

Jobsheet Konfigurasi OwnCloud



Disusun oleh:

Khairunnisa Alwita, S.Kom



22.04 LTS

**Ubuntu Server
22.04**

Konfigurasi OwnCloud di Ubuntu Server

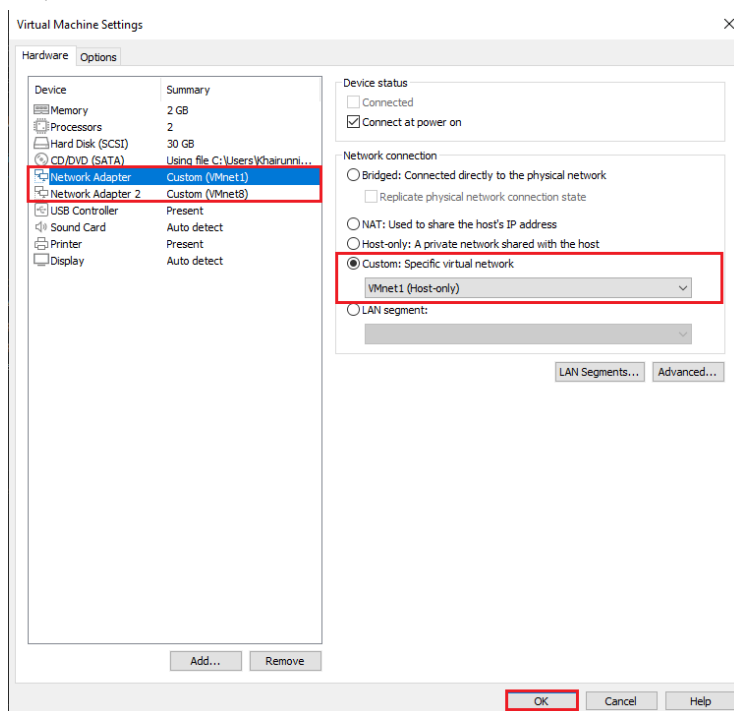
1. Silahkan masuk ke server Ubuntu dengan mode super user dengan tanda pagar. Jika sudah terlanjur masuk menggunakan user yang ditandai dengan tanda dolar “\$” silahkan masuk ke mode super user dengan perintah **sudo su -l** lalu ketikkan *password* **root**.

```
tkj@tkj:~$
```

Ketikkan **sudo su -l** untuk masuk sebagai super user.

```
tkj@tkj:~$ sudo su -l
[sudo] password for tkj:
root@tkj:~#
```

2. Sebelum melakukan konfigurasi silahkan cek terlebih dahulu nama interface pada server Ubuntu kamu. Pada konfigurasi menggunakan 2 network interface. Yakni *interface* pertama untuk lokal server dan *interface* kedua untuk koneksi ke internet. Untuk lebih jelas tentang pengaturan konfigurasi *interface* silahkan simak pada tutorial ini.



3. Silahkan cek nama *interface virtual* di server Ubuntu dengan perintah “**ip a**” dan akan terlihat nama *interface* misalnya **ens33**, **ens37** dan lainnya.

```
root@tkj:~# ip a
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: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:9b:bc:5c brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet6 fe80::20c:29ff:fe9b:bc5c/64 scope link
        valid_lft forever preferred_lft forever
3: ens37: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 00:0c:29:9b:bc:66 brd ff:ff:ff:ff:ff:ff
    altname enp2s5
root@tkj:~#
```

4. Kemudian masuk ke konfigurasi network dengan perintah berikut “**vi /etc/netplan/ 00-installer-config.yaml**” .

```
root@tkj:~# vi /etc/netplan/00-installer-config.yaml
```

5. Lakukan konfigurasi pada *netplan* dengan konfigurasi seperti dibawah ini. Pada konfigurasi ini terdapat 2 network *interface* yang digunakan. *Interface* pertama untuk lokal server dalam hal ini menggunakan “**ens33**” dengan mode *static* yang dalam hal ini digunakan perintah **dhcp:false** setelah itu masukkan alamat IP yang ingin digunakan. Untuk bisa mengkonfigurasi dengan perintah di tahapan ke 3 silahkan ketik **i** pada keyboard agar kita bisa mengedit isi file. Dan untuk *interface* kedua “**ens37**” digunakan untuk koneksi ke jaringan internet dengan mode *dhcp* yang dalam hal ini digunakan perintah **dhcp:true**. Untuk menyimpan konfigurasi gunakan **shift+zz** pada keyboard.

```
# This is the network config written by 'subiquity'
network:
  ethernets:
    ens33:
      dhcp4: false
      addresses :
        - 192.168.22.1/24
    ens37:
      dhcp4: true
  version: 2
~
```

6. Masukkan perintah “**sudo netplan apply**” untuk menyimpan konfigurasi tersebut di *netplan* yang baru.

```
root@tkj:~# sudo netplan apply
```

7. Setelah itu lakukan pengecekan kembali pada IP Address yang sudah dikonfