

# **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.

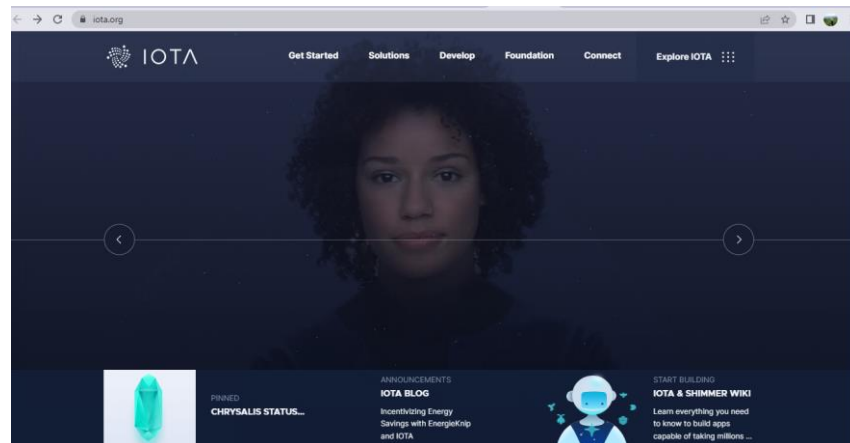
Huda Ubaya, M.T.

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

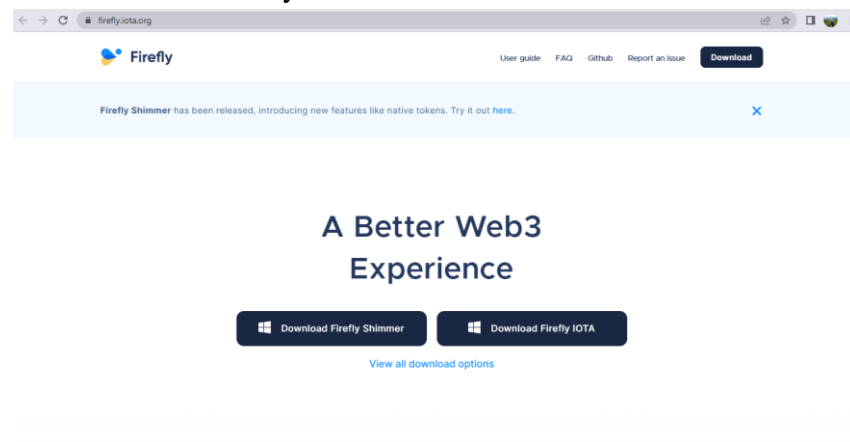
# INSTALASI IOTA, DOCKER, DAN PROXMOX

## A. Install IOTA

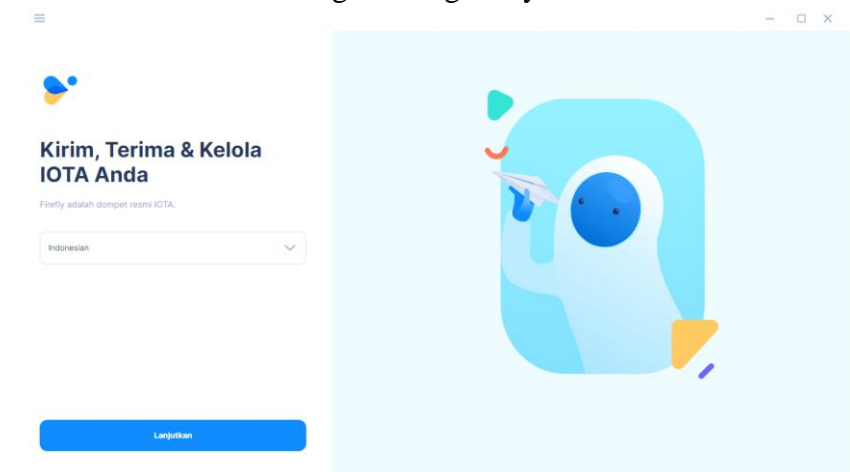
1. Pertama buka web IOTA di browser



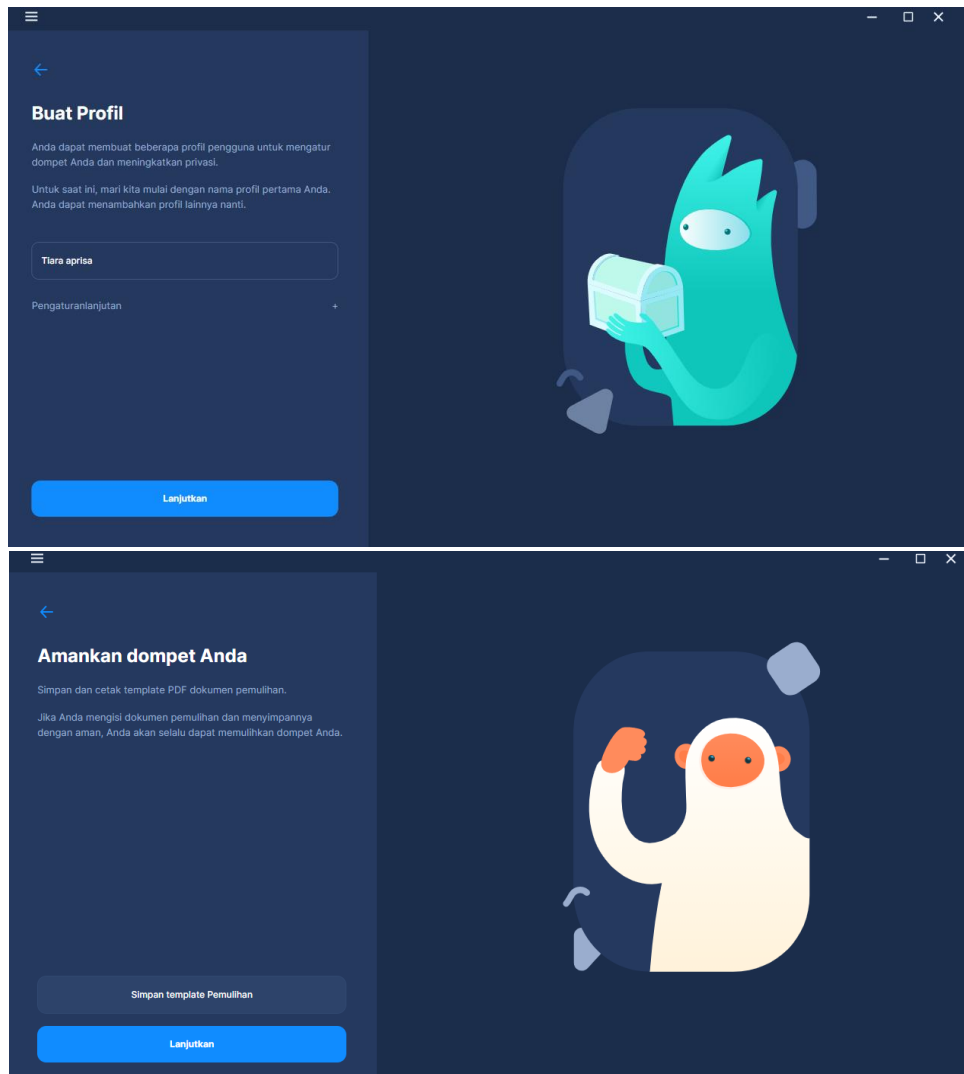
2. Kemudian Download Firefly IOTA



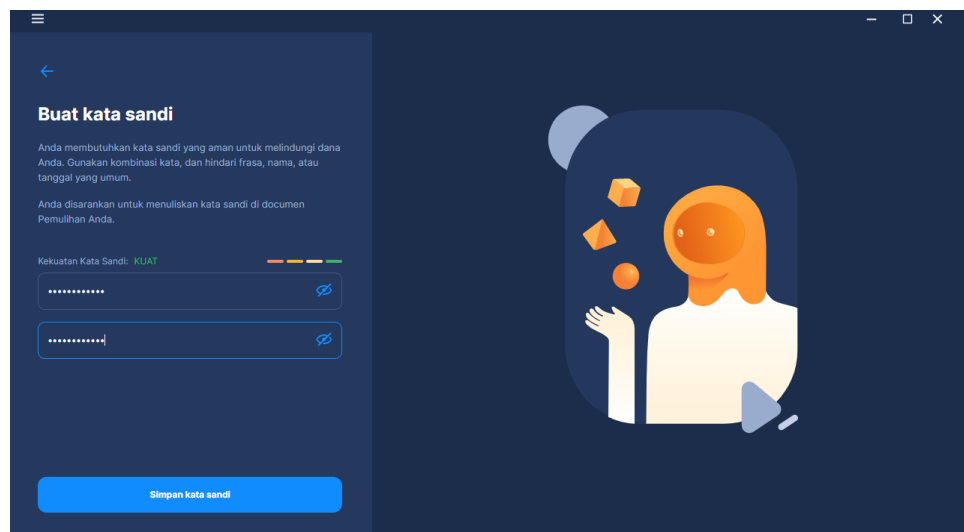
3. Setelah IOTA terinstal ikuti langkah-langkahnya



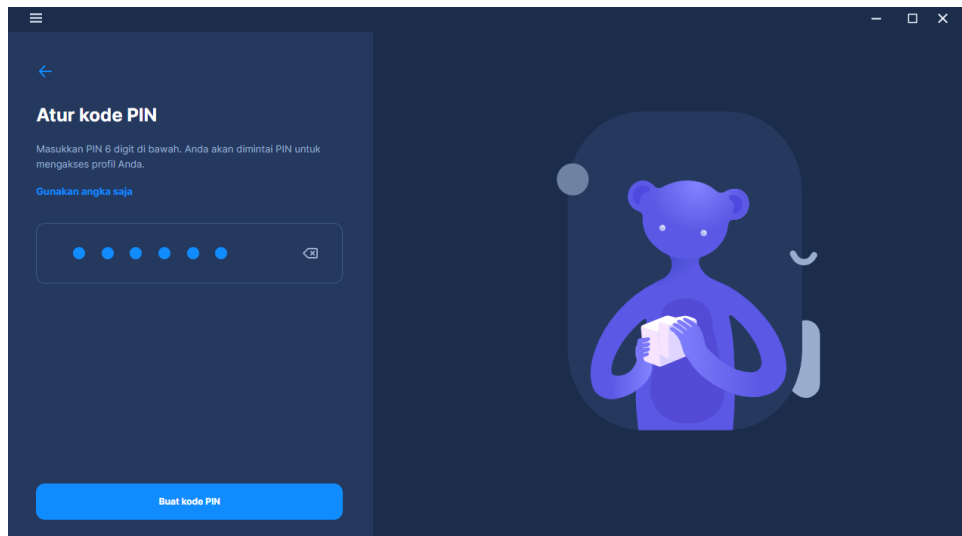
4. Buat akun Profil terlebih dahulu



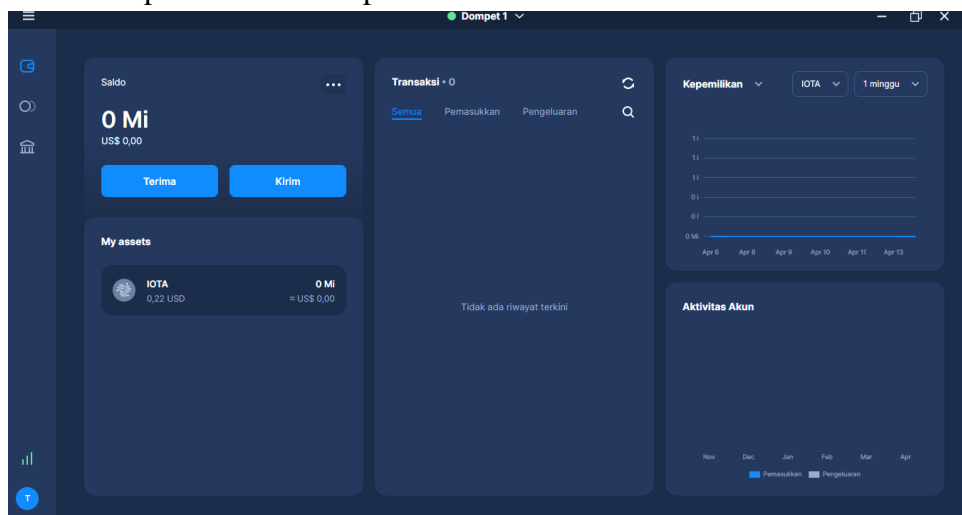
## 5. Kemudian Membuat Kata Sandi



## 6. Lalu Atur kode Pin

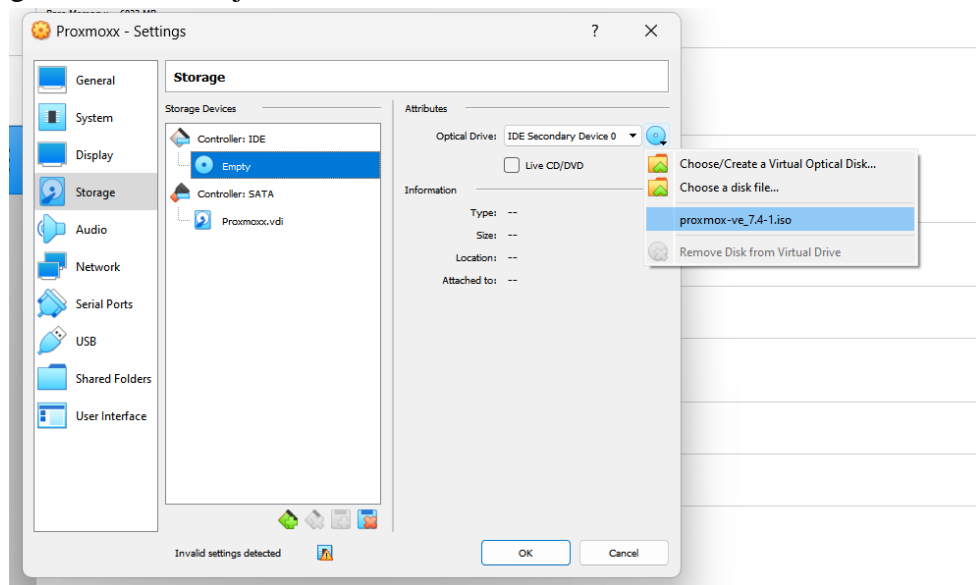


## 7. Berikut Tampilan Dashboard pada IOTA

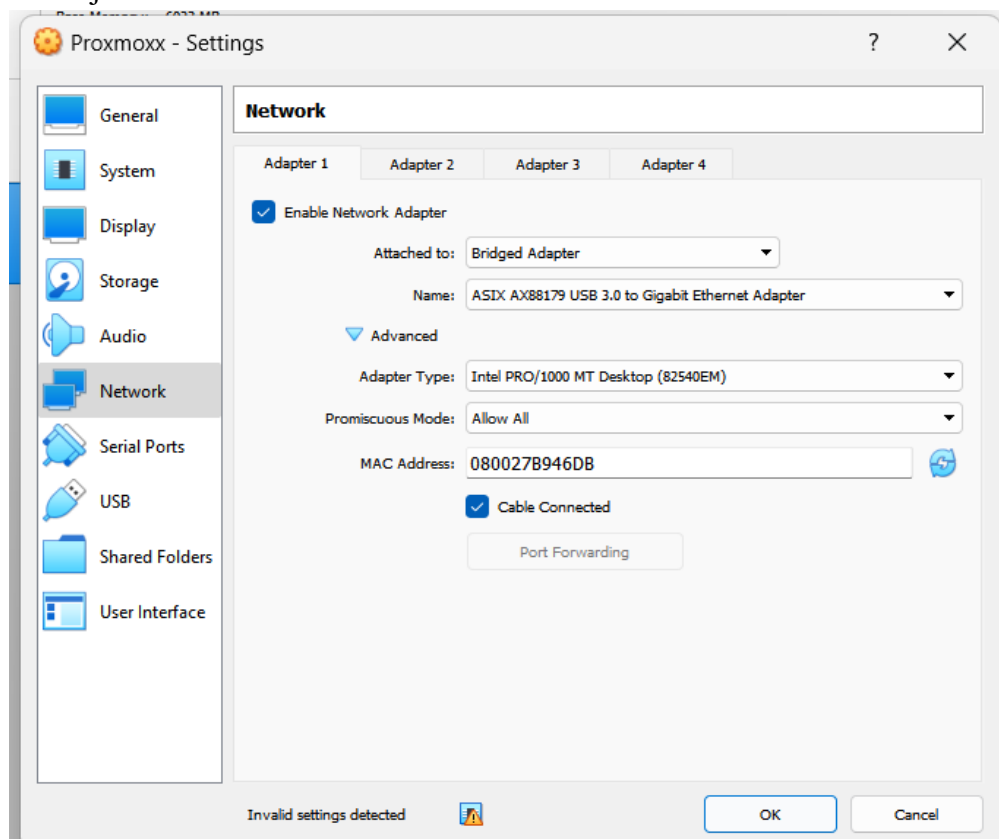


## B. Install Proxmox VE for VirtualBox Manager

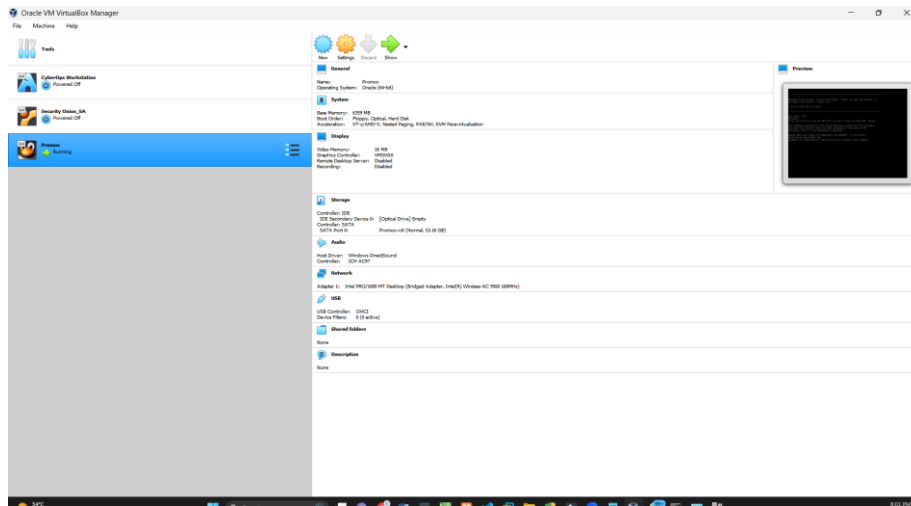
### 1. Storage ubah IDE menjadi Proxmox



### 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



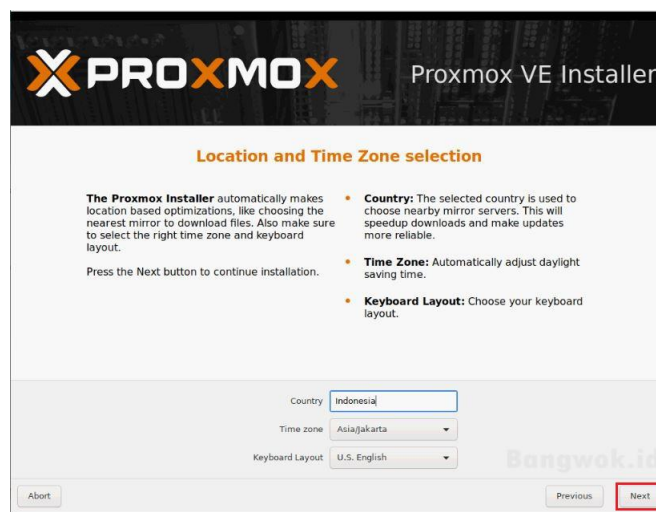
### 5. lalu klik I Agree



6. Klik Next



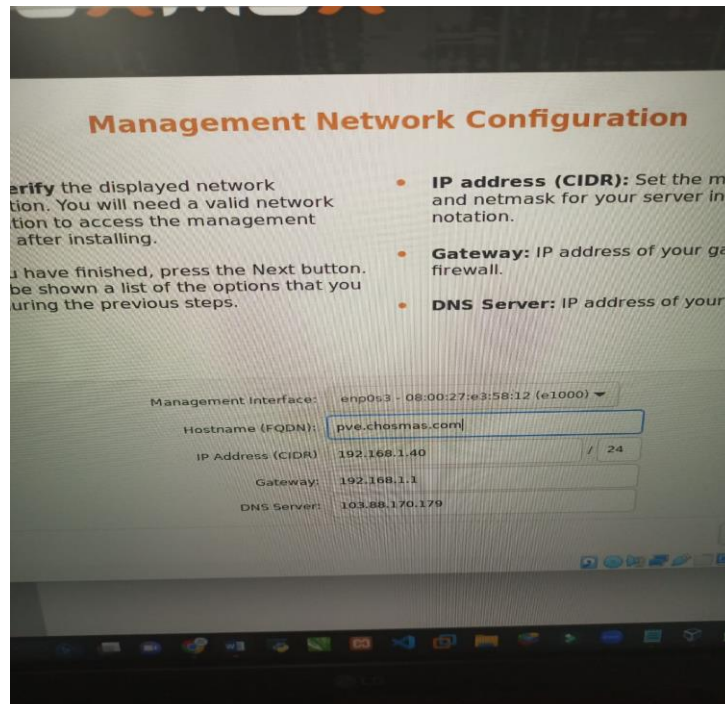
7. Pilih Indonesia lalu next



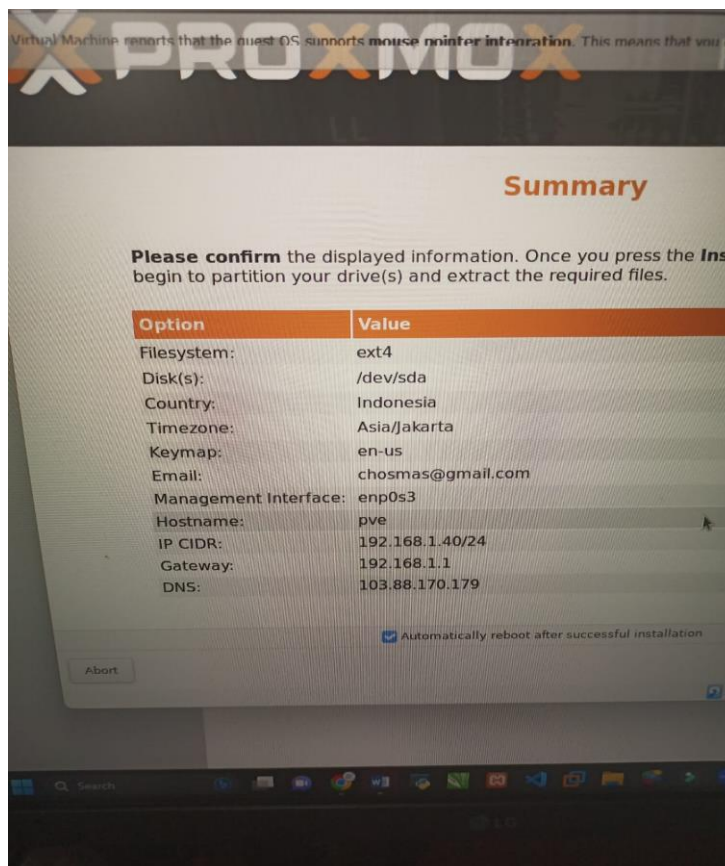
8. Masukkan Password dan email anda lalu next



9. Masukkan Hostname anda lalu next



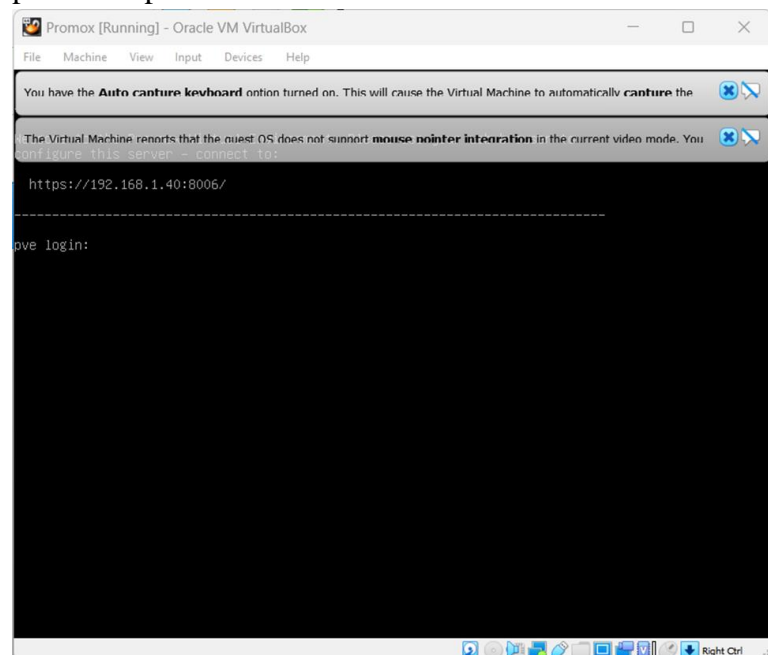
10. Klik Install



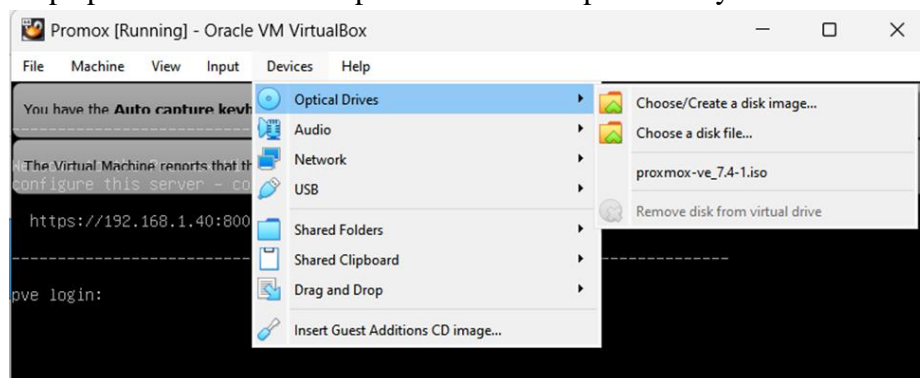




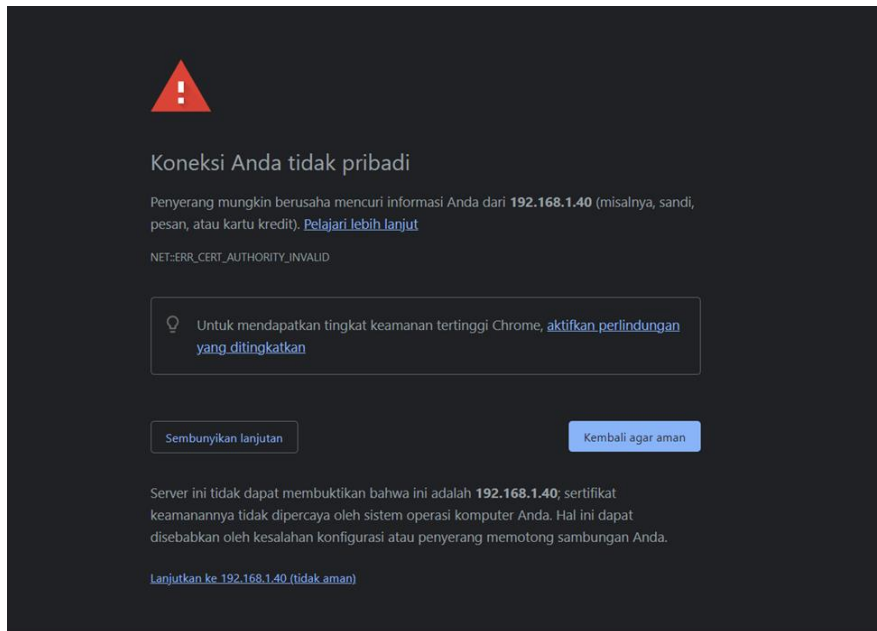
Berikut merupakan tampilan Proxmox IP



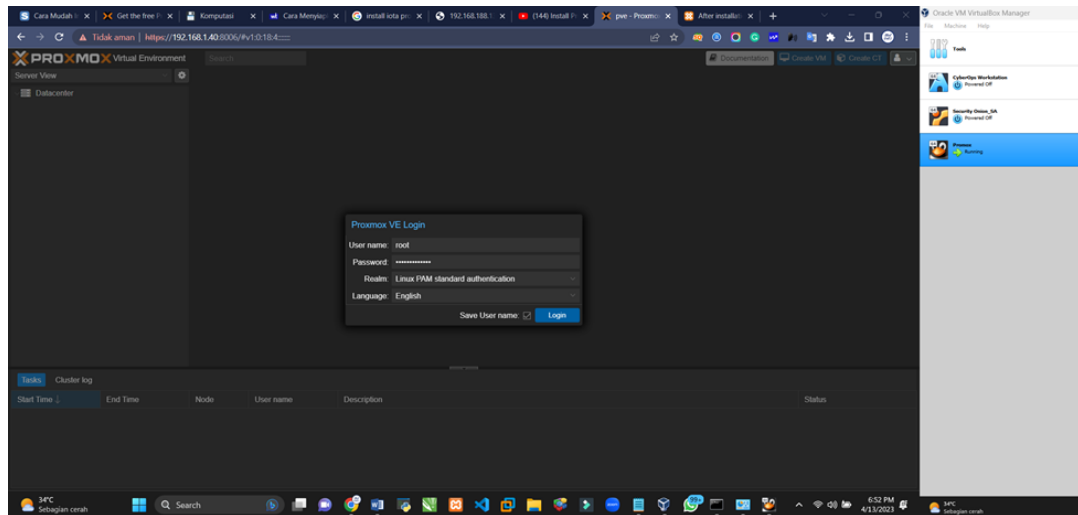
11. Jangan Lupa pada Devices Klik Option lalu cheklis proxmoxnya



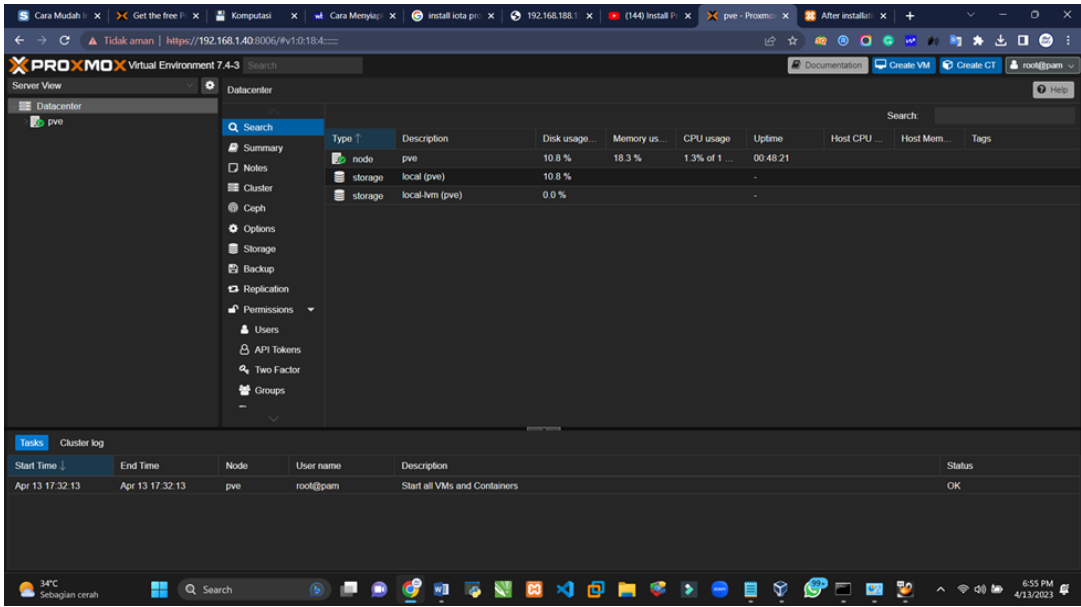
## 12. Masukkan IP yang tertera pada Proxmox ke web browser



