

KOMPUTASI UBIQUITOUS DAN PERVASIF INSTALASI IOTA, DOCKER, DAN PROXMOX



222L1

Disusun Oleh :

Chosmas Marzuki	09021182025003
Karinda Amelia	09021282025054
Tiara Aprisa	09021182025005

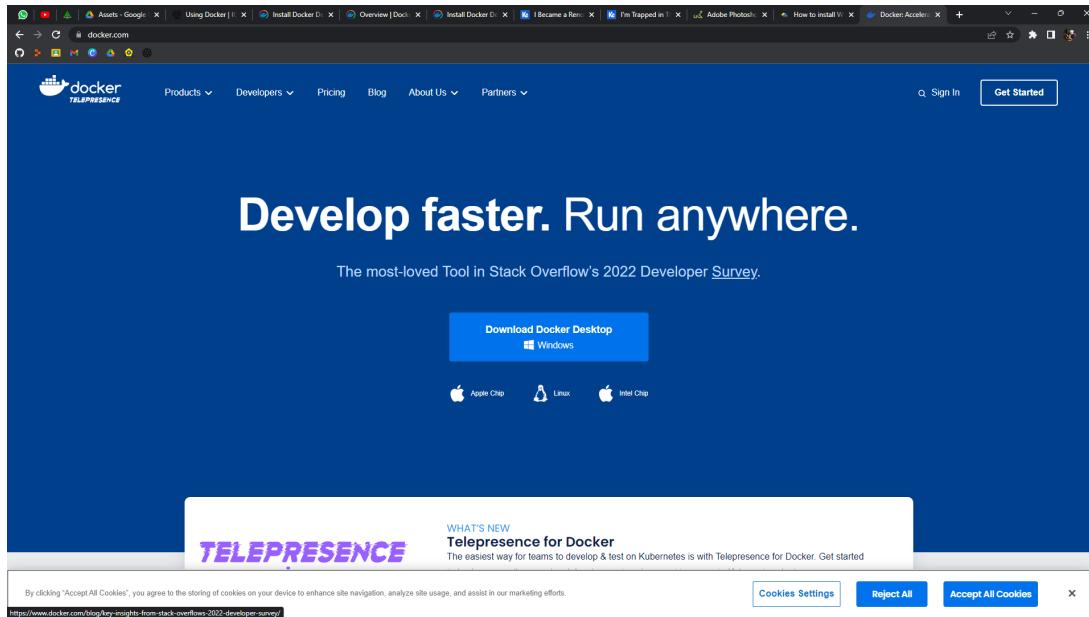
Dosen Pengampu:

Adi Hermansyah, S.Kom., M.T.
Huda Ubaya, M.T.

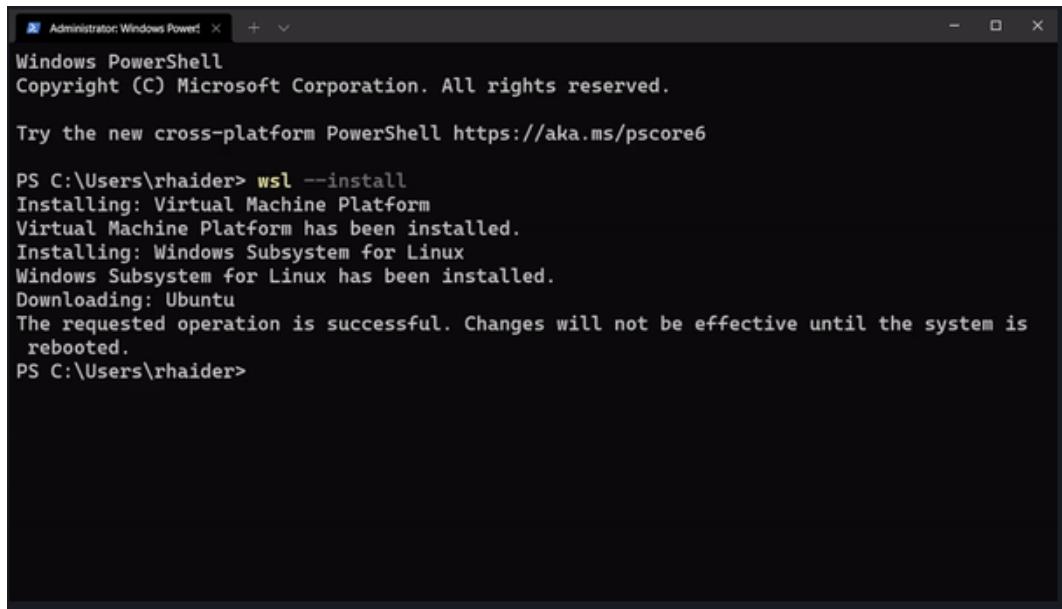
**SEMESTER GENAP 2022/2023
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SRIWIJAYA
2023**

INSTALASI IOTA, DOCKER, DAN PROXMOX

A. Install IOTA



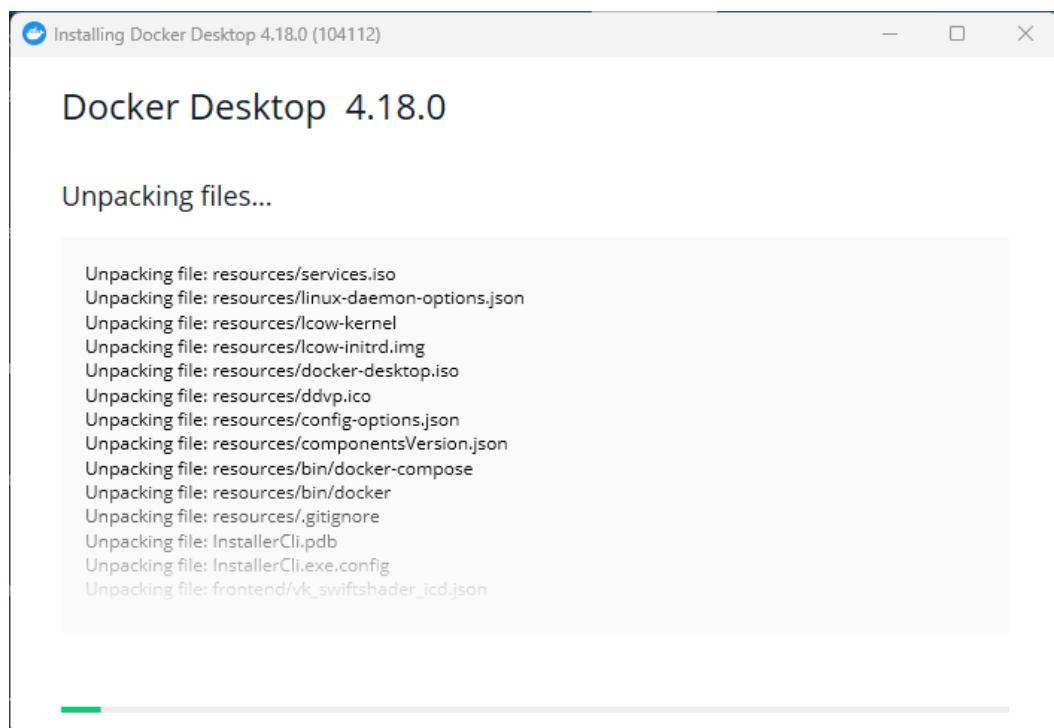
A screenshot of a Microsoft Learn article titled 'Install Linux on Windows with WSL'. The page has a dark theme. On the left, there's a sidebar with navigation links for WSL Documentation, Overview, Install (selected), Install WSL, Manual install steps for older versions, Install on Windows Server, Tutorials, Concepts, How-to, Frequently Asked Questions, Troubleshooting, and Release Notes. The main content area shows the article title, a brief description, and a 'Prerequisites' section. It includes a code snippet for running 'wsl --install' in PowerShell. A feedback form at the bottom right asks for user satisfaction with Microsoft Learn.

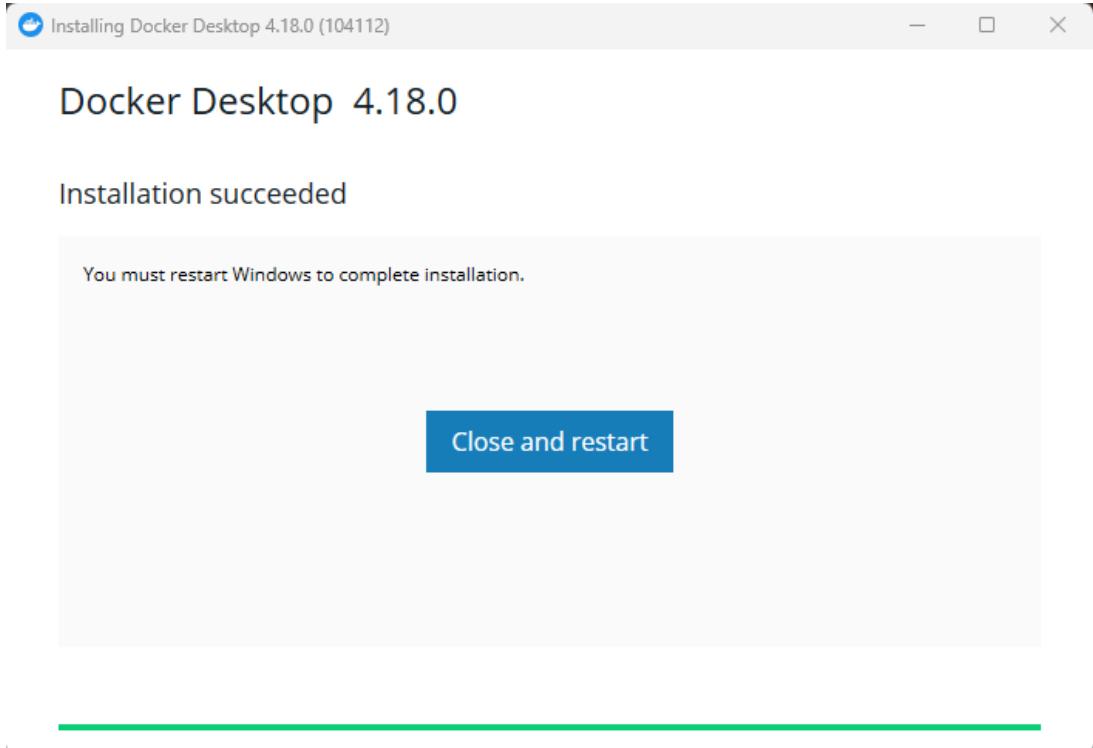


```
Administrator: Windows PowerShell -> + <
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

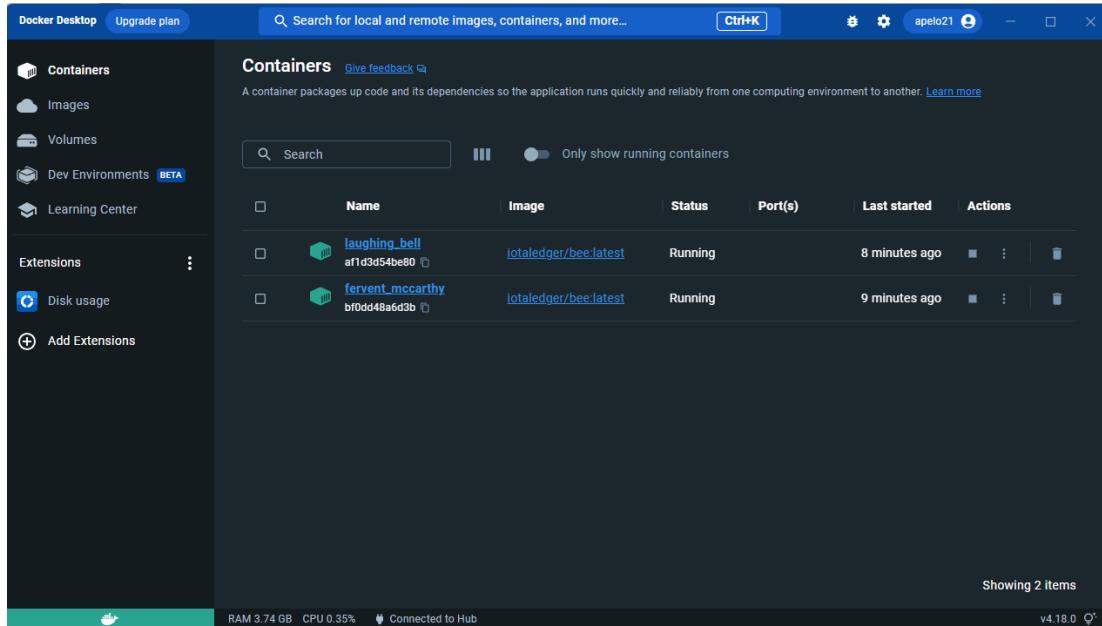
PS C:\Users\rhaider> wsl --install
Installing: Virtual Machine Platform
Virtual Machine Platform has been installed.
Installing: Windows Subsystem for Linux
Windows Subsystem for Linux has been installed.
Downloading: Ubuntu
The requested operation is successful. Changes will not be effective until the system is
rebooted.
PS C:\Users\rhaider>
```





```
Cloning into 'bee'...
remote: Enumerating objects: 33340, done.
remote: Counting objects: 100% (285/285), done.
remote: Compressing objects: 100% (205/205), done.
remote: Total 33340 (delta 117), reused 204 (delta 78), pack-reused 33055
Receiving objects: 100% (33340/33340), 15.52 MiB | 1.15 MiB/s, done.
Resolving deltas: 100% (23457/23457), done.
```

```
latest: Pulling from iotaledger/bee
36698cfa5275: Pull complete
6a8659ec8836: Pull complete
a1f1879bb7de: Pull complete
bb14e507de00: Pull complete
023d1fb2c19b: Pull complete
aa9fe1f7c3a9: Pull complete
183d7be5fbfd: Pull complete
2e4d624d5d2a: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:b3216b65a2190e766886de679e9e864a98f30be05b5427a77ee986c15f94415f
Status: Downloaded newer image for iotaledger/bee:latest
docker.io/iotaledger/bee:latest
```



set up a node for IOTA Transaction

Untuk adanya transaksi pada IOTA, maka diperlukannya node IOTA.

Installing from Source :

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> Install-Chocolatey -Force -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
PS C:\Windows\system32> Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072
Forcing web requests to allow TLS v1.2 (Required for requests to Chocolatey.org)
Getting latest version of the Chocolatey package for download.
Not using proxy.
Getting Chocolatey from https://community.chocolatey.org/api/v2/package/chocolatey/1.3.1.
Downloaded https://community.chocolatey.org/api/v2/package/chocolatey/1.3.1 to C:\Users\Zen\AppData\Local\Temp\chocoInstall\chocolatey.zip
Extracting C:\Users\Zen\AppData\Local\Temp\chocoInstall\chocolatey.zip to C:\Users\Zen\AppData\Local\Temp\chocoInstall
Installing Chocolatey on the local machine
Creating ChocolateyInstall as an environment variable (targeting 'Machine')
  This will affect all users on this machine
ChocolateyInstall points to 'C:\ProgramData\chocolatey\bin'
WARNING: It's very likely you will need to close and reopen your shell
before you can use choco.
Restricting write permissions to Administrators
You and your team can use Chocolatey package repository.
The packages themselves go to 'C:\ProgramData\chocolatey\lib'.
(i.e. C:\ProgramData\chocolatey\lib\yourPackageName)
And the executables go to 'C:\ProgramData\chocolatey\bin'
and points to an executable in 'C:\ProgramData\chocolatey\lib\yourPackageName'.
Creating Chocolatey folders if they do not already exist.

WARNING: You can safely ignore errors related to missing log files when
upgrading from a version of Chocolatey less than 0.9.9.
Any errors should be safe to ignore.
The system cannot find the file specified. also safe.
chocolatey.nupkg file not installed in lib.
Attempting to locate the file at bootstrap...
Path: C:\ProgramData\chocolatey\lib\chocolatey\bootstrap\chocolatey.nupkg
Attempting to copy file C:\ProgramData\chocolatey\bin in it. Adding...
WARNING: Not setting tab completion: Profile file does not exist at
D:\Documents\WindowsPowerShell\Microsoft.PowerShell_profile.ps1'.
Chocolatey has been successfully installed!
You can call choco from anywhere, command line or powershell by typing choco.
Run choco --? for a list of functions.
You may need to shutdown and restart powershell and/or consoles
First, after the update.
Ensuring Chocolatey commands are on the path
Ensuring chocolatey.nupkg is in the lib folder
PS C:\Windows\system32>
```

Chocolatey installed 7/7 packages.

See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).

Installed:

- llvm v16.0.1
- chocolatey-core.extension v1.4.0
- cmake v3.26.3
- cmake.install v3.26.3
- git.install v2.40.0
- chocolatey-compatibility.extension v1.0.0
- git v2.40.0

PS C:\Windows\system32>

```
PS C:\WINDOWS\system32> rustup update
info: syncing channel updates for 'stable-x86_64-pc-windows-msvc'
info: checking for self-updates

stable-x86_64-pc-windows-msvc unchanged - rustc 1.68.2 (9eb3afe9e 2023-03-27)

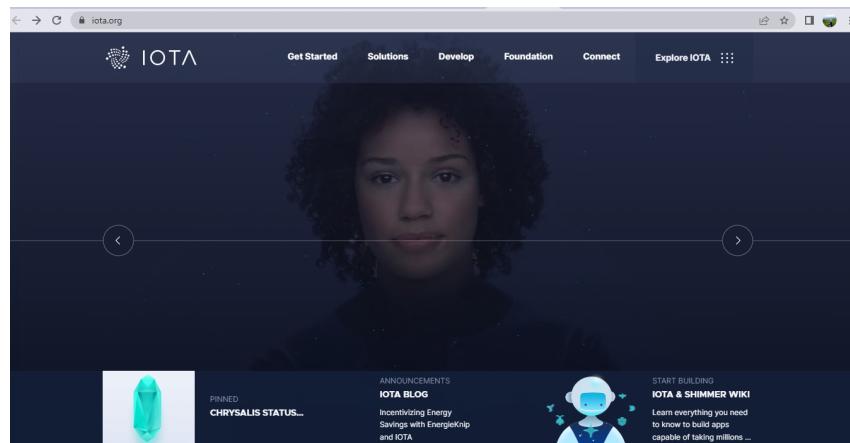
info: cleaning up downloads & tmp directories
```

Compiling the Bee Node :

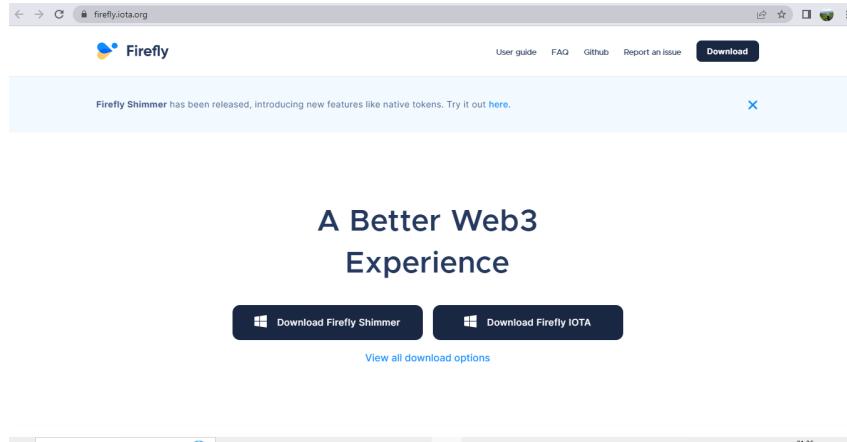
```
PS C:\WINDOWS\system32> git clone https://github.com/iotaledger/bee.git --branch mainnet-develop
Cloning into 'bee'...
remote: Enumerating objects: 33340, done.
remote: Counting objects: 100% (285/285), done.
remote: Compressing objects: 100% (205/205), done.
remote: Total 33340 (delta 117), reused 204 (delta 78), pack-reused 33055
Receiving objects: 100% (33340/33340), 15.52 MiB | 1.21 MiB/s, done.
Resolving deltas: 100% (23457/23457), done.
Updating files: 100% (794/794), done.
```

```
latest: Pulling from iotaledger/bee
36698cfa5275: Pull complete
6a8659ec8836: Pull complete
a1f1879bb7de: Pull complete
bb14e507de00: Pull complete
023d1fb2c19b: Pull complete
aa9fe1f7c3a9: Pull complete
183d7be5fbfd: Pull complete
2e4d624d5d2a: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:b3216b65a2190e766886de679e9e864a98f30be05b5427a77ee986c15f94415f
Status: Downloaded newer image for iotaledger/bee:latest
docker.io/iotaledger/bee:latest
```

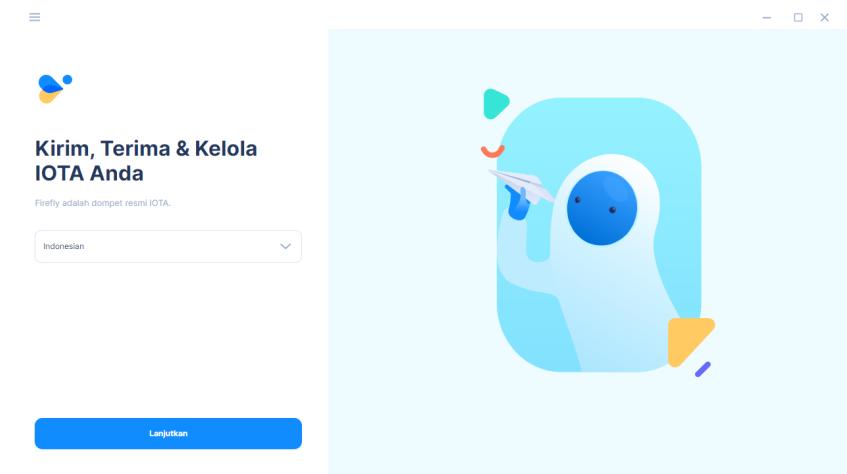
1. Pertama buka web IOTA di browser



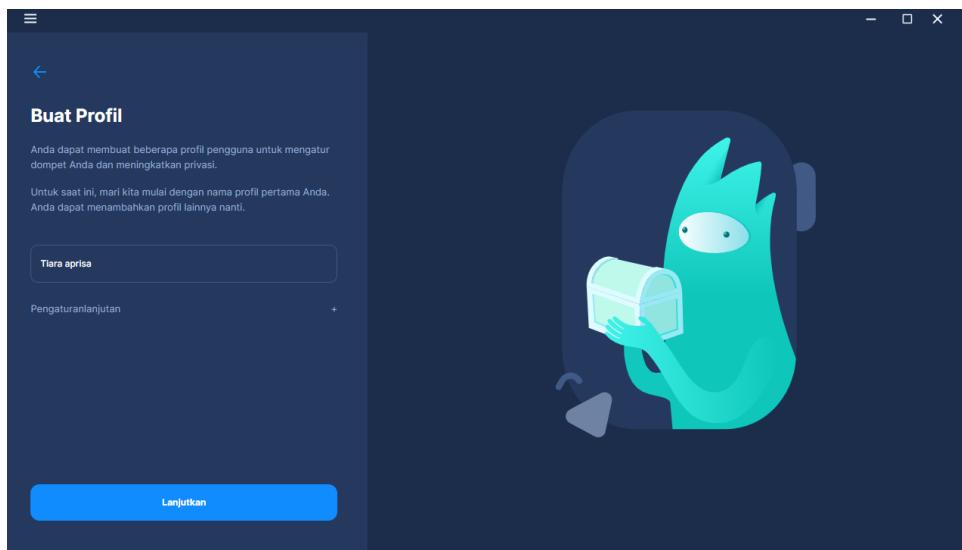
2. Kemudian Download Firefly IOTA

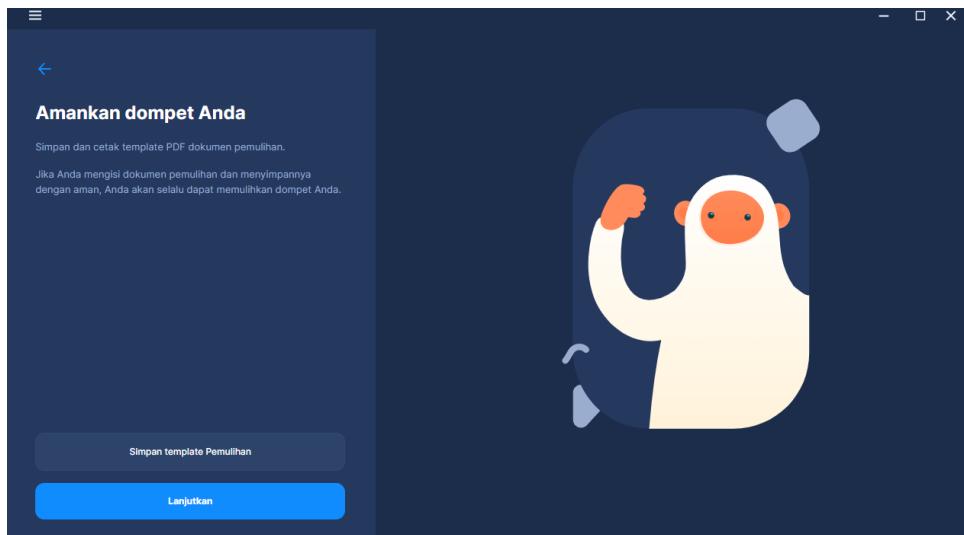


3. Setelah IOTA terinstal ikuti langkah-langkahnya

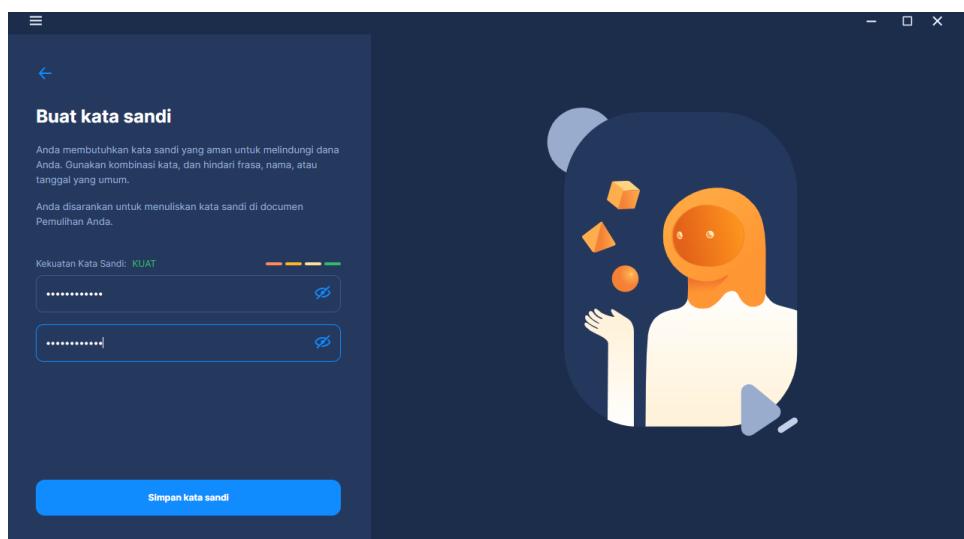


4. Buat akun Profil terlebih dahulu

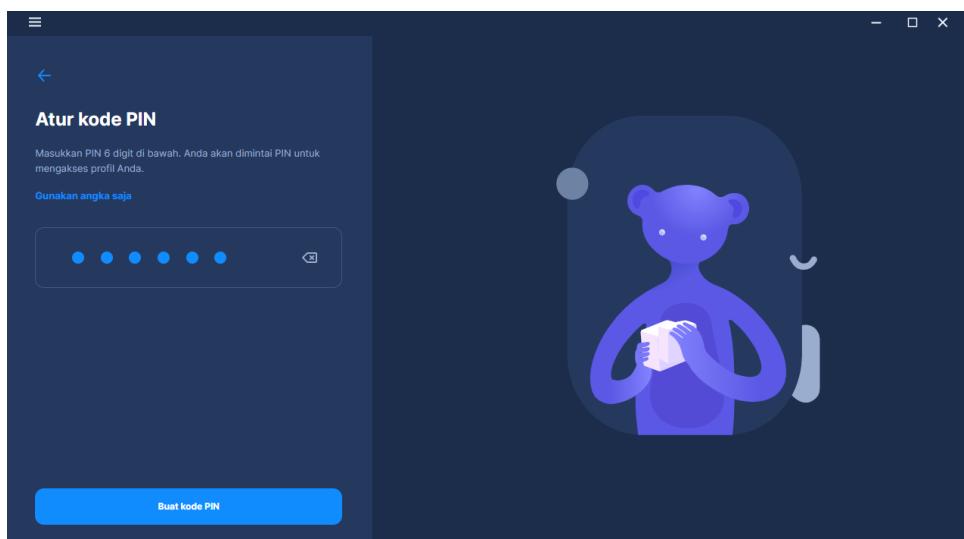




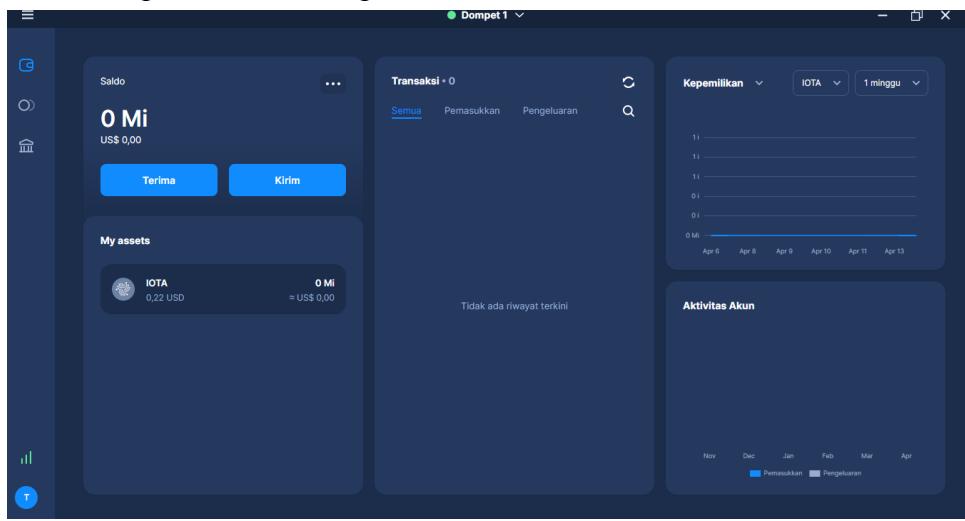
5. Kemudian Membuat Kata Sandi



6. Lalu Atur kode Pin

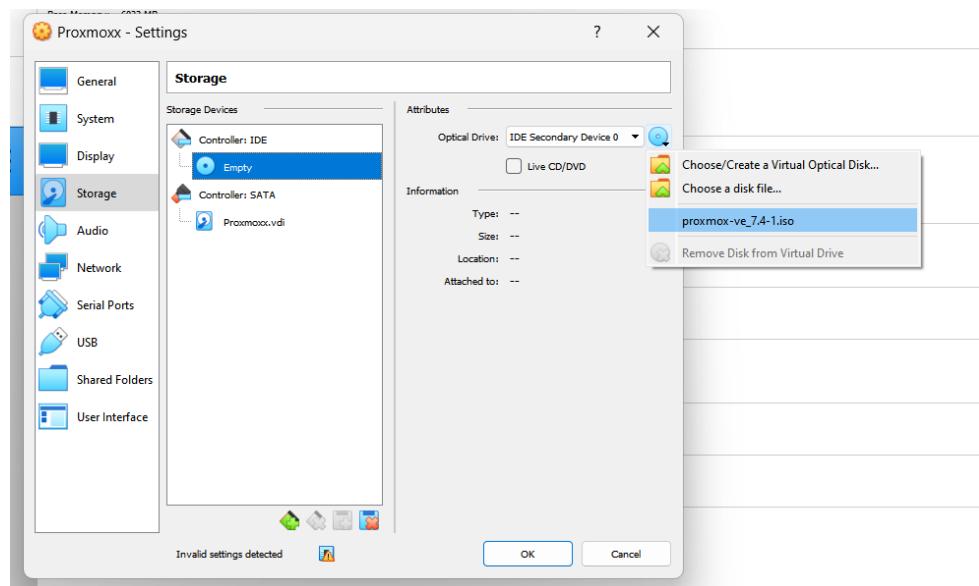


7. Berikut Tampilan Dashboard pada IOTA

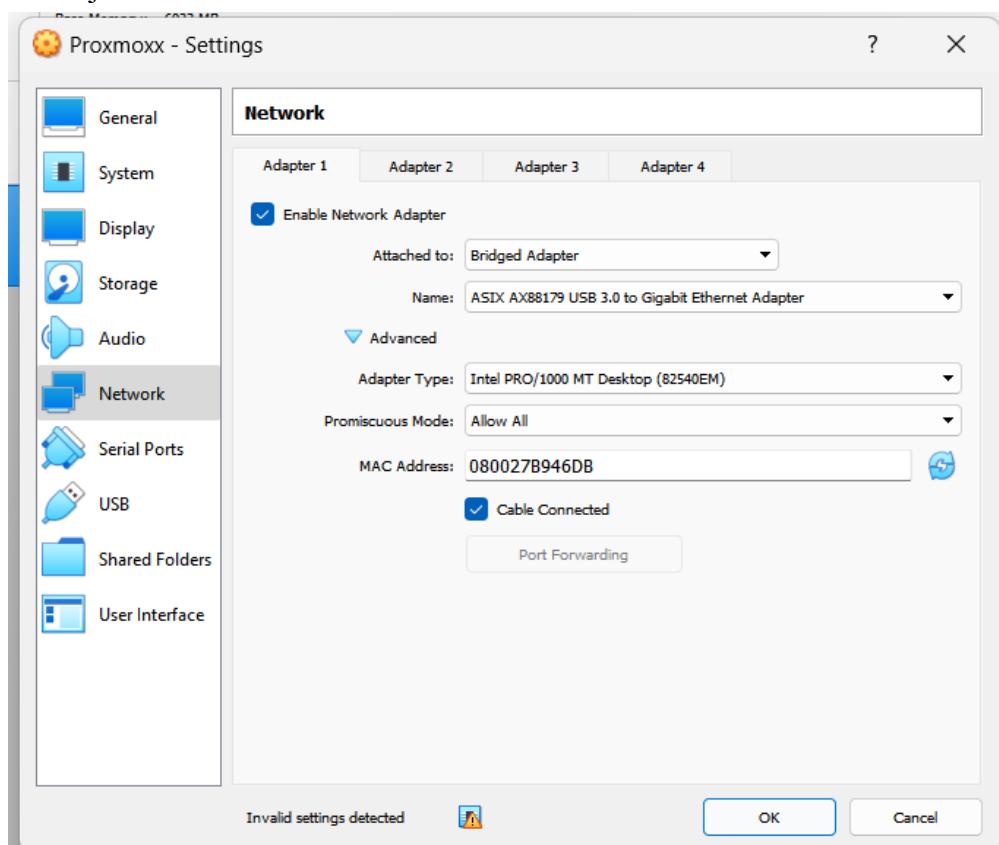


B. Install Proxmox VE for VirtualBox Manager

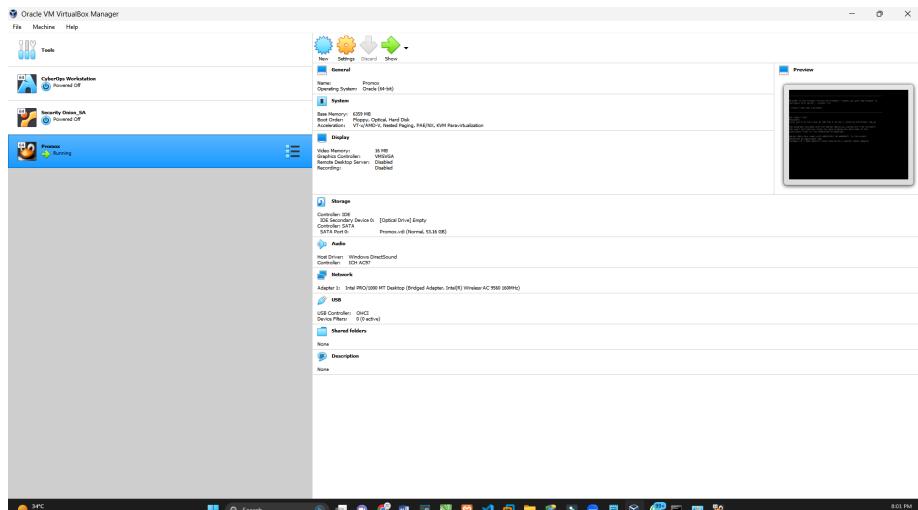
1. Storage ubah IDE menjadi Proxmox



2. Ubah Attached to menjadi Bridged Adapter dan pada Advanced untuk Promiscuous Mode menjadi Allow All lalu OK



3. Start Proxmox



4. Klik Install atau langsung enter



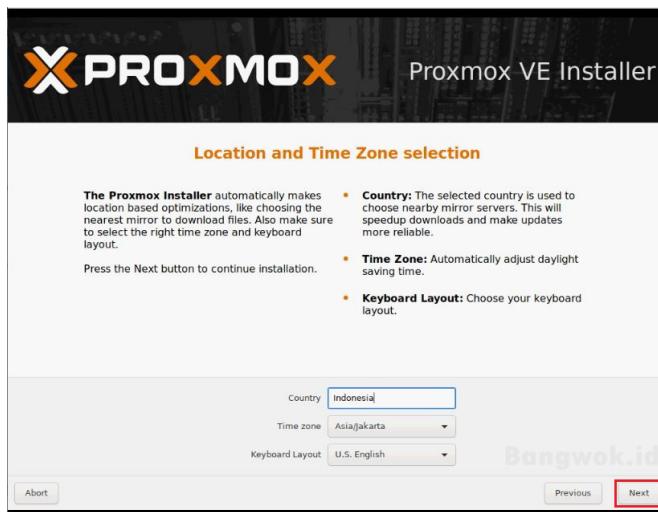
5. lalu klik I Agree



6. Klik Next



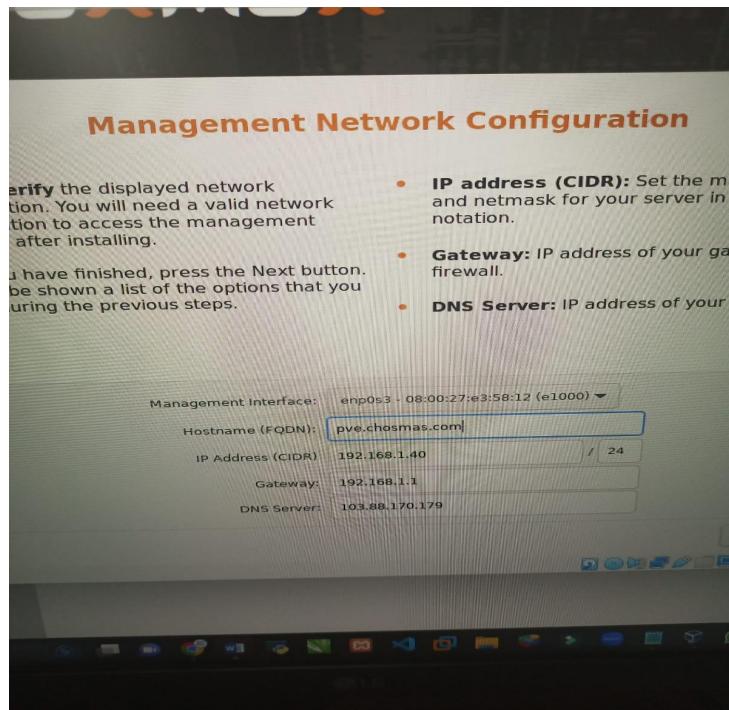
7. Pilih Indonesia lalu next



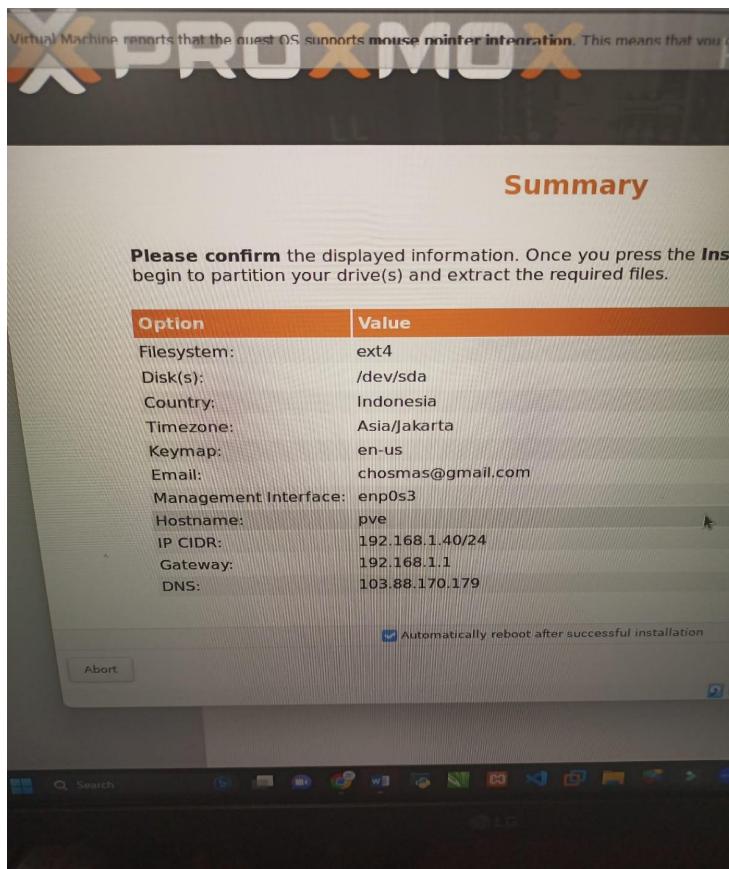
8. Masukan Password dan email anda lalu next

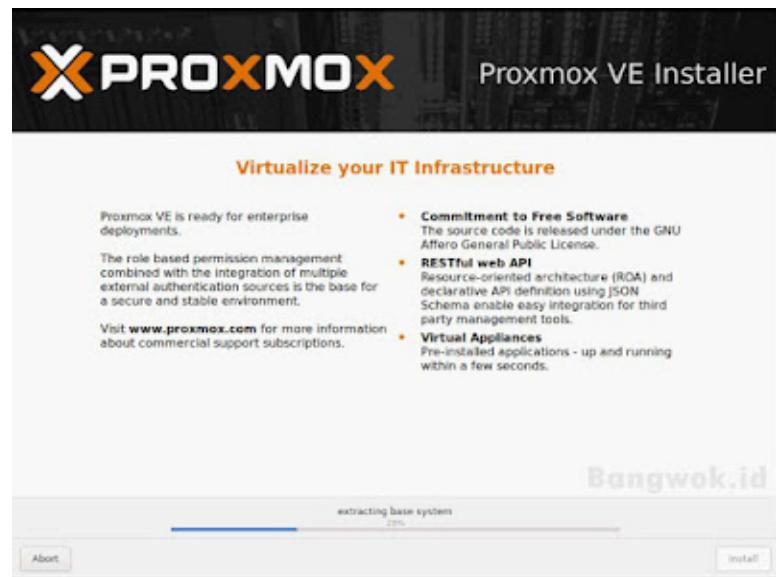


9. Masukan Hostname anda lalu next

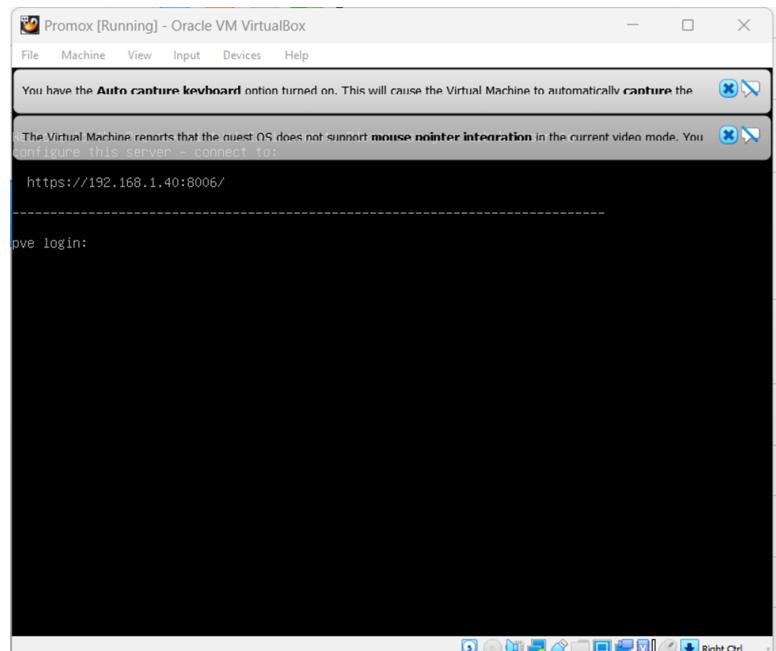


10. Klik Install

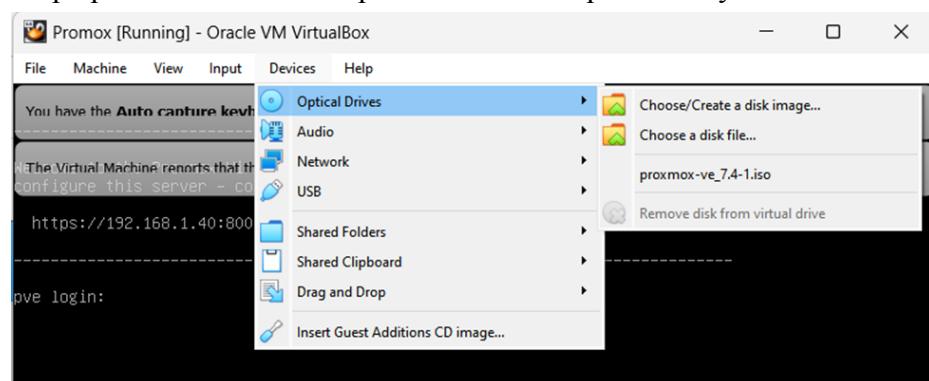




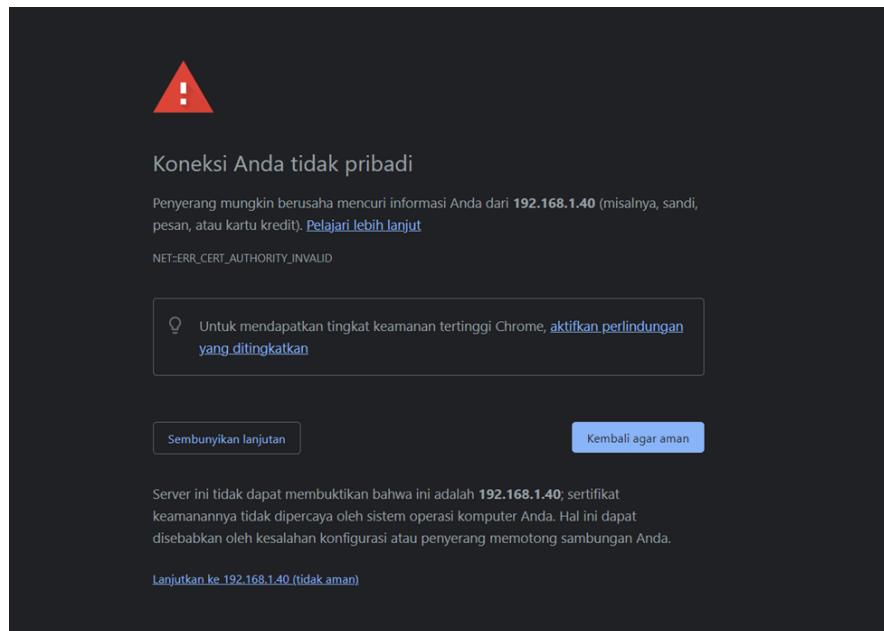
Berikut merupakan tampilan Proxmox IP



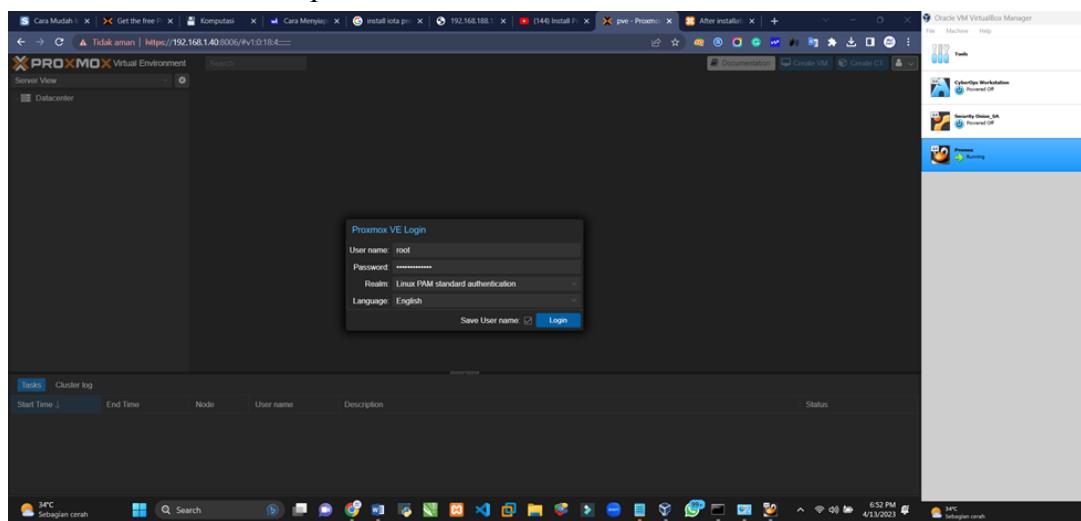
11. Jangan Lupa pada Devices Klik Option lalu cheklis proxmoxnya



12. Masukkan IP yang tertera pada Proxmox ke web browser



13. Masukan Username dan password



14. Tampilan Proxmox

The screenshot shows the Proxmox VE 7.4-3 interface. The main window displays a table of resources under the 'Datacenter' tab. The table includes columns for Type, Description, Disk usage, Memory us., CPU usage, Uptime, Host CPU, Host Mem, and Tags. The resources listed are:

Type	Description	Disk usage...	Memory us...	CPU usage	Uptime	Host CPU ...	Host Mem...	Tags
