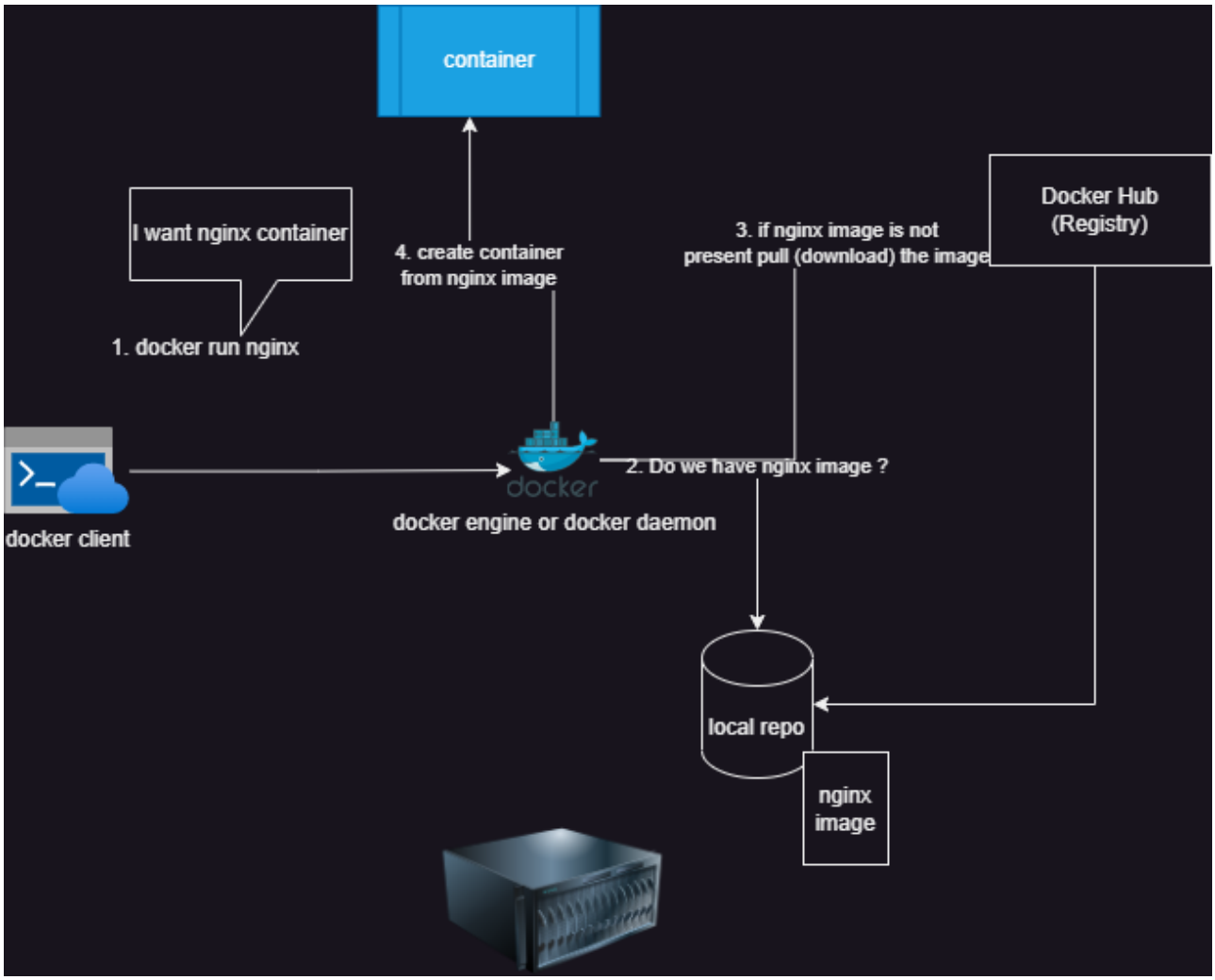
Docker

- High level Overview of Components:



