**Bridge networking using user defined bridge networks**

1.

Structure

Start 2 alphine containers - connected to user created network alpine-net

3rd alpine container - connected to bridge container(default)

4th alpine container connected to both networks

2.

Create a user defined network called alpine-net

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker network create --driver bridge alpine-net

467d76d368b9c6d91dac4fc395ed4571c4f6b31d0f59dcb86da4cbb62a02fef3

You don’t need the flag --driver bridge since it is the default

3.

List all docker networks

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker network ls

NETWORK ID NAME DRIVER SCOPE

467d76d368b9 alpine-net bridge local

1f8ada3b2cce bridge bridge local

1ba4df7d51a1 host host local

adbff08e1d50 none null local

4.

Inspect the alpine-net network

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker network inspect alpine-net

[

{

"Name": "alpine-net",

"Id": "467d76d368b9c6d91dac4fc395ed4571c4f6b31d0f59dcb86da4cbb62a02fef3",

"Created": "2019-01-12T17:01:15.656983335+05:30",

"Scope": "local",

"Driver": "bridge",

"EnableIPv6": false,

"IPAM": {

"Driver": "default",

"Options": {},

"Config": [

{

"Subnet": "172.18.0.0/16",

"Gateway": "172.18.0.1" → network’s gateway ip address

}

]

},

"Internal": false,

"Attachable": false,

"Ingress": false,

"ConfigFrom": {

"Network": ""

},

"ConfigOnly": false,

"Containers": {},

"Options": {},

"Labels": {}

}

]

No containers connected to this network

5.

Create & connect the containers to different networks

Alpine1 → alpine-net

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker run -dit --name alpine1 --network alpine-net alpine ash

E8233a9d2050464d5f4f4954ba013f49a883f65333bc1a434c44561ee9bb798c

Alpine2 → alpine-net

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker run -dit --name alpine2 --network alpine-net alpine ash

a7556d47eba4b3f0f3acad798fe2b4d42c32900fb296df1fafcbb5722e14ddb7

Alpine3 → default bridge

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker run -dit --name alpine3 alpine ash

C61465a09ed0738f77651d52c1e172c15f0d18bd3b4de587cc25f3f65557b551

Alpine4 → alpine-net

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker run -dit --name alpine4 --network alpine-net alpine ash

5f90f82ddc5f182288bc09991a8e392b8005bce861bce1d65baabea2326cc5b0

Alpine4 → default bridge

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker network connect bridge alpine4

6.

Check whether the containers are running

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker container ls

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

5f90f82ddc5f alpine "ash" 3 minutes ago Up 3 minutes alpine4

c61465a09ed0 alpine "ash" 4 minutes ago Up 4 minutes alpine3

a7556d47eba4 alpine "ash" 5 minutes ago Up 5 minutes alpine2

e8233a9d2050 alpine "ash" 6 minutes ago Up 6 minutes

7.

Inspect the networks

Inspect default bridge network

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker network inspect bridge

[

{

"Name": "bridge",

"Id": "1f8ada3b2ccefcc2fed6351d87a5453545725377fbabbd57daf620ca732413a7",

"Created": "2019-01-12T12:42:49.223610422+05:30",

"Scope": "local",

"Driver": "bridge",

"EnableIPv6": false,

"IPAM": {

"Driver": "default",

"Options": null,

"Config": [

{

"Subnet": "172.17.0.0/16",

"Gateway": "172.17.0.1"

}

]

},

"Internal": false,

"Attachable": false,

"Ingress": false,

"ConfigFrom": {

"Network": ""

},

"ConfigOnly": false,

"Containers": {

"5f90f82ddc5f182288bc09991a8e392b8005bce861bce1d65baabea2326cc5b0": {

"Name": "alpine4",

"EndpointID": "363b282cf30583ba54420f4c49994c66914de927a68ad2e6123695c1468ddabf",

"MacAddress": "02:42:ac:11:00:03",

"IPv4Address": "172.17.0.3/16",

"IPv6Address": ""

},

"c61465a09ed0738f77651d52c1e172c15f0d18bd3b4de587cc25f3f65557b551": {

"Name": "alpine3",

"EndpointID": "f63d15b5b3607f83b25253989bc6c32e1e27b223f9198dbee639c191ef8117b9",

"MacAddress": "02:42:ac:11:00:02",

"IPv4Address": "172.17.0.2/16",

"IPv6Address": ""

