[**https://www.baeldung.com/ops/docker-reduce-build-context**](https://www.baeldung.com/ops/docker-reduce-build-context)

**Change Docker default directory & Image Location:**

<https://linuxconfig.org/how-to-move-docker-s-default-var-lib-docker-to-another-directory-on-ubuntu-debian-linux>

Using Docker Volumes for Mongo DB:

docker volume create mongodb

/var/lib/docker/volumes/mongodb/\_data/

**#Deploy mongodb Database**

docker run --rm -d --name mongodb -v mongodb:/data/db \

-e MONGO\_INITDB\_ROOT\_USERNAME=mongoadmin \

-e MONGO\_INITDB\_ROOT\_PASSWORD=secret \

-p 27017:27017 mongo:latest

**#Deploy mongo-express to check access to MongoDB**

docker run -d --name mongo-express \

-e ME\_CONFIG\_MONGODB\_ADMINUSERNAME=mongoadmin \

-e ME\_CONFIG\_MONGODB\_ADMINPASSWORD=secret \

-e ME\_CONFIG\_MONGODB\_URL=mongodb://mongoadmin:secret@172.17.0.2:27017/ \

-p 8081:8081 mongo-express:latest

docker exec -it mongodb bash

mongosh -u mongoadmin -p

db.helo.insertMany([

{ "\_id" : 1, "name" : "Matt", "status": "active", "level": 12, "score":202},

{ "\_id" : 2, "name" : "Frank", "status": "inactive", "level": 2, "score":9},

{ "\_id" : 3, "name" : "Karen", "status": "active", "level": 7, "score":87},

{ "\_id" : 4, "name" : "Katie", "status": "active", "level": 3, "score":27, "status": "married", "emp": "yes", "kids": 3},

{ "\_id" : 5, "name" : "Matt1", "status": "active", "level": 12, "score":202},

{ "\_id" : 6, "name" : "Frank2", "status": "inactive", "level": 2, "score":9},

{ "\_id" : 7, "name" : "Karen3", "status": "active", "level": 7, "score":87},

{ "\_id" : 8, "name" : "Katie4", "status": "active", "level": 3, "score":27, "status": "married", "emp": "yes", "kids": 3}

])

db.helo.find({name: "Katie"})

Using Docker Volumes for MySQL:

docker volume create mysqlvol

docker run --rm -dit --name mysql1 -v mysqlvol:/var/lib/mysql -e MYSQL\_ROOT\_PASSWORD=India@123456 -p 3306:3306 mysql:latest

docker exec -i mysql1 mysql -uroot -pIndia@123456 myflixdb < myflixdb .sql

Bind Mounts:

docker run --rm -dit --name mysql1 --mount type=bind,source=/bindpath,target=/var/lib/mysql -e MYSQL\_ROOT\_PASSWORD=India@123456 -p 3306:3306 mysql:latest

**Changing Docker Volumes Default Location:**

nano /lib/systemd/system/docker.service

ExecStart=/usr/bin/dockerd --data-root /mydocekrvolumes/ -H fd:// --containerd=/run/containerd/containerd.sock

sudo systemctl daemon-reload

sudo systemctl restart docker

**Using Bind Mount:**

docker run --rm -dit --name utils -v /var/run/docker.sock:/var/run/docker.sock sreeharshav/utils

**Using Bind Mount With None Network:**

docker run --rm -d --name utils -v /var/run/docker.sock:/var/run/docker.sock --network none sreeharshav/utils:latest

**Using cAdvisor with Docker Bind Mount:**

sudo docker run \

-v /:/rootfs:ro \

-v /var/run:/var/run:ro \

-v /sys:/sys:ro \

-v /var/lib/docker/:/var/lib/docker:ro \

-v /dev/disk/:/dev/disk:ro \

--publish=8080:8080 \

--detach=true \

--name=cadvisor \

google/cadvisor:latest

**Using cAdvisor as Docker Service In Global Mode:**

docker service create --name cadvisor -l prometheus-job=cadvisor \

--mode=global --publish target=8080,mode=host \

--mount type=bind,src=/var/run/docker.sock,dst=/var/run/docker.sock,ro \

--mount type=bind,src=/,dst=/rootfs,ro \

--mount type=bind,src=/var/run,dst=/var/run \

--mount type=bind,src=/sys,dst=/sys,ro \

--mount type=bind,src=/var/lib/docker,dst=/var/lib/docker,ro \

google/cadvisor -docker\_only

**In the above cAdvisor command we provided target port as 8080 but no hostport is defined. This allows docker to select a random port from 30000. Example below.**

*docker service create --name hello1 --replicas 3 --publish target=80 nginx:latest*

