

LAPORAN PRAKTIKUM VIRTUALISASI KOMPUTER

DOCKER NETWORKS



Agus Pranata Marpaung

13323033

DIII TEKNOLOGI KOMPUTER

**INSTITUT TEKNOLOGI DEL
FAKULTAS VOKASI**

Judul Praktikum

Minggu/Sesi	:	XI/3
Kode Mata Kuliah	:	1032101
Nama Mata Kuliah	:	VIRTUALISASI KOMPUTER
Setoran	:	Jawaban dalam bentuk <i>softcopy</i>
Batas Waktu Setoran	:	<i>11 November 2024 jam 21:30</i>
Tujuan	:	1. Mahasiswa mampu membuat dan mengkonfigurasi jaringan antar container pada Docker

Petunjuk

Docker Networks

Teori

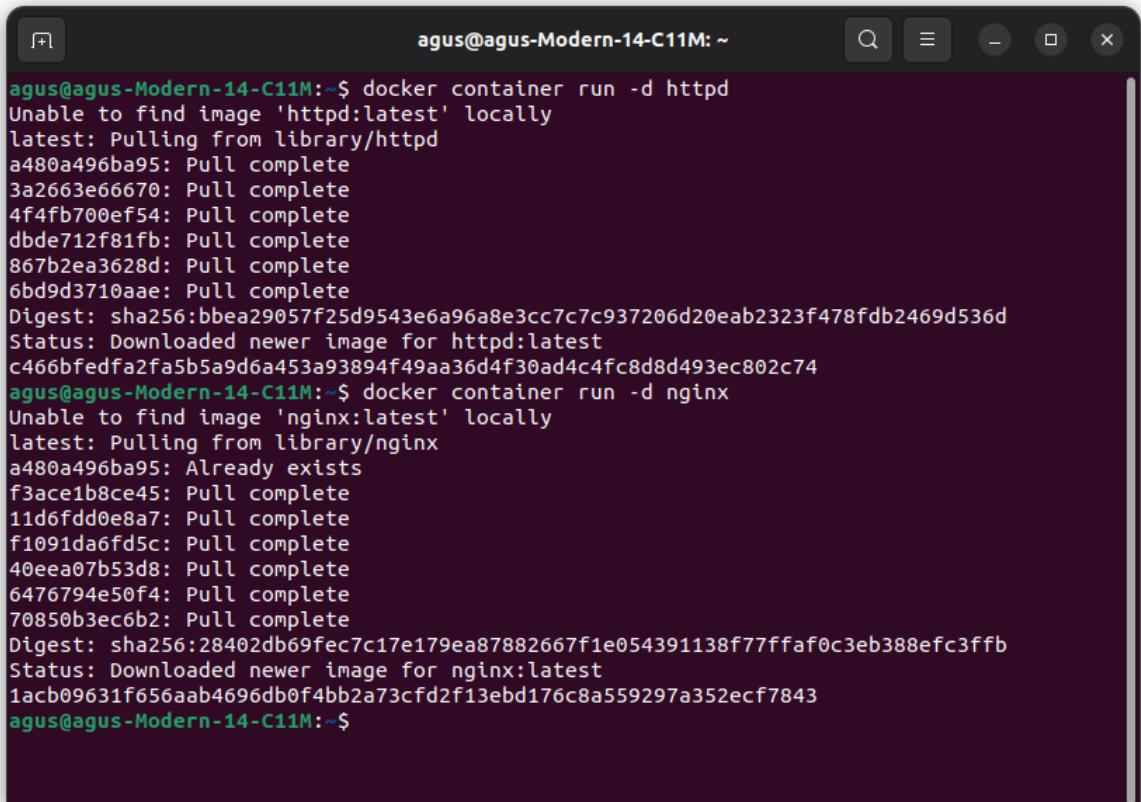
4 dari 6 Tipe Jaringan yang tersedia oleh Docker.

- **bridge**, jenis ini adalah jaringan default. Docker menghubungkan container ke jaringan ini secara default, kecuali memilih opsi driver yang berbeda saat menjalankan docker run -- network = <NETWORK>.
- **none**, container tidak memiliki antarmuka jaringan.
- **host**, container yang berada di jaringan host. Tidak ada batas antara container dan host. Port yang ada yang dibuka di host dapat diakses langsung oleh container, dan sebaliknya.
- **macvlan**, jaringan ini memungkinkan pembuatan alamat MAC (alamat jaringan fisik).

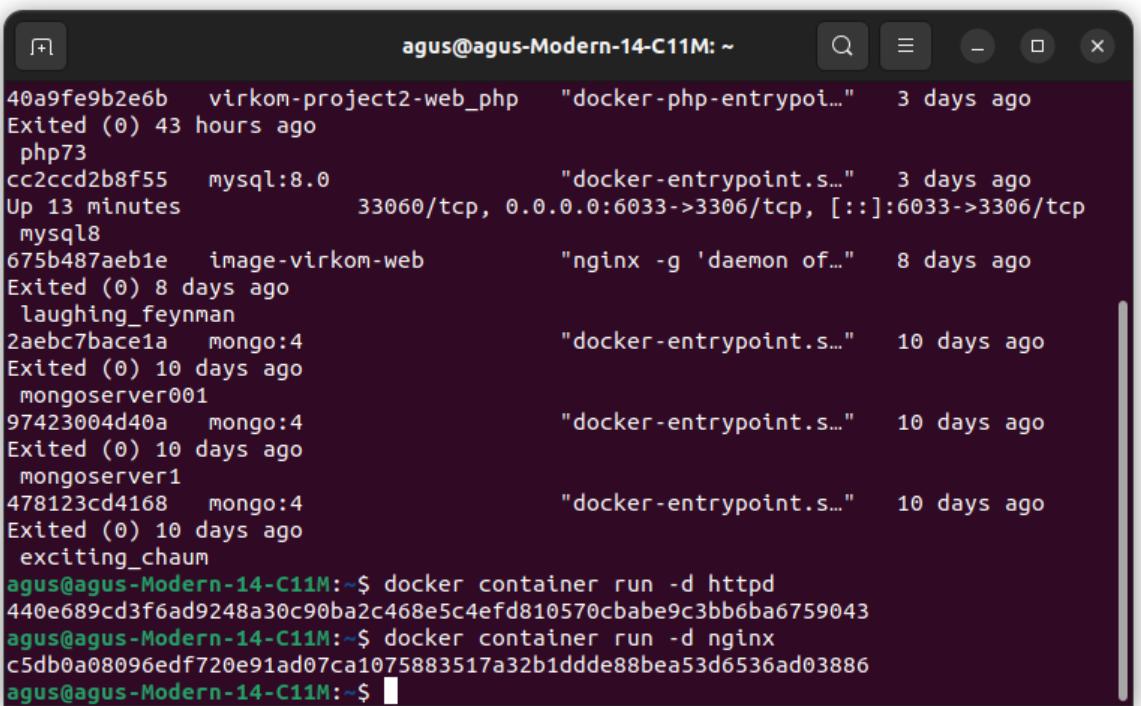
Praktek

Dapatkan container berkomunikasi satu sama lain? Mari kita cari tahu.

1. Jalankan nginx dan httpd pada container

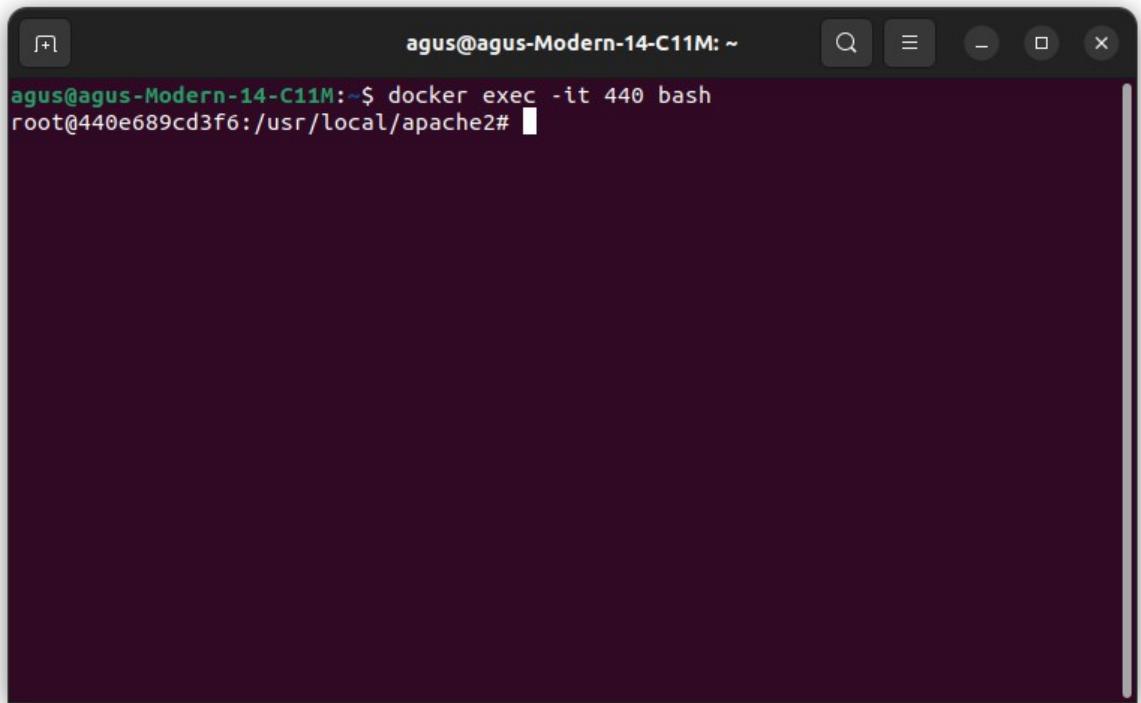


```
agus@agus-Modern-14-C11M:~$ docker container run -d httpd
Unable to find image 'httpd:latest' locally
latest: Pulling from library/httpd
a480a496ba95: Pull complete
3a2663e66670: Pull complete
4f4fb700ef54: Pull complete
dbde712f81fb: Pull complete
867b2ea3628d: Pull complete
6bd9d3710aae: Pull complete
Digest: sha256:bbea29057f25d9543e6a96a8e3cc7c7c937206d20eab2323f478fdb2469d536d
Status: Downloaded newer image for httpd:latest
c466bfedfa2fa5b5a9d6a453a93894f49aa36d4f30ad4c4fc8d8d493ec802c74
agus@agus-Modern-14-C11M:~$ docker container run -d nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
a480a496ba95: Already exists
f3ace1b8ce45: Pull complete
11d6fdd0e8a7: Pull complete
f1091da6fd5c: Pull complete
40eea07b53d8: Pull complete
6476794e50f4: Pull complete
70850b3ec6b2: Pull complete
Digest: sha256:28402db69fec7c17e179ea87882667f1e054391138f77ffaf0c3eb388efc3ffb
Status: Downloaded newer image for nginx:latest
1acb09631f656aab4696db0f4bb2a73cf2f13ebd176c8a559297a352ecf7843
agus@agus-Modern-14-C11M:~$
```



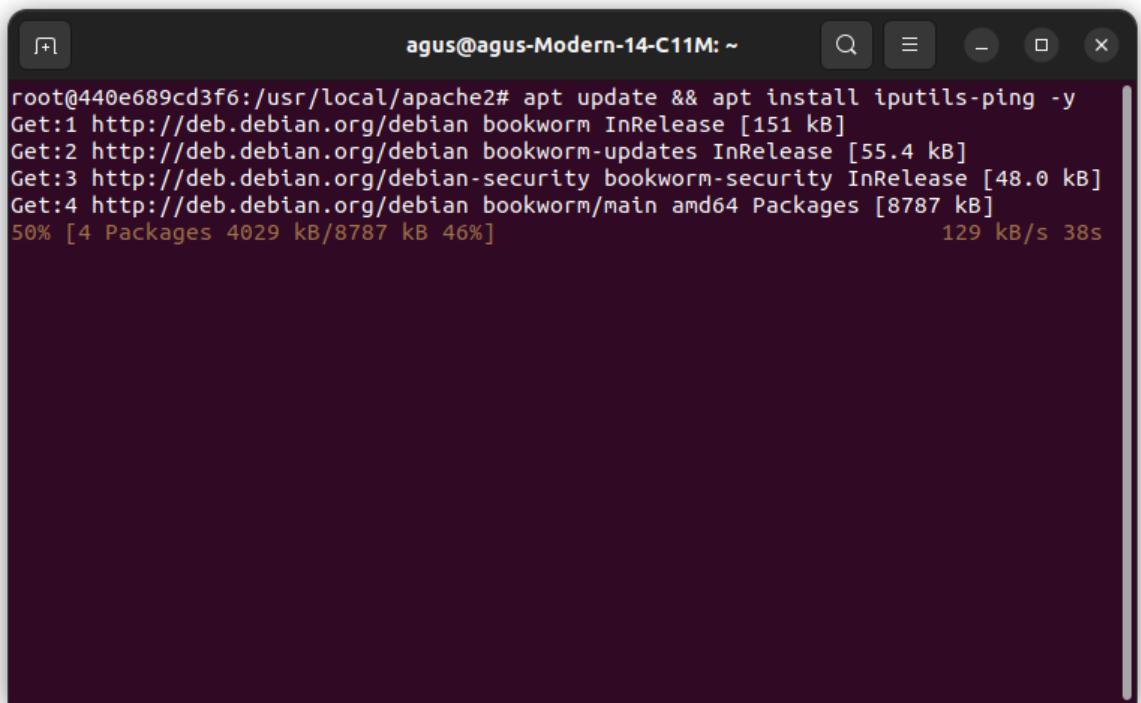
```
40a9fe9b2e6b    virkom-project2-web_php    "docker-php-entrypoi..."   3 days ago
Exited (0) 43 hours ago
php73
cc2ccd2b8f55    mysql:8.0                  "docker-entrypoint.s..."  3 days ago
Up 13 minutes      33060/tcp, 0.0.0.0:6033->3306/tcp, [::]:6033->3306/tcp
mysql8
675b487aeb1e    image-virkom-web        "nginx -g 'daemon of..."   8 days ago
Exited (0) 8 days ago
laughing_feynman
2aebc7bace1a    mongo:4                  "docker-entrypoint.s..."  10 days ago
Exited (0) 10 days ago
mongoserver001
97423004d40a    mongo:4                  "docker-entrypoint.s..."  10 days ago
Exited (0) 10 days ago
mongoserver1
478123cd4168    mongo:4                  "docker-entrypoint.s..."  10 days ago
Exited (0) 10 days ago
exciting_chaum
agus@agus-Modern-14-C11M:~$ docker container run -d httpd
440e689cd3f6ad9248a30c90ba2c468e5c4efd810570cbabe9c3bb6ba6759043
agus@agus-Modern-14-C11M:~$ docker container run -d nginx
c5db0a08096edf720e91ad07ca1075883517a32b1ddde88bea53d6536ad03886
agus@agus-Modern-14-C11M:~$
```

2. Mulaikan sesi bash didalam apache.



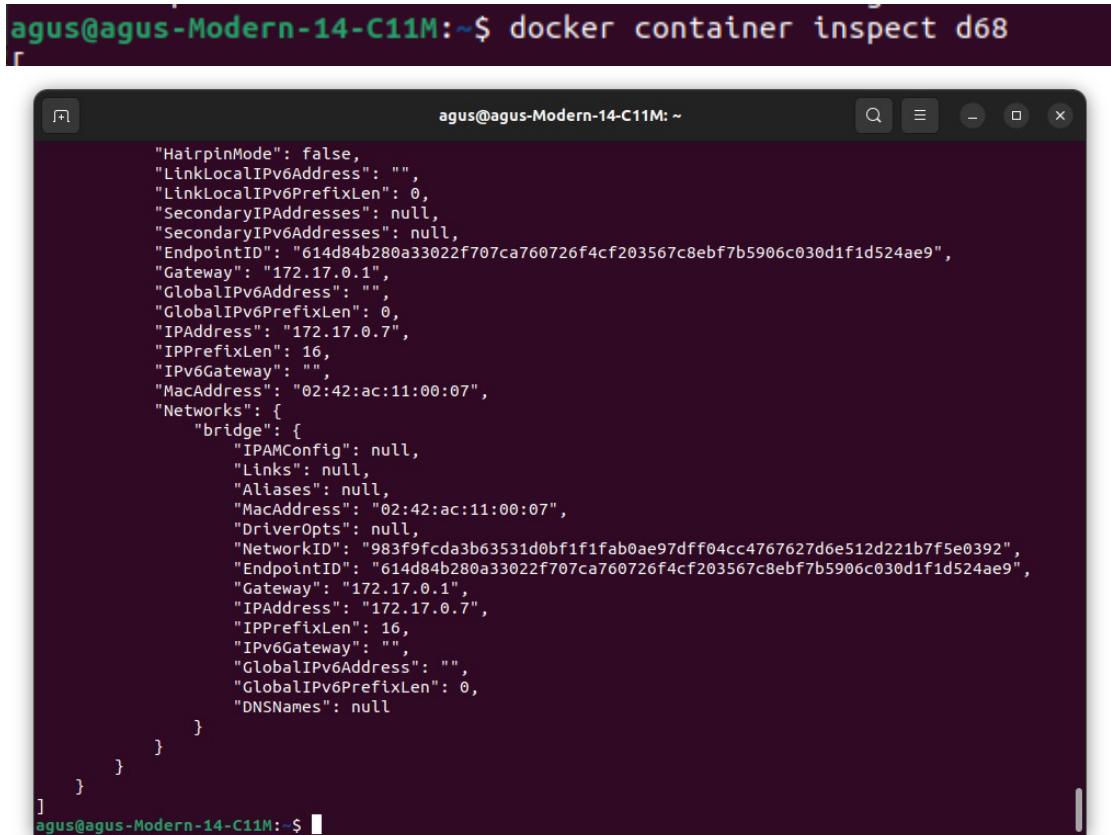
A screenshot of a terminal window titled "agus@agus-Modern-14-C11M: ~". The window shows a root shell in a Docker container. The command "docker exec -it 440 bash" was run to enter the container. The prompt "root@440e689cd3f6:/usr/local/apache2#" is visible, indicating the user is now in the Apache directory within the container.

3. Install IPUtils-ping untuk mencoba berkomunikasi dengan nginx container.



A screenshot of a terminal window titled "agus@agus-Modern-14-C11M: ~". The window shows a root shell in a Docker container. The command "apt update & apt install iputils-ping -y" was run to update the package index and install the iputils-ping package. The output shows the progress of the download and installation, with 50% of 4 packages at 4029 kB, a download speed of 129 kB/s, and a total time of 38s.