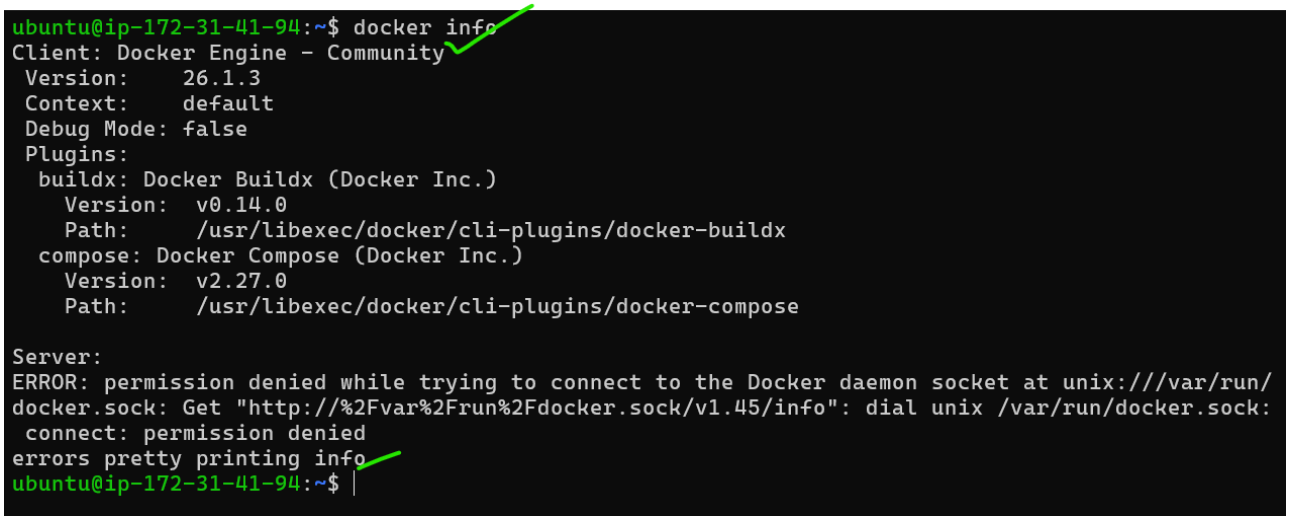
- Docker when installed on a system will have two major components
    - Docker client
    - Docker Engine (Daemon/Server)
- To create container we need docker image. Generally image represents application or application component (developed by your team)
- Using one image we can create multiple contianers
- Images are stored and made available using Registries and DockerHub is default registry.
- Container is a runtime in an isolated area which gets
    - virtual cpu
    - virtual ram
    - virtual disk (contents will be filled from image)
    - virtual network interface
- Docker cheatsheet: Refer Here

Install Docker

- Docker can be installed on
    - Windows Refer Here
    - Mac Refer Here
- Create a ubuntu Linux VM and ssh into it
- Run the following commands

```
curl -fsSL https://get.docker.com -o install-docker.sh
sh install-docker.sh
```

- Run the command `docker info`



```
ubuntu@ip-172-31-41-94:~$ sudo docker info
Client: Docker Engine - Community
 Version:    26.1.3
 Context:    default
 Debug Mode: false
 Plugins:
  buildx: Docker Buildx (Docker Inc.)
    Version:  v0.14.0
    Path:     /usr/libexec/docker/cli-plugins/docker-buildx
  compose: Docker Compose (Docker Inc.)
    Version:  v2.27.0
    Path:     /usr/libexec/docker/cli-plugins/docker-compose

Server:
 Containers: 0
  Running: 0
  Paused: 0
  Stopped: 0
 Images: 0
 Server Version: 26.1.3
 Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Using metacopy: false
  Native Overlay Diff: true
```

- NOw add you user to docker group

```
sudo usermod -aG docker ubuntu
exit
# relogin
docker info
```

```
ubuntu@ip-172-31-41-94:~$ docker info
Client: Docker Engine - Community
 Version:    26.1.3
 Context:    default
 Debug Mode: false
 Plugins:
  buildx: Docker Buildx (Docker Inc.)
    Version:  v0.14.0
    Path:     /usr/libexec/docker/cli-plugins/docker-buildx
  compose: Docker Compose (Docker Inc.)
    Version:  v2.27.0
    Path:     /usr/libexec/docker/cli-plugins/docker-compose

Server:
 Containers: 0
  Running: 0
  Paused: 0
  Stopped: 0
 Images: 0
 Server Version: 26.1.3
 Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Using metacopy: false
  Native Overlay Diff: true
```

Lets create our first docker container

- Docker hub is default image registry Refer Here
- image: hello-world Refer Here

- command `docker container run hello-world`

```
ubuntu@ip-172-31-41-94:~$ docker container run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
c1ec31eb5944: Pull complete
Digest: sha256:266b191e926f65542fa8daaec01a192c4d292bff79426f47300a046e1bc576fd
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/

ubuntu@ip-172-31-41-94:~$
```

- how to give a name to container
- what would happen if i dont give name to the container