node	pve	10.8 %	18.3 %	1.3% of 1 ...	00:48:21			
storage	local (pve)				-			
storage	local-lvm (pve)				-			

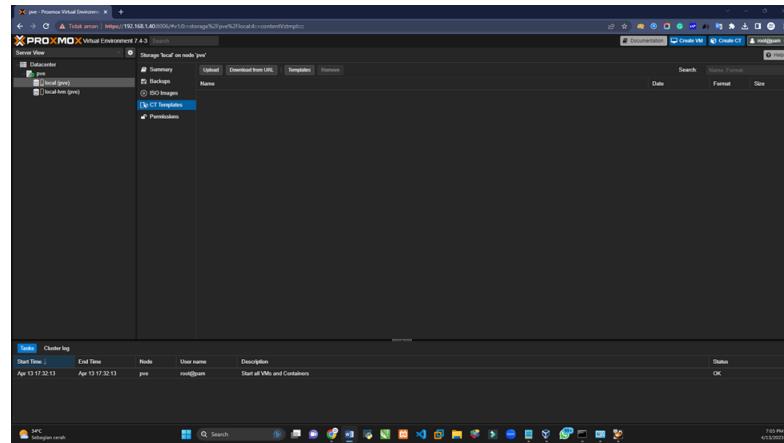
Below the table, there is a 'Tasks' section showing a single task entry:

Start Time	End Time	Node	User name	Description	Status
Apr 13 17:32:13	Apr 13 17:32:13	pve	root@pam	Start all VMs and Containers	OK

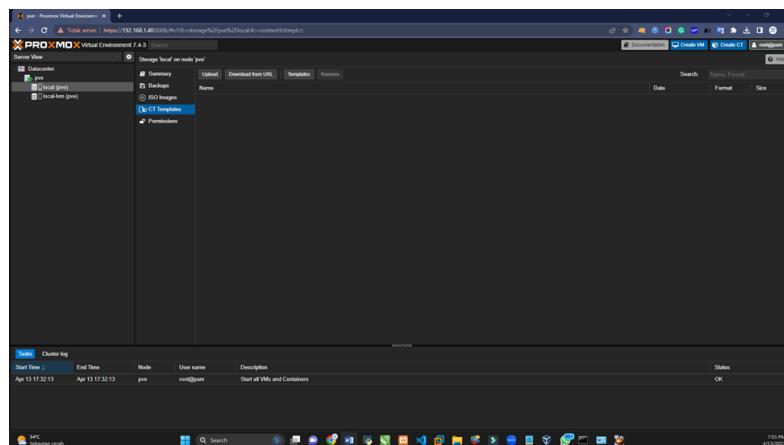
The bottom of the screen shows a Windows-style taskbar with various icons and system status information.

C. Install Docker in Proxmox

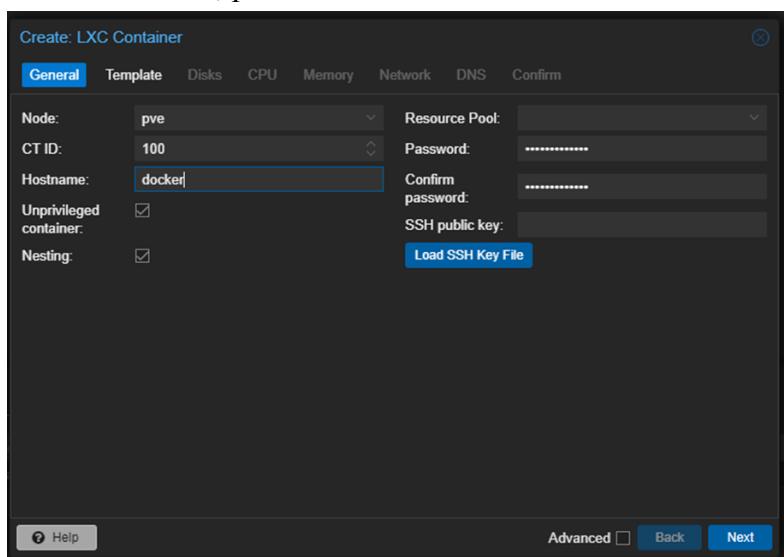
1. Masuk ke Proxmox, pilih lokasi penyimpanan tempat Anda ingin menyimpan template container, pilih CT Templates , lalu pilih Templates.



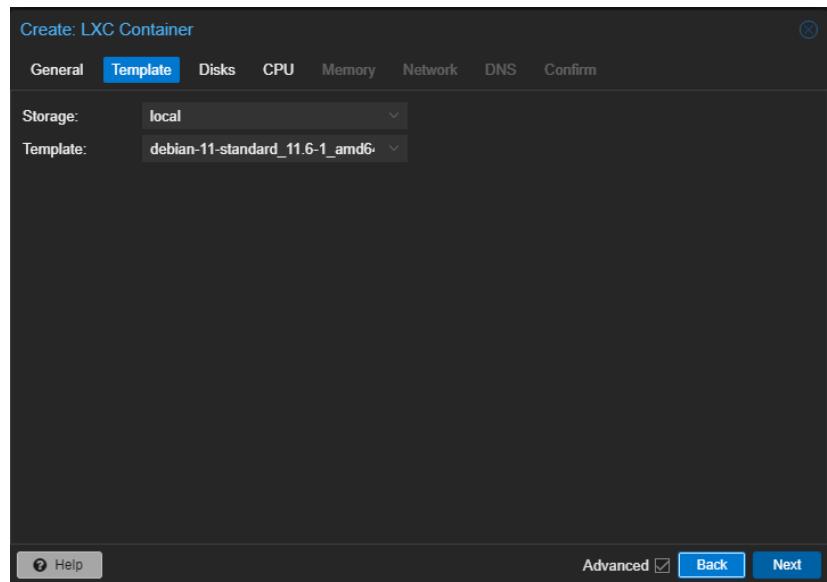
2. Cari Debian , lalu pilih debian-11-standard dan Download.



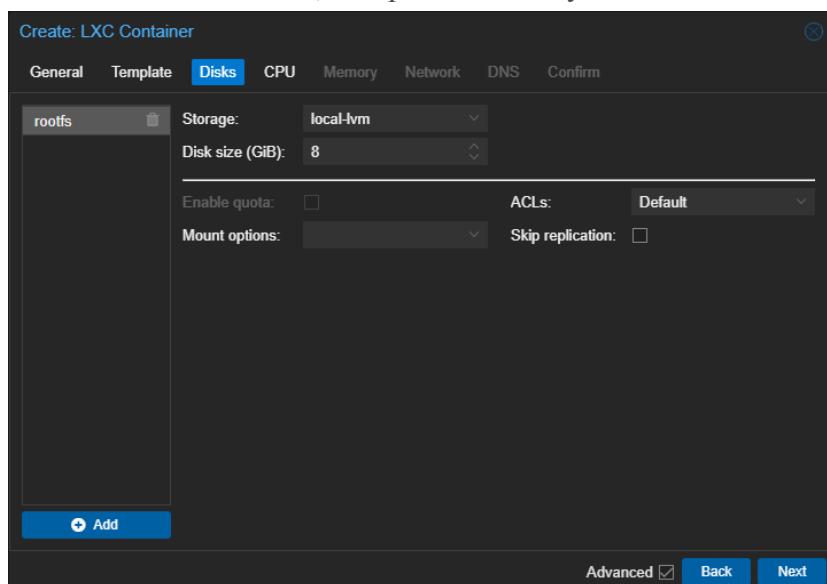
3. Buat CT, masukkan Hostname , lalu masukkan Kata Sandi yang ingin Anda gunakan. Kata sandi ini akan digunakan untuk masuk ke akun pengguna root . Setelah semua pengaturan sudah ditentukan, pilih Next.



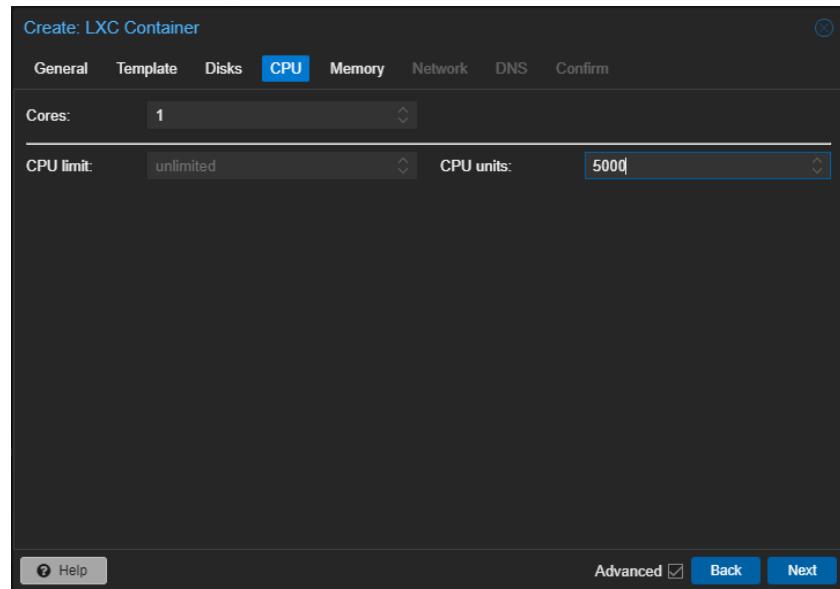
4. Pilih Template , lalu pilih Next untuk melanjutkan.



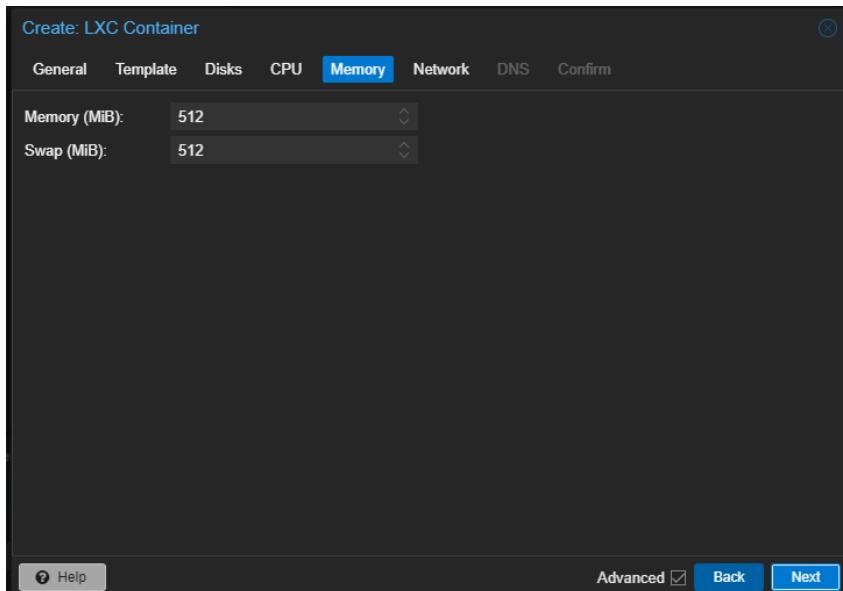
5. Pilih Ukuran Disk untuk wadah ini, lalu pilih Berikutnya.



6. Pilih total Core untuk CPU, lalu pilih Next.



7. Atur total Memory , lalu pilih Next .



8. Ubah Jaringan untuk menggunakan DHCP untuk IPv4 dan IPv6 (kecuali jika Anda ingin menentukannya secara manual), lalu pilih Berikutnya hingga Anda mendapatkan Konfirmasi.

Create: LXC Container

Network

Name:	eth0	IPv4:	Static	DHCP
MAC address:	auto	IPv4/CIDR:		
Bridge:	vmbr0	Gateway (IPv4):		
VLAN Tag:	no VLAN	IPv6:	Static	DHCP
Firewall:	<input checked="" type="checkbox"/>	IPv6/CIDR:	None	
Disconnect:	<input type="checkbox"/>	Rate limit (MB/s):	unlimited	
MTU:	Same as bridge			

Help Advanced Back Next

9. Konfirmasikan pengaturan, lalu pilih Selesai untuk membuat penampung!

Create: LXC Container

Confirm

Key ↑	Value
cores	1
cpuunits	5000
features	nesting=1
hostname	docker
memory	512
net0	name=eth0,bridge=vmbr0,firewall=1,ip=dhcp
nodename	pve
ostemplate	local.vztmpl/debian-11-standard_11.6-1_amd64.tar.zst
pool	
rootfs	local-lvm:8
swap	512
unprivileged	1
vmid	100

Start after created

Advanced Back **Finish**

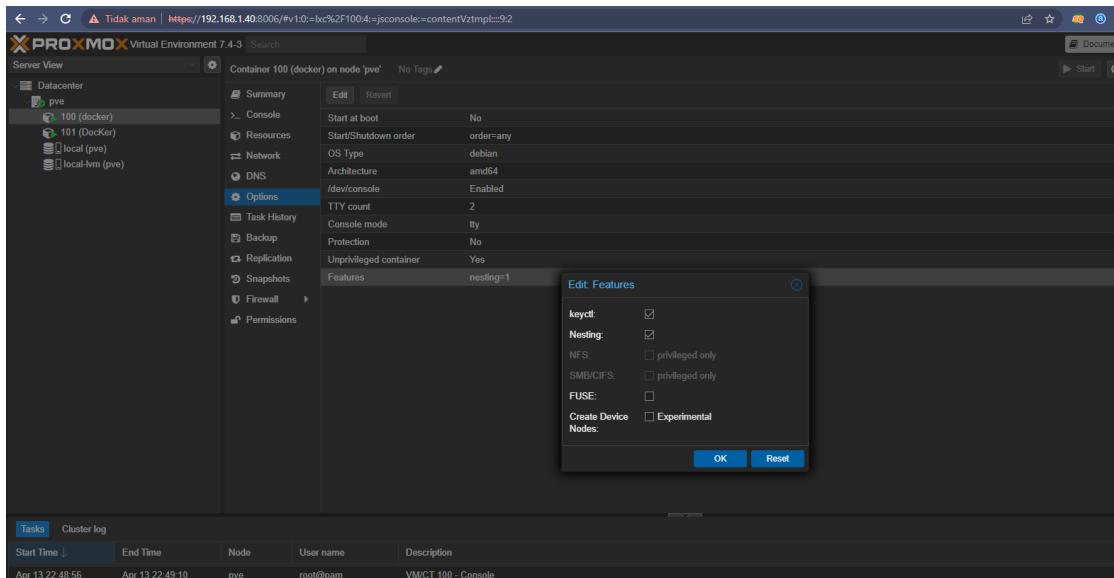
Task viewer: CT 100 - Create

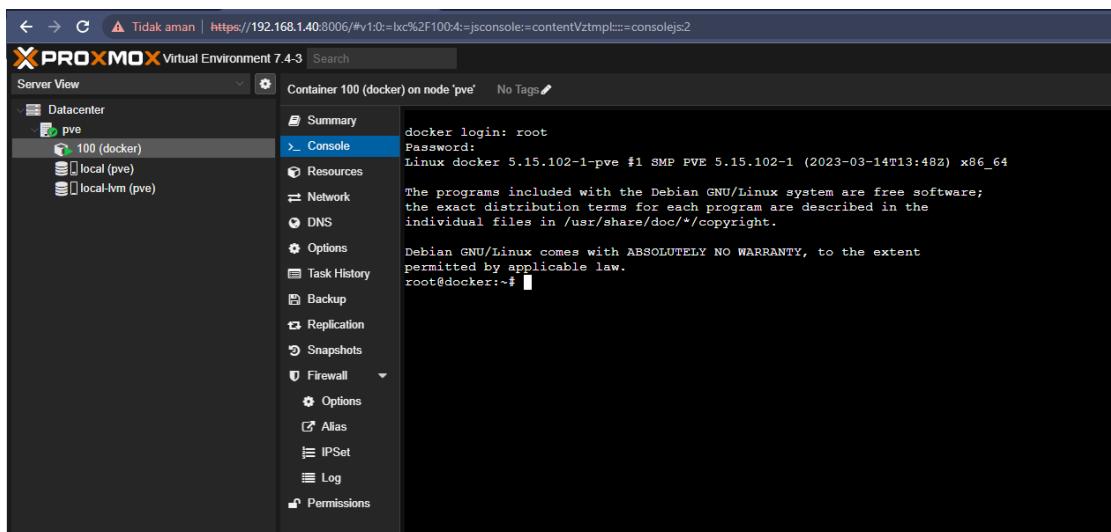
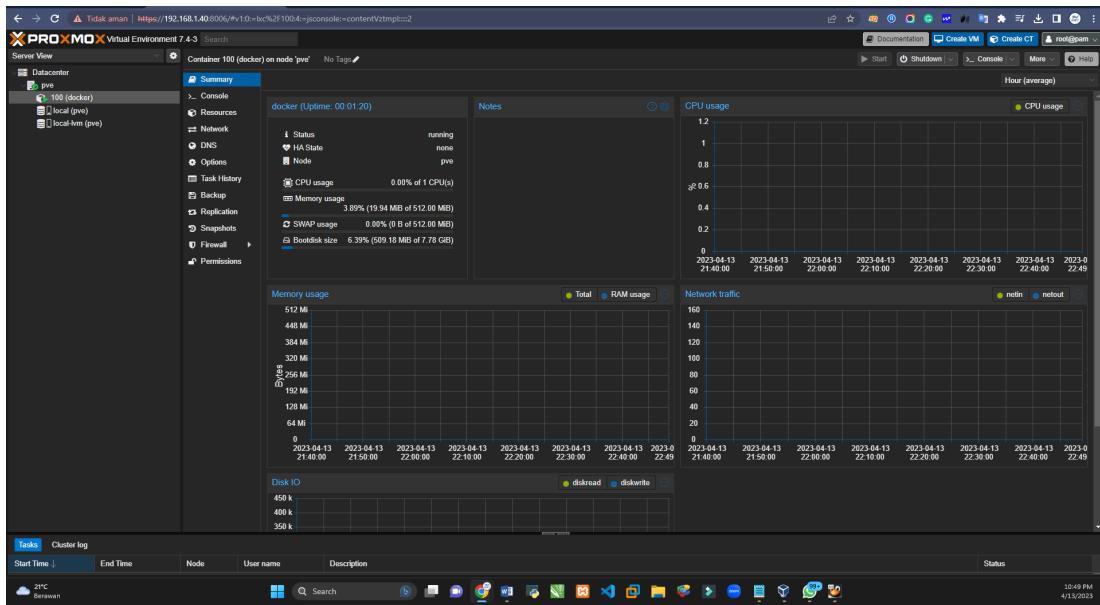
Output Status

Stop Download