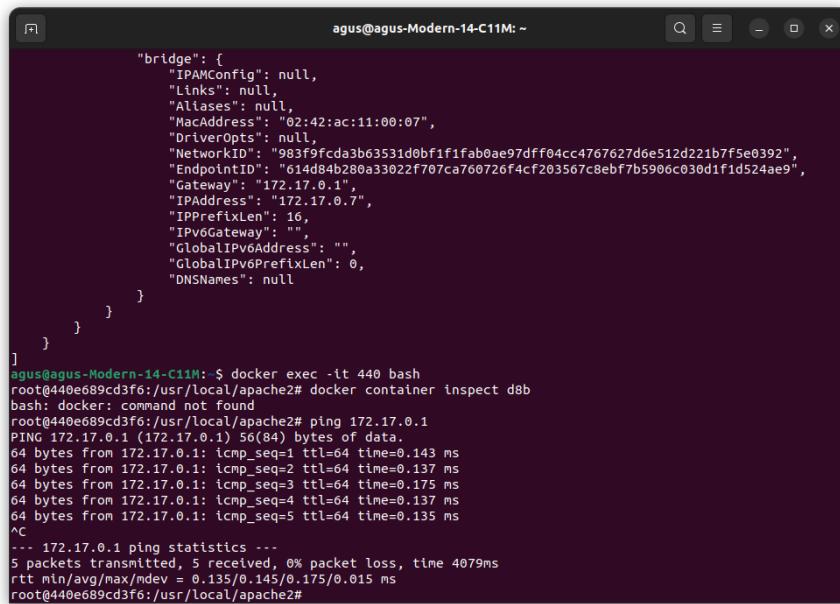
- Mari kita cari tahu IP Address dari nginx container.



```
agus@agus-Modern-14-C11M:~$ docker container inspect d68
[{"Name": "nginx", "Id": "614d84b280a33022f707ca760726f4cf203567c8ebf7b5906c030d1f1d524ae9", "Created": "2023-09-14T10:42:42.000Z", "Status": "running", "Image": "nginx:latest", "Ports": [{"HostPort": "80", "ContainerPort": 80}, {"HostPort": "443", "ContainerPort": 443}], "HostConfig": {"Binds": ["/var/run/docker.sock:/var/run/docker.sock"], "CgroupParent": "", "Links": null, "PortBindings": null, "Privileged": false, "ReadonlyRootfs": false, "SecurityOpt": null, "Tty": false, "Ulimits": null, "VolumeDriver": null, "VolumeDriverOptions": null}, "NetworkSettings": {"Bridge": "bridge", "GlobalIPv6Address": null, "GlobalIPv6PrefixLen": 0, "Gateway": "172.17.0.1", "IPAddress": "172.17.0.7", "IPPrefixLen": 16, "IPv6Gateway": "", "MacAddress": "02:42:ac:11:00:07", "NetworkID": "983f9fcda3b63531d0bf1fab0ae97dff04cc4767627d6e512d221b7f5e0392", "EndpointID": "614d84b280a33022f707ca760726f4cf203567c8ebf7b5906c030d1f1d524ae9", "Gateway": "172.17.0.1", "IPAddress": "172.17.0.7", "IPPrefixLen": 16, "IPv6Gateway": "", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "DNSNames": null}, {"Name": "bridge", "Id": "983f9fcda3b63531d0bf1fab0ae97dff04cc4767627d6e512d221b7f5e0392", "Created": "2023-09-14T10:42:42.000Z", "Status": "running", "Image": "nginx:latest", "Ports": [{"HostPort": "80", "ContainerPort": 80}, {"HostPort": "443", "ContainerPort": 443}], "HostConfig": {"Binds": ["/var/run/docker.sock:/var/run/docker.sock"], "CgroupParent": "", "Links": null, "PortBindings": null, "Privileged": false, "ReadonlyRootfs": false, "SecurityOpt": null, "Tty": false, "Ulimits": null, "VolumeDriver": null, "VolumeDriverOptions": null}, "NetworkSettings": {"Bridge": "bridge", "GlobalIPv6Address": null, "GlobalIPv6PrefixLen": 0, "Gateway": "172.17.0.1", "IPAddress": "172.17.0.7", "IPPrefixLen": 16, "IPv6Gateway": "", "MacAddress": "02:42:ac:11:00:07", "NetworkID": "983f9fcda3b63531d0bf1fab0ae97dff04cc4767627d6e512d221b7f5e0392", "EndpointID": "614d84b280a33022f707ca760726f4cf203567c8ebf7b5906c030d1f1d524ae9", "Gateway": "172.17.0.1", "IPAddress": "172.17.0.7", "IPPrefixLen": 16, "IPv6Gateway": "", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "DNSNames": null}}]
agus@agus-Modern-14-C11M:~$
```

Catatan: docker inspect mengembalikan informasi terperinci tentang sumber daya yang kita tentukan dalam format JSON.

- Ping nginx container menggunakan IP Address yang kita lihat sebelumnya.



```
agus@agus-Modern-14-C11M:~$ docker exec -it 440e689cd3f6 bash
root@440e689cd3f6:/usr/local/apache2# docker container inspect d68
bash: docker: command not found
root@440e689cd3f6:/usr/local/apache2# ping 172.17.0.1
PING 172.17.0.1 (172.17.0.1) 56(84) bytes of data.
64 bytes from 172.17.0.1: icmp_seq=1 ttl=64 time=0.143 ms
64 bytes from 172.17.0.1: icmp_seq=2 ttl=64 time=0.137 ms
64 bytes from 172.17.0.1: icmp_seq=3 ttl=64 time=0.175 ms
64 bytes from 172.17.0.1: icmp_seq=4 ttl=64 time=0.137 ms
64 bytes from 172.17.0.1: icmp_seq=5 ttl=64 time=0.135 ms
^C
--- 172.17.0.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4079ms
rtt min/avg/max/mdev = 0.135/0.145/0.175/0.015 ms
root@440e689cd3f6:/usr/local/apache2#
```

Hal ini dimungkinkan karena adanya jaringan docker. Apa yang dimaksud dengan jaringan?

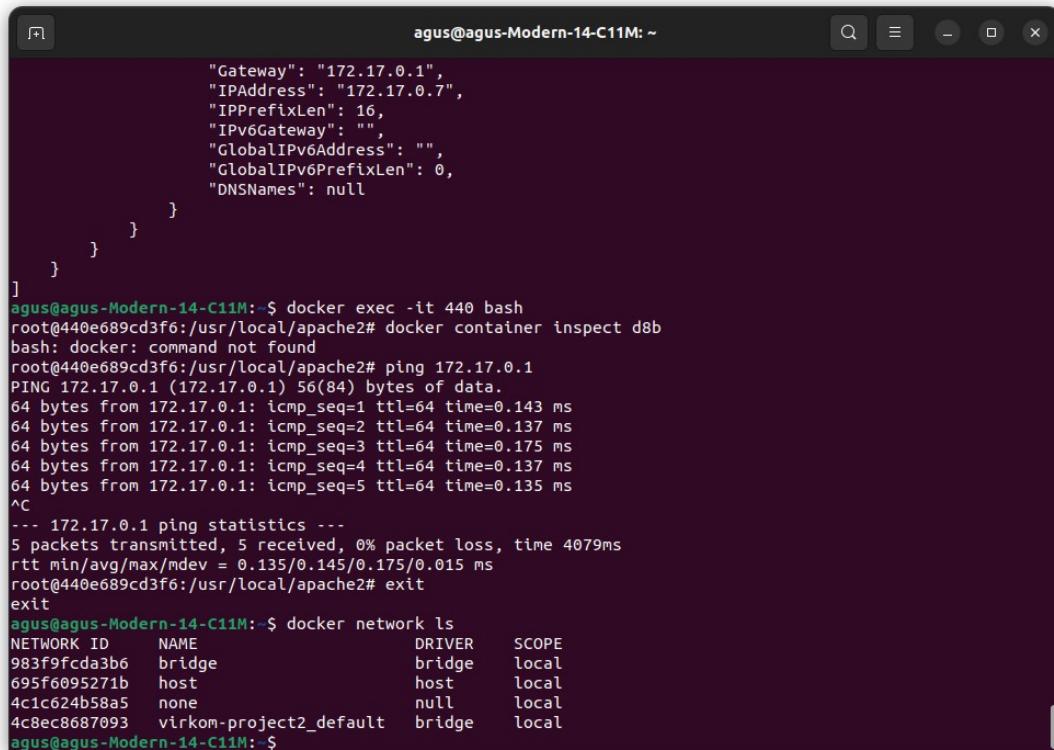
Sebuah jaringan dapat dianggap sebagai kumpulan container/simpul yang terhubung satu sama lain sedemikian rupa sehingga mereka dapat berbicara dengan container/simpul mana pun yang ada di dalam jaringan ini.

Tipe Jaringan

Docker memiliki 3 tipe:

- Bridge/docker0
- Host
- None

6. Mari kita lihat semua jaringan yang ada di docker.



```
agus@agus-Modern-14-C11M:~$ docker exec -it 440 bash
root@440e689cd3f6:/usr/local/apache2# docker container inspect d8b
bash: docker: command not found
root@440e689cd3f6:/usr/local/apache2# ping 172.17.0.1
PING 172.17.0.1 (172.17.0.1) 56(84) bytes of data.
64 bytes from 172.17.0.1: icmp_seq=1 ttl=64 time=0.143 ms
64 bytes from 172.17.0.1: icmp_seq=2 ttl=64 time=0.137 ms
64 bytes from 172.17.0.1: icmp_seq=3 ttl=64 time=0.175 ms
64 bytes from 172.17.0.1: icmp_seq=4 ttl=64 time=0.137 ms
64 bytes from 172.17.0.1: icmp_seq=5 ttl=64 time=0.135 ms
^C
--- 172.17.0.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4079ms
rtt min/avg/max/mdev = 0.135/0.145/0.175/0.015 ms
root@440e689cd3f6:/usr/local/apache2# exit
agus@agus-Modern-14-C11M:~$ docker network ls
NETWORK ID     NAME      DRIVER      SCOPE
983f9fcda3b6   bridge    bridge      local
695f6095271b   host      host      local
4c1c624b58a5   none      null      local
4c8ec8687093   virkom-project2_default  bridge      local
agus@agus-Modern-14-C11M:~$
```

Mari kita pahami satu per satu.

- **bridge/docker0**

Anda pasti menyadari bahwa kami tidak perlu melakukan apa pun untuk membuat komunikasi menjadi mungkin. Ini karena :

Secara default, setiap container menggunakan jaringan bridge kecuali ditentukan secara eksplisit.

Ini adalah jaringan virtual pribadi yang bersifat internal untuk host sehingga container pada jaringan ini dapat berkomunikasi satu sama lain.

Anda dapat menginspect jaringan bridge untuk melihat semua container Anda dapat memeriksa jaringan bridge untuk melihat semua container yang menjadi bagian darinya.

```
agus@agus-Modern-14-C11M:~$ docker network inspect bridge
[{"Name": "bridge",
 "Id": "983f9fcda3b63531d0bf1f1fab0ae97dff04cc4767627d6e512d221b7f5e0392",
 "Created": "2024-11-09T02:46:59.067115793+07:00",
 "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": {
     "195a4ba3022c0e736fb2a0d2c2141fa4c30dcce5e4f56e47aafb5b2f2d022a3f": {
         "Name": "vibrant_einstein",
         "EndpointID": "ee2b5b0bdf401cea14334ed777833ac0aeaec3034322b8f21acd95cb7b60e501",
         "MacAddress": "02:42:ac:11:00:06",
         "IPv4Address": "172.17.0.6/16",
         "IPv6Address": ""
     },
     "1acb09631f656aab4696db0f4bb2a73cf2f13ebd176c8a559297a352ecf7843": {
         "Name": "elegant_lederberg",
         "EndpointID": "98eb930880300806377f0a1d6198bffb88a70510d100b046b6f6898b58067ef",
         "MacAddress": "02:42:ac:11:00:03",
         "IPv4Address": "172.17.0.3/16",
         "IPv6Address": ""
     },
     "440e689cd3f6ad9248a30c90ba2c468e5c4efd810570cbabe9c3bb6ba6759043": {
         "Name": "xenodochial_shtrley",
         "EndpointID": "aa6230c056de2fb75d5f31b3146d5e7588df2f1feb57eea39eadde62ceeeb99f",
         "MacAddress": "02:42:ac:11:00:04",
         "IPv4Address": "172.17.0.4/16",
         "IPv6Address": ""
     },
     "c466bfedfa2fa5b5a9d6a453a93894f49aa36d4f30ad4c4fc8d8d493ec802c74": {
         "Name": "serene_hopper",
         "EndpointID": "038600438aea83928075e62bd5bc64066be9d92a7aedc94109392a55f772bff1",
         "MacAddress": "02:42:ac:11:00:02",
         "IPv4Address": "172.17.0.2/16",
         "IPv6Address": ""
     },
     "c5db0a08096edf720e91ad07ca1075883517a32b1ddde88bea53d6536ad03886": {
         "Name": "pedantic_yonath",
         "EndpointID": "e92cf70c8431b01dbf99020935d21b9a315cf4ebb1d4d8495a7aea503b472ee8",
         "MacAddress": "02:42:ac:11:00:05",
         "IPv4Address": "172.17.0.5/16",
         "IPv6Address": ""
     },
     "d680d1e181a525b1e91a54a31498adbc0390325bdf8539e19c343ae7ae3612b": {
         "Name": "hardcore_chateau",
         "EndpointID": "614d84b280a33022f707ca760726f4cf203567c8ebf7b5906c030d1f1d524ae9",
         "MacAddress": "02:42:ac:11:00:07",
         "IPv4Address": "172.17.0.7/16",
         "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": {}
}]
```

Anda dapat melihat container apache dan nginx sebagai bagian dari jaringan bridge default.

Kita juga akan melihat bahwa container menggunakan IP yang berbeda dari host.

Kita akan memverifikasi di bagian selanjutnya.

1. Alamat IP Host dan docker0

docker0 adalah jaringan yang dikonfigurasi pada host itu sendiri. Anda dapat memverifikasi ini dengan menjalankan perintah berikut.

```
agus@agus-Modern-14-C1M: $ ip address show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: wlo1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 74:04:f1:6f:4c:b7 brd ff:ff:ff:ff:ff:ff
        altname wlp0s20f3
        inet 192.168.77.219/24 brd 192.168.77.255 scope global dynamic noprefixroute wlo1
            valid_lft 3317sec preferred_lft 3317sec
        inet6 2400:9800:471:32cc:c508:c3f0:386e:9429/64 scope global temporary dynamic
            valid_lft 7137sec preferred_lft 7137sec
        inet6 2400:9800:471:32cc:9be0:6d27:ee04:877a/64 scope global dynamic mngtmpaddr noprefixroute
            valid_lft 7137sec preferred_lft 7137sec
        inet6 fe80::d337:1a05:3bd4:247f/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
3: br-4c8ec8687093: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:33:02:7c:cb brd ff:ff:ff:ff:ff:ff
    inet 172.18.0.1/16 brd 172.18.255.255 scope global br-4c8ec8687093
        valid_lft forever preferred_lft forever
    inet6 fe80::42:33ff:fe02:7cbc/64 scope link
        valid_lft forever preferred_lft forever
4: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:f5:cb:3e:1e brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
    inet6 fe80::42:f5ff:fe02:3e1e/64 scope link
        valid_lft forever preferred_lft forever
6: veth79a288c@if5: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-4c8ec8687093 state UP group default
    link/ether 9a:7c:de:90:ca:29 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet6 fe80::987c:deff:fe90:ca29/64 scope link
        valid_lft forever preferred_lft forever
8: veth2a9ece3@if7: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP group default
    link/ether 8a:47:85:c4:3e:63 brd ff:ff:ff:ff:ff:ff link-netnsid 1
    inet6 fe80::8847:85ff:fec4:3e63/64 scope link
        valid_lft forever preferred_lft forever
10: veth2e6e4360@if9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP group default
    link/ether 16:74:ff:db:16:1d brd ff:ff:ff:ff:ff:ff link-netnsid 2
    inet6 fe80::1474:ffff:fedb:161d/64 scope link
        valid_lft forever preferred_lft forever
12: veth3eecc1e@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP group default
    link/ether ca:28:56:8e:c7:51 brd ff:ff:ff:ff:ff:ff link-netnsid 3
    inet6 fe80::c828:56ff:fe8e:c751/64 scope link
        valid_lft forever preferred_lft forever
14: veth03fef760@if13: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP group default
    link/ether 92:aea:7:97:69:32 brd ff:ff:ff:ff:ff:ff link-netnsid 4
    inet6 fe80::90ae:a7ff:fe97:6932/64 scope link
        valid_lft forever preferred_lft forever
16: vetha11a6a90@if15: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP group default
    link/ether 3a:45:8f:20:d2:5c brd ff:ff:ff:ff:ff:ff link-netnsid 5
    inet6 fe80::3845:8fff:fe02:d25c/64 scope link
        valid_lft forever preferred_lft forever
18: veth4b1dd050@if17: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP group default
    link/ether 8e:0a:30:02:b5:ca brd ff:ff:ff:ff:ff:ff link-netnsid 6
    inet6 fe80::8c0a:30ff:fe02:b5ca/64 scope link
        valid_lft forever preferred_lft forever
agus@agus-Modern-14-C1M: $
```

Di sini kita dapat melihat dengan jelas bahwa host memiliki IP yang berbeda dengan IP container nginx yang kita buat dan docker0.

2. Membuat jaringan kita sendiri

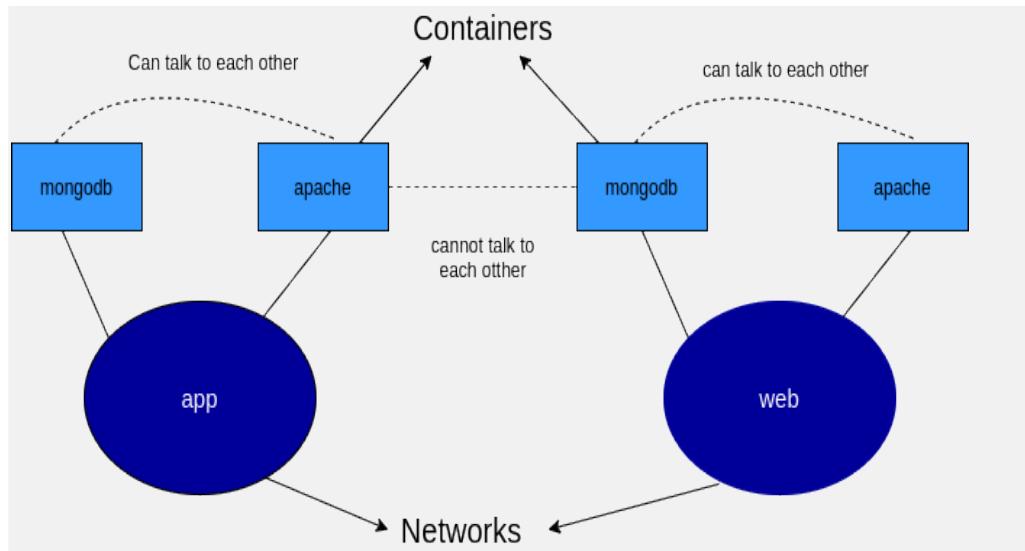
Sekarang, kita telah mengetahui bahwa container dapat berbicara satu sama lain pada jaringan yang sama.

Mari kita pertimbangkan skenario di mana kita memiliki dua aplikasi berbeda yang berjalan, yang tidak terkait satu sama lain. Bagaimana kita memastikan tidak ada komunikasi yang diizinkan di antara container dari aplikasi yang berbeda dalam skenario seperti itu?

Mari kita lihat solusinya.

Kita dapat membuat jaringan sendiri untuk mengisolasi container secara logis.

Kontainer dalam jaringan yang sama dapat berbicara satu sama lain.

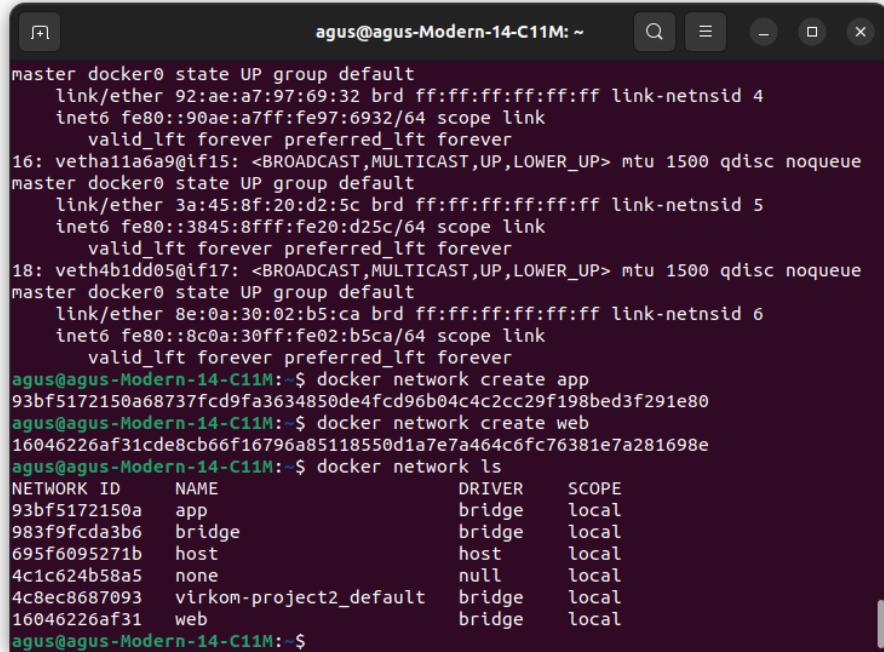


Mari kita lihat ini dalam tindakan.