*root@ip-10-38-1-232:~# docker service ls | grep -i hello*

*mlaennn9ke9g hello1 replicated 3/3 nginx:latest*  ***\*:30000->80/tcp***

*f6qvz5hd051h hello2 replicated 3/3 nginx:latest*  ***\*:30001->80/tcp***

**Using both Docker volumes and bindmount With Portainer:**

docker volume create portainer\_data

docker run -d -p 8000:8000 -p 9443:9443 --name portainer \

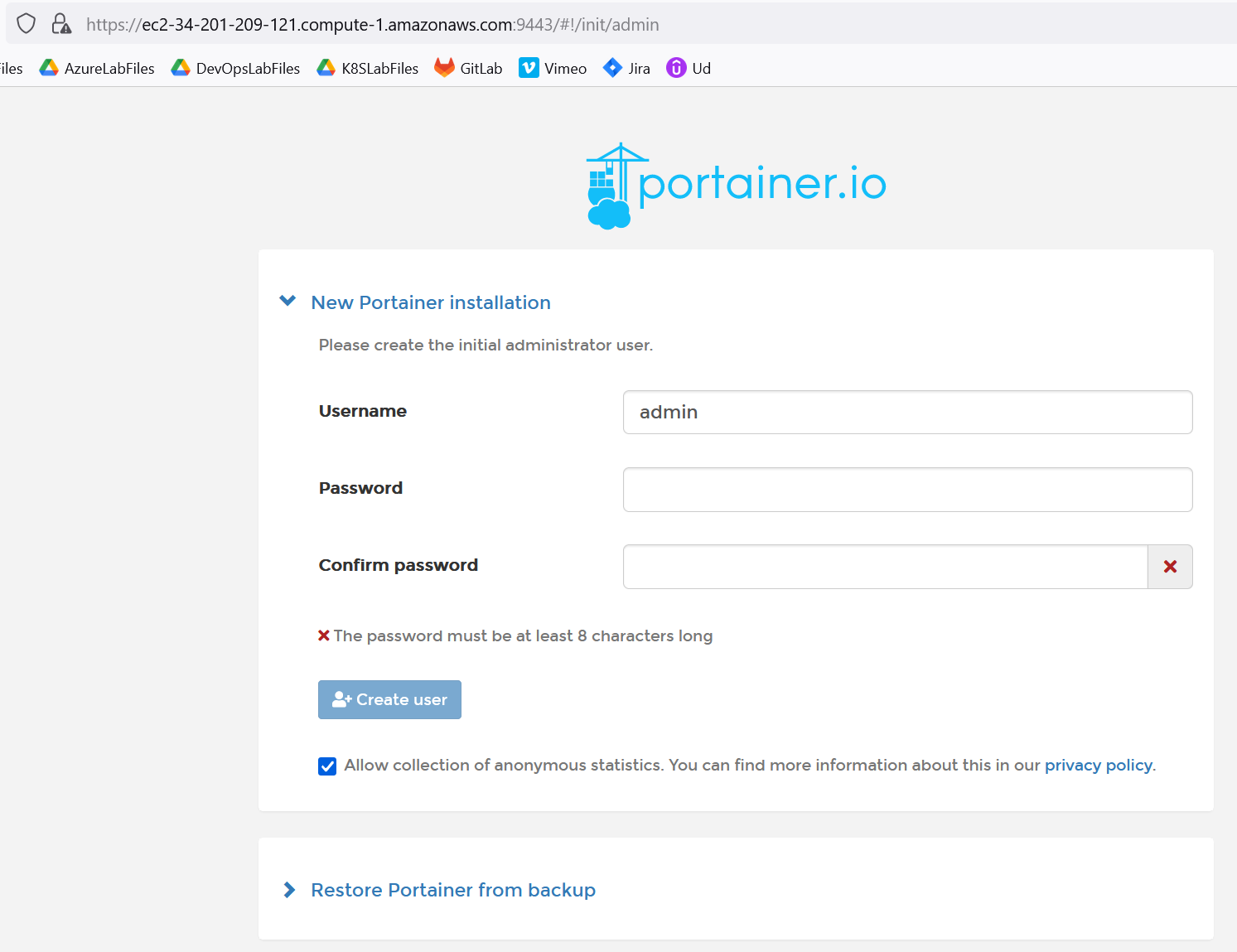
--restart=always \

-v /var/run/docker.sock:/var/run/docker.sock \

-v portainer\_data:/data \

portainer/portainer-ce:2.11.1

https://<public-ip-address-docker-host>:9443/



**Using Prometheus NodeExporter with Docker Bind Mount:**

docker run -d \

--pid="host" \

-v "/:/host:ro,rslave" \

--publish=9100:9100 \

quay.io/prometheus/node-exporter:latest \

--path.rootfs=/host

