

Dokumentasi Docker

by. Dimar Hanung P. | XII TKJ 1 | 35

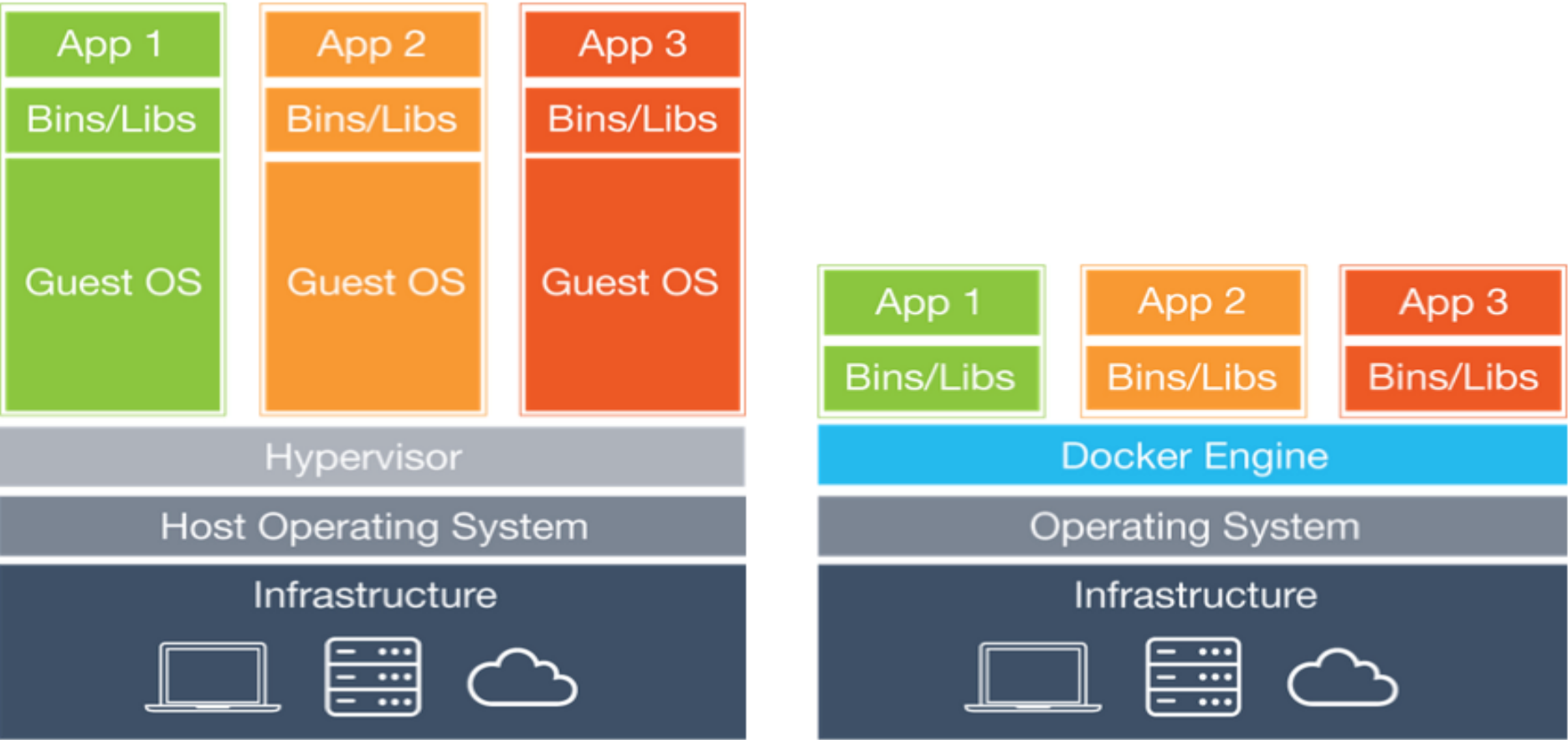
apabila menemui kata asing bisa scroll ke paling bawah sebelum daftar pustaka, ada glosarium

kotak berwarna hitam berisi perintah yang bisa dijalankan, kotak putih berisi output maupun syntax