```
Logical volume "vm-100-disk-0" created.
Creating filesystem with 2097152 4k blocks and 524288 inodes
Filesystem UUID: 8fa3eb75-1206-4b91-bacd-b8635f243755
Superblock backup stored on blocks:
32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632
extracting archive '/var/lib/vz/template/cache/debian-11-standard_11.6-1_amd64.tar.zst'
Total bytes read: 489799680 (468MiB, 151MiB/s)
Detected container architecture: amd64
Creating SSH host key ssh_host_ecdsa_key' - this may take some time ...
done: SHA256:uJ/CkVLb7cnrVhAA6PIDu9pFSvQoncrzj+0JEbj4 root@docker
Creating SSH host key 'ssh_host_dsa_key' - this may take some time ...
done: SHA256:PsZ6gtfC19j0tCHFxKvAziPoRA6eWBNipg2vnWMs root@docker
Creating SSH host key ssh_host_rsa_key' - this may take some time ...
done: SHA256:mfTmWuKU7cheabzO2hN522Sd3B7K1z7tbGJqzbPM root@docker
Creating SSH host key 'ssh_host_ed25519_key' - this may take some time ...
done: SHA256:GM6gTFiCrYSLbMe2xKKkZzE0sTk0ula/bQDxsKUBc root@docker
TASK OK
```

- Pilih LXC Container yang baru kita buat, lalu pilih Options dan Edit the Features Aktifkan keyctl , lalu pilih OK. Anda sekarang dapat memulai wadah!





11. Setelah penampung dimulai, login dengan nama pengguna root dan kata sandi yang diatur di langkah empat. Jalankan perintah di bawah ini untuk memperbarui sistem.