1. Buat jaringan yang disebut aplikasi dan jaringan lain yang disebut web.

```
agus@agus-Modern-14-C11M: ~
inet6 fe80::1474:ffff:fedb:161d/64 scope link
    valid_lft forever preferred_lft forever
12: veth3eec1ce@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
master docker0 state UP group default
    link/ether ca:28:56:8e:c7:51 brd ff:ff:ff:ff:ff:ff link-netnsid 3
    inet6 fe80::c828:56ff:fe8e:c751/64 scope link
        valid_lft forever preferred_lft forever
14: veth03fef76@if13: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
master docker0 state UP group default
    link/ether 92:ae:a7:97:69:32 brd ff:ff:ff:ff:ff:ff link-netnsid 4
    inet6 fe80::90ae:a7ff:fe97:6932/64 scope link
        valid_lft forever preferred_lft forever
16: vetha11a6a9@if15: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
master docker0 state UP group default
    link/ether 3a:45:8f:20:d2:5c brd ff:ff:ff:ff:ff:ff link-netnsid 5
    inet6 fe80::3845:8fff:fe20:d25c/64 scope link
        valid_lft forever preferred_lft forever
18: veth4b1dd05@if17: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
master docker0 state UP group default
    link/ether 8e:0a:30:02:b5:ca brd ff:ff:ff:ff:ff:ff link-netnsid 6
    inet6 fe80::8c0a:30ff:fe02:b5ca/64 scope link
        valid_lft forever preferred_lft forever
agus@agus-Modern-14-C11M: ~$ docker network create app
93bf5172150a68737fcdf9fa3634850de4fc96b04c4c2cc29f198bed3f291e80
agus@agus-Modern-14-C11M: ~$ docker network create web
16046226af31cd8cb66f16796a85118550d1a7e7a464c6fc76381e7a281698e
agus@agus-Modern-14-C11M: ~$
```

2. Mari kita verifikasi bahwa jaringan telah dibuat.

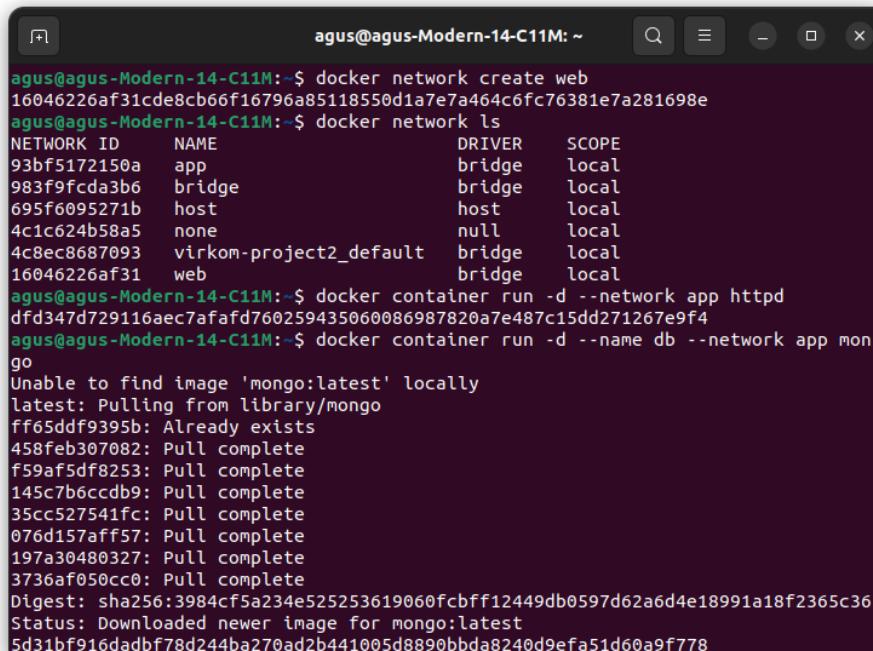


```
master docker0 state UP group default
    link/ether 92:ae:a7:97:69:32 brd ff:ff:ff:ff:ff:ff link-netnsid 4
        inet6 fe80::90ae:a7ff:fe97:6932/64 scope link
            valid_lft forever preferred_lft forever
16: vetha11a6a9@if15: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
master docker0 state UP group default
    link/ether 3a:45:8f:20:d2:5c brd ff:ff:ff:ff:ff:ff link-netnsid 5
        inet6 fe80::3845:8fff:fe20:d25c/64 scope link
            valid_lft forever preferred_lft forever
18: veth4b1dd05@if17: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
master docker0 state UP group default
    link/ether 8e:0a:30:02:b5:ca brd ff:ff:ff:ff:ff:ff link-netnsid 6
        inet6 fe80::8c0a:30ff:fe02:b5ca/64 scope link
            valid_lft forever preferred_lft forever
agus@agus-Modern-14-C11M:~$ docker network create app
93bf5172150a68737fc9fa3634850de4fc96b04c4c2cc29f198bed3f291e80
agus@agus-Modern-14-C11M:~$ docker network create web
16046226af31cde8cb66f16796a85118550dia7e7a464c6fc76381e7a281698e
agus@agus-Modern-14-C11M:~$ docker network ls
NETWORK ID      NAME      DRIVER      SCOPE
93bf5172150a    app       bridge      local
983f9fcda3b6    bridge     bridge      local
695f6095271b    host       host       local
4c1c624b58a5    none      null       local
4c8ec8687093    virkcom-project2_default  bridge      local
16046226af31    web       bridge      local
agus@agus-Modern-14-C11M:~$
```

3. Mulai container apache dan menambahkannya ke jaringan aplikasi.

```
agus@agus-Modern-14-C11M:~$ docker container run -d --network app httpd
dfd347d729116aec7afafdf760259435060086987820a7e487c15dd271267e9f4
```

4. Mulai container mongo db dan tambahkan juga ke jaringan aplikasi.



```
agus@agus-Modern-14-C11M:~$ docker network create web
16046226af31cde8cb66f16796a85118550dia7e7a464c6fc76381e7a281698e
agus@agus-Modern-14-C11M:~$ docker network ls
NETWORK ID      NAME      DRIVER      SCOPE
93bf5172150a    app       bridge      local
983f9fcda3b6    bridge     bridge      local
695f6095271b    host       host       local
4c1c624b58a5    none      null       local
4c8ec8687093    virkcom-project2_default  bridge      local
16046226af31    web       bridge      local
agus@agus-Modern-14-C11M:~$ docker container run -d --network app httpd
dfd347d729116aec7afafdf760259435060086987820a7e487c15dd271267e9f4
agus@agus-Modern-14-C11M:~$ docker container run -d --name db --network app mongo
Unable to find image 'mongo:latest' locally
latest: Pulling from library/mongo
ff65ddf9395b: Already exists
458feb307082: Pull complete
f59af5df8253: Pull complete
145c7b6ccdb9: Pull complete
35cc527541fc: Pull complete
076d157aff57: Pull complete
197a30480327: Pull complete
3736af050cc0: Pull complete
Digest: sha256:3984cf5a234e525253619060fcbff12449db0597d62a6d4e18991a18f2365c36
Status: Downloaded newer image for mongo:latest
5d31bf916dadbf78d244ba270ad2b441005d8890bbda8240d9efa51d60a9f778
```

5. Mulai container mongo db yang lain dan tambahkan ke jaringan web sebagai berikut.

```
f59af5df8253: Pull complete
145c7b6ccdb9: Pull complete
35cc527541fc: Pull complete
076d157aff57: Pull complete
197a30480327: Pull complete
3736af050cc0: Pull complete
Digest: sha256:3984cf5a234e525253619060fcbff12449db0597d62a6d4e18991a18f2365c36
Status: Downloaded newer image for mongo:latest
5d31bf916dadbf78d244ba270ad2b441005d8890bbda8240d9efa51d60a9f778
agus@agus-Modern-14-C11M: $ docker container run -d --name db --network app mongo
docker: Error response from daemon: Conflict. The container name "/db" is already in use by container "5d31bf916dadbf78d244ba270ad2b441005d8890bbda8240d9efa51d60a9f778". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.
agus@agus-Modern-14-C11M: $ docker container run -d --name db --network app mongo
docker: Error response from daemon: Conflict. The container name "/db" is already in use by container "5d31bf916dadbf78d244ba270ad2b441005d8890bbda8240d9efa51d60a9f778". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.
agus@agus-Modern-14-C11M: $ docker container run -d --name webdb --network web mongo
657ebcf74f9e689f55387700b9f63a3a8aa3bcd164a901fe826d70d19bc78b6a
agus@agus-Modern-14-C11M: $
```

6. Memulai sesi bash di dalam container apache.

```
5d31bf916dadbf78d244ba270ad2b441005d8890bbda8240d9efa51d60a9f778
agus@agus-Modern-14-C11M: $ docker container run -d --name db --network app mongo
docker: Error response from daemon: Conflict. The container name "/db" is already in use by container "5d31bf916dadbf78d244ba270ad2b441005d8890bbda8240d9efa51d60a9f778". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.
agus@agus-Modern-14-C11M: $ docker container run -d --name db --network app mongo
docker: Error response from daemon: Conflict. The container name "/db" is already in use by container "5d31bf916dadbf78d244ba270ad2b441005d8890bbda8240d9efa51d60a9f778". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.
agus@agus-Modern-14-C11M: $ docker container run -d --name webdb --network web mongo
657ebcf74f9e689f55387700b9f63a3a8aa3bcd164a901fe826d70d19bc78b6a
agus@agus-Modern-14-C11M: $ docker exec -it fd4bash
"docker exec" requires at least 2 arguments.
See 'docker exec --help'.

Usage: docker exec [OPTIONS] CONTAINER COMMAND [ARG...]

Execute a command in a running container
agus@agus-Modern-14-C11M: $ docker exec -it dfd bash
root@dfd347d72911:/usr/local/apache2#
```

7. Instal IPutils-ping untuk melakukan ping untuk berbicara dengan container mongo.

```
agu@agus-Modern-14-C11M:~$ docker exec -it dfd bash
root@dfd347d72911:/usr/local/apache# apt update && apt install iputils-ping -y
Get:1 http://deb.debian.org/debian bookworm InRelease [151 kB]
Get:2 http://deb.debian.org/debian bookworm-updates InRelease [55.4 kB]
Get:3 http://deb.debian.org/debian-security bookworm-security InRelease [48.0 kB]
Get:4 http://deb.debian.org/debian bookworm/main amd64 Packages [8787 kB]
Get:5 http://deb.debian.org/debian/bookworm-updates/main amd64 Packages [468 B]
Get:6 http://deb.debian.org/debian-security/bookworm-security/main amd64 Packages [198 kB]
Fetched 9234 kB in 8s (1107 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libcap2-bin libpam-cap
The following NEW packages will be installed:
  libcap2-bin libpam-cap
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 96.2 kB of archives.
After this operation, 311 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian bookworm/main amd64 libcap2-bin amd64 1:2.66-4 [34.7 kB]
Get:2 http://deb.debian.org/debian bookworm/main amd64 iputils-ping amd64 3:20221126-1 [47.1 kB]
Get:3 http://deb.debian.org/debian bookworm/main amd64 libpam-cap amd64 1:2.66-4 [14.5 kB]
Fetched 96.2 kB in 0s (280 kB/s)
dpkg: warning: package configuration, since apt-utils is not installed
Selecting previously unselected package libcap2-bin.
(Reading database ... 6791 files and directories currently installed.)
Preparing to unpack .../libcap2-bin_1%k3a2.66-4_amd64.deb ...
Unpacking libcap2-bin (1:2.66-4) ...
Selecting previously unselected package iputils-ping.
Preparing to unpack .../iputils-ping_3%k3a2.66-1_amd64.deb ...
Unpacking iputils-ping (3:20221126-1) ...
Selecting previously unselected package libpam-cap:amd64.
Preparing to unpack .../libpam-cap_3%k3a2.66-4_amd64.deb ...
Unpacking libpam-cap:amd64 (1:2.66-4) ...
Setting up libcap2-bin (1:2.66-4) ...
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based front end cannot be used. at /usr/share/perl5/DebConf/FrontEnd/Dialog.pm line 78.)
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (Can't locate Term/Readline.pm in @INC (you may need to install the Term::Readline module) (@INC contains: /usr/lib/x86_64-linux-gnu/perl/5.36.0 /usr/local/share/perl/5.36.0 /usr/lib/x86_64-linux-gnu/perl-base /usr/lib/x86_64-linux-gnu/perl/5.36 /usr/share/perl /usr/lib/x86_64-linux-gnu/perl-base /usr/lib/x86_64-linux-gnu/perl /usr/perl5.36 /usr/share/perl/5.36 /usr/local/lib/site_perl) at /usr/share/perl5/DebConf/FrontEnd/Readline.pm line 7.)
debconf: falling back to frontend: Teletype
Setting up iputils-ping (3:20221126-1) ...
root@dfd347d72911:/usr/local/apache#
```

8. Cari tahu alamat IP dari db container.

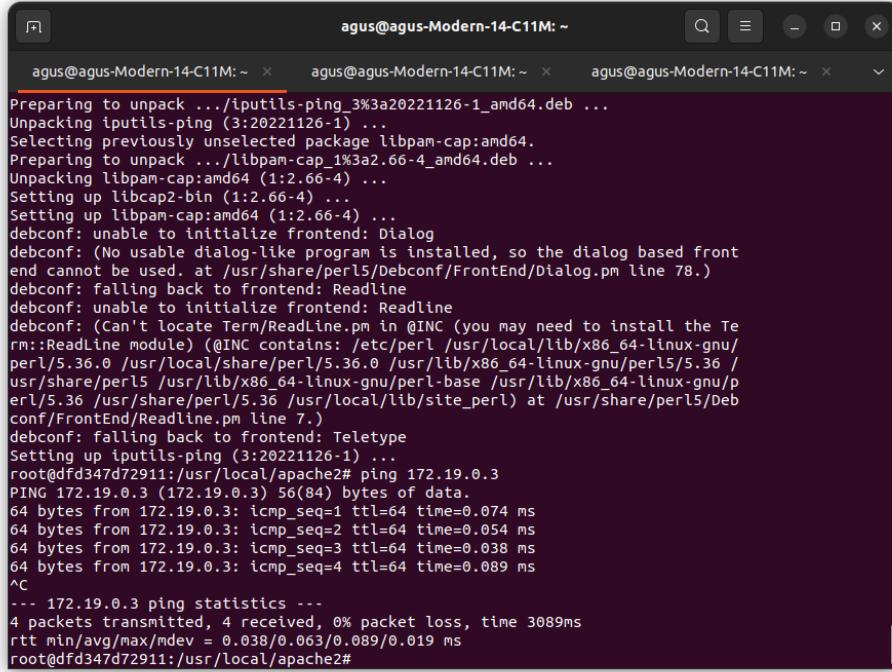
```
agu@agus-Modern-14-C11M:~ agus@agus-Modern-14-C11M:~ agus@agus-Modern-14-C11M:~
```

```
Usage: docker inspect [OPTIONS] NAME|ID [NAME|ID...]
```

```
Return low-level information on Docker objects
```

