

HA Redis cluster

Za implementaciju high availability Redis clustera odabrala sam master-slave setup sa Sentinel nodeovima.

Cluster podnosi gubitak najviše jednog nodea bez gubitaka podataka.

Master i slave nodeovi oboje slušaju na portu 6379, koji je unutar Docker-a mapiran na sistemski port 6379.

Deployane su 3 Sentinel instance, s obzirom da je to preporučeni broj za robustan setup, iako u konkretnom slučaju sa dva redis-server nodea nije prijeko potrebno imati više instanci.

Konfiguracije Redis servera i Sentinela su vrlo jednostavne i bazne, bez dodatnih optimizacijskih parametara, primjerice:

```
redis-slave:
  container_name: redis-slave
  image: redis:latest
  command: redis-server --slaveof redis-master 6379
  volumes:
    - ./app

sentinel-1:
  container_name: sentinel-1
  build: sentinel
  environment:
    - SENTINEL_DOWN_AFTER=5000
    - SENTINEL_FAILOVER=5000
```

Python aplikacija

Kao proizvoljne vrijednosti za upisivanje u Redis odabrani su timestamp i random broj izgeneriran uz pomoć random library-a.

Python komunicira sa Redisom preko objekta vraćenog pozivom na Sentinel.

Primarno sam implementirala komunikaciju direktno s Redisom, međutim postojalo je par problema prilikom failoveranja na slave kada je master nedostupan.

Python se spaja na jedan od tri Sentinel nodea, dobiva informaciju o trenutnom masteru, te zapisuje podatke na trenutni master.

Za upis podataka odabrala sam for petlju u 1000 iteracija.

Razmatrala sam while petlju, međutim odabrala sam for zbog bolje predvidljivosti i konačnosti, a količina upisanih podataka je i ovako dovoljna da se izgenerira demonstrativan load na clusteru.

Grafana/monitoriranje

Kao datasource Grafani odabrala sam Prometheus. Prometheus je konfiguriran da scrapea metrike iz cAdvisora (koji agregira, procesira i exportira metrike o containerima).

Od Grafana verzije 5.0, novi API/provisioning sustav je dostupan, stoga sam koristila datasource.yml za Grafanu, kako bi se generirao defaultni Prometheus datasource.

Također, koristeći provisioning API, inicijalno kreirani dashboard je exportan u provisioning/dashboards direktorij, te se loada prilikom kreiranja containera.