ENGINEERING BLOG

DOCKER DYNAMIC PORT

admin July 18, 2019 Leave a comment Edit

REQUEST CYCLE (earlier)
ALB > OCTOPUS > EUREKA > HOST > CONTAINER > CONTROLLER

In production we run multiple tasks in a single instance. In order to determine the port mapping for a container with its host (useful while registering to eureka), following approach has been taken.

1. AWS Task Definition has port mapping as follows : ***Host Port : 0, Container Port : 8080***
2. Provide ***volume mount*** of host machine docker socket to container
3. ECS Agent respects this configuration and assigns a random available ephemeral port for configuration. In order to fetch the exact details of the configured port, we run following commands as part of the base docker file

- ***export CONTAINER_ID=\$(cat /etc/hostname)***
- ***export CONTAINER_PORT=\$(docker port \$CONTAINER_ID | awk 'END {print \$NF}' | awk -F'0.0.0.0:' '{print \$2}')***

The first command when run inside the container fetches container-id and exports it as system variable CONTAINER_ID and the second fetches configured dynamic port and exports it as a system variable
CONTAINER_PORT

As part of the Application.java in each service following shall be performed to register to eureka with the fetched port.

```
@Bean("eurekaInstanceConfig")
        @Profile("docker")
        public EurekaInstanceConfigBean
eurekaInstanceConfig(InetUtils inetUtils) {
                System.out.println("in
eurekaInstanceConfig");
                EurekaInstanceConfigBean config = new
EurekaInstanceConfigBean(inetUtils);
                AmazonInfo info =
AmazonInfo.Builder.newBuilder().autoBuild("eureka");
                config.setDataCenterInfo(info);

info.getMetadata().put(MetaDataKey.localHostname.getName(),
info.get(AmazonInfo.MetaDataKey.localIpv4));

config.setHostname(info.get(AmazonInfo.MetaDataKey.localHost
name));

config.setIpAddress(info.get(AmazonInfo.MetaDataKey.localIpv
4));
                System.out.println("in  eurekaInstanceConfig
" + config.getHostname() + " " + config.getIpAddress());
                String port =
System.getenv("CONTAINER_PORT");
                if (port != null && !port.trim().equals(""))
{

config.setNonSecurePort(Integer.parseInt(port));
                        System.out.println("Eureka port set
to :" + port);
                } else {
                        config.setNonSecurePort(80);
                        System.out.println("Eureka port set
to :" + port);
```

```
                }

config.setInstanceId(info.get(MetaDataKey.instanceId)+"_"+co
nfig.getNonSecurePort());
                return config;
        }
```

Leave a comment

Comment

Post Comment