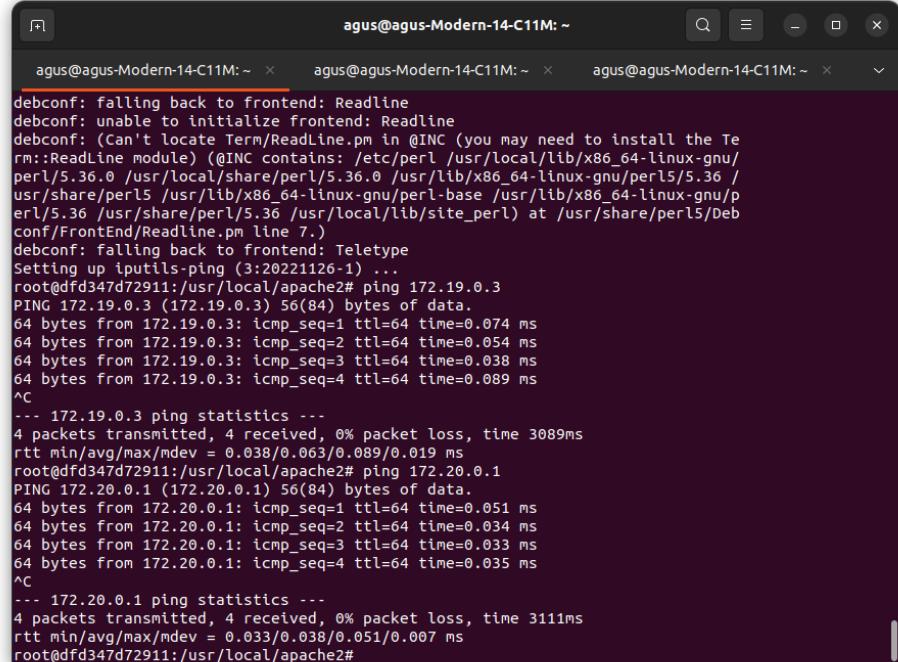
```
agu@agus-Modern-14-C11M:~$ docker inspect --format='json' 5d31
[{"Id": "5d31bf916dadbf78d44ba278ad2b441005d8890bd8240d9e51d68a9f778", "Created": "2024-11-08T20:32:06.485462951Z", "Path": "docker-entrypoint-init.d", "Args": ["mongod"], "State": {"Status": "running", "Running": true, "Paused": false, "Restarting": false, "OOMKilled": false, "Dead": false, "ExitCode": "0", "Error": ""}, "Image": "sha256:77c9b6384122f2efafdb739ff5f84ed662289e2973c7ff09", "ResolvConfPath": "/var/lib/docker/containers/5d31bf916dadbf78d44ba278ad2b441005d8890bd8240d9e51d68a9f778/hosts", "LogPath": "/var/lib/docker/containers/5d31bf916dadbf78d44ba278ad2b441005d8890bd8240d9e51d68a9f778/json.log", "Name": "/db", "RestartCount": 0, "Driver": "overlay2", "Platform": "linux", "MountLabel": "", "ProcessLabel": "", "AppArmorProfile": "docker-default", "ExecIDs": null, "HostConfig": {"Binds": null, "ContainerSize": 0, "LogConfig": {"Type": "json-file", "Config": {}}, "NetworkMode": "ap0", "PortBindings": {}, "RestartPolicy": {"Name": "no", "MaximumRetryCount": 0}, "AutoRemove": false, "VolumeDriver": "", "VolumesFrom": null, "ConsoleSize": [127, 79], "CapAdd": null, "CapDrop": null, "GroupsMode": "private", "Group": "", "Links": null, "OomScoreAdj": 0, "PidMode": "", "Privileged": false, "PublishAllPorts": false, "ReadonlyRootfs": false, "SecurityOpt": null, "UTSMode": "", "ShmSize": "67108864", "Runtime": "runc", "Isolation": "cgroups", "CpusShares": 0, "Memory": 0, "NanoCpus": 0, "GroupPermissions": "", "BlkiowriteWeightDevice": [], "BlkiodeviceRead": [], "BlkiodeviceWrite": [], "BlkiodeviceReadIops": [], "BlkiodeviceWriteIops": [], "CpuPerf": 0, "CpuQuota": 0, "CpuRealtimePriority": 0, "CpuRealtimeRuntime": 0, "CousetCpus": "0", "CousetMems": "", "Devices": null, "DeviceGroupRules": null, "MemoryReservation": 0, "MemorySwap": 0, "MemorySwappiness": null, "OomKillDisable": null, "PidsLimit": null, "Limits": null, "Ulimits": 0, "CpuCount": 0, "CpuPercent": 0, "IOMaximumIOPS": 0, "IOMaximumBandwidth": 0, "MaskedPaths": "[\"/proc/asmount\", \"/proc/acpi\", \"/proc/core\", \"/proc/latency_stats\", \"/proc/timer_list\", \"/proc/timer_stats\", \"/proc/dns\", \"/proc/fs\", \"/proc/latency\", \"/proc/meminfo\", \"/proc/mounts\", \"/proc/net/dev\", \"/proc/pulseaudio\", \"/proc/reboot\", \"/proc/sound\", \"/proc/sys/firmware\", \"/proc/sys/virtual-powercontrol\", \"/proc/sysrq-trigger\"]", "GraphDriver": {"Data": {"LowerDir": "/var/lib/docker/overlay2", "UpperDir": "/var/lib/docker/overlay2/1f748cbfa794f948bc8a70b4fccebf3e16d314ae25231422b2e710ee7354/diff", "Workdir": "/var/lib/docker/overlay2/742bdca147210d55212aa00663d903938a5e0179d134e4891452990d5fbff/diff", "ContentConfig": "742bdca147210d55212aa00663d903938a5e0179d134e4891452990d5fbff"}, "Name": "aufs"}, "Mounts": [{"Type": "volume", "Name": "c58baccc5bb44788a8703777ba9bdc4fb693fcab7d641765f28cc080", "Source": "/var/lib/docker/volumes/c58baccc5bb44788a8703777ba9bdc4fb693fcab7d641765f28cc080/_data", "Destination": "/data/db", "Driver": "local", "Mode": "rw", "RW": true, "Propagation": "host"}, {"Type": "volume", "Name": "cd2d05e0bd2d96b1886d2e24aa6c873709892595173cabf7b6ccf0a787", "Source": "/var/lib/docker/volume/cd2d05e0bd2d96b1886d2e24aa6c873709892595173cabf7b6ccf0a787/_data", "Destination": "/data/configdb", "Driver": "local", "Mode": "rw", "RW": true, "Propagation": "host"}, {"Type": "volume", "Name": "5d31bf916dadbf78d44ba278ad2b441005d8890bd8240d9e51d68a9f778", "Source": "/var/lib/docker/volume/5d31bf916dadbf78d44ba278ad2b441005d8890bd8240d9e51d68a9f778/_data", "Destination": "/data/configdb", "Driver": "local", "Mode": "rw", "RW": true, "Propagation": "host"}], "Config": {"Hostname": "5d31bf916dadbf78d44ba278ad2b441005d8890bd8240d9e51d68a9f778", "Domainname": "", "User": "root", "AttachStdin": false, "AttachStdout": false, "AttachStderr": false, "ExposedPorts": {"27017/tcp": {"Type": "tcp"}, "27018/tcp": {"Type": "tcp"}, "27019/tcp": {"Type": "tcp"}, "27020/tcp": {"Type": "tcp"}, "27021/tcp": {"Type": "tcp"}, "27022/tcp": {"Type": "tcp"}, "27023/tcp": {"Type": "tcp"}, "27024/tcp": {"Type": "tcp"}, "27025/tcp": {"Type": "tcp"}, "27026/tcp": {"Type": "tcp"}, "27027/tcp": {"Type": "tcp"}, "27028/tcp": {"Type": "tcp"}, "27029/tcp": {"Type": "tcp"}, "27030/tcp": {"Type": "tcp"}, "27031/tcp": {"Type": "tcp"}, "27032/tcp": {"Type": "tcp"}, "27033/tcp": {"Type": "tcp"}, "27034/tcp": {"Type": "tcp"}, "27035/tcp": {"Type": "tcp"}, "27036/tcp": {"Type": "tcp"}, "27037/tcp": {"Type": "tcp"}, "27038/tcp": {"Type": "tcp"}, "27039/tcp": {"Type": "tcp"}, "27040/tcp": {"Type": "tcp"}, "27041/tcp": {"Type": "tcp"}, "27042/tcp": {"Type": "tcp"}, "27043/tcp": {"Type": "tcp"}, "27044/tcp": {"Type": "tcp"}, "27045/tcp": {"Type": "tcp"}, "27046/tcp": {"Type": "tcp"}, "27047/tcp": {"Type": "tcp"}, "27048/tcp": {"Type": "tcp"}, "27049/tcp": {"Type": "tcp"}, "27050/tcp": {"Type": "tcp"}, "27051/tcp": {"Type": "tcp"}, "27052/tcp": {"Type": "tcp"}, "27053/tcp": {"Type": "tcp"}, "27054/tcp": {"Type": "tcp"}, "27055/tcp": {"Type": "tcp"}, "27056/tcp": {"Type": "tcp"}, "27057/tcp": {"Type": "tcp"}, "27058/tcp": {"Type": "tcp"}, "27059/tcp": {"Type": "tcp"}, "27060/tcp": {"Type": "tcp"}, "27061/tcp": {"Type": "tcp"}, "27062/tcp": {"Type": "tcp"}, "27063/tcp": {"Type": "tcp"}, "27064/tcp": {"Type": "tcp"}, "27065/tcp": {"Type": "tcp"}, "27066/tcp": {"Type": "tcp"}, "27067/tcp": {"Type": "tcp"}, "27068/tcp": {"Type": "tcp"}, "27069/tcp": {"Type": "tcp"}, "27070/tcp": {"Type": "tcp"}, "27071/tcp": {"Type": "tcp"}, "27072/tcp": {"Type": "tcp"}, "27073/tcp": {"Type": "tcp"}, "27074/tcp": {"Type": "tcp"}, "27075/tcp": {"Type": "tcp"}, "27076/tcp": {"Type": "tcp"}, "27077/tcp": {"Type": "tcp"}, "27078/tcp": {"Type": "tcp"}, "27079/tcp": {"Type": "tcp"}, "27080/tcp": {"Type": "tcp"}, "27081/tcp": {"Type": "tcp"}, "27082/tcp": {"Type": "tcp"}, "27083/tcp": {"Type": "tcp"}, "27084/tcp": {"Type": "tcp"}, "27085/tcp": {"Type": "tcp"}, "27086/tcp": {"Type": "tcp"}, "27087/tcp": {"Type": "tcp"}, "27088/tcp": {"Type": "tcp"}, "27089/tcp": {"Type": "tcp"}, "27090/tcp": {"Type": "tcp"}, "27091/tcp": {"Type": "tcp"}, "27092/tcp": {"Type": "tcp"}, "27093/tcp": {"Type": "tcp"}, "27094/tcp": {"Type": "tcp"}, "27095/tcp": {"Type": "tcp"}, "27096/tcp": {"Type": "tcp"}, "27097/tcp": {"Type": "tcp"}, "27098/tcp": {"Type": "tcp"}, "27099/tcp": {"Type": "tcp"}, "27100/tcp": {"Type": "tcp"}, "27101/tcp": {"Type": "tcp"}, "27102/tcp": {"Type": "tcp"}, "27103/tcp": {"Type": "tcp"}, "27104/tcp": {"Type": "tcp"}, "27105/tcp": {"Type": "tcp"}, "27106/tcp": {"Type": "tcp"}, "27107/tcp": {"Type": "tcp"}, "27108/tcp": {"Type": "tcp"}, "27109/tcp": {"Type": "tcp"}, "27110/tcp": {"Type": "tcp"}, "27111/tcp": {"Type": "tcp"}, "27112/tcp": {"Type": "tcp"}, "27113/tcp": {"Type": "tcp"}, "27114/tcp": {"Type": "tcp"}, "27115/tcp": {"Type": "tcp"}, "27116/tcp": {"Type": "tcp"}, "27117/tcp": {"Type": "tcp"}, "27118/tcp": {"Type": "tcp"}, "27119/tcp": {"Type": "tcp"}, "27120/tcp": {"Type": "tcp"}, "27121/tcp": {"Type": "tcp"}, "27122/tcp": {"Type": "tcp"}, "27123/tcp": {"Type": "tcp"}, "27124/tcp": {"Type": "tcp"}, "27125/tcp": {"Type": "tcp"}, "27126/tcp": {"Type": "tcp"}, "27127/tcp": {"Type": "tcp"}, "27128/tcp": {"Type": "tcp"}, "27129/tcp": {"Type": "tcp"}, "27130/tcp": {"Type": "tcp"}, "27131/tcp": {"Type": "tcp"}, "27132/tcp": {"Type": "tcp"}, "27133/tcp": {"Type": "tcp"}, "27134/tcp": {"Type": "tcp"}, "27135/tcp": {"Type": "tcp"}, "27136/tcp": {"Type": "tcp"}, "27137/tcp": {"Type": "tcp"}, "27138/tcp": {"Type": "tcp"}, "27139/tcp": {"Type": "tcp"}, "27140/tcp": {"Type": "tcp"}, "27141/tcp": {"Type": "tcp"}, "27142/tcp": {"Type": "tcp"}, "27143/tcp": {"Type": "tcp"}, "27144/tcp": {"Type": "tcp"}, "27145/tcp": {"Type": "tcp"}, "27146/tcp": {"Type": "tcp"}, "27147/tcp": {"Type": "tcp"}, "27148/tcp": {"Type": "tcp"}, "27149/tcp": {"Type": "tcp"}, "27150/tcp": {"Type": "tcp"}, "27151/tcp": {"Type": "tcp"}, "27152/tcp": {"Type": "tcp"}, "27153/tcp": {"Type": "tcp"}, "27154/tcp": {"Type": "tcp"}, "27155/tcp": {"Type": "tcp"}, "27156/tcp": {"Type": "tcp"}, "27157/tcp": {"Type": "tcp"}, "27158/tcp": {"Type": "tcp"}, "27159/tcp": {"Type": "tcp"}, "27160/tcp": {"Type": "tcp"}, "27161/tcp": {"Type": "tcp"}, "27162/tcp": {"Type": "tcp"}, "27163/tcp": {"Type": "tcp"}, "27164/tcp": {"Type": "tcp"}, "27165/tcp": {"Type": "tcp"}, "27166/tcp": {"Type": "tcp"}, "27167/tcp": {"Type": "tcp"}, "27168/tcp": {"Type": "tcp"}, "27169/tcp": {"Type": "tcp"}, "27170/tcp": {"Type": "tcp"}, "27171/tcp": {"Type": "tcp"}, "27172/tcp": {"Type": "tcp"}, "27173/tcp": {"Type": "tcp"}, "27174/tcp": {"Type": "tcp"}, "27175/tcp": {"Type": "tcp"}, "27176/tcp": {"Type": "tcp"}, "27177/tcp": {"Type": "tcp"}, "27178/tcp": {"Type": "tcp"}, "27179/tcp": {"Type": "tcp"}, "27180/tcp": {"Type": "tcp"}, "27181/tcp": {"Type": "tcp"}, "27182/tcp": {"Type": "tcp"}, "27183/tcp": {"Type": "tcp"}, "27184/tcp": {"Type": "tcp"}, "27185/tcp": {"Type": "tcp"}, "27186/tcp": {"Type": "tcp"}, "27187/tcp": {"Type": "tcp"}, "27188/tcp": {"Type": "tcp"}, "27189/tcp": {"Type": "tcp"}, "27190/tcp": {"Type": "tcp"}, "27191/tcp": {"Type": "tcp"}, "27192/tcp": {"Type": "tcp"}, "27193/tcp": {"Type": "tcp"}, "27194/tcp": {"Type": "tcp"}, "27195/tcp": {"Type": "tcp"}, "27196/tcp": {"Type": "tcp"}, "27197/tcp": {"Type": "tcp"}, "27198/tcp": {"Type": "tcp"}, "27199/tcp": {"Type": "tcp"}, "27200/tcp": {"Type": "tcp"}, "27201/tcp": {"Type": "tcp"}, "27202/tcp": {"Type": "tcp"}, "27203/tcp": {"Type": "tcp"}, "27204/tcp": {"Type": "tcp"}, "27205/tcp": {"Type": "tcp"}, "27206/tcp": {"Type": "tcp"}, "27207/tcp": {"Type": "tcp"}, "27208/tcp": {"Type": "tcp"}, "27209/tcp": {"Type": "tcp"}, "27210/tcp": {"Type": "tcp"}, "27211/tcp": {"Type": "tcp"}, "27212/tcp": {"Type": "tcp"}, "27213/tcp": {"Type": "tcp"}, "27214/tcp": {"Type": "tcp"}, "27215/tcp": {"Type": "tcp"}, "27216/tcp": {"Type": "tcp"}, "27217/tcp": {"Type": "tcp"}, "27218/tcp": {"Type": "tcp"}, "27219/tcp": {"Type": "tcp"}, "27220/tcp": {"Type": "tcp"}, "27221/tcp": {"Type": "tcp"}, "27222/tcp": {"Type": "tcp"}, "27223/tcp": {"Type": "tcp"}, "27224/tcp": {"Type": "tcp"}, "27225/tcp": {"Type": "tcp"}, "27226/tcp": {"Type": "tcp"}, "27227/tcp": {"Type": "tcp"}, "27228/tcp": {"Type": "tcp"}, "27229/tcp": {"Type": "tcp"}, "27230/tcp": {"Type": "tcp"}, "27231/tcp": {"Type": "tcp"}, "27232/tcp": {"Type": "tcp"}, "27233/tcp": {"Type": "tcp"}, "27234/tcp": {"Type": "tcp"}, "27235/tcp": {"Type": "tcp"}, "27236/tcp": {"Type": "tcp"}, "27237/tcp": {"Type": "tcp"}, "27238/tcp": {"Type": "tcp"}, "27239/tcp": {"Type": "tcp"}, "27240/tcp": {"Type": "tcp"}, "27241/tcp": {"Type": "tcp"}, "27242/tcp": {"Type": "tcp"}, "27243/tcp": {"Type": "tcp"}, "27244/tcp": {"Type": "tcp"}, "27245/tcp": {"Type": "tcp"}, "27246/tcp": {"Type": "tcp"}, "27247/tcp": {"Type": "tcp"}, "27248/tcp": {"Type": "tcp"}, "27249/tcp": {"Type": "tcp"}, "27250/tcp": {"Type": "tcp"}, "27251/tcp": {"Type": "tcp"}, "27252/tcp": {"Type": "tcp"}, "27253/tcp": {"Type": "tcp"}, "27254/tcp": {"Type": "tcp"}, "27255/tcp": {"Type": "tcp"}, "27256/tcp": {"Type": "tcp"}, "27257/tcp": {"Type": "tcp"}, "27258/tcp": {"Type": "tcp"}, "27259/tcp": {"Type": "tcp"}, "27260/tcp": {"Type": "tcp"}, "27261/tcp": {"Type": "tcp"}, "27262/tcp": {"Type": "tcp"}, "27263/tcp": {"Type": "tcp"}, "27264/tcp": {"Type": "tcp"}, "27265/tcp": {"Type": "tcp"}, "27266/tcp": {"Type": "tcp"}, "27267/tcp": {"Type": "tcp"}, "27268/tcp": {"Type": "tcp"}, "27269/tcp": {"Type": "tcp"}, "27270/tcp": {"Type": "tcp"}, "27271/tcp": {"Type": "tcp"}, "27272/tcp": {"Type": "tcp"}, "27273/tcp": {"Type": "tcp"}, "27274/tcp": {"Type": "tcp"}, "27275/tcp": {"Type": "tcp"}, "27276/tcp": {"Type": "tcp"}, "27277/tcp": {"Type": "tcp"}, "27278/tcp": {"Type": "tcp"}, "27279/tcp": {"Type": "tcp"}, "27280/tcp": {"Type": "tcp"}, "27281/tcp": {"Type": "tcp"}, "27282/tcp": {"Type": "tcp"}, "27283/tcp": {"Type": "tcp"}, "27284/tcp": {"Type": "tcp"}, "27285/tcp": {"Type": "tcp"}, "27286/tcp": {"Type": "tcp"}, "27287/tcp": {"Type": "tcp"}, "27288/tcp": {"Type": "tcp"}, "27289/tcp": {"Type": "tcp"}, "27290/tcp": {"Type": "tcp"}, "27291/tcp": {"Type": "tcp"}, "27292/tcp": {"Type": "tcp"}, "27293/tcp": {"Type": "tcp"}, "27294/tcp": {"Type": "tcp"}, "27295/tcp": {"Type": "tcp"}, "27296/tcp": {"Type": "tcp"}, "27297/tcp": {"Type": "tcp"}, "27298/tcp": {"Type": "tcp"}, "27299/tcp": {"Type": "tcp"}, "27300/tcp": {"Type": "tcp"}, "27301/tcp": {"Type": "tcp"}, "27302/tcp": {"Type": "tcp"}, "27303/tcp": {"Type": "tcp"}, "27304/tcp": {"Type": "tcp"}, "27305/tcp": {"Type": "tcp"}, "27306/tcp": {"Type": "tcp"}, "27307/tcp": {"Type": "tcp"}, "27308/tcp": {"Type": "tcp"}, "27309/tcp": {"Type": "tcp"}, "27310/tcp": {"Type": "tcp"}, "27311/tcp": {"Type": "tcp"}, "27312/tcp": {"Type": "tcp"}, "27313/tcp": {"Type": "tcp"}, "27314/tcp": {"Type": "tcp"}, "27315/tcp": {"Type": "tcp"}, "27316/tcp": {"Type": "tcp"}, "27317/tcp": {"Type": "tcp"}, "27318/tcp": {"Type": "tcp"}, "27319/tcp": {"Type": "tcp"}, "27320/tcp": {"Type": "tcp"}, "27321/tcp": {"Type": "tcp"}, "27322/tcp": {"Type": "tcp"}, "27323/tcp": {"Type": "tcp"}, "27324/tcp": {"Type": "tcp"}, "27325/tcp": {"Type": "tcp"}, "27326/tcp": {"Type": "tcp"}, "27327/tcp": {"Type": "tcp"}, "27328/tcp": {"Type": "tcp"}, "27329/tcp": {"Type": "tcp"}, "27330/tcp": {"Type": "tcp"}, "27331/tcp": {"Type": "tcp"}, "27332/tcp": {"Type": "tcp"}, "27333/tcp": {"Type": "tcp"}, "27334/tcp": {"Type": "tcp"}, "27335/tcp": {"Type": "tcp"}, "27336/tcp": {"Type": "tcp"}, "27337/tcp": {"Type": "tcp"}, "27338/tcp": {"Type": "tcp"}, "27339/tcp": {"Type": "tcp"}, "27340/tcp": {"Type": "tcp"}, "27341/tcp": {"Type": "tcp"}, "27342/tcp": {"Type": "tcp"}, "27343/tcp": {"Type": "tcp"}, "27344/tcp": {"Type": "tcp"}, "27345/tcp": {"Type": "tcp"}, "27346/tcp": {"Type": "tcp"}, "27347/tcp": {"Type": "tcp"}, "27348/tcp": {"Type": "tcp"}, "27349/tcp": {"Type": "tcp"}, "27350/tcp": {"Type": "tcp"}, "27351/tcp": {"Type": "tcp"}, "27352/tcp": {"Type": "tcp"}, "27353/tcp": {"Type": "tcp"}, "27354/tcp": {"Type": "tcp"}, "27355/tcp": {"Type": "tcp"}, "27356/tcp": {"Type": "tcp"}, "27357/tcp": {"Type": "tcp"}, "27358/tcp": {"Type": "tcp"}, "27359/tcp": {"Type": "tcp"}, "27360/tcp": {"Type": "tcp"}, "27361/tcp": {"Type": "tcp"}, "27362/tcp": {"Type": "tcp"}, "27363/tcp": {"Type": "tcp"}, "27364/tcp": {"Type": "tcp"}, "27365/tcp": {"Type": "tcp"}, "27366/tcp": {"Type": "tcp"}, "27367/tcp": {"Type": "tcp"}, "27368/tcp": {"Type": "tcp"}, "27369/tcp": {"Type": "tcp"}, "27370/tcp": {"Type": "tcp"}, "27371/tcp": {"Type": "tcp"}, "27372/tcp": {"Type": "tcp"}, "27373/tcp": {"Type": "tcp"}, "27374/tcp": {"Type": "tcp"}, "27375/tcp": {"Type": "tcp"}, "27376/tcp": {"Type": "tcp"}, "27377/tcp": {"Type": "tcp"}, "27378/tcp": {"Type": "tcp"}, "27379/tcp": {"Type": "tcp"}, "27380/tcp": {"Type": "tcp"}, "27381/tcp": {"Type": "tcp"}, "27382/tcp": {"Type": "tcp"}, "27383/tcp": {"Type": "tcp"}, "27384/tcp": {"Type": "tcp"}, "27385/tcp": {"Type": "tcp"}, "27386/tcp": {"Type": "tcp"}, "27387/tcp": {"Type": "tcp"}, "27388/tcp": {"Type": "tcp"}, "27389/tcp": {"Type": "tcp"}, "27390/tcp": {"Type": "tcp"}, "27391/tcp": {"Type": "tcp"}, "27392/tcp": {"Type": "tcp"}, "27393/tcp": {"Type": "tcp"}, "27394/tcp": {"Type": "tcp"}, "27395/tcp": {"Type": "tcp"}, "27396/tcp": {"Type": "tcp"}, "27397/tcp": {"Type": "tcp"}, "27398/tcp": {"Type": "tcp"}, "27399/tcp": {"Type": "tcp"}, "27400/tcp": {"Type": "tcp"}, "27401/tcp": {"Type": "tcp"}, "27402/tcp": {"Type": "tcp"}, "27403/tcp": {"Type": "tcp"}, "27404/tcp": {"Type": "tcp"}, "27405/tcp": {"Type": "tcp"}, "27406/tcp": {"Type": "tcp"}, "27407/tcp": {"Type": "tcp"}, "27408/tcp": {"Type": "tcp"}, "27409/tcp": {"Type": "tcp"}, "27410/tcp": {"Type": "tcp"}, "27411/tcp": {"Type": "tcp"}, "27412/tcp": {"Type": "tcp"}, "27413/tcp": {"Type": "tcp"}, "27414/tcp": {"Type": "tcp"}, "27415/tcp": {"Type": "tcp"}, "27416/tcp": {"Type": "tcp"}, "27417/tcp": {"Type": "tcp"}, "27418/tcp": {"Type": "tcp"}, "27419/tcp": {"Type": "tcp"}, "27420/tcp": {"Type": "tcp"}, "27421/tcp": {"Type": "tcp"}, "27422/tcp": {"Type": "tcp"}, "27423/tcp": {"Type": "tcp"}, "27424/tcp": {"Type": "tcp"}, "27425/tcp": {"Type": "tcp"}, "27426/tcp": {"Type": "tcp"}, "27427/tcp": {"Type": "tcp"}, "27428/tcp": {"Type": "tcp"}, "27429/tcp": {"Type": "tcp"}, "27430/tcp": {"Type": "tcp"}, "27431/tcp": {"Type": "tcp"}, "27432/tcp": {"Type": "tcp"}, "27433/tcp": {"Type": "tcp"}, "27434/tcp": {"Type": "tcp"}, "27435/tcp": {"Type": "tcp"}, "27436/tcp": {"Type": "tcp"}, "27437/tcp": {"Type": "tcp"}, "27438/tcp": {"Type": "tcp"}, "27439/tcp": {"Type": "tcp"}, "27440/tcp": {"Type": "tcp"}, "27441/tcp": {"Type": "tcp"}, "27442/tcp": {"Type": "tcp"}, "27443/tcp": {"Type": "tcp"}, "27444/tcp": {"Type": "tcp"}, "27445/tcp": {"Type": "tcp"}, "27446/tcp": {"Type": "tcp"}, "27447/tcp": {"Type": "tcp"}, "27448/tcp": {"Type": "tcp"}, "27449/tcp": {"Type": "tcp"}, "27450/tcp": {"Type": "tcp"}, "27451/tcp": {"Type": "tcp"}, "27452/tcp": {"Type": "tcp"}, "27453/tcp": {"Type": "tcp"}, "27454/tcp": {"Type": "tcp"}, "27455/tcp": {"Type": "tcp"}, "27456/tcp": {"Type": "tcp"}, "27457/tcp": {"Type": "tcp"}, "27458/tcp": {"Type": "tcp"}, "27459/tcp": {"Type": "tcp"}, "27460/tcp": {"Type": "tcp"}, "27461/tcp": {"Type": "tcp"}, "27462/tcp": {"Type": "tcp"}, "27463/tcp": {"Type": "tcp"}, "27464/tcp": {"Type": "tcp"}, "27465/tcp": {"Type": "tcp"}, "27466/tcp": {"Type": "tcp"}, "27467/tcp": {"Type": "tcp"}, "27468/tcp": {"Type": "tcp"}, "27469/tcp": {"Type": "tcp"}, "27470/tcp": {"Type": "tcp"}, "27471/tcp": {"Type": "tcp"}, "27472/tcp": {"Type": "tcp"}, "27473/tcp": {"Type": "tcp"}, "27474/tcp": {"Type": "tcp"}, "27475/tcp": {"Type": "tcp"}, "27476/tcp": {"Type": "tcp"}, "27477/tcp": {"Type": "tcp"}, "27478/tcp": {"Type": "tcp"}, "27479/tcp": {"Type": "tcp"}, "27480/tcp": {"Type": "tcp"}, "27481/tcp": {"Type": "tcp"}, "27482/tcp": {"Type": "tcp"}, "27483/tcp": {"Type": "tcp"}, "27484/tcp": {"Type": "tcp"}, "27485/tcp": {"Type": "tcp"}, "27486/tcp": {"Type": "tcp"}, "27487/tcp": {"Type": "tcp"}, "27488/tcp": {"Type": "tcp"}, "27489/tcp": {"Type": "tcp"}, "27490/tcp": {"Type": "tcp"}, "27491/tcp": {"Type": "tcp"}, "27492/tcp": {"Type": "tcp"}, "27493/tcp": {"Type": "tcp"}, "27494/tcp": {"Type": "tcp"}, "27495/tcp": {"Type": "tcp"}, "27496/tcp": {"Type": "tcp"}, "27497/tcp": {"Type": "tcp"}, "27498/tcp": {"Type": "tcp"}, "27499/tcp": {"Type": "tcp"}, "27500/tcp": {"Type": "tcp"}, "27501/tcp": {"Type": "tcp"}, "27502/tcp": {"Type": "tcp"}, "27503/tcp": {"Type": "tcp"}, "27504/tcp": {"Type": "tcp"}, "27505/tcp": {"Type": "tcp"}, "27506/tcp": {"Type": "tcp"}, "27507/tcp": {"Type": "tcp"}, "27508/tcp": {"Type": "tcp"}, "27509/tcp": {"Type": "tcp"}, "27510/tcp": {"Type": "tcp"}, "27511/tcp": {"Type": "tcp"}, "27512/tcp": {"Type": "tcp"}, "27513/tcp": {"Type": "tcp"}, "27514/tcp": {"Type": "tcp"}, "27515/tcp": {"Type": "tcp"}, "27516/tcp": {"Type": "tcp"}, "27517/tcp": {"Type": "tcp"}, "27518/tcp": {"Type": "tcp"}, "27519/tcp": {"Type": "tcp"}, "27520/tcp": {"Type": "tcp"}, "27521/tcp": {"Type": "tcp"}, "27522/tcp": {"Type": "tcp"}, "27523/tcp": {"Type": "tcp"}, "27524/tcp": {"Type": "tcp"}, "27525/tcp": {"Type": "tcp"}, "27526/tcp": {"Type": "tcp"}, "27527/tcp": {"Type": "tcp"}, "27528/tcp": {"Type": "tcp"}, "27529/tcp": {"Type": "tcp"}, "27530/tcp": {"Type": "tcp"}, "27531/tcp": {"Type": "tcp"}, "27532/tcp": {"Type": "tcp"}, "27533/tcp": {"Type": "tcp"}, "27534/tcp": {"Type": "tcp"}, "27535/tcp": {"Type": "tcp"}, "27536/tcp": {"Type": "tcp"}, "27537/tcp": {"Type": "tcp"}, "27538/tcp": {"Type": "tcp"}, "27539/tcp": {"Type": "tcp"}, "27540/tcp": {"Type": "tcp"}, "27541/tcp": {"Type": "tcp"}, "27542/tcp": {"Type": "tcp"}, "27543/tcp": {"Type": "tcp"}, "27544/tcp": {"Type": "tcp"}, "27545/tcp": {"Type": "tcp"}, "27546/tcp": {"Type": "tcp"}, "27547/tcp": {"Type": "tcp"}, "27548/tcp": {"Type": "tcp"}, "27549/tcp": {"Type": "tcp"}, "27550/tcp": {"Type": "tcp"}, "27551/tcp": {"Type": "tcp"}, "27552/tcp": {"Type": "tcp"}, "27553/tcp": {"Type": "tcp"}, "27554/tcp": {"Type": "tcp"}, "27555/tcp": {"Type": "tcp"}, "27556/tcp": {"Type": "tcp"}, "27557/tcp": {"Type": "tcp"}, "27558/tcp": {"Type": "tcp"}, "27559/tcp": {"Type": "tcp"}, "27560/tcp": {"Type": "tcp"}, "27561/tcp": {"Type": "tcp"}, "27562/tcp": {"Type": "tcp"}, "27563/tcp": {"Type": "tcp"}, "27564/tcp": {"Type": "tcp"}, "27565/tcp": {"Type": "tcp"}, "27566/tcp": {"Type": "tcp"}, "27567/tcp": {"Type": "tcp"}, "27568/tcp": {"Type": "tcp"}, "27569/tcp": {"Type": "tcp"}, "27570/tcp": {"Type": "tcp"}, "27571/tcp": {"Type": "tcp"}, "27572/tcp": {"Type": "tcp"}, "27573/tcp": {"Type": "tcp"}, "27574/tcp": {"Type": "tcp"}, "27575/tcp": {"Type": "tcp"}, "27576/tcp": {"Type": "tcp"}, "27577/tcp": {"Type": "tcp"}, "27578/tcp": {"Type": "tcp"}, "27579/tcp": {"Type": "tcp"}, "27580/tcp": {"Type": "tcp"}, "27581/tcp": {"Type": "tcp"}, "27582/tcp": {"Type": "tcp"}, "27583/tcp": {"Type": "tcp"}, "27584/tcp": {"Type": "tcp"}, "27585/tcp": {"Type": "tcp"}, "27586/tcp": {"Type": "tcp"}, "27587/tcp": {"Type": "tcp"}, "27588/tcp": {"Type": "tcp"}, "27589/tcp": {"Type": "tcp"}, "27590/tcp": {"Type": "tcp"}, "27591/tcp": {"Type": "tcp"}, "27592/tcp": {"Type": "tcp"}, "27593/tcp": {"Type": "tcp"}, "27594/tcp": {"Type": "tcp"}, "27595/tcp": {"Type": "tcp"}, "27596/tcp": {"Type": "tcp"}, "27597/tcp": {"Type": "tcp"}, "27598/tcp": {"Type": "tcp"},
```

