

Vežba4

Docker

Cilj

- Upoznavanje sa DOCKER tehnologijom

1. Docker

Docker je softverska platforma koja omogućava da se brzo prave, testiraju i postavljaju aplikacije. Docker pakuje softver u kontejnere koji imaju sve što je potrebno za pokretanje uključujući biblioteke, sistemske alate, kod i vreme izvođenja. Docker se na CentOS-u instalira pomoću komade:

```
sudo yum install docker
```

Pokretanje Docker servisa na CentOS mašini se izvršava pomoću komande:

```
sudo systemctl start docker
```

Status Docker servisa na CentOS mašini moguće je saznati pomoću komade:

```
sudo systemctl status docker
```

Za definisanje jednog Docker kontejnera koristi se tekstualna datoteka **Dockerfile** gde se nalaze skripte i naredbe. Kontejneri su izolovani okviri koji omogućavaju pokretanje aplikacija i servisa nezavisno od okruženja na host-u.

Osnovne strukture Dockerfile-a:

- **FROM:** Ova naredba određuje osnovnu Docker sliku na kojoj će se bazirati vaš kontejner. Na primer, FROM ubuntu:latest će koristiti najnoviju verziju Ubuntu slike kao osnovu.
- **WORKDIR:** Ova naredba postavlja radni direktorijum unutar kontejnera u koji će se smestiti ostatak fajlova i izvršavati komande.
- **COPY** ili **ADD:** Služe da kopiraju fajlove iz lokalnog sistema (host) u kontejner.
- **RUN:** Ova naredba pokreće komande unutar kontejnera u toku faze izgradnje. Npr, RUN yum update && yum install -y httpd će ažurirati listu paketa i instalirati paket httpd.
- **CMD:** Definiše komandu koja se pokreće kada se kontejner pokrene. Npr. komanda CMD ["python", "app1.py"] pokreće Python aplikaciju koristeći **python app1.py**. Često se servisi pokreću u prvom planu sa opcijom "-D foreground", izlazne poruke (logovi), će biti direktno vidljivi u konzoli. - **CMD ["xxx", "-D", "BACKGROUND"]**

- **EXPOSE:** se koristi za označavanje portova na kojima će aplikacija unutar Docker kontejnera slušati. Naredba ne otvara portove niti ih čini dostupnim spoljašnjem svetu već samo dokumentuje koji portovi će biti korišćeni od strane aplikacije unutar kontejnera.
- **MAINTAINER:** se koristi za označavanje autora ili osobe koja održava Docker sliku. Sve manje se koristi od verzije 1.13 Docker. Npr. MAINTAINER Ime Prezime <email@>.

Dockerfile može sadržava i druge naredbe i opcije koje se koriste za specifične potrebe aplikacije ili servisa koji se pokreće unutar kontejnera.

Komande

Komanda **docker build** služi za izgradnju Docker kontejnera na osnovu specifikacija definisanih u Dockerfile-u, navodi se putanja do Dockerfile tj. direktorijum gde se nalazi.

Osnovni oblik docker build komande: **docker build [OPCIJE] PUTANJA_DO_DOCKERFILE**

Primer (ako se Dockerfile u trenutnom direktorijumu): **docker build -t ime-slike .**

Komanda **docker run** komanda se koristi za pokretanje novog Docker kontejnera na osnovu određene Docker slike. Ova komanda omogućava kreiranje i pokretanje kontejnera sa specifičnim opcijama i konfiguracijama.

Osnovni oblik docker run komande: **docker run [OPCIJE] IME_SLIKE [KOMANDA] [ARGUMENTI]**

OPCIJE: Ovo su opcionalni parametri koji se koriste za konfigurisanje ponašanja kontejnera. Ove opcije mogu uključivati opcije kao što su mapiranje portova, postavljanje okruženja, montiranje volumena, ograničavanje resursa i druge. Na primer, -p se koristi za mapiranje portova, -v se koristi za montiranje volumena, -e se koristi za postavljanje okruženjskih varijabli, itd.

IME_SLIKE: Ovo je ime Docker slike na osnovu koje se kreira kontejner. Ako slika nije dostupna lokalno, Docker će je automatski preuzeti sa Docker Hub-a ili drugog registra slika ako je javno dostupna.

KOMANDA (opciono): Ovo je komanda koja će se izvršiti unutar kontejnera nakon pokretanja. Ako nije navedena, pokreće se podrazumevana komanda definisana u Dockerfile-u slike.

ARGUMENTI (opciono): Ovo su argumenti koji se prosleđuju komandi unutar kontejnera.

Primer: **docker run -it ubuntu /bin/bash**

Ova komanda će pokrenuti novi kontejner na osnovu Ubuntu slike i otvoriti interaktivni bash terminal unutar tog kontejnera. -it opcija se koristi za pokretanje interaktivnog terminala.