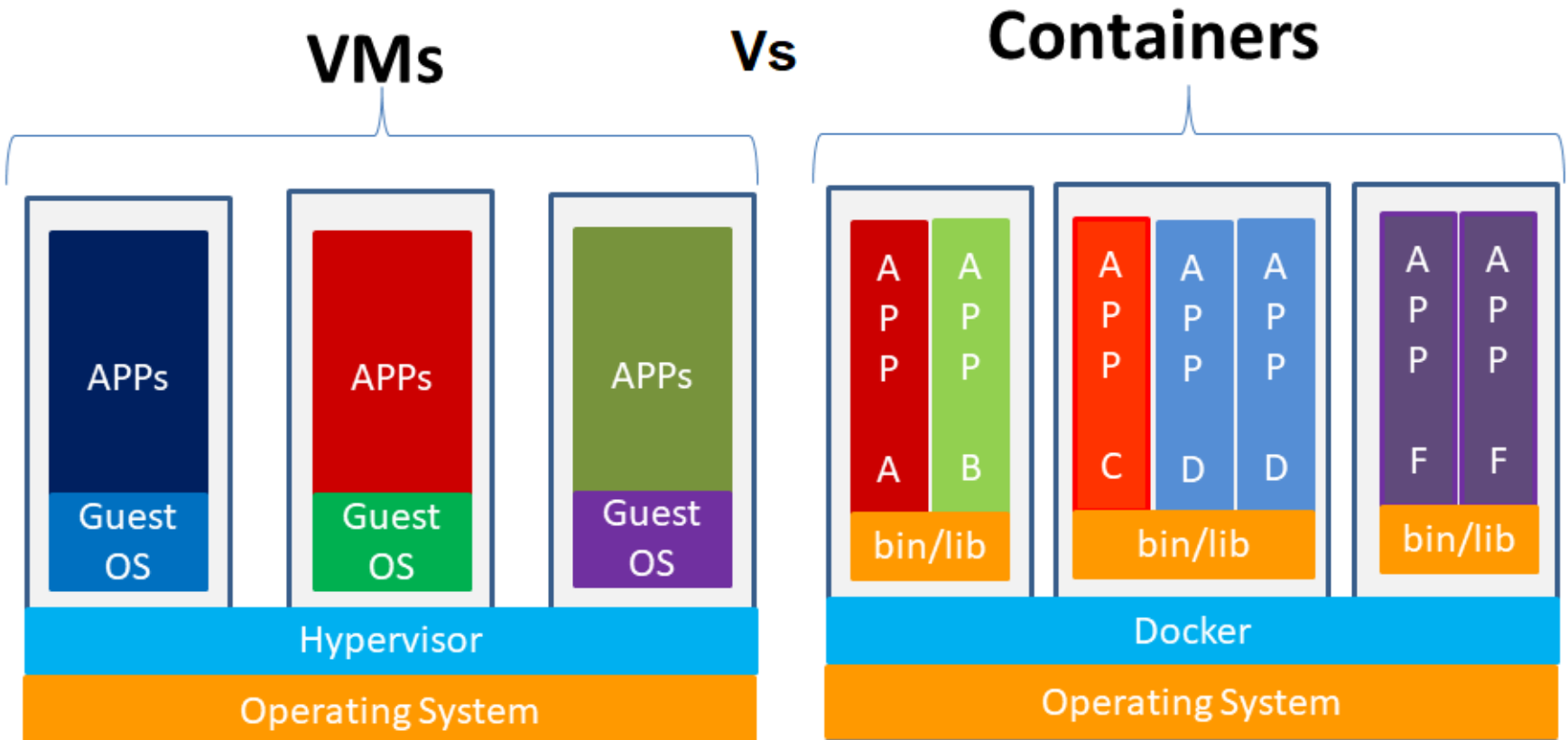
Daftar Isi

Pengenalan

Ilustrasi 1 perbedaan vm dan container (docker)