9. Ping mongo container menggunakan alamat IP yang kita lihat di atas.



```
Preparing to unpack .../iputils-ping_3%3a20221126-1_amd64.deb ...
Unpacking iputils-ping (3:20221126-1) ...
Selecting previously unselected package libpam-cap:amd64.
Preparing to unpack .../libpam-cap_1%3a2.66-4_amd64.deb ...
Unpacking libpam-cap:amd64 (1:2.66-4) ...
Setting up libcap2-bin (1:2.66-4) ...
Setting up libpam-cap:amd64 (1:2.66-4) ...
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based front end cannot be used. at /usr/share/perl5/Debconf/FrontEnd/Dialog.pm line 78.)
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (Can't locate Term/ReadLine.pm in @INC (you may need to install the Term::ReadLine module) (@INC contains: /etc/perl /usr/local/lib/x86_64-linux-gnu/perl/5.36.0 /usr/local/share/perl/5.36.0 /usr/lib/x86_64-linux-gnu/perl5/5.36 /usr/share/perl5 /usr/lib/x86_64-linux-gnu/perl-base /usr/lib/x86_64-linux-gnu/perl/5.36 /usr/share/perl/5.36 /usr/local/lib/site_perl) at /usr/share/perl5/Debconf/FrontEnd/Readline.pm line 7.)
debconf: falling back to frontend: Teletype
Setting up iputils-ping (3:20221126-1) ...
root@dfd347d72911:/usr/local/apache2# ping 172.19.0.3
PING 172.19.0.3 (172.19.0.3) 56(84) bytes of data.
64 bytes from 172.19.0.3: icmp_seq=1 ttl=64 time=0.074 ms
64 bytes from 172.19.0.3: icmp_seq=2 ttl=64 time=0.054 ms
64 bytes from 172.19.0.3: icmp_seq=3 ttl=64 time=0.038 ms
64 bytes from 172.19.0.3: icmp_seq=4 ttl=64 time=0.089 ms
^C
--- 172.19.0.3 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3089ms
rtt min/avg/max/mdev = 0.038/0.063/0.089/0.019 ms
root@dfd347d72911:/usr/local/apache2#
```

10. Sekarang, mari kita coba melakukan ping ke container webdb dari jaringan lain. Cari tahu alamat IP dengan cara yang sama yang sama seperti yang kita lakukan sebelumnya.

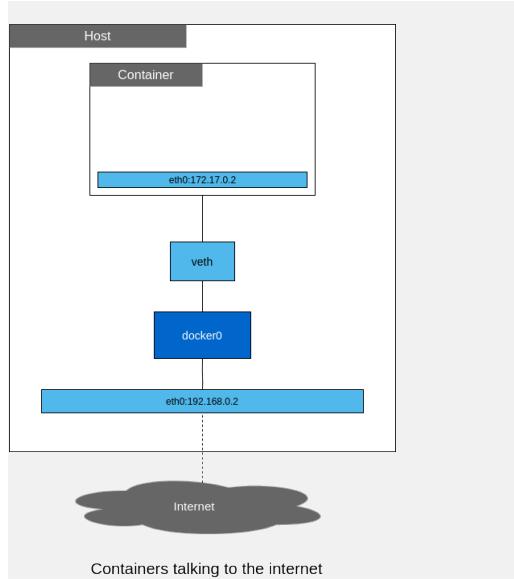


```
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (Can't locate Term/ReadLine.pm in @INC (you may need to install the Term::ReadLine module) (@INC contains: /etc/perl /usr/local/lib/x86_64-linux-gnu/perl/5.36.0 /usr/local/share/perl/5.36.0 /usr/lib/x86_64-linux-gnu/perl5/5.36 /usr/share/perl5 /usr/lib/x86_64-linux-gnu/perl-base /usr/lib/x86_64-linux-gnu/perl/5.36 /usr/share/perl/5.36 /usr/local/lib/site_perl) at /usr/share/perl5/Debconf/FrontEnd/Readline.pm line 7.)
debconf: falling back to frontend: Teletype
Setting up iputils-ping (3:20221126-1) ...
root@dfd347d72911:/usr/local/apache2# ping 172.19.0.3
PING 172.19.0.3 (172.19.0.3) 56(84) bytes of data.
64 bytes from 172.19.0.3: icmp_seq=1 ttl=64 time=0.074 ms
64 bytes from 172.19.0.3: icmp_seq=2 ttl=64 time=0.054 ms
64 bytes from 172.19.0.3: icmp_seq=3 ttl=64 time=0.038 ms
64 bytes from 172.19.0.3: icmp_seq=4 ttl=64 time=0.089 ms
^C
--- 172.19.0.3 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3089ms
rtt min/avg/max/mdev = 0.038/0.063/0.089/0.019 ms
root@dfd347d72911:/usr/local/apache2# ping 172.20.0.1
PING 172.20.0.1 (172.20.0.1) 56(84) bytes of data.
64 bytes from 172.20.0.1: icmp_seq=1 ttl=64 time=0.051 ms
64 bytes from 172.20.0.1: icmp_seq=2 ttl=64 time=0.034 ms
64 bytes from 172.20.0.1: icmp_seq=3 ttl=64 time=0.033 ms
64 bytes from 172.20.0.1: icmp_seq=4 ttl=64 time=0.035 ms
^C
--- 172.20.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3111ms
rtt min/avg/max/mdev = 0.033/0.038/0.051/0.007 ms
root@dfd347d72911:/usr/local/apache2#
```

Dengan demikian, kami dapat memastikan container dari aplikasi yang berbeda tidak dapat berbicara satu sama lain.

Info Bonus: Wadah di jaringan yang berbeda tidak dapat berkomunikasi satu sama lain karena mereka berada di subnet yang berbeda.

- **Bagaimana cara container berbicara dengan internet?**



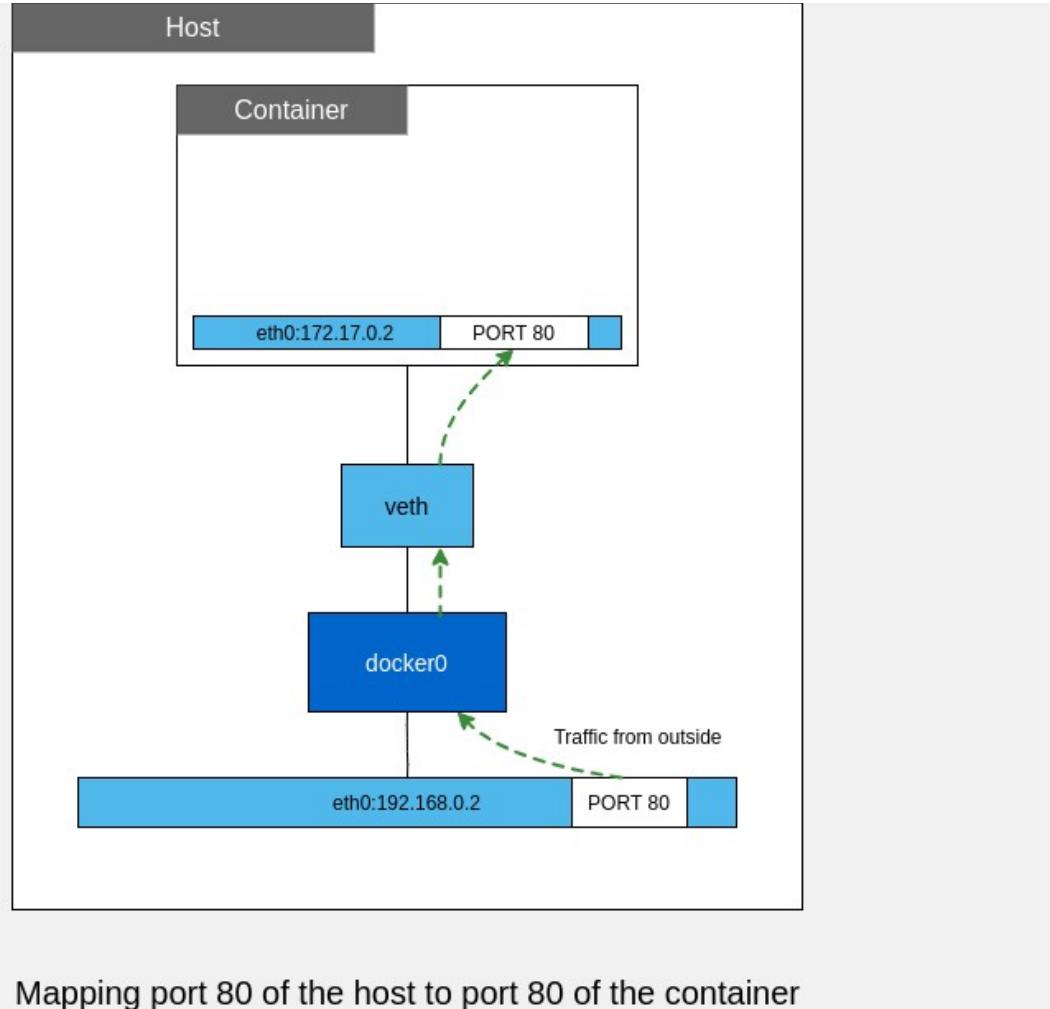
Kontainer berbicara dengan jaringan docker0/jaringan yang ditentukan pengguna menggunakan IP yang ditetapkan oleh jaringan ini.

