

Docker Advanced Concepts: Naming, Port Forwarding & Volumes

1. Docker Container Naming

When you run a container, you can specify a custom name or let Docker assign a random one.

- Run a container with a name:

```
docker run -d --name mynginx nginx
```

- List all running containers:

```
docker ps
```

- Inspect a container by name:

```
docker inspect mynginx
```

- Remove a container by name:

```
docker rm mynginx
```

```
buntu@ip-172-31-19-167:~$ docker run -dt --name jithu1 hello-world  
291a3995cc5540a480ed8f01f60518bb78a7051c29ab3138acdd5ec7986c327
```

```
ubuntu@ip-172-31-19-167:~/sai$ docker images  
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE  
httpd          latest    6a4fe18d08d2  8 days ago   117MB  
nginx          latest    d261fd19cb63  8 days ago   152MB  
hello-world    latest    1b44b5a3e06a  3 months ago  10.1kB  
ubuntu@ip-172-31-19-167:~/sai$ docker run -d --name sai-container
```

```
ubuntu@ip-172-31-19-167:~$ docker ps  
CONTAINER ID  IMAGE      COMMAND      NAMES      CREATED      STATUS      PORTS  
TS          nginx      "/dock...tmp"    jithu1      14 minutes ago  Up 14 minutes  80/  
tcp        nginx      "/dock...tmp"    jithu2      25 minutes ago  Up 25 minutes  80/  
tcp        nginx      "/dock...sai-container"  jithu3      35 minutes ago  Up 35 minutes  80/  
tcp        nginx      "/dock...jithu4"    jithu4      39 minutes ago  Up 39 minutes  80/  
tcp        nginx      "/dock...jithu5"    jithu5      47 minutes ago  Up 47 minutes  0.0  
.0.0:32770->80/tcp, [::]:32770->80/tcp  jithu6      49 minutes ago  Up 49 minutes  0.0  
.0.0:8080->80/tcp, [::]:8080->80/tcp  jithu7      About an hour ago  Up About an hour  80/  
tcp        nginx      "/dock...jithu8"    zealous_hermann  jithu8      About an hour ago  Up About an hour  80/  
ubuntu@ip-172-31-19-167:~$
```

```
ubuntu@ip-172-31-19-167:~$ docker run -dt nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
d7ecded7702a: Pull complete
266626526d42: Pull complete
320b0949be89: Pull complete
d921c57c6a81: Pull complete
9def903993e4: Pull complete
52bc359bcbd7: Pull complete
e2f8e296d9df: Pull complete
```

2. Docker Port Forwarding and Assignment

Docker allows you to map container ports to host machine ports for access from outside the container.

- Automatically assign random ports for exposed ports:

```
docker run -dt -P nginx
```

- Assign specific host port to container port:

```
docker run -dt -p 8080:80 nginx
```

- Map multiple ports:

```
docker run -dt -p 8080:80 -p 443:443 nginx
```

- Check port mappings:

```
docker ps
```

- Inspect detailed port information:

```
docker inspect <container_name> | grep Port
```

[EC2](#) > [Security Groups](#) > [sg-0e056dd52d09819fe - launch-wizard-1](#) > Edit inbound rules

Edit inbound rules [Info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info	Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info
sgr-0ce8db1c553f6f47e	SSH	TCP	22	Custom	<input type="text"/> 0.0.0.0/0 X Delete
-	All traffic	All	All	Anywh...	<input type="text"/> 0.0.0.0/0 X Delete

[Add rule](#)

⚠️ Rules with source of 0.0.0.0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. [X](#)

[Cancel](#) [Preview changes](#) [Save rules](#)

```
ubuntu@ip-172-31-19-167:~$ docker run -dt --name jithu3 -P nginx
275d9ab90adafba14941d6f016582dd8503751ab48c44e8b7959298025f560db
ubuntu@ip-172-31-19-167:~$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
 NAMES
275d9ab90ada nginx "/docker-entrypoint...." 17 seconds ago Up 16 seconds 0.0.0.0:3
2768->80/tcp, [::]:32768->80/tcp jithu3
2f88c188c9bc nginx "/docker-entrypoint...." 3 minutes ago Up 3 minutes 80/tcp
jithu2
9b1f4f86f7fb nginx "/docker-entrypoint...." 17 minutes ago Up 17 minutes 80/tcp
zealous_hermann
ubuntu@ip-172-31-19-167:~$
```

3. Docker Volumes Overview

Volumes are used to persist data beyond the container lifecycle. There are several types of volumes:

3.1 Named Volumes

- Create and attach a named volume:

```
docker run -d --name web1 -v mydata:/app/data nginx
```

- List volumes:

```
docker volume ls
```

- Inspect a specific volume:

```
docker volume inspect mydata
```

```
ubuntu@ip-172-31-19-167:~$ docker run -dt --name jithu7 -v jithu7:/app/data nginx
ff4364e1addb94a242f5c776c7fb829ca89fc209c7d21e175e92e151c5e056e4
ubuntu@ip-172-31-19-167:~$ docker volume ls
DRIVER      VOLUME NAME
local      859d4c7c1532d5e3037d2d7ce7a64431b8fe94b176c51508a5a6fa7205856b87
local      jithu7
ubuntu@ip-172-31-19-167:~$ █
```

```
ubuntu@ip-172-31-19-167:~$ docker run -dt --name jithu2 nginx
2f88c188c9bc14b98d0a2bb67ca78472b6b62c05137ff845d86e43149acca0b2
ubuntu@ip-172-31-19-167:~$ docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED             STATUS              PORTS
 NAMES
2f88c188c9bc   nginx      "/docker-entrypoint...."   8 seconds ago    Up 8 seconds     80/tcp
jithu2
0b1f4f86f7fb   nginx      "/docker-entrypoint...."   14 minutes ago   Up 14 minutes   80/tcp
ubuntu@ip-172-31-19-167:~$ █
```