Ilustrasi 2 perbedaan vm dan container (docker)



Server

Server

Bedanya vm dan container

Supaya mudah kita bandingkan saja dengan vm software seperti virtual box,

s

virtual machine	docker
Berat	Ringan
Performa terbatas karena sudah menjalankan host os	performa sesuai atau murni hardware karena menjalankan dari host os
tidak bisa menjalankan 2 aplikasi sama atau lebih seperti ftp dan webserver	bisa menjalankan 2 aplikasi sama atau lebih seperti ftp dan webserver
Memiliki Os nya sendiri	Semua container berbagi dengan host os
hardware-level virtualization	OS virtualization
Startup dalam hitungan menit	Startup dalam hitungan detik
membutuhkan alokasi memory	membutuhkan memory yang lebih kecil
lebih aman karena terisolasi penuh	bisa jadi kurang aman karena isolasi hanya batas proses

Docker dapat di Os apa saja?

Docker tersedia di linux, windows, mac os

1. linux

paling disarankan untuk diinstall di linux karena docker mengambil os linux sebagai host nya

2. windows

untuk menggunakan docker yang host linux di windows harus **memiliki ram setidaknya nya melebihi 4 gb**, karena secara tidak langsung sebenarnya saat di windows docker menjalankan linux virtual untuk digunakan sebagai host os

docker juga memiliki windows container (windows sebagai host os), namun masih belum versi stabil jadi tidak disarankan

3. mac os

ga punya mac dong

Instalasi

Instalasi Docker pada windows

untuk cara install nya mudah, bisa lihat di dokumentasinya langsung di :

<https://docs.docker.com/docker-for-windows/install/>

buat akun docker nya dulu ya :D

Penting

Saat Install Docker Desktop, maka virtual box yang di desktop akan mengalami crash, dikarenakan hyper-v menyala, tapi jangan panik bisa diatasi dengan melihat link berikut: <https://fredrikaverpil.github.io/2018/03/15/switching-between-docker-and-virtualbox-on-windows-10/>

jadi kesimpulanya docker desktop dan virtual box tidak dapat jalan bersamaan

Instalasi Docker pada ubuntu 18

A. Menggunakan Apt

Step 1 Melakukan update paket

```
sudo apt-get update
```

Step 2 hapus versi seblemunya dari paket, (untuk memastikan juga apa bila docker benar benar tidak ada di ubuntu kamu)

```
sudo apt-get remove docker docker-engine docker.io
```

Step 3 Install docker inti

```
sudo apt install docker.io
```

Step 4 jalankan docker

```
sudo systemctl start docker
```

Step 5 gunakan perintah enable supaya saat boot pertama kali service docker langsung berjalan

```
sudo systemctl enable docker
```

```
Synchronizing state of docker.service with SysV service script with /lib/systemd/systemd-sysv-install.  
Executing: /lib/systemd/systemd-sysv-install enable docker
```

Step 6 cek docker version

```
sudo docker --version
```

Step 7 mencoba hello world

```
docker run hello-world
```

untuk pertama kali mencoba membutuhkan internet, karena mendownload images nya dulu

```
root@ip-172-31-80-102:/home/ubuntu# docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
1b930d010525: Pull complete
Digest: sha256:9572f7cdcee8591948c2963463447a53466950b3fc15a247fcad1917ca215a2f
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

root@ip-172-31-80-102:/home/ubuntu#
```

Docker pada Aws (ubuntu - ec2)

untuk apa docker di aws? ya biar bisa public dong

A. Tahap - Tahap Instalasi (persiapan)

1. Sambungkan terlebih dahulu aws ec2 dengan ssh
2. atur security group pada aws untuk membuka port 80

B. Tahap Tahap Instalasi (install docker)

1. `sudo su`

```
tes tes
```

- 2.
- 3.
- 4.