## 13. Masukan Username dan password

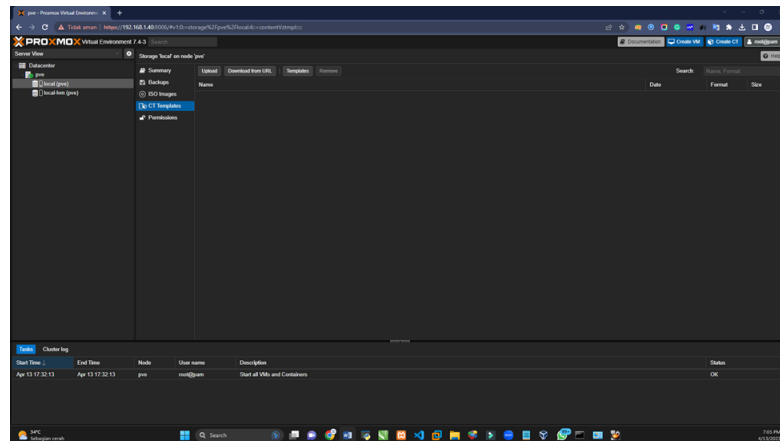


# 14. Tampilan Proxmox

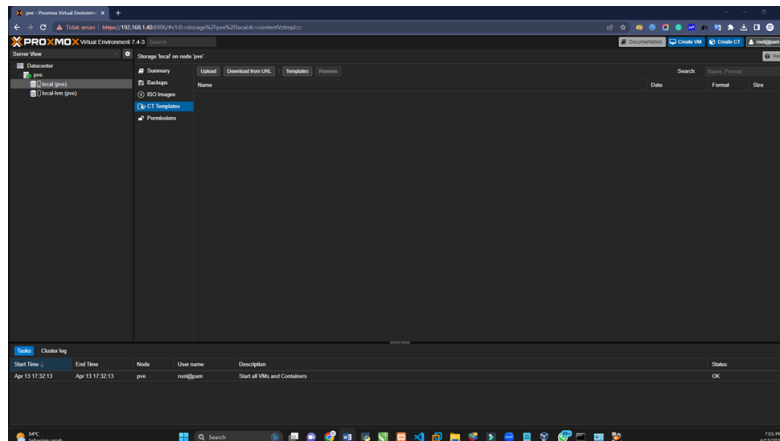


## C. Install Docker in Proxmox

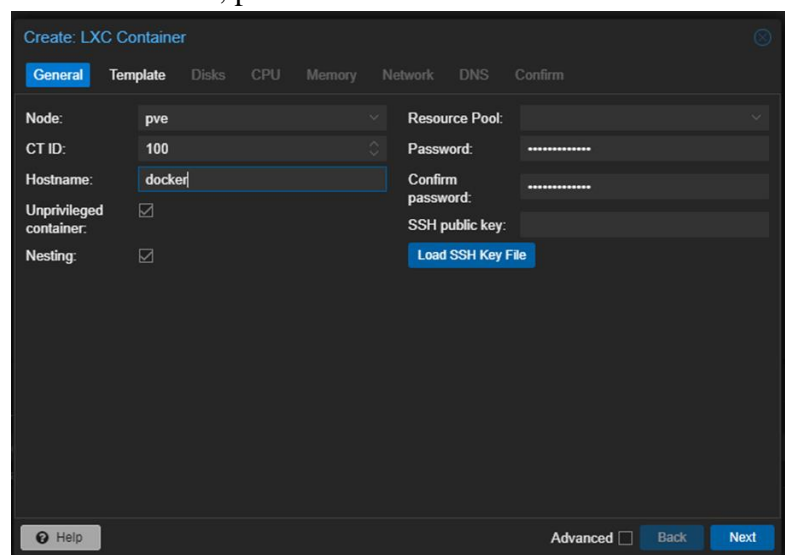
1. Masuk ke Proxmox, pilih lokasi penyimpanan tempat Anda ingin menyimpan template container, pilih CT Templates , lalu pilih Templates.



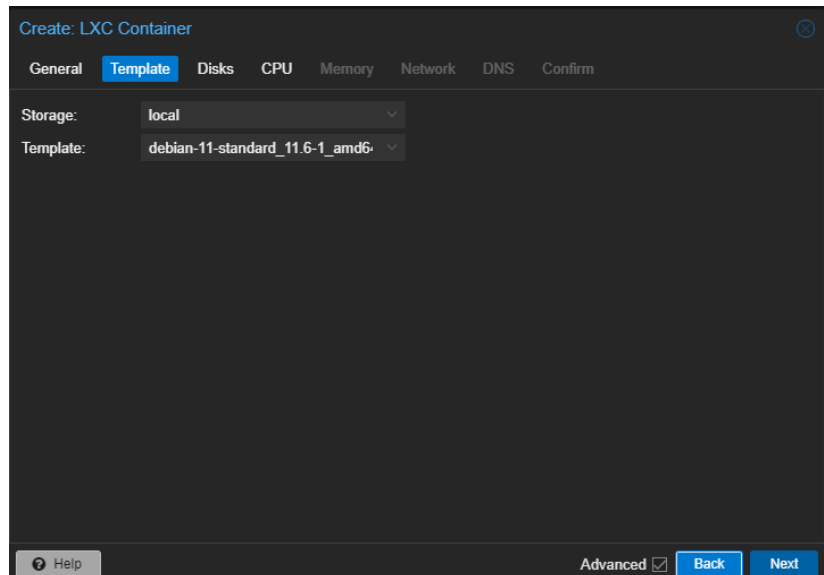
2. Cari Debian , lalu pilih debian-11-standard dan Download.