```
docker login: root
Password:
Welcome to Ubuntu 22.10 (GNU/Linux 5.15.102-1-pve x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

```
apt update
```

```
root@docker:~# apt update
Ign:1 http://archive.ubuntu.com/ubuntu kinetic InRelease
Ign:2 http://archive.ubuntu.com/ubuntu kinetic-updates InRelease
Ign:3 http://archive.ubuntu.com/ubuntu kinetic-security InRelease
Ign:1 http://archive.ubuntu.com/ubuntu kinetic InRelease
Ign:2 http://archive.ubuntu.com/ubuntu kinetic-updates InRelease
Ign:3 http://archive.ubuntu.com/ubuntu kinetic-security InRelease
Ign:1 http://archive.ubuntu.com/ubuntu kinetic InRelease
Ign:2 http://archive.ubuntu.com/ubuntu kinetic-updates InRelease
Ign:3 http://archive.ubuntu.com/ubuntu kinetic-security InRelease
Err:1 http://archive.ubuntu.com/ubuntu kinetic InRelease
  Temporary failure resolving 'archive.ubuntu.com'
Err:2 http://archive.ubuntu.com/ubuntu kinetic-updates InRelease
  Temporary failure resolving 'archive.ubuntu.com'
Err:3 http://archive.ubuntu.com/ubuntu kinetic-security InRelease
  Temporary failure resolving 'archive.ubuntu.com'
Reading package lists... Done
Building dependency tree... Done
```

```
update -ya
```

```
root@docker:~# apt upgrade -y
Reading package lists... Done
Building dependency tree... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

12. Setelah sistem diperbarui, jalankan perintah di bawah ini untuk menginstal curl . Kami akan menggunakan curl untuk menjalankan skrip yang menginstal buruh pelabuhan.

