginx:latest
podman container ls
podman psA
podman logs new_con
history
podman container ls
podman exec -it new_con bash
podman exec -it new_con date
ls
podman exec -it new_con ls
history
```

TASK:

Pull an MySQL Image and create a Container using necessary Environment Variables and it should run in the background.
Then enter the container do ls.
Remove the container completely.