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.

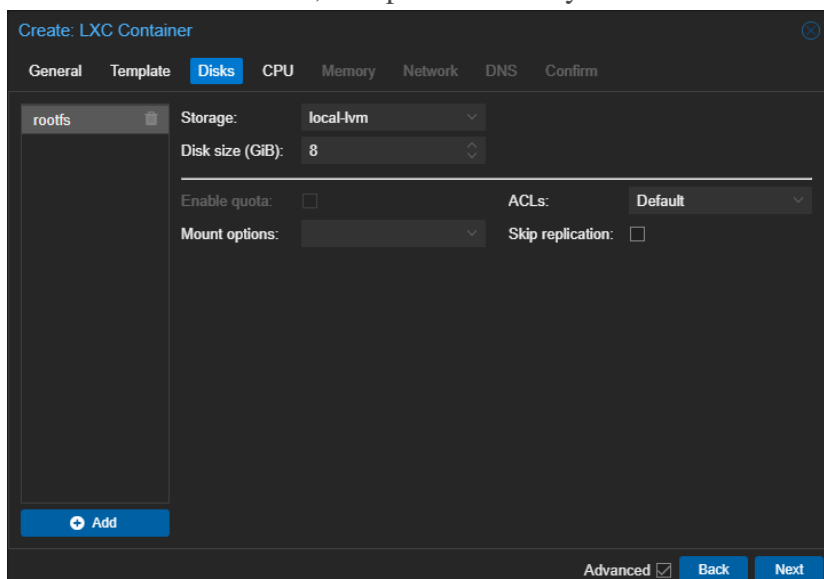


- Pilih Template , lalu pilih Next untuk melanjutkan.



The screenshot shows the 'Create: LXC Container' dialog box with the 'Template' tab selected. The 'Storage' dropdown is set to 'local' and the 'Template' dropdown is set to 'debian-11-standard\_11.6-1\_amd64'. At the bottom, there is a 'Help' button, an 'Advanced' checkbox that is checked, and 'Back' and 'Next' buttons.

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



The screenshot shows the 'Create: LXC Container' dialog box with the 'Disks' tab selected. On the left, there is a list of disks with 'rootfs' selected. On the right, the 'Storage' dropdown is set to 'local-lvm', the 'Disk size (GiB)' is set to '8', 'Enable quota' is unchecked, 'ACLs' is set to 'Default', 'Mount options' is empty, and 'Skip replication' is unchecked. At the bottom left, there is an 'Add' button. At the bottom right, there is an 'Advanced' checkbox that is checked, and 'Back' and 'Next' buttons.

- Pilih total Core untuk CPU, lalu pilih Next.

Create: LXC Container

General Template Disks **CPU** Memory Network DNS Confirm

Cores: 1

CPU limit: unlimited CPU units: 5000

Help Advanced ☒ Back Next

7. Atur total Memory , lalu pilih Next .

Create: LXC Container

General Template Disks CPU **Memory** Network DNS Confirm

Memory (MiB): 512

Swap (MiB): 512

Help Advanced ☒ Back 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

General Template Disks CPU Memory **Network** DNS Confirm

Name: eth0 IPv4: Static DHCP  
 MAC address: auto IPv4/CIDR:  
 Bridge: vmbri0 Gateway (IPv4):  
 VLAN Tag: no VLAN IPv6: Static DHCP SLAAC  
 Firewall: ☒ IPv6/CIDR: None  
 Gateway (IPv6):  
 Disconnect: ☐ 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

General Template Disks CPU Memory Network DNS **Confirm**

Key ↑	Value
cores	1
cpunits	5000
features	nesting=1
hostname	docker
memory	512
net0	name=eth0,bridge=vmbri0,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

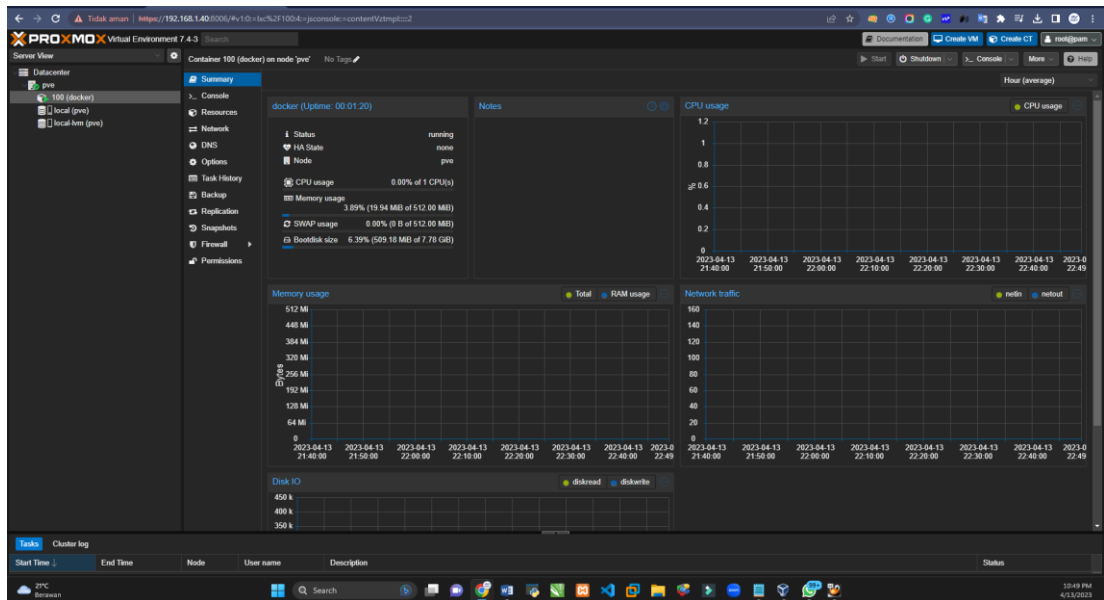
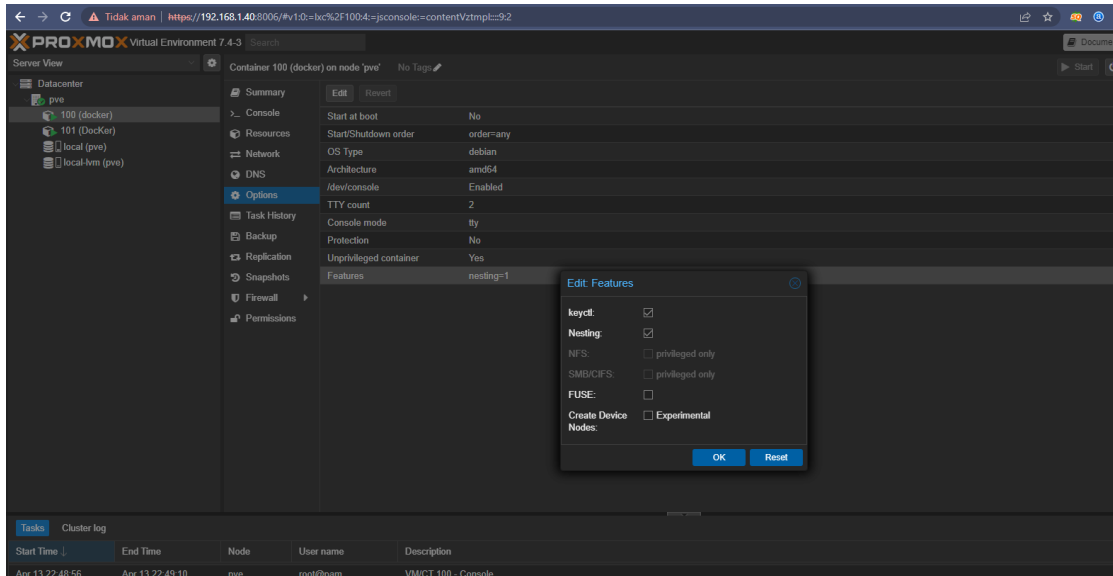
Output Status