Jaringan docker0/jaringan bridge menggunakan IP host pada antarmuka defaultnya untuk berbicara dengan internet.

Jaringan menjaga agar container benar-benar terisolasi dari apa pun yang tidak berada di jaringan yang sama.

- **Lalu, bagaimana dunia luar berinteraksi dengan container tersebut?**

Hal ini dapat dicapai dengan membuka port pada container dan memetakannya ke port host.



Mapping port 80 of the host to port 80 of the container

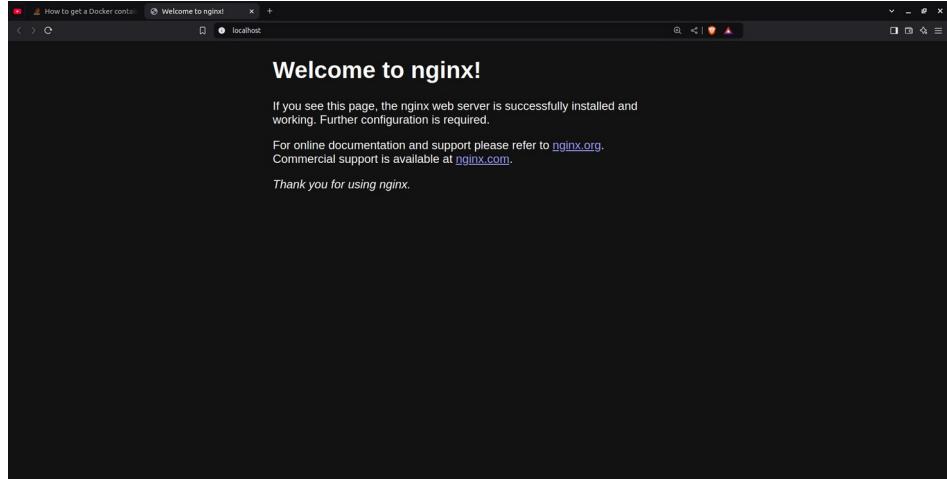
Mari kita lihat bagaimana cara melakukannya.

1. Jalankan nginx container dengan dan petakan port 80 host ke port 80 container.

```
agus@agus-Modern-14-C11M: ~ x agus@agus-Modern-14-C11M: ~ x agus@agus-Modern-14-C11M: ~ x
/ diff", "MergedDir": "/var/lib/docker/overlay2/b6020cc3bd502212dbe2cbea15977ea6f9a6cdde615b06fd7fb969c3cdcd/merged", "UpperDir": "/var/lib/docker/overlay2/b6020cc3bd502212dbe2cbea15977ea6f9a6cdde615b06fd7fb969c3cdcd/diff", "WorkDir": "/var/lib/docker/overlay2/b6020cc3bd502212dbe2cbea15977ea6f9a6cdde615b06fd7fb969c3cdcd/work", "Name": "Overlay2"}, "Mounts": [{"Type": "volume", "Name": "c58baccbb5be4478488037737ba99bd4fc693fcabc7d041765f728cc60b8", "Source": "/var/lib/docker/volumes/c58baccbb5be4478488037737ba99bd4fc693fcabc7d041765f728cc60b8", "Destination": "/data"}, {"Type": "volume", "Name": "cd25d055e0bbd290b1b1886a2624eac673709892595173f2ab7fbcc0fa787", "Source": "/var/lib/docker/volumes/cd25d055e0bbd290b1b1886a2624eac673709892595173f2ab7fbcc0fa787", "Destination": "/data"}, {"Type": "volume", "Name": "bf91f916dad", "Driver": "local", "Mode": "rw", "RW": "Propagation"}, {"Type": "volume", "Name": "3", "Driver": "local", "Mode": "rw", "RW": "Propagation"}, {"Type": "volume", "Name": "bf91f916dad", "Driver": "local", "Mode": "rw", "RW": "Propagation"}], "Config": {"Hostname": "5d31bf91f916dad", "Domainname": "User", "AttachStdin": false, "AttachStdout": false, "AttachStderr": false, "ExposedPorts": {"27017/tcp": {}}, "Tty": false, "OpenStdin": false, "StdinOnce": false, "Env": ["PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin"], "OS": "Ubuntu_18.04", "Version": "1.17", "JSYML_VERSION=3.13.1", "CHECKSUM=62e2319bbd378e9f1675678e5a34954ba0e33ac1a170a0bf4826fa24b941", "MONGO_PACKAGE=mongodb-org", "MONGO_REPO=repo.mongodb.org", "MONGO_MAJOR=8.0", "MONGO_VERSION=8.0.3", "HOME=/data/db", "Cnd": ["mongod"], "Image": "mongo", "Volumes": {"data/configdb": {}, "data/db": {}, "workdir": ""}, "Entrypoint": ["docker-entrypoint.sh"], "OnBuild": null, "Labels": {"org.opencontainers.image.ref.fqdn": "agus-Modern-14-C11M", "org.opencontainers.image.version": "24.04"}, "Networking": {"Bridge": "", "SandboxID": "8a3d12588bfe17a9b507fb373a2f2763d47a16ee886d02890df5d7834cb553", "SandboxKey": "/var/run/docker/netns/8a3d12588bfe17a9b507fb373a2f2763d47a16ee886d02890df5d7834cb553", "Address": "172.19.0.1", "Port": "27017/tcp", "HostIP": "172.19.0.1", "HostPort": "27017", "LinkLocalIPV6Address": "", "LinkLocalIPV6PrefixLen": 0, "SecondaryIPAddresses": null, "SecondaryIPV6Addresses": null, "EndpointID": "bf91f916dad", "Gateway": "172.19.0.1", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "IPAddress": "", "IPPrefixLen": 0, "IPv6Gateway": "", "MacAddress": "", "Networks": {"app": {"IPAMConfig": null, "Links": null, "Aliases": null, "MacAddress": "02:42:ac:13:00:03", "Driver": "bridge", "EndpointID": "896354959b47178593fb5172150a68737fcdfa3634850de4fc9d604c42c29f198bbed3f291e80", "DriverName": "bridge", "IP": "172.19.0.1", "IPPrefixLen": 16, "IPV6Gateway": "", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "DNSNames": [{"db": "5d31bf91f916dad"}]}}, "Labels": {"com.docker.compose.project": "agus-Modern-14-C11M", "com.docker.compose.service": "nginx", "com.docker.compose.config-hash": "169411351877ff0424ff5a7d22e15e824bfdbca07d0ca43fc6ff8a675abd05", "com.docker.compose.version": "1.28.1", "com.docker.compose.volume": "data"}, "Mounts": [{"Type": "volume", "Name": "bf91f916dad", "Source": "/var/lib/docker/volumes/bf91f916dad", "Destination": "/data"}, {"Type": "volume", "Name": "3", "Source": "/var/lib/docker/volumes/3", "Destination": "/data"}, {"Type": "volume", "Name": "bf91f916dad", "Source": "/var/lib/docker/volumes/bf91f916dad", "Destination": "/data"}, {"Type": "volume", "Name": "bf91f916dad", "Source": "/var/lib/docker/volumes/bf91f916dad", "Destination": "/data"}]
```

2. Mari kita lihat apakah pemetaan berhasil dilakukan.

3. Mari kita juga memverifikasi apakah kita dapat melihat server nginx yang berjalan pada port 80 host dengan menavigasi ke <http://localhost:80>



• Penamaan DNS

Dalam lingkungan docker, alamat IP tidak dapat diandalkan.

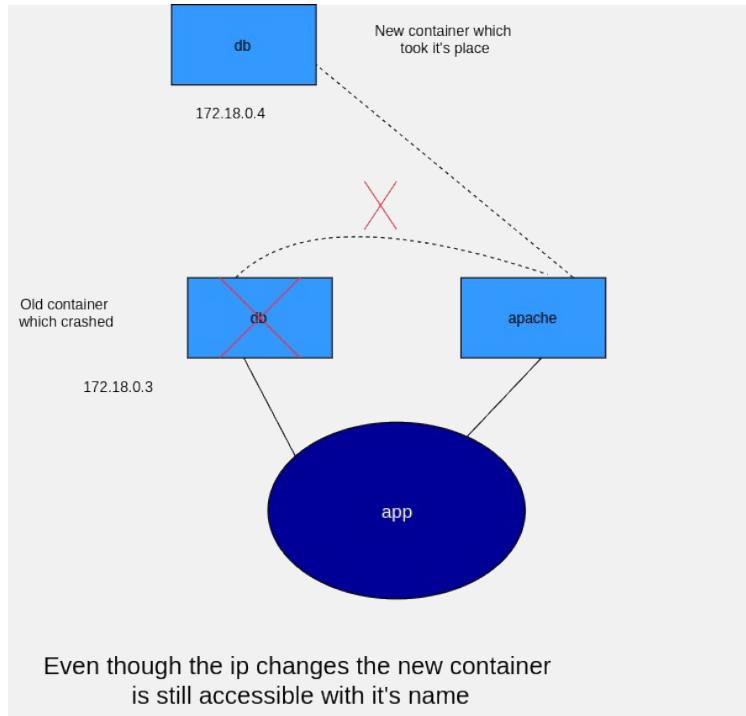
Ini karena container dapat macet/berhenti secara tidak terduga dan kemudian container baru menggantikannya dengan IP yang berbeda.

Jadi, bagaimana kita memastikan komunikasi dalam situasi seperti itu?

Di sinilah penamaan DNS berperan. Docker secara default menggunakan nama container sebagai nama host yang dapat digunakan oleh container untuk berbicara satu sama lain.

Catatan: Ini hanya berlaku untuk jaringan yang dibuat oleh pengguna. Jaringan bridge default tidak memiliki fitur ini.

Baca [artikel](#) ini untuk mengetahui perbedaan antara jaringan yang ditentukan pengguna dan jaringan default.



Mari kita lakukan latihan untuk melihat bagaimana hal ini bermanfaat.

1. Dalam sesi bash yang sama, ping container mongo menggunakan namanya.

```
agus@agus-Modern-14-C11M: ~  agus@agus-Modern-14-C11M: ~  agus@agus-Modern-14-C11M: ~
root@df347d72911:/usr/local/apache2# ping 172.19.0.3
PING 172.19.0.3 (172.19.0.3) 56(84) bytes of data.
64 bytes from 172.19.0.3: icmp_seq=1 ttl=64 time=0.074 ms
64 bytes from 172.19.0.3: icmp_seq=2 ttl=64 time=0.054 ms
64 bytes from 172.19.0.3: icmp_seq=3 ttl=64 time=0.038 ms
64 bytes from 172.19.0.3: icmp_seq=4 ttl=64 time=0.089 ms
^C
--- 172.19.0.3 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3089ms
rtt min/avg/max/mdev = 0.038/0.063/0.089/0.019 ms
root@df347d72911:/usr/local/apache2# ping 172.20.0.1
PING 172.20.0.1 (172.20.0.1) 56(84) bytes of data.
64 bytes from 172.20.0.1: icmp_seq=1 ttl=64 time=0.051 ms
64 bytes from 172.20.0.1: icmp_seq=2 ttl=64 time=0.034 ms
64 bytes from 172.20.0.1: icmp_seq=3 ttl=64 time=0.033 ms
64 bytes from 172.20.0.1: icmp_seq=4 ttl=64 time=0.035 ms
^C
--- 172.20.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3111ms
rtt min/avg/max/mdev = 0.033/0.038/0.051/0.007 ms
root@df347d72911:/usr/local/apache2# ping db
PING db (172.19.0.3) 56(84) bytes of data.
64 bytes from db.app (172.19.0.3): icmp_seq=1 ttl=64 time=0.053 ms
64 bytes from db.app (172.19.0.3): icmp_seq=2 ttl=64 time=0.052 ms
64 bytes from db.app (172.19.0.3): icmp_seq=3 ttl=64 time=0.046 ms
64 bytes from db.app (172.19.0.3): icmp_seq=4 ttl=64 time=0.085 ms
^C
--- db ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3106ms
rtt min/avg/max/mdev = 0.046/0.059/0.085/0.015 ms
root@df347d72911:/usr/local/apache2#
```

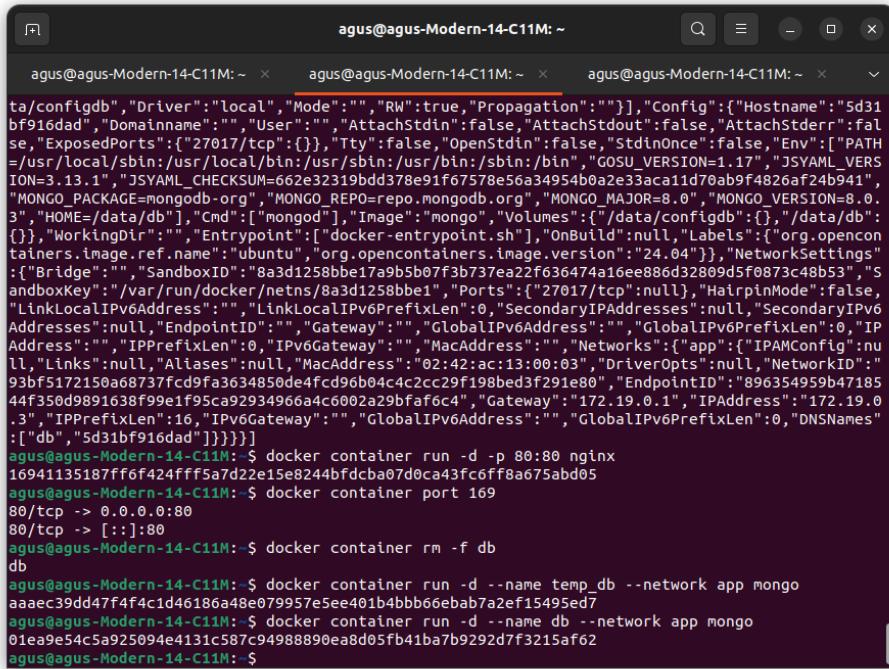
2. Hentikan dan hapus container ini di tab yang berbeda.

```
agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~
cd25d055e0bdb290b11886a2624eeac873709892595173f2ab7fb6ccfa0787/_data", "Destination": "/data/db", "Driver": "local", "Mode": "", "RW": true, "Propagation": ""}, {"Type": "volume", "Name": "cd25d055e0bdb290b11886a2624eeac873709892595173f2ab7fb6ccfa0787", "Source": "/var/lib/docker/volumes/cd25d055e0bdb290b11886a2624eeac873709892595173f2ab7fb6ccfa0787/_data", "Destination": "/data/configdb", "Driver": "local", "Mode": "", "RW": true, "Propagation": ""}], "Config": {"Hostname": "5d31bf916dad", "Domainname": "", "User": "", "AttachStdin": false, "AttachStdout": false, "AttachStderr": false, "ExposedPorts": [{"27017/tcp": {}}], "Env": [{"PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/bin", "GOSU_VERSION=1.17", "JSYAML_VERSION=3.13.1"}, {"MONGO_PACKAGE=mongodb-org", "MONGO_REPO=repo.mongodb.org", "MONGO_MAJOR=8.0", "MONGO_VERSION=8.0.3", "HOME=/data/db"}, {"Cmd": ["mongod"], "Image": "mongo", "Volumes": [{"/data/configdb": {}, "/data/db": {}}, {"WorkingDir": "", "Entrypoint": ["docker-entrypoint.sh"], "OnBuild": null, "Labels": {"org.opencontainers.image.ref.name": "ubuntu", "org.opencontainers.image.version": "24.04"}], "NetworkSettings": {"Bridge": "", "SandboxID": "8a3d1258bbe17a9b5b07f3b737ea22f636474a16ee886d32809d5f0873c48b53", "SandboxKey": "/var/run/docker/netns/8a3d1258bbe1", "Ports": [{"27017/tcp": null}, {"HairpinMode": false, "LinkLocalIPv6Address": "", "LinkLocalIPv6PrefixLen": 0, "SecondaryIPAddresses": null, "SecondaryIPv6Addresses": null, "EndpointID": "", "Gateway": "", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "IPAddress": "", "IPPrefixLen": 0, "IPv6Gateway": "", "MacAddress": "", "Networks": {"app": {"IPAMConfig": null, "Links": null, "Aliases": null, "MacAddress": "02:42:ac:13:00:03", "DriverOpts": null, "NetworkID": "93bf5172150a68737fcdf9fa3634850de4fc96b04c4c2cc29f198bed3f291e80"}, "EndpointID": "896354959b4718544f350d9891638f99ef95ca92934966a4c6002a29bfaf6c4", "Gateway": "172.19.0.1", "IPAddress": "172.19.0.3", "IPPrefixLen": 16, "IPv6Gateway": "", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "DNSNames": [{"db": "5d31bf916dad"}]}]}]}], "agus@agus-Modern-14-C11M: $ docker container run -d -p 80:80 nginx 16941135187ff6f424fff5a7d22e15e8244bfdcba07d0ca43fc6ff8a675abd05 agus@agus-Modern-14-C11M: $ docker container port 169 80/tcp -> 0.0.0.0:80 80/tcp -> [::]:80 agus@agus-Modern-14-C11M: $ docker container rm -f db db agus@agus-Modern-14-C11M: $
```

3. Mulai container apa pun di jaringan yang sama sehingga IP 172.20.0.3 dialokasikan untuknya.

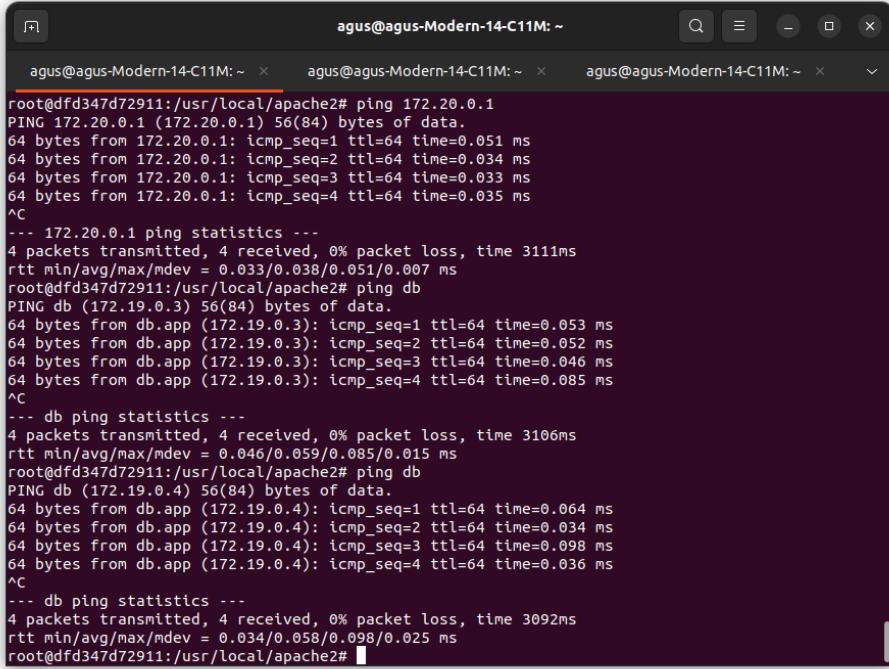
```
agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~
cd25d055e0bdb290b11886a2624eeac873709892595173f2ab7fb6ccfa0787/_data", "Source": "/var/lib/docker/volumes/cd25d055e0bdb290b11886a2624eeac873709892595173f2ab7fb6ccfa0787/_data", "Destination": "/data/db", "Driver": "local", "Mode": "", "RW": true, "Propagation": ""}, {"Type": "volume", "Name": "cd25d055e0bdb290b11886a2624eeac873709892595173f2ab7fb6ccfa0787", "Source": "/var/lib/docker/volumes/cd25d055e0bdb290b11886a2624eeac873709892595173f2ab7fb6ccfa0787/_data", "Destination": "/data/configdb", "Driver": "local", "Mode": "", "RW": true, "Propagation": ""}], "Config": {"Hostname": "5d31bf916dad", "Domainname": "", "User": "", "AttachStdin": false, "AttachStdout": false, "AttachStderr": false, "ExposedPorts": [{"27017/tcp": {}}], "Env": [{"PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/bin", "GOSU_VERSION=1.17", "JSYAML_VERSION=3.13.1"}, {"MONGO_PACKAGE=mongodb-org", "MONGO_REPO=repo.mongodb.org", "MONGO_MAJOR=8.0", "MONGO_VERSION=8.0.3", "HOME=/data/db"}, {"Cmd": ["mongod"], "Image": "mongo", "Volumes": [{"/data/configdb": {}, "/data/db": {}}, {"WorkingDir": "", "Entrypoint": ["docker-entrypoint.sh"], "OnBuild": null, "Labels": {"org.opencontainers.image.ref.name": "ubuntu", "org.opencontainers.image.version": "24.04"}], "NetworkSettings": {"Bridge": "", "SandboxID": "8a3d1258bbe17a9b5b07f3b737ea22f636474a16ee886d32809d5f0873c48b53", "SandboxKey": "/var/run/docker/netns/8a3d1258bbe1", "Ports": [{"27017/tcp": null}, {"HairpinMode": false, "LinkLocalIPv6Address": "", "LinkLocalIPv6PrefixLen": 0, "SecondaryIPAddresses": null, "SecondaryIPv6Addresses": null, "EndpointID": "", "Gateway": "", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "IPAddress": "", "IPPrefixLen": 0, "IPv6Gateway": "", "MacAddress": "", "Networks": {"app": {"IPAMConfig": null, "Links": null, "Aliases": null, "MacAddress": "02:42:ac:13:00:03", "DriverOpts": null, "NetworkID": "93bf5172150a68737fcdf9fa3634850de4fc96b04c4c2cc29f198bed3f291e80"}, "EndpointID": "896354959b4718544f350d9891638f99ef95ca92934966a4c6002a29bfaf6c4", "Gateway": "172.19.0.1", "IPAddress": "172.19.0.3", "IPPrefixLen": 16, "IPv6Gateway": "", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "DNSNames": [{"db": "5d31bf916dad"}]}]}]}], "agus@agus-Modern-14-C11M: $ docker container run -d -p 80:80 nginx 16941135187ff6f424fff5a7d22e15e8244bfdcba07d0ca43fc6ff8a675abd05 agus@agus-Modern-14-C11M: $ docker container port 169 80/tcp -> 0.0.0.0:80 80/tcp -> [::]:80 agus@agus-Modern-14-C11M: $ docker container rm -f db db agus@agus-Modern-14-C11M: $ docker container run -d --name temp_db --network app mongo aaaec39dd47f4f4c1d46186448e079957e5ee401b4bbb6ebab7a2ef15495ed7 agus@agus-Modern-14-C11M: $
```