```
root@docker:~# apt install curl -y
Reading package lists... Done
Building dependency tree... Done
The following additional packages will be installed:
  libbrotli1 libcurl4 libldap-2.5-0 libldap-common librtmp1 libssh-4
The following NEW packages will be installed:
  curl libbrotli1 libcurl4 libldap-2.5-0 libldap-common librtmp1 libssh-4
0 upgraded, 7 newly installed, 0 to remove and 0 not upgraded.
Need to get 1253 kB of archives.
After this operation, 3500 kB of additional disk space will be used.
Ign:1 http://archive.ubuntu.com/ubuntu kinetic/main amd64 libbrotli1 amd64 1.0.9-2build6
Ign:2 http://archive.ubuntu.com/ubuntu kinetic/main amd64 libldap-2.5-0 amd64 2.5.13+dfsg-1ubuntul
Ign:3 http://archive.ubuntu.com/ubuntu kinetic/main amd64 librtmp1 amd64 2.4+20151223.gitfa8646d.1-2build4
Ign:4 http://archive.ubuntu.com/ubuntu kinetic/main amd64 libssh-4 amd64 0.9.6-2build1
Ign:5 http://archive.ubuntu.com/ubuntu kinetic/main amd64 libcurl4 amd64 7.85.0-1
Ign:6 http://archive.ubuntu.com/ubuntu kinetic/main amd64 curl amd64 7.85.0-1
Ign:7 http://archive.ubuntu.com/ubuntu kinetic/main amd64 libldap-common all 2.5.13+dfsg-1ubuntul
Ign:1 http://archive.ubuntu.com/ubuntu kinetic/main amd64 libbrotli1 amd64 1.0.9-2build6
Ign:2 http://archive.ubuntu.com/ubuntu kinetic/main amd64 librtmp-2.5-0 amd64 2.5.13+dfsg-1ubuntul
Ign:3 http://archive.ubuntu.com/ubuntu kinetic/main amd64 librtmp1 amd64 2.4+20151223.gitfa8646d.1-2build4
Ign:4 http://archive.ubuntu.com/ubuntu kinetic/main amd64 libssh-4 amd64 0.9.6-2build1
Ign:5 http://archive.ubuntu.com/ubuntu kinetic/main amd64 libcurl4 amd64 7.85.0-1
Ign:6 http://archive.ubuntu.com/ubuntu kinetic/main amd64 curl amd64 7.85.0-1
Ign:7 http://archive.ubuntu.com/ubuntu kinetic/main amd64 libldap-common all 2.5.13+dfsg-1ubuntul
Ign:1 http://archive.ubuntu.com/ubuntu kinetic/main amd64 libbrotli1 amd64 1.0.9-2build6
Ign:2 http://archive.ubuntu.com/ubuntu kinetic/main amd64 libldap-2.5-0 amd64 2.5.13+dfsg-1ubuntul
```