Praktik Dasar

Membuat Container / docker run | praktik #1

Penting Sebelumnya pahami dulu sifat perintah docker run ini

1. otomatis pull image
apabila images tidak tersedia, maka akan mendownload image nya terlebih dahulu
2. docker run bukan untuk menjalankan
docker run bukan untuk menjalankan container, melainkan membuat container

3. docker apa bila tidak disertai argumen `--name` maka akan membuat nama container otomatis
4. akan terjadi error apabila nama container sudah dibuat sebelumnya

Syntax:

```
docker container run --name [nama container yg akan dibuat] [nama image/paket aplikasi]
```

A. Menjalankan Hello World

Jalankan Perintah dibawah ini

```
docker container run --name halo1 hello-world
```

atau bisa disingkat menjadi:

```
docker run --name halo1 hello-world
```

```
Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/
```

Dalam 1 perintah diatas sebenarnya ada proses yang terjadi

Proses 1 ketika enter maka docker mengecek apakah container sudah ada apa belum

Proses 2 kalo ada, muncul pesan error sudah ada container dengan nama yang sama
kalo tidak ada , maka docker akan pull image dari docker store / docker hub

Proses 3 setelah image ter download, docker langsung menjalankan container

Proses 4 ada container yang setelah dijalankan langsung berhenti, ada yang setelah dijalankan tetap berjalan tergantung dari container nya

B. Menjalankan nginx (testing port 80)

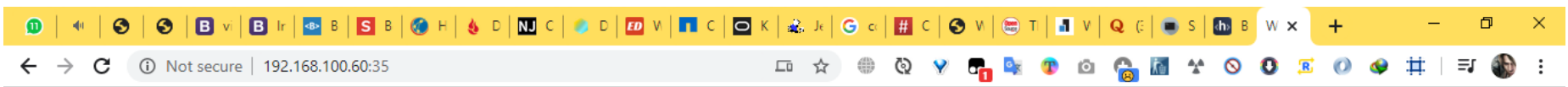
Syntax:

```
docker run -d -p [port forward ke]:80 --name [nama container] [nama paket images]:[versi spesifik]
```

jalankan perintah berikut :

```
docker run -d -p 35:80 --name webserver1 nginx
```

didasar tidak menggunakan tag / versi spesifik, maka otomatis mendownload versi latest
cek di web apakah sudah bisa di panggil, contoh dibawah ini



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org. Commercial support is available at nginx.com.

Thank you for using nginx.



C. Menjalankan Alpine OS

penting ingat kembali sifat docker run bagian argumen `--name`

penting apabila tidak menggunakan argumen `-it` maka container cuma sekali pakai / tidak bisa dipakai / distart lagi

contoh 1 melihat isi direktori os alpine

```
docker container run alpine ls -l
```

```
total 56
drwxr-xr-x  2 root    root      4096 Jan 16 21:52 bin
drwxr-xr-x  5 root    root      340 Jan 30 13:27 dev
drwxr-xr-x  1 root    root     4096 Jan 30 13:27 etc
drwxr-xr-x  2 root    root     4096 Jan 16 21:52 home
drwxr-xr-x  5 root    root     4096 Jan 16 21:52 lib
drwxr-xr-x  5 root    root     4096 Jan 16 21:52 media
drwxr-xr-x  2 root    root     4096 Jan 16 21:52 mnt
drwxr-xr-x  2 root    root     4096 Jan 16 21:52 opt
dr-xr-xr-x 96 root    root        0 Jan 30 13:27 proc
drwx----- 2 root    root     4096 Jan 16 21:52 root
drwxr-xr-x  2 root    root     4096 Jan 16 21:52 run
drwxr-xr-x  2 root    root     4096 Jan 16 21:52 sbin
drwxr-xr-x  2 root    root     4096 Jan 16 21:52 srv
dr-xr-xr-x 13 root    root        0 Jan 30 13:27 sys
drwxrwxrwt  2 root    root     4096 Jan 16 21:52 tmp
drwxr-xr-x  7 root    root     4096 Jan 16 21:52 usr
drwxr-xr-x 12 root    root     4096 Jan 16 21:52 var
```

