

Dockerizing Python Microservices

Sanchit Balchandani
Ramanathan R



About Us



```
def __sanchit__():  
    core_team_member@hydPython  
    backend_developer  
    works@PramatiConnect  
    twitter_@inovizz
```

```
def __ramanathan__():  
    core_team_member@hydPython  
    director@ZentropyTech  
    twitter_@ramanathanhari
```

What this talk is about?



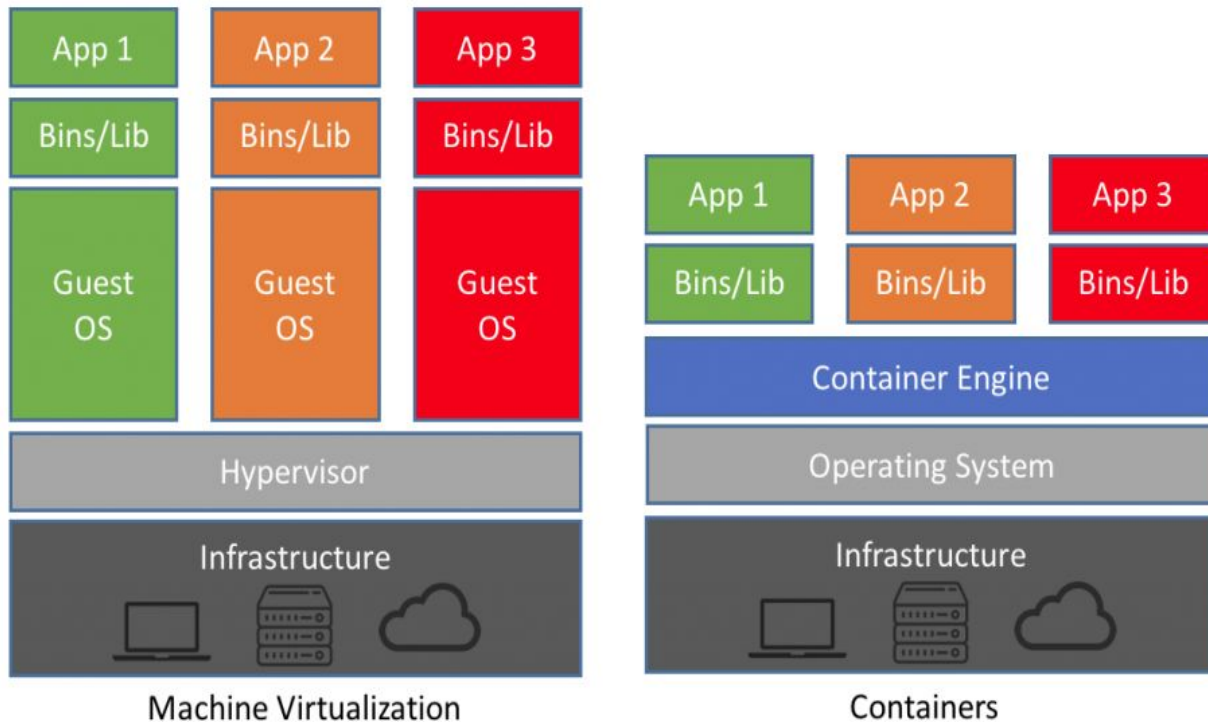
- Containers?
- Why docker?
- Docker Architecture
- Docker Images & Layers
- Creating a Dockerfile
- Important Docker Commands
- Docker Compose
- Docker Swarm & Kubernetes

What this talk will not cover?



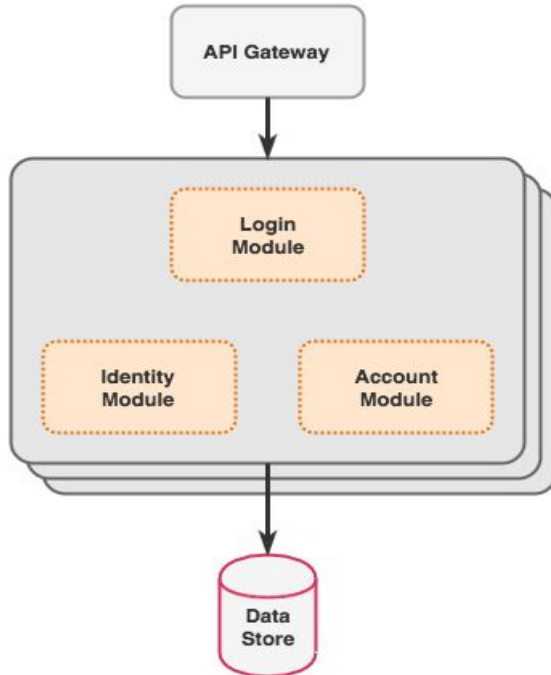
- Micro Level details of Docker Components
- What & Why of Microservices Architecture
- Detailed info on Docker Swarm or Kubernetes