3.2 Anonymous (Unnamed) Volumes

- Run a container with an unnamed volume (Docker auto-generates name):

```
docker run -d -v /app/data nginx
```

- Check generated volume name:

```
docker volume ls
```

These are automatically deleted when the container is removed (unless you use `--volumes-from` or manually retain them).

```
ubuntu@ip-172-31-19-167:~$ docker run -dt --name jithu7 -v jithu7:/app/data nginx  
ff4364e1addb94a242f5c776c7fb829ca89fc209c7d21e175e92e151c5e056e4  
ubuntu@ip-172-31-19-167:~$ docker volume ls  
DRIVER      VOLUME NAME  
local      859d4c7c1532d5e3037d2d7ce7a64431b8fe94b176c51508a5a6fa7205856b87  
local      jithu7  
ubuntu@ip-172-31-19-167:~$ █
```

3.3 Host / Bind Mount

Bind mounts allow you to use a specific directory from your host machine inside a container.

- Run a container with bind mount:

```
docker run -d --name sai-container -v /home/ubuntu/sai:/app/data nginx
```

- Any change inside /home/ubuntu/sai on host reflects inside /app/data in the container.

- Verify mount inside container:

```
docker exec -it sai-container /bin/bash  
cd /app/data  
ls
```

```
ubuntu@ip-172-31-19-167:~$ ls  
ubuntu@ip-172-31-19-167:~$ sudo mkdir sai  
ubuntu@ip-172-31-19-167:~$ cd sai/  
ubuntu@ip-172-31-19-167:~/sai$ touch sai.txt  
touch: cannot touch 'sai.txt': Permission denied  
ubuntu@ip-172-31-19-167:~/sai$ sudo touch sai.txt  
ubuntu@ip-172-31-19-167:~/sai$ ls  
sai.txt  
ubuntu@ip-172-31-19-167:~/sai$ █
```

```
ubuntu@ip-172-31-19-167:~/sai$ docker run -d --name sai-container -v /home/ubuntu/sai:/app/  
data nginx  
dfb684d6ad129c680f0e57836326c29328e234138e49e149010da534af353344  
ubuntu@ip-172-31-19-167:~/sai$ █
```

```
sai.txt
root@dfb684d6ad12:/app/data# cat sai.txt
root@dfb684d6ad12:/app/data# ls
sai.txt
root@dfb684d6ad12:/app/data# ls
sai.txt
root@dfb684d6ad12:/app/data# cat sai.txt
root@dfb684d6ad12:/app/data# cat > sai.txt
i am a bad boy
^C
root@dfb684d6ad12:/app/data# ls
sai.txt
root@dfb684d6ad12:/app/data# cat sai.txt
i am a bad boy
root@dfb684d6ad12:/app/data# exit
exit
ubuntu@ip-172-31-19-167:~/sai$ ls
sai.txt
ubuntu@ip-172-31-19-167:~/sai$ cat sai.txt
i am a bad boy
```

```
"HostConfig": {
    "Binds": [
        "/home/ubuntu/sai/:/app/data"
    ],
    "ContainerIDFile": "",
    "LogConfig": {
        "Type": "json-file",
        "Config": {}
    },
    "NetworkMode": "bridge",
    "PortBindings": {},
    "RestartPolicy": {
        "Name": "no",
        "MaximumRetryCount": 0
    },
    "AutoRemove": false,
    "VolumeDriver": "",
    "VolumesFrom": null,
    "ConsoleSize": [
        21,
        92
    ]
}
```

3.4 tmpfs Mounts

tmpfs mounts are in-memory storage locations that disappear when the container stops.

- Run a container with tmpfs mount:

```
docker run -dt --name tmp --mount type=tmpfs,target=/app/data,tmpfs-size=3m nginx
```

- Verify memory mount inside container:

```
docker exec tmp df -h /app/data
```

```
ubuntu@ip-172-31-19-167:~$ docker run -dt --name tmp --mount type=tmpfs,target=/app/data,tmpfs-size=3m nginx
d1cec291aec08b627583998b5922cdc53c872691dce6a70bbdae971dc62582e9
ubuntu@ip-172-31-19-167:~$ docker volume ls
DRIVER      VOLUME NAME
local      859d4c7c1532d5e3037d2d7ce7a64431b8fe94b176c51508a5a6fa7205856b87
local      jithu7
```

```
ubuntu@ip-172-31-19-167:~$ docker exec tmp df -h /app/data
Filesystem      Size  Used Avail Use% Mounted on
tmpfs           3.0M   0    3.0M  0% /app/data
ubuntu@ip-172-31-19-167:~$
```

4. Summary Table of Commands

Concept	Example Command
Container Naming	docker run -d --name mynginx nginx
Auto Port Forwarding	docker run -dt -P nginx
Custom Port Mapping	docker run -dt -p 8080:80 nginx
Named Volume	docker run -v mydata:/app/data nginx
Anonymous Volume	docker run -v /app/data nginx
Bind Mount	docker run -v /home/user/data:/app nginx
tmpfs Volume	docker run --mount type=tmpfs,target=/app/data,tmpfs-size=64m nginx