

Enabling IPV6

Configure the docker daemon to use IPV6

The following two parameters shall be added to the docker engine json configuration.

For example via the Docker desktop, preferences.

The address range 2001.... below is just an example. Containers will be allocated addresses starting with this address.

```
"ipv6": true,  
"fixed-cidr-v6": "2001:db8:13b:330:ffff::/80"
```

Create a docker private network with IPV6 enabled

Use the following command to create a private docker network

```
docker network create --ipv6 <network_name>
```

Adapting container images for IPV6

For example, Java Spring Boot apps seem to already listen on http requests on IPV6.

However, python apps need to be configured to listen to IPV6. If accepting calls from any IP then a python server shall listen the ip "::" - which mean any ipv6 address (and any ipv4 address as well).

Example python server start:

```
app.run(port=HOST_PORT, host="::")
```

Start containers using a docker IPV6 network

Start containers using the IPV6 enabled private network. Same procedure as for an IPV4 network.

Note: Containers may still IPV4 addresses if that is what the docker internal host lookup returns.

Example from tests

Snippet from a 'docker inspect <network-name>' when running an IPV6 enabled network. The containers in the network are assigned IPV6 addresses from the address range given in the configuration in the docker daemon.

Note that IPV4 addresses are also assigned since the network also support IPV4.

```

"Containers": {
  "085e6193e716d14b9d3653a591720c3d9463617e5731638130cc59b47414c8cd": {
    "Name": "policy-agent",
    "EndpointID": "18a7a0fdbbfc47a3bb45f0571870e3b6fe2fb0d958f949a4eb74d752039a05083",
    "MacAddress": "02:42:ac:14:00:0c",
    "IPv4Address": "172.20.0.12/16",
    "IPv6Address": "2001:db8:13b:330:dd::b/80"
  },
  "133a9bff0e850dd6bc2c28f2fb724d2f36cbfeb936f249b40e238883a126166d": {
    "Name": "control-panel",
    "EndpointID": "2642d9106e876def5482ebad1462ceac9f0be0017ffb0c743b99e84c8cf0607b",
    "MacAddress": "02:42:ac:14:00:09",
    "IPv4Address": "172.20.0.9/16",
    "IPv6Address": "2001:db8:13b:330:dd::8/80"
  },
  "2c439c5ce7f23350bc54af7c3f728d72ab84e06d28d976c5c282d8609c92dea3": {
    "Name": "ricsim_g2_1",
    "EndpointID": "e657b1091faa4d89943d3af7cdb0e7e31de6b0a339a28d6e28ee2b800df7c6c9",
    "MacAddress": "02:42:ac:14:00:03",
    "IPv4Address": "172.20.0.3/16",
    "IPv6Address": "2001:db8:13b:330:dd::3/80"
  },
  "430592191e76115b920aab5ca438764b1e5d5fe9a87c23d367c299e1cfff8c8e": {
    "Name": "message-router",
    "EndpointID": "d39db4a82cc138d1adc668a67627d044e03e4806a7dac28368b3ba2f2c06e252",
    "MacAddress": "02:42:ac:14:00:04",
    "IPv4Address": "172.20.0.4/16",
    "IPv6Address": "2001:db8:13b:330:dd::4/80"
  }
},

```

Making a curl using an IPV6 address, to container ricsim_g2_1 from the list above

```
curl http://[2001:db8:13b:330:dd::3]:8085/
```

OK

In the server the IPV6 address of the client shows up, proving IPV6 connectivity.

```
2001:db8:13b:330:dd::7 - - [27/Apr/2020 14:35:08] "GET / HTTP/1.1" 200 -
```