Containers - lightweight VMs

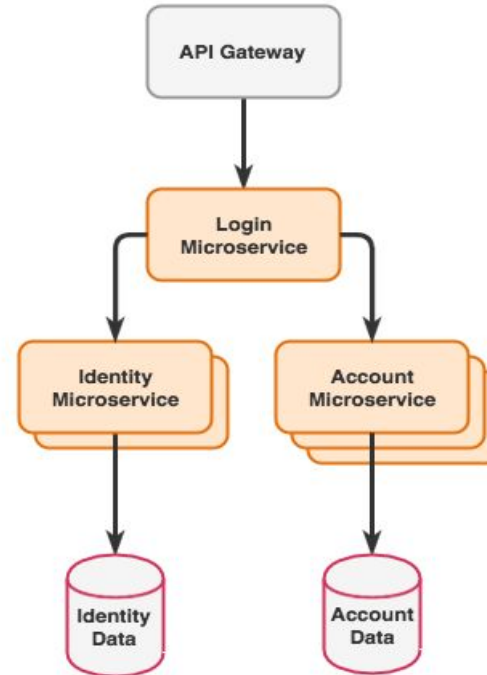


Monolith vs Microservices

Monolithic Application



Microservices Architecture

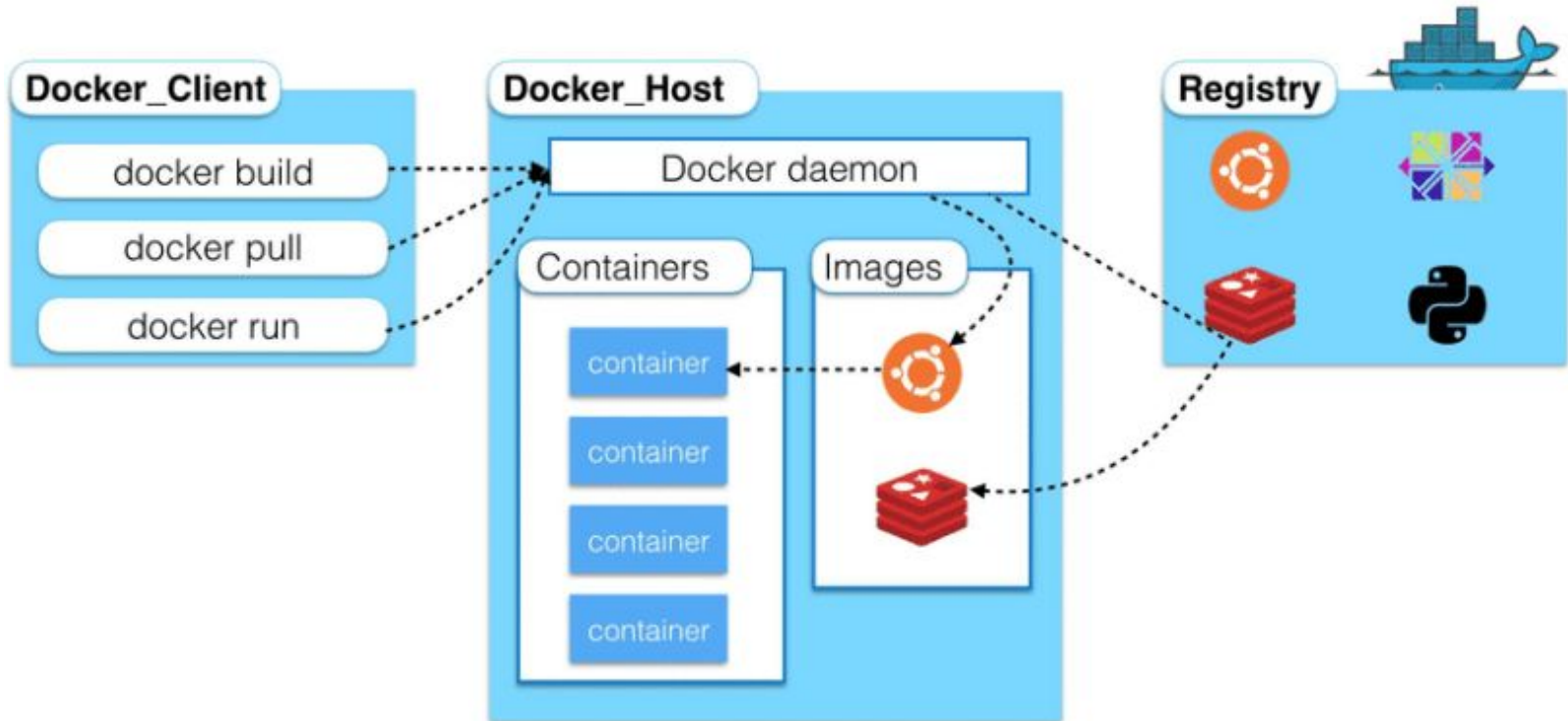


Why Docker?



- Containers (lighter than Virtual Machines)
- Fast and iterative development
- Rapid deployment
- Easy to ship
- Portability across machines
- Version control
- Multi-cloud platform support

Docker Architecture



Docker Image & Layers



- A Dockerfile is a recipe for creating Docker images
- A Docker image gets built by running a Docker command (which uses that Dockerfile)
- A Docker image is built up from a series of layers.
- Each layer represents an instruction in the image's Dockerfile.
- Each layer is only a set of differences from the layer before it.
- The layers are stacked on top of each other.



Creating a Dockerfile - Demo

Docker Commands



- [docker image ls](#) list images.
- [docker rmi](#) Remove one or more images.
- [docker rm](#) Remove one or more containers.
- [docker container ls](#) List containers.
- [docker run](#) creates and starts a container in one operation.
- [docker start](#)/[stop](#) starts/stops a running container.
- [docker pause](#) pauses a running container, "freezing" it in place.
- [docker unpause](#) will unpause a running container.
- [docker attach](#) will connect to a running container.
- [docker exec](#) Run a command in a running container
- [docker logs](#) Fetch the logs of a container

Docker Compose



- Compose is a tool for defining and running multi-container Docker applications.
- It uses YAML file to configure your application's services.
- With a single command, you create and start all the services from your configuration.



Docker Compose Demo

Docker Swarm



- A swarm is a group of machines that runs multiple Docker containers, on a cluster
- Docker commands can be executed on a cluster by a **swarm manager**.
- The machines in a swarm can be physical or virtual.



Any Questions?

Thank You!

Interested to Join Python Community
in Hyderabad (HydPy) ?

Connect with us at -

Twitter - @hydPython

Facebook - @HydPy