4. Memulai container mongo lain dengan nama yang sama.



```
agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~
[{"id": "1", "text": "cat /configdb", "type": "text"}, {"id": "2", "text": "Driver": "local", "type": "text"}, {"id": "3", "text": "Mode": "", "type": "text"}, {"id": "4", "text": "RW": true, "type": "text"}, {"id": "5", "text": "Propagation": ""}], "Config": [{"Hostname": "5d31bf916dad", "Domainname": "", "User": "", "AttachStdin": false, "AttachStdout": false, "AttachStderr": false, "ExposedPorts": ["27017/tcp"]}, {"Tty": false, "OpenStdin": false, "StdinOnce": false, "Env": ["PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/bin"], "Image": "mongo", "Volumes": ["/data/configdb:/data/db"], "WorkingDir": "", "Entrypoint": ["docker-entrypoint.sh"], "OnBuild": null, "Labels": {"org.opencontainers.image.ref.name": "ubuntu", "org.opencontainers.image.version": "24.04"}, "NetworkSettings": {"Bridge": "", "SandboxID": "8a3d1258bbe17a9b5b07f3b737ea22f636474a16ee886d32809d5f0873c48b53", "SandboxKey": "/var/run/docker/netns/8a3d1258bbe1", "Ports": [{"27017/tcp": null}], "HairpinMode": false, "LinkLocalIPv6Address": null, "LinkLocalIPv6PrefixLen": 0, "SecondaryIPv6Addresses": null, "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "IP Address": "", "IPPrefixLen": 0, "IPv6Gateway": "", "MacAddress": "", "Networks": [{"app": {"TPAMConfig": null, "Aliases": null, "MacAddress": null, "DriverOpts": null, "NetworkID": "93bf5172150a68737fcdf9fa3634850de4fc96b04c4c2cc29f198bed3f291e80"}, "EndpointID": "896354959b4718544f350d9891638ff99e1f95ca92934966a4c6002a29bfaf6c4", "Gateway": "172.19.0.1", "IPAddress": "172.19.0.3", "IPPrefixLen": 16, "IPv6Gateway": "", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "DNSNames": ["db", "5d31bf916dad"]}}}], "agus@agus-Modern-14-C11M: $ docker container run -d -p 80:80 nginx 16941135187ff6f424fff5a7d22e15e8244bfdcb07d0ca43fc6ff8a675abd05", "agus@agus-Modern-14-C11M: $ docker container port 169 80/tcp -> 0.0.0.0:80 80/tcp -> [::]:80", "agus@agus-Modern-14-C11M: $ docker container rm -f db db", "agus@agus-Modern-14-C11M: $ docker container run -d --name temp_db --network app mongo aaaec39dd47f4f4c1d46186a48e079957e5ee401b4bb66ebab7a2ef15495ed7", "agus@agus-Modern-14-C11M: $ docker container run -d --name db --network app mongo 01ea9e54c5a925094e4131c587c94988890ea8d05fb41ba7b9292d7f3215af62", "agus@agus-Modern-14-C11M: $"}]
```

5. Coba ping container db lagi dari container apache.



```
agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~
[{"id": "1", "text": "root@dfd347d72911:/usr/local/apache2# ping 172.20.0.1", "type": "text"}, {"id": "2", "text": "PING 172.20.0.1 (172.20.0.1) 56(84) bytes of data.", "type": "text"}, {"id": "3", "text": "64 bytes from 172.20.0.1: icmp_seq=1 ttl=64 time=0.051 ms", "type": "text"}, {"id": "4", "text": "64 bytes from 172.20.0.1: icmp_seq=2 ttl=64 time=0.034 ms", "type": "text"}, {"id": "5", "text": "64 bytes from 172.20.0.1: icmp_seq=3 ttl=64 time=0.033 ms", "type": "text"}, {"id": "6", "text": "64 bytes from 172.20.0.1: icmp_seq=4 ttl=64 time=0.035 ms", "type": "text"}, {"id": "7", "text": "^C", "type": "text"}, {"id": "8", "text": "... 172.20.0.1 ping statistics ...", "type": "text"}, {"id": "9", "text": "4 packets transmitted, 4 received, 0% packet loss, time 3111ms", "type": "text"}, {"id": "10", "text": "rtt min/avg/max/mdev = 0.033/0.038/0.051/0.007 ms", "type": "text"}, {"id": "11", "text": "root@dfd347d72911:/usr/local/apache2# ping db", "type": "text"}, {"id": "12", "text": "PING db (172.19.0.3) 56(84) bytes of data.", "type": "text"}, {"id": "13", "text": "64 bytes from db.app (172.19.0.3): icmp_seq=1 ttl=64 time=0.053 ms", "type": "text"}, {"id": "14", "text": "64 bytes from db.app (172.19.0.3): icmp_seq=2 ttl=64 time=0.052 ms", "type": "text"}, {"id": "15", "text": "64 bytes from db.app (172.19.0.3): icmp_seq=3 ttl=64 time=0.046 ms", "type": "text"}, {"id": "16", "text": "64 bytes from db.app (172.19.0.3): icmp_seq=4 ttl=64 time=0.085 ms", "type": "text"}, {"id": "17", "text": "^C", "type": "text"}, {"id": "18", "text": "... db ping statistics ...", "type": "text"}, {"id": "19", "text": "4 packets transmitted, 4 received, 0% packet loss, time 3106ms", "type": "text"}, {"id": "20", "text": "rtt min/avg/max/mdev = 0.046/0.059/0.085/0.015 ms", "type": "text"}, {"id": "21", "text": "root@dfd347d72911:/usr/local/apache2# ping db", "type": "text"}, {"id": "22", "text": "PING db (172.19.0.4) 56(84) bytes of data.", "type": "text"}, {"id": "23", "text": "64 bytes from db.app (172.19.0.4): icmp_seq=1 ttl=64 time=0.064 ms", "type": "text"}, {"id": "24", "text": "64 bytes from db.app (172.19.0.4): icmp_seq=2 ttl=64 time=0.034 ms", "type": "text"}, {"id": "25", "text": "64 bytes from db.app (172.19.0.4): icmp_seq=3 ttl=64 time=0.098 ms", "type": "text"}, {"id": "26", "text": "64 bytes from db.app (172.19.0.4): icmp_seq=4 ttl=64 time=0.036 ms", "type": "text"}, {"id": "27", "text": "^C", "type": "text"}, {"id": "28", "text": "... db ping statistics ...", "type": "text"}, {"id": "29", "text": "4 packets transmitted, 4 received, 0% packet loss, time 3092ms", "type": "text"}, {"id": "30", "text": "rtt min/avg/max/mdev = 0.034/0.058/0.098/0.025 ms", "type": "text"}, {"id": "31", "text": "root@dfd347d72911:/usr/local/apache2#", "type": "text"}]
```

Anda akan melihat bahwa meskipun IP container telah berubah , kami dapat melakukan ping menggunakan DNS (nama container).

Hal ini memastikan komunikasi yang andal dalam lingkungan yang sangat dinamis.

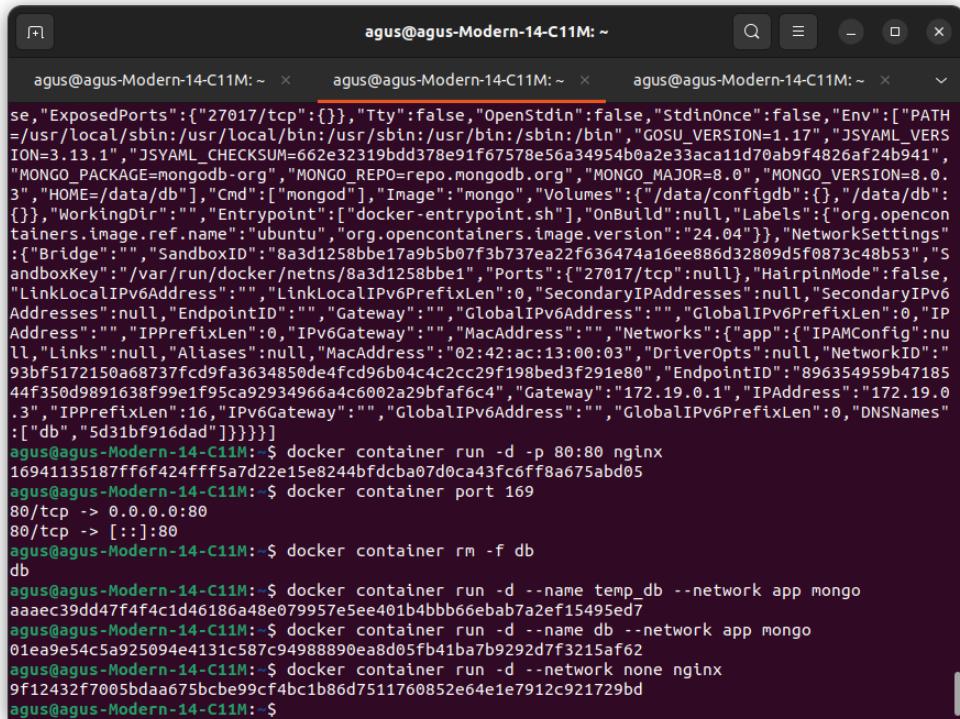
Catatan : Pada jaringan docker0 untuk berkomunikasi menggunakan DNS, kita harus menggunakan flag tautan saat membuat container yang sudah tidak digunakan lagi dan tidak boleh digunakan. Anda sebaiknya membuat jaringan baru.

Baca [artikel](#) ini untuk mengetahui lebih lanjut tentang hal ini.

- **Jaringan None**

Ketika Anda menentukan container untuk tidak menggunakan jaringan, maka container tersebut akan sepenuhnya terisolasi. Ia tidak akan dapat berbicara dengan container lain atau internet.

Tidak ada alamat IP yang diberikan ke container tersebut.



```
agus@agus-Modern-14-C11M: ~      agus@agus-Modern-14-C11M: ~      agus@agus-Modern-14-C11M: ~
se,"ExposedPorts":{"27017/tcp":{}}, "Tty":false, "OpenStdin":false, "StdinOnce":false, "Env": ["PATH =/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin", "GOSU_VERSION=1.17", "JSYAML_VERS
ION=3.13.1", "JSYAML_CHECKSUM=662e32319bdd378e91f67578e56a34954b0a2e33aca11d70ab9f4826af24b941",
"MONGO_PACKAGE=mongodb-org", "MONGO_REPO=repo.mongodb.org", "MONGO_MAJOR=8.0", "MONGO_VERSION=8.0.
3", "HOME=/data/db"], "Cmd": ["mongod"], "Image": "mongo", "Volumes": {"/data/configdb": {}, "/data/db": {}}, "WorkingDir": "", "Entrypoint": ["docker-entrypoint.sh"], "OnBuild": null, "Labels": {"org.opencontainers.image.ref.name": "ubuntu", "org.opencontainers.image.version": "24.04"}, "NetworkSettings": {"Bridge": "", "SandboxID": "8a3d1258bbe17a9b5b07f3b737ea22f636474a16ee886d32809d5f0873c48b53", "SandboxKey": "/var/run/docker/netns/8a3d1258bbe1", "Ports": [{"27017/tcp": null}], "HairpinMode": false, "LinkLocalIPv6Address": "", "LinkLocalIPv6PrefixLen": 0, "SecondaryIPv6Addresses": null, "SecondaryIPv6
Addresses": null, "EndpointID": "", "Gateway": "", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "IP
Address": "", "IPPrefixLen": 0, "IPv6Gateway": "", "MacAddress": "", "Networks": {"app": {"IPAMConfig": null, "Links": null, "Aliases": null, "MacAddress": "02:42:ac:13:00:03", "DriverOpts": null, "NetworkID": "93bf5172150a68737fcdf9fa3634850de4fcfd96b04c4c2cc29f198bedf291e80", "EndpointID": "896354959b47185
44f350d9891638f99e1f95ca92934966a4c6002a29bfaf6c4", "Gateway": "172.19.0.1", "IPAddress": "172.19.0
.3", "IPPrefixLen": 16, "IPv6Gateway": "", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "DNSNames": [{"db": "5d31bf916dad"}]}]}
agus@agus-Modern-14-C11M: $ docker container run -d -p 80:80 nginx
16941135187ff6t424fff5a7d22e15e8244bfdfcba07d0ca43fc6ff8a675abd05
agus@agus-Modern-14-C11M: $ docker container port 169
80/tcp -> 0.0.0.0:80
80/tcp -> [::]:80
agus@agus-Modern-14-C11M: $ docker container rm -f db
db
agus@agus-Modern-14-C11M: $ docker container run -d --name temp_db --network app mongo
aaaec39dd47f4fc41d46186a48e07957e5ee401b4bbb66ebab7a2ef15495ed7
agus@agus-Modern-14-C11M: $ docker container run -d --name db --network app mongo
01ea9e54c5a925094e4131c587c94988890ea8d05fb41ba7b9292d7f3215af62
agus@agus-Modern-14-C11M: $ docker container run -d --network none nginx
9f12432f7005bdaa675bcbe99cf4bc1b86d7511760852e64e1e7912c921729bd
agus@agus-Modern-14-C11M: $
```

Mari kita coba mencatat alamat IP dari container ini.

```
agus@agus-Modern-14-C11M:~$ docker container inspect 9f12
```

Anda akan melihat tidak ada alamat IP yang ditetapkan untuk container ini.

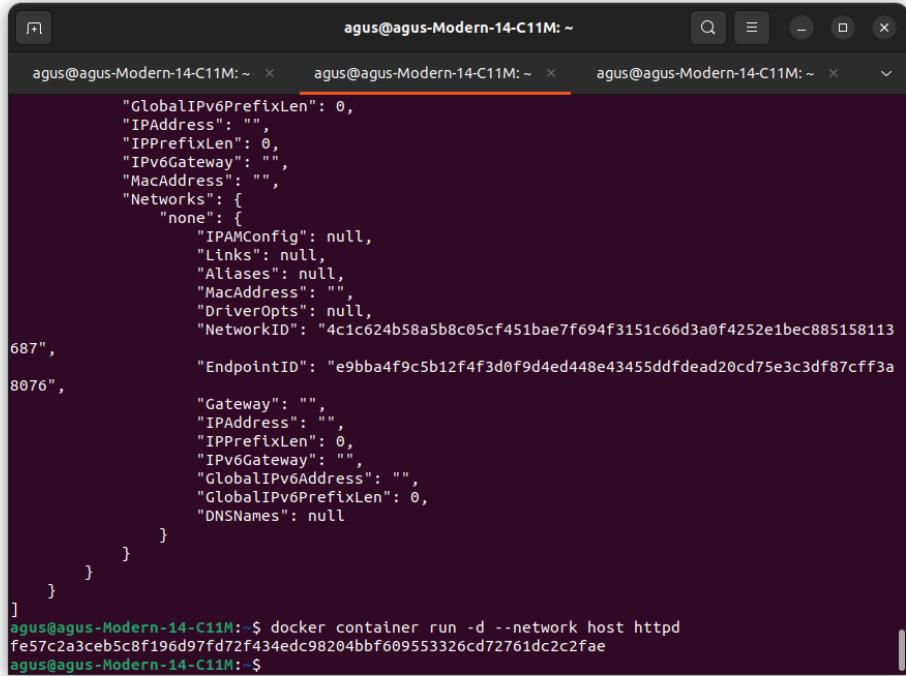
• Jaringan host

Terkadang ada kebutuhan untuk menggunakan jaringan host itu sendiri. Dalam hal ini Anda kehilangan beberapa manfaat containerisasi tetapi jaringan menjadi cepat.

Dalam hal ini tidak ada jaringan baru atau alamat IP baru yang ditetapkan. Kontainer menggunakan IP host itu sendiri.

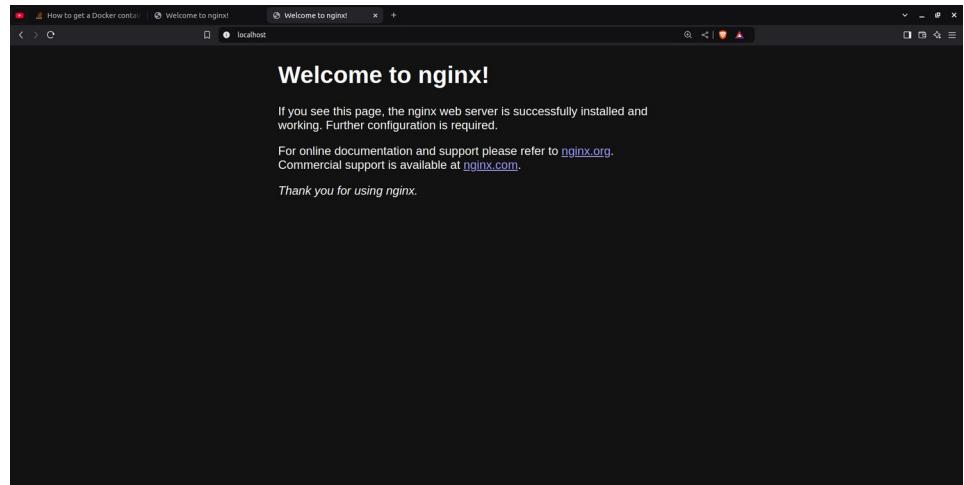
Catatan: Driver jaringan host hanya bekerja pada host Linux, dan tidak didukung pada Docker Desktop untuk Mac, Docker Desktop untuk Windows, atau Docker EE untuk Windows Server.