}

},

"Options": {

"com.docker.network.bridge.default\_bridge": "true",

"com.docker.network.bridge.enable\_icc": "true",

"com.docker.network.bridge.enable\_ip\_masquerade": "true",

"com.docker.network.bridge.host\_binding\_ipv4": "0.0.0.0",

"com.docker.network.bridge.name": "docker0",

"com.docker.network.driver.mtu": "1500"

},

"Labels": {}

}

]

Alpine3 & alpine4 are connected to the default bridge network

Inspect alpine-net user defined bridge network

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker network inspect alpine-net

[

{

"Name": "alpine-net",

"Id": "467d76d368b9c6d91dac4fc395ed4571c4f6b31d0f59dcb86da4cbb62a02fef3",

"Created": "2019-01-12T17:01:15.656983335+05:30",

"Scope": "local",

"Driver": "bridge",

"EnableIPv6": false,

"IPAM": {

"Driver": "default",

"Options": {},

"Config": [

{

"Subnet": "172.18.0.0/16",

"Gateway": "172.18.0.1"

}

]

},

"Internal": false,

"Attachable": false,

"Ingress": false,

"ConfigFrom": {

"Network": ""

},

"ConfigOnly": false,

"Containers": {

"5f90f82ddc5f182288bc09991a8e392b8005bce861bce1d65baabea2326cc5b0": {

"Name": "alpine4",

"EndpointID": "279cd41016e5acecec9ffd75a5632c7265960c25dd342466312d13500f79c688",

"MacAddress": "02:42:ac:12:00:04",

"IPv4Address": "172.18.0.4/16",

"IPv6Address": ""

},

"a7556d47eba4b3f0f3acad798fe2b4d42c32900fb296df1fafcbb5722e14ddb7": {

"Name": "alpine2",

"EndpointID": "03b36f551d3b5085ec427aa6b31ac5993672b03457c4ec0c59bcfb49c0d5400b",

"MacAddress": "02:42:ac:12:00:03",

"IPv4Address": "172.18.0.3/16",

"IPv6Address": ""

},

"e8233a9d2050464d5f4f4954ba013f49a883f65333bc1a434c44561ee9bb798c": {

"Name": "alpine1",

"EndpointID": "967dac088fe4cc3e6619a4126843976a3c8766e014a333ddd494c79e61b16260",

"MacAddress": "02:42:ac:12:00:02",

"IPv4Address": "172.18.0.2/16",

"IPv6Address": ""

}

},

"Options": {},

"Labels": {}

}

]

alpine 1 , alpine2 & alpine4 are connected to alpine-net user defined bridge network

8.

**Automatic service discovery**

User defined networks like alpine-net can communicate by both using their ip addresses & resolving container name in to the ip address

Connect to alpine1

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker container attach alpine1

/ #

alpine 1 , alpine2 & alpine4 are connected to alpine-net user defined bridge network

Connect to alpine2

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker container attach alpine1

/ # ping -c 2 alpine2

PING alpine2 (172.18.0.3): 56 data bytes

64 bytes from 172.18.0.3: seq=0 ttl=64 time=0.150 ms

64 bytes from 172.18.0.3: seq=1 ttl=64 time=0.087 ms

--- alpine2 ping statistics ---

2 packets transmitted, 2 packets received, 0% packet loss

round-trip min/avg/max = 0.087/0.118/0.150 ms

/ #

Connect to alpine4

/ # ping -c 2 alpine4

PING alpine4 (172.18.0.4): 56 data bytes

64 bytes from 172.18.0.4: seq=0 ttl=64 time=0.075 ms

64 bytes from 172.18.0.4: seq=1 ttl=64 time=0.052 ms

--- alpine4 ping statistics ---

2 packets transmitted, 2 packets received, 0% packet loss

round-trip min/avg/max = 0.052/0.063/0.075 ms

/ #

Connect to alpine1 (itself)

/ # ping -c 2 alpine1

PING alpine1 (172.18.0.2): 56 data bytes

64 bytes from 172.18.0.2: seq=0 ttl=64 time=0.029 ms

64 bytes from 172.18.0.2: seq=1 ttl=64 time=0.038 ms

--- alpine1 ping statistics ---

2 packets transmitted, 2 packets received, 0% packet loss

round-trip min/avg/max = 0.029/0.033/0.038 ms

/ #

9.

/ # ping -c 2 alpine3

ping: bad address 'alpine3'

/ #

Can’t resolve the name

Alpine3 is in the default bridge network & you are in the alpine1 which is in the alpine-net user defined network

You can’t connect using the ip address even because it is in a different network

/ # ping -c 2 172.17.0.2

PING 172.17.0.2 (172.17.0.2): 56 data bytes

--- 172.17.0.2 ping statistics ---

2 packets transmitted, 0 packets received, 100% packet loss

/ #

10.

detach from alpine1

CTRL p,q

11.