Stop Download

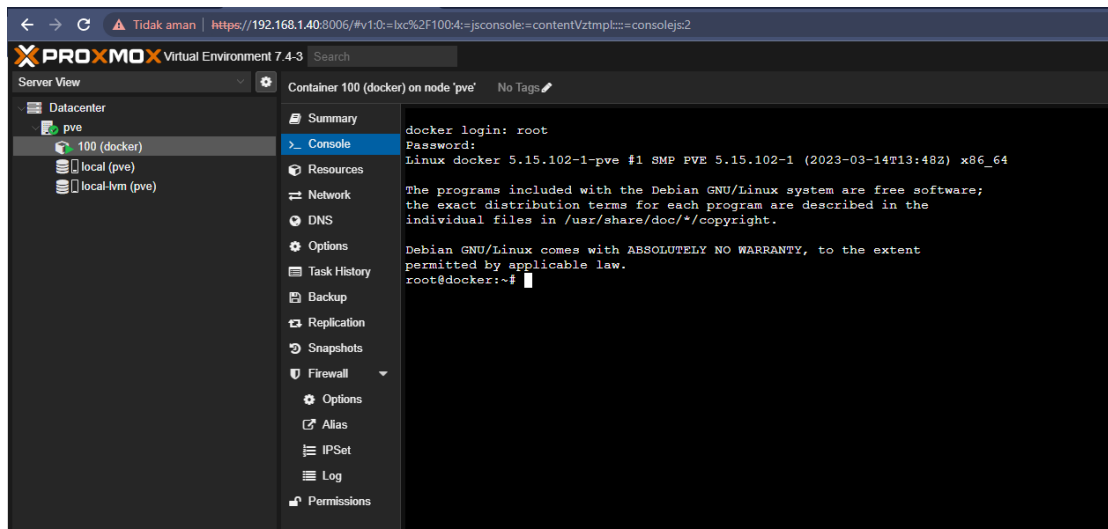
```

Logical volume "vm-100-disk-0" created.
Creating filesystem with 2097152 4k blocks and 524288 inodes
Filesystem UUID: 8fa3eb75-1206-4b91-bacd-b8635f243755
Superblock backups 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:uL/c/kVLb7cnrVlhAA6PIDu9pF5vQoncrz+OIEbj4 root@docker
Creating SSH host key 'ssh_host_dsa_key' - this may take some time ...
done: SHA256:PeZ6gtfCI9j00tChkF3XvAZfPoRA6elWBNlpg2vmWMs root@docker
Creating SSH host key 'ssh_host_rsa_key' - this may take some time ...
done: SHA256:imTmWuKU7cheabz02hNSZ2Z5d367KzI7tbG3qzBPM root@docker
Creating SSH host key 'ssh_host_ed25519_key' - this may take some time ...
done: SHA256:GM6gTfCfYSLbMeZxKkKzE0sTkula/bQDxksKUBc root@docker
TASK OK
  
```

10. 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:
Linux docker 5.15.102-1-pve #1 SMP PVE 5.15.102-1 (2023-03-14T13:48Z) x86_64

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to
the extent
permitted by applicable law.
```

apt update

```
root@Test-CT:~# apt update
Get:1 http://archive.ubuntu.com/ubuntu focal InRelease [265 kB]
Get:2 http://archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:3 http://archive.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:4 http://archive.ubuntu.com/ubuntu focal/main Translation-en [506 kB]
Get:5 http://archive.ubuntu.com/ubuntu focal/main amd64 c-n-f Metadata [29.5 kB]
Get:6 http://archive.ubuntu.com/ubuntu focal/restricted Translation-en [6212 B]
Get:7 http://archive.ubuntu.com/ubuntu focal/restricted amd64 c-n-f Metadata [392 B]
Get:8 http://archive.ubuntu.com/ubuntu focal/universe Translation-en [5124 kB]
18% [8 Translation-en 248 kB/5124 kB 5%]
```

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 libbrotli1 libcurl3-gnutls liberror-perl lib
  libhcrypto4-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-fa
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-readl
  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 libbrotli1 libcurl3-gnutls liberr
  libgssapi3-heimdal libhcrypto4-heimdal libheimbase1-heimdal libheimntlm0-heimdal libhx509-5-heimdal libidn11 libkrb5-26-heim
  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 20.10.12-0ubuntu2~20.04.1
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 pyth
  python3-importlib-metadata python3-jsonschema python3-lib2to3 python3-more-itertools python3-pyrsistent python3-requests pyth
  python3-urllib3 python3-websocket python3-zipp
Suggested packages:
  python-attr-doc python-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-pyrsistent 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]

```

## 12. 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: Pull complete
Digest: sha256:80f31dalac7b312ba29d65080fddf797dd76acfb870e677f390d5acba9741b17
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/

```