Komanda **docker ps** se koristi za prikaz trenutno pokrenutih Docker kontejnera na sistemu. Kada se izvrši bez ikakvih dodatnih opcija, ova komanda prikazuje osnovne informacije o pokrenutim kontejnerima, kao što su njihovi ID, imena, statusi, pokrenuti procesi unutar njih, vreme pokretanja i portovi koji su mapirani između kontejnera i host sistema. Sve kontejnere na sistemu, uključujući i one koji su završeni, prikazuju se dodavanjem opcije -a (ili --all).

Osnovni oblik komande docker ps prikazuje samo aktivne kontejnere. Ako želite videti sve kontejnere na sistemu, uključujući i one koji su završeni, možete koristiti opciju -a (ili --all).

Komada **docker exec** se koristi za izvršavanje komandi unutar postojećeg Docker kontejnera. Ova komanda omogućava korisnicima da komuniciraju sa pokrenutim kontejnerom, izvršavajući komande unutar njega, pregledajući stanje, debugirajući probleme ili održavajući aplikacije koje se izvršavaju unutar kontejnera.

Osnovni oblik: **docker exec [OPCIJE] KONTEJNER KOMANDA [ARGUMENTI]**

OPCIJE su opcionalni parametri koji se koriste za konfigurisanje izvršavanja komande unutar kontejnera.

KONTEJNER je ime ili ID kontejnera unutar kojeg želite izvršiti komandu. (docker ps)

KOMANDA je komanda koju želite izvršiti unutar kontejnera.

ARGUMENTI su argumenti koji se prosleđuju komandi unutar kontejnera.

Primer: **docker exec moj-kontejner ls -l** (lista sadžaj iz kontejnera)

Zadatak 0: Pokrenuti CentOS mašinu. [\\fs\cloud\CentOS.7Z](https://fs.cloud/CentOS.7Z)

Zadatak 1: Instalirati paket **docker**. Proveriti status servisa docker. Pokrenuti servis ukoliko nije aktivan i postaviti da bude trajno aktivan.

Zadatak 2: U radnom direktorijumu po želji kreirati stranu **index.html**.

Zadatak 3: U radnom direktorijumu kreirati Dockerfile koji će služiti za kreiranje kontejnera sa sledećim karakteristikama: Docker sliku koja se zasniva na CentOS Linux operativnom sistemu verzije 7, ažurira listu paketa i instalira paket httpd, kopira lokalni fajl (Zadatak 2) u kontejner na lokaciju /var/www/html i eksponira port 80 unutar kontejnera.

Zadatak 4: Pomoću **docker build** izgraditi kontejner na osnovu kreiranog Dockerfile, koristiti naziv za sliku **centos-apache**.

Zadatak 5: Pomoću komande **docker run** pokrenuti Docker kontejnera na osnovu Docker slike iz prethodnog zadatka. Mapirati port između host sistema i Docker kontejnera 80 -> 80. Koristiti opcije -itd.

Zadatak 6: Izlistati trenutno pokrenute Docker kontejnere na sistemu.

Zadatak 7: Obezbediti da je port 80 dostupan na VM. Pristupiti iz browser-a (Ip adresa VM). Koja je strana dostupna?

Zadatak 8: Pristupiti kreiranom kontejneru pomoću komande **docker exec** (sa opcijama -it, komanda: /bin/bash - da se buduće komande direktno izvršavaju u kontejneru). Pristupiti logovima apache web servera.

Zadatak 9: U kontejneru pomoću **Vi** editora izmeniti sadržaj stranice **/var/www/htm/index.html**. Pristupiti izmenjenoj stranici iz browser-a (Ip adresa VM). Koristiti komandu **exit** za povratak iz kontejnera.

Zadatak 10: Na CentOS-u kreirati direktorijum **/root/docker2** i u njemu **Dockerfile** koji će služiti za kreiranje kontejnera sa sledećim karakteristikama: Docker sliku koja se zasniva na Ubuntu Linux operativnom sistemu verzije 16:04, ažurira listu paketa i instalira paket apache2, instalira nano editor i eksponira port 80 unutar kontejnera.

Zadatak 11: Pomoću **docker build** izgraditi kontejner na osnovu kreiranog Dockerfile, koristiti naziv za sliku **ubunt-apache**.

Zadatak 12: Pomoću komande **docker run** pokrenuti Docker kontejnera na osnovu Docker slike iz prethodnog zadatka. Mapirati port između host sistema i Docker kontejnera 81 -> 80. Koristiti opcije -itd.

Zadatak 13: Izlistati trenutno pokrenute Docker kontejnere na sistemu.

Zadatak 14: Obezbediti da je port 81 dostupan na VM. Pristupiti iz browser-a (Ip adresa VM i port 81).
Koja je strana dostupna?

Zadatak 15: Pristupiti kreiranom kontejneru pomoću komande **docker exec**. Da li je dostupan nano editor? Izmeniti sadržaj stranice **/var/www/htm/index.html**. Testirati.