Connect to alpine4

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker container attach alpine4

/ #

Alpine4 is connected to both default & user defined bridge alpine-net networks

So all the containers can be accessed with in this container.

Only the alpine3 should be connected using it’s ip address since it is in the default bridge network

Others can be accessed by using their names

Connect to Alpine 1

/ # ping -c 2 alpine1

PING alpine1 (172.18.0.2): 56 data bytes

64 bytes from 172.18.0.2: seq=0 ttl=64 time=0.087 ms

64 bytes from 172.18.0.2: seq=1 ttl=64 time=0.093 ms

--- alpine1 ping statistics ---

2 packets transmitted, 2 packets received, 0% packet loss

round-trip min/avg/max = 0.087/0.090/0.093 ms

/ #

Connect to Alpine 2

/ # ping -c 2 alpine2

PING alpine2 (172.18.0.3): 56 data bytes

64 bytes from 172.18.0.3: seq=0 ttl=64 time=0.118 ms

64 bytes from 172.18.0.3: seq=1 ttl=64 time=0.090 ms

--- alpine2 ping statistics ---

2 packets transmitted, 2 packets received, 0% packet loss

round-trip min/avg/max = 0.090/0.104/0.118 ms

/ #

Connect to Alpine 4 (itself)

/ # ping -c 2 alpine4

PING alpine4 (172.18.0.4): 56 data bytes

64 bytes from 172.18.0.4: seq=0 ttl=64 time=0.048 ms

64 bytes from 172.18.0.4: seq=1 ttl=64 time=0.068 ms

--- alpine4 ping statistics ---

2 packets transmitted, 2 packets received, 0% packet loss

round-trip min/avg/max = 0.048/0.058/0.068 ms

/ #

Connect to Alpine 3

/ # ping -c 2 alpine3

ping: bad address 'alpine3'

/ #

/ # ping -c 2 172.17.0.2

PING 172.17.0.2 (172.17.0.2): 56 data bytes

64 bytes from 172.17.0.2: seq=0 ttl=64 time=0.124 ms

64 bytes from 172.17.0.2: seq=1 ttl=64 time=0.090 ms

--- 172.17.0.2 ping statistics ---

2 packets transmitted, 2 packets received, 0% packet loss

round-trip min/avg/max = 0.090/0.107/0.124 ms

/ #

12.

Check whether all the containers can connect to internet

From alpine4

/ # ping -c 2 google.com

PING google.com (216.58.203.206): 56 data bytes

64 bytes from 216.58.203.206: seq=0 ttl=54 time=67.334 ms

64 bytes from 216.58.203.206: seq=1 ttl=54 time=71.447 ms

--- google.com ping statistics ---

2 packets transmitted, 2 packets received, 0% packet loss

round-trip min/avg/max = 67.334/69.390/71.447 ms

/ #

CTRL p,q

From alpine3

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker container attach alpine3

/ #

/ # ping -c 2 google.com

PING google.com (172.217.167.174): 56 data bytes

64 bytes from 172.217.167.174: seq=0 ttl=54 time=64.542 ms

64 bytes from 172.217.167.174: seq=1 ttl=54 time=52.055 ms

--- google.com ping statistics ---

2 packets transmitted, 2 packets received, 0% packet loss

round-trip min/avg/max = 52.055/58.298/64.542 ms

/ #

CTRL p,q

From alpine1

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker container attach alpine1

/ #

/ # ping -c 2 google.com

PING google.com (172.217.167.174): 56 data bytes

64 bytes from 172.217.167.174: seq=0 ttl=54 time=344.052 ms

64 bytes from 172.217.167.174: seq=1 ttl=54 time=69.334 ms

--- google.com ping statistics ---

2 packets transmitted, 2 packets received, 0% packet loss

round-trip min/avg/max = 69.334/206.693/344.052 ms

/ #

CTRL p,q

13.

Stop & remove all the containers

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker container stop alpine1 alpine2 alpine3 alpine4

alpine1

alpine2

alpine3

alpine4

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker container rm alpine1 alpine2 alpine3 alpine4

alpine1

alpine2

alpine3

alpine4

14.

Remove the alpine-net network

sarala@sarala-pc:~/SARALA/BL/spark-2.4docker/docker-spark-master$ sudo docker network rm alpine-net

alpine-net