```
apt install docker.io
```

```

root@Test-CT:~# apt install docker.io
Reading package lists... Done
Building dependency tree... Done
The following additional packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base git git-man libasn1-8-heimdal libbrotlii libcurl3-gnutls liberror-perl lib
  libcrypto4-heimdal libheimbase1-heimdal libheimntlm0-heimdal libhx509-5-heimdal libidn11 libkrb5-26-heimdal libldap-2.4-2 lib
  libroken18-heimdal librtmp1 libseccomp2 libssh-4 libwind0-heimdal patch perl perl-base perl-modules-5.30 pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools btrfs-progs cgroupfs-mount | cgroup-lite debootstrap docker-doc rinse zfs-fuse | zfsutils git-daemon-run
  git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn diffutils-doc perl-doc libterm-readline-gnu-perl | libterm-readli
  liblocale-codes-perl
The following NEW packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base docker.io git git-man libasn1-8-heimdal libbrotlii libcurl3-gnutls liberr
  libgssapi3-heimdal libcrypto4-heimdal libheimbase1-heimdal libheimntlm0-heimdal libhx509-5-heimdal libidn11 libkrb5-26-heimda
  libnhttp2-14 libperl5.30 libroken18-heimdal librtmp1 libssh-4 libwind0-heimdal patch perl perl-modules-5.30 pigz runc ubuntu
The following packages will be upgraded:

```

apt install docker-compose

```

root@Test-CT:~# docker --version
Docker version 20.10.12, build 20.10.12-Ubuntu2~20.04.1
root@Test-CT:~# docker-compose

Command 'docker-compose' not found, but can be installed with:

apt install docker-compose

root@Test-CT:~# apt install docker-compose
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  python3-attr python3-cached-property python3-certifi python3-chardet python3-distutils python3-docker python3-dockerpty python3-
  python3-importlib-metadata python3-jsonschema python3-lib2to3 python3-more-itertools python3-persistent python3-requests pyth
  python3-urllib3 python3-websocket python3-zipp
Suggested packages:
  python3-attr-doc python3-jsonschema-doc python3-cryptography python3-openssl python3-socks python-setuptools-doc
The following NEW packages will be installed:
  docker-compose python3-attr python3-cached-property python3-certifi python3-chardet python3-distutils python3-docker python3-
  python3-importlib-metadata python3-jsonschema python3-lib2to3 python3-more-itertools python3-persistent python3-requests pyth
  python3-urllib3 python3-websocket python3-zipp
0 upgraded, 21 newly installed, 0 to remove and 166 not upgraded.
Need to get 1392 kB of archives.
After this operation, 7856 kB of additional disk space will be used.
Do you want to continue? [Y/n] 

```

13. Run Docker

```

root@Test-CT:~# docker-compose --version
docker-compose version 1.29.2, build 5becea4c
root@Test-CT:~# docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Full complete
Digest: sha256:80f31dalac7b312ba29d65080fdddf797dd76acf870e677f390d5acba9741b17
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/

```