contoh 2 menjalankan perintah echo

```
docker container run alpine echo "halo saya hanung"
```

```
halo saya hanung
```

contoh 3 masuk direktori

```
docker container run -it --name latihan-os alpine /bin/sh
```

kamu sekarang berada dalam linux shell alpine, kamu bisa menjalankan perintah dari os alpine sekarang, kamu bisa mencoba `ls -l`, `uname -a`, dan perintah linux alpine lainnya, untuk masuk shell harus menggunakan argunmen `-it`, apabila tidak maka akan exit otomatis, gunakan perintah `exit` untuk keluar dari shell

Melihat list container / docker container ls | praktik #2

contoh 1 melihat container yang berjalan

```
docker container ls
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
db351ab12d47	nginx	"nginx -g 'daemon of..."	About an hour ago	Up About an hour	0.0.0.0:35->80/
tcp	webserver2				
9ae179958589	nginx	"nginx -g 'daemon of..."	About an hour ago	Up About an hour	0.0.0.0:8080->8
0/tcp	webserver1				

contoh 2 Meihat semua list container yang ada

```
docker container ls --all
```

contoh 3 menjalankan container yang berhenti

Syntax:

```
docker container start [nama / id container]
```

contoh:

```
docker container start halo1
```

Menjalankan perintah container / docker container exec | praktik #3

Syntax:

```
docker container exec [id/nama container] [perintah]
```

untuk menjalankan perintah exec, container harus berjalan dulu menggunakan

```
docker container start [id/nama container]
```

contoh 1 echo alpine os

```
docker container exec latihan-os echo "halo saya hanung"
```

contoh 2 Melihat list folder alpine os

```
docker container exec latihan-os ls
```

contoh 3 Masuk bash

```
docker container exec -it latihan-os bin/sh
```

Perintah rm (remove) | praktik #4

perintah rm bisa digunakan untuk menghapus image maupun container

syntax:

```
docker [container/image] rm [nama container/image]
```

contoh 1

```
docker container rm halo1
```

contoh 2

```
docker image rm nginx
```

Penutup

Glosarium

Hypervisor - platform / aplikasi untuk menjalankan sebuah virtualisasi , contoh:

1. Microsoft Hyper-V
2. Oracle VirtualBox
3. VMware Workstation Player

apt - Advanced Package Tool, untuk install paket linux yang base nya debian

alpine OS - adalah os linux yang berdiri sendiri , bukan dari debian , arch, redhat atau yang lain

nginx - adalah webserver seperti apache

echo - perintah dasar kebanyakan bahasa pemrograman untuk print suatu string / integer

Daftar Pustaka

<https://www.backblaze.com/blog/vm-vs-containers/>

<https://readwindow.blogspot.com/2015/03/pengertian-hypervisor.html>

<https://phoenixnap.com/kb/how-to-install-docker-on-ubuntu-18-04>

<https://opensourceforu.com/2019/04/the-top-open-source-hypervisor-technologies/>

<https://mjaglan.github.io/docs/why-containerize-apps.html>

<https://training.play-with-docker.com/ops-s1-hello/>

<https://www.adminschoice.com/containers-vs-vms-top-5-differences-you-must-know/>

https://upload.wikimedia.org/wikipedia/commons/1/1b/Linux_Distribution_Timeline.svg

<https://www.backblaze.com/blog/vm-vs-containers/>