Mari kita verifikasi hal ini.



```
agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~
"GlobalIPv6PrefixLen": 0,
"IPAddress": "",
"IPPrefixLen": 0,
"IPv6Gateway": "",
"MacAddress": "",
"Networks": [
    "none": {
        "IPAMConfig": null,
        "Links": null,
        "Aliases": null,
        "MacAddress": "",
        "DriverOpts": null,
        "NetworkID": "4c1c624b58a5b8c05cf451bae7f694f3151c66d3a0f4252e1bec885158113
687",
        "EndpointID": "e9bba4f9c5b12f4f3d0f9d4ed448e43455ddfdead20cd75e3c3df87cff3a
8076",
        "Gateway": "",
        "IPAddress": "",
        "IPPrefixLen": 0,
        "IPv6Gateway": "",
        "GlobalIPv6Address": "",
        "GlobalIPv6PrefixLen": 0,
        "DNSNames": null
    }
}
]
agus@agus-Modern-14-C11M: ~ docker container run -d --network host httpd
fe57c2a3ceb5c8f196d97fd72f434edc98204bbf609553326cd72761dc2c2fae
agus@agus-Modern-14-C11M: ~
```

Navigasikan ke <http://localhost>.



Anda akan melihat bahwa server apache berjalan pada port host 80 tanpa mempublikasikan port ini. Ini karena container menggunakan jaringan host itu sendiri.

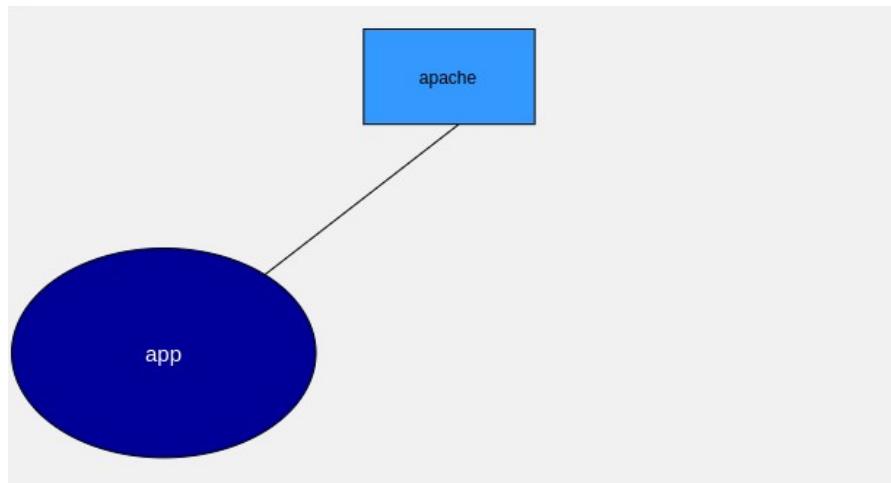
- **Menambahkan dan menghapus container ke jaringan secara dinamis.**

Kontainer dapat dIPasang dan dihapus dari jaringan mana pun secara dinamis.

Mari kita lihat bagaimana cara melakukannya.

1. Jalankan container apache dan lampirkan ke jaringan aplikasi.

```
agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~
[{"IPPrefixLen": 0,
 "IPv6Gateway": "",
 "MacAddress": "",
 "Networks": [
     {"none": {
         "IPAMConfig": null,
         "Links": null,
         "Aliases": null,
         "MacAddress": "",
         "DriverOpts": null,
         "NetworkID": "4c1c624b58a5b8c05cf451bae7f694f3151c66d3a0f4252e1bec885158113
687",
         "EndpointID": "e9bba4f9c5b12f4f3d0f9d4ed448e43455ddfded20cd75e3c3df87cff3a
8076",
         "Gateway": "",
         "IPAddress": "",
         "IPPrefixLen": 0,
         "IPv6Gateway": "",
         "GlobalIPv6Address": "",
         "GlobalIPv6PrefixLen": 0,
         "DNSNames": null
     }
    }
   ]
}
]
agus@agus-Modern-14-C11M: ~$ docker container run -d --network host httpd
fes7c2a3ceb5c8f196d97fd72f434edc98204bbf609553326cd72761dc2c2fae
agus@agus-Modern-14-C11M: ~$ docker container run -d --network app httpd
de056b5c807b5cdf1ac07cd058b58b48ac4be3876559a6d13ebe1aaife857a55
agus@agus-Modern-14-C11M: ~$
```



2. Buat jaringan baru dengan nama <web>.

```
agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~
[{"IPPrefixLen": 0,
 "Links": null,
 "Aliases": null,
 "MacAddress": "",
 "DriverOpts": null,
 "NetworkID": "4c1c624b58a5b8c05cf451bae7f694f3151c66d3a0f4252e1bec885158113
687",
 "EndpointID": "e9bba4f9c5b12f4f3d0f9d4ed448e43455ddfded20cd75e3c3df87cff3a
8076",
 "Gateway": "",
 "IPAddress": "",
 "IPPrefixLen": 0,
 "IPv6Gateway": "",
 "GlobalIPv6Address": "",
 "GlobalIPv6PrefixLen": 0,
 "DNSNames": null
}
]
]
agus@agus-Modern-14-C11M: ~$ docker container run -d --network host httpd
fes7c2a3ceb5c8f196d97fd72f434edc98204bbf609553326cd72761dc2c2fae
agus@agus-Modern-14-C11M: ~$ docker container run -d --network web httpd
de056b5c807b5cdf1ac07cd058b58b48ac4be3876559a6d13ebe1aaife857a55
agus@agus-Modern-14-C11M: ~$ docker network create web
Error response from daemon: network with name web already exists
agus@agus-Modern-14-C11M: ~$ docker network create agus
cda76a95d6285967a98043b7a78c6d651c10c54e82495325d2af0b14e630d9
agus@agus-Modern-14-C11M: ~$
```

3. Verifikasi bahwa jaringan telah dibuat.

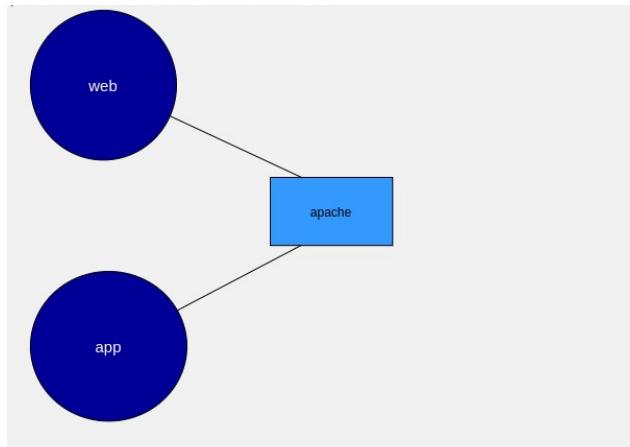
```
agus@agus-Modern-14-C11M: ~ x agus@agus-Modern-14-C11M: ~ x agus@agus-Modern-14-C11M: ~ x
8076",
        "Gateway": "",
        "IPAddress": "",
        "IPPrefixLen": 0,
        "IPv6Gateway": "",
        "GlobalIPv6Address": "",
        "GlobalIPv6PrefixLen": 0,
        "DNSNames": null
    }
}
}
]
]

agus@agus-Modern-14-C11M: $ docker container run -d --network host httpd
fe57c2a3ceb5c8f196d97fd72f434edc98204bbf609553326cd72761dc2c2fae
agus@agus-Modern-14-C11M: $ docker container run -d --network app httpd
de056b5c807b5cd1ac07cd058b58b48ac4be387659a6d13eb1aa1fe857a55
agus@agus-Modern-14-C11M: $ docker network create web
Error response from daemon: network with name web already exists
agus@agus-Modern-14-C11M: $ docker network create agus
cda76a95d6285967a98043b7a70c6d651c10c54e82495325d2af0b014e630d9
agus@agus-Modern-14-C11M: $ docker network ls
NETWORK ID      NAME      DRIVER      SCOPE
cda76a95d62     agus      bridge      local
93bf5172150a    app       bridge      local
983f9fcda3b6   bridge     bridge      local
695f6095271b   host      host       local
4c1c624b58a5   none      null       local
4c8ec8687093   virkom-project2_default  bridge      local
16046226af31   web       bridge      local
agus@agus-Modern-14-C11M: ~$
```

4. Memasang container apache ke jaringan web.

```
agus@agus-Modern-14-C11M: ~ x agus@agus-Modern-14-C11M: ~ x agus@agus-Modern-14-C11M: ~ x
        "Gateway": "",
        "IPAddress": "",
        "IPPrefixLen": 0,
        "IPv6Gateway": "",
        "GlobalIPv6Address": "",
        "GlobalIPv6PrefixLen": 0,
        "DNSNames": null
    }
}
]
]

agus@agus-Modern-14-C11M: $ docker container run -d --network host httpd
fe57c2a3ceb5c8f196d97fd72f434edc98204bbf609553326cd72761dc2c2fae
agus@agus-Modern-14-C11M: $ docker container run -d --network app httpd
de056b5c807b5cd1ac07cd058b58b48ac4be387659a6d13eb1aa1fe857a55
agus@agus-Modern-14-C11M: $ docker network create web
Error response from daemon: network with name web already exists
agus@agus-Modern-14-C11M: $ docker network create agus
cda76a95d6285967a98043b7a70c6d651c10c54e82495325d2af0b014e630d9
agus@agus-Modern-14-C11M: $ docker network ls
NETWORK ID      NAME      DRIVER      SCOPE
cda76a95d62     agus      bridge      local
93bf5172150a    app       bridge      local
983f9fcda3b6   bridge     bridge      local
695f6095271b   host      host       local
4c1c624b58a5   none      null       local
4c8ec8687093   virkom-project2_default  bridge      local
16046226af31   web       bridge      local
agus@agus-Modern-14-C11M: ~$ docker network connect agus de05
agus@agus-Modern-14-C11M: ~$
```



5. Verifikasi bahwa container terpasang ke kedua jaringan.

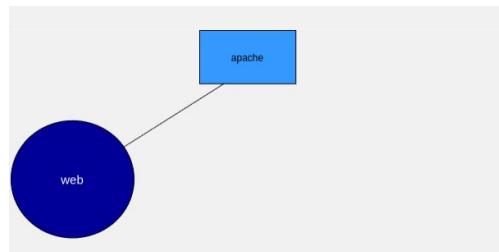
```
agus@agus-Modern-14-C11M:~$ docker container inspect de05
```

```
"Networks": [
    {
        "tag": "eth0",
        "IPAMConfig": {},
        "Links": null,
        "Aliases": [],
        "MacAddress": "02:42:ac:15:00:02",
        "DriverOpts": {},
        "NetworkID": "cdaf76a95d6285967a98043b7a70c6d651c054e82495325d2af0b8014e630d9",
        "EndpointID": "8bcab2da7bc44f5563ef74565b13d748f067139f955d2e57a3b076f8181dc",
        "Gateway": "172.21.0.1",
        "IPAddress": "172.21.0.2",
        "IPPrefixLen": 16,
        "IPv6Gateway": "",
        "GlobalIPv6Address": "",
        "GlobalIPv6PrefixLen": 0,
        "DNSNames": [
            "relaxed_cannon",
            "de05eb5c807b"
        ]
    },
    {
        "app": {
            "IPAMConfig": null,
            "Links": null,
            "Aliases": null,
            "MacAddress": "02:42:ac:13:00:05",
            "DriverOpts": null,
            "NetworkID": "93bf5172150a68737cd9fa3634850de4fc96804c4c2cc29f198bed3f291e80",
            "EndpointID": "d1b13f62be832bb6e40e8d751d85c109754d1c268f74e4abf0055221310eafc9",
            "Gateway": "172.19.0.1",
            "IPAddress": "172.19.0.5",
            "IPPrefixLen": 16,
            "IPv6Gateway": "",
            "GlobalIPv6Address": "",
            "GlobalIPv6PrefixLen": 0,
            "DNSNames": [
                "relaxed_cannon",
                "de05eb5c807b"
            ]
        }
    }
]
```

Anda akan melihat container container yang sama berada di dua jaringan yang berbeda dan memiliki dua alamat IP yang berbeda.

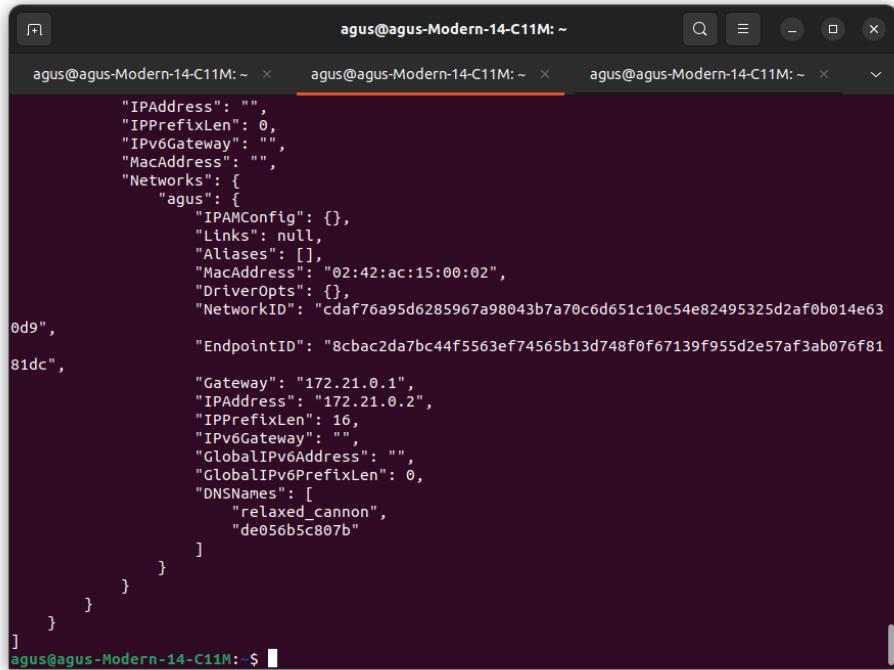
6. Mari hapus container dari jaringan aplikasi sekarang

```
agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~ agus@agus-Modern-14-C11M: ~
[{"id": "de056b5c807b", "name": "relaxed_cannon", "IPAMConfig": null, "Links": null, "Aliases": null, "MacAddress": "02:42:ac:13:00:05", "DriverOpts": null, "NetworkID": "93bf5172150a68737fc9fa3634850de4fc96b04c4cc2c29f198bed3f291e80", "EndpointID": "d1b11f2be832b66e40e8d751d85c109754d1c268f72e4abf0055221310eafc9", "Gateway": "172.19.0.1", "IPAddress": "172.19.0.5", "IPPrefixLen": 16, "IPv6Gateway": "", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "DNSNames": [{"name": "relaxed_cannon", "IP": "de056b5c807b"}]}, {"id": "de056b5c807b", "name": "relaxed_cannon", "IPAMConfig": null, "Links": null, "Aliases": null, "MacAddress": "02:42:ac:13:00:05", "DriverOpts": null, "NetworkID": "93bf5172150a68737fc9fa3634850de4fc96b04c4cc2c29f198bed3f291e80", "EndpointID": "d1b11f2be832b66e40e8d751d85c109754d1c268f72e4abf0055221310eafc9", "Gateway": "172.19.0.1", "IPAddress": "172.19.0.5", "IPPrefixLen": 16, "IPv6Gateway": "", "GlobalIPv6Address": "", "GlobalIPv6PrefixLen": 0, "DNSNames": [{"name": "relaxed_cannon", "IP": "de056b5c807b"}]}]
agus@agus-Modern-14-C11M: ~ docker network disconnect app de05
agus@agus-Modern-14-C11M: ~
```



7. Mari kita verifikasi bahwa aplikasi tersebut telah dihapus dari jaringan aplikasi.

```
agus@agus-Modern-14-C11M: $ docker container inspect de05
```



```
    "IPAddress": "",  
    "IPPrefixLen": 0,  
    "IPv6Gateway": "",  
    "MacAddress": "",  
    "Networks": {  
        "agus": {  
            "IPAMConfig": {},  
            "Links": null,  
            "Aliases": [],  
            "MacAddress": "02:42:ac:15:00:02",  
            "DriverOpts": {},  
            "NetworkID": "cdaf76a95d6285967a98043b7a70c6d651c10c54e82495325d2af0b014e63  
0d9",  
            "EndpointID": "8cbc2da7bc44f5563ef74565b13d748f0f67139f955d2e57af3ab076f81  
81dc",  
            "Gateway": "172.21.0.1",  
            "IPAddress": "172.21.0.2",  
            "IPPrefixLen": 16,  
            "IPv6Gateway": "",  
            "GlobalIPv6Address": "",  
            "GlobalIPv6PrefixLen": 0,  
            "DNSNames": [  
                "relaxed_cannon",  
                "de056b5c807b"  
            ]  
        }  
    }  
}
```

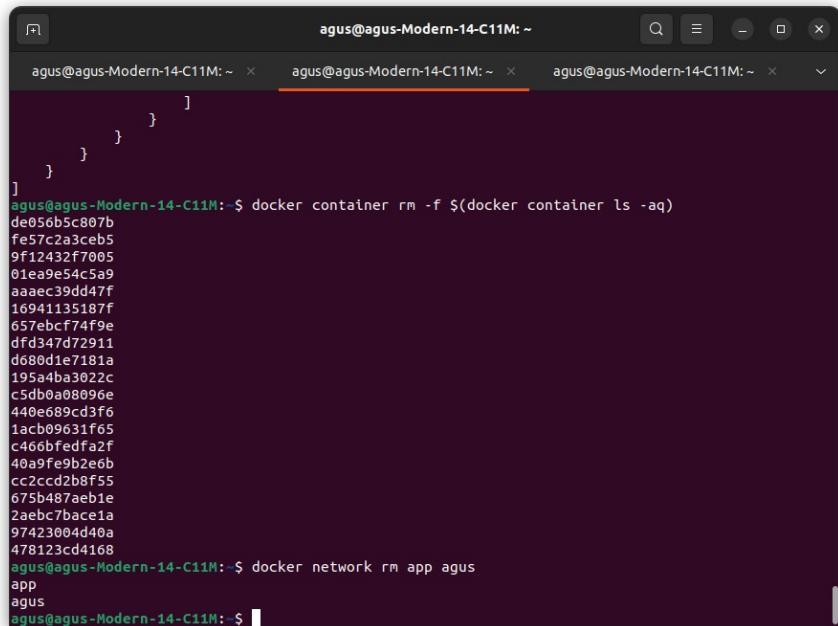
- **Menghapus jaringan yang ada**

Sebagai langkah terakhir, mari kita hapus semua container dan hapus semua jaringan yang telah kita buat.

1. Keluarkan semua container sekaligus.

```
agus@agus-Modern-14-C11M: ~ $ docker container rm -f $(docker container ls -aq)  
de056b5c807b  
fe57c2a3ce5  
9f12432f7005  
01ea9e54c5a9  
aaaec39dd47f  
16941135187f  
657ebcf74f9e  
df1d347d72911  
d680d1e7181a  
195494ba3022c  
c5db6a08096e  
440e689cd3f6  
1acb09631f65  
c466bfedfa2f  
40a9fe9b2e6b  
cc2ccd2b8f55  
675b487aeb1e  
2aebc7bace1a  
97423004d40a  
478123cd4168  
agus@agus-Modern-14-C11M: ~
```

2. Hapus semua jaringan yang kita buat.



```
agus@agus-Modern-14-C11M:~$ docker container rm -f $(docker container ls -aq)
de056b5c007b
fe57c2a3ceb5
9f12432f7005
01ea9e54c5a9
aaaec39dd47f
16941135187f
057ebcf74f9e
dfd347d72911
d688d1e7181a
195a4ba3022c
c5db0a08096e
440e689cd3f6
1acb09631f65
c466bfedfa2f
40a9fe9b2e6b
cc2ccd2b8f55
675b487aeb1e
2aebc7bace1a
97423004d40a
478123cd4168
agus@agus-Modern-14-C11M:~$ docker network rm app agus
app
agus
agus@agus-Modern-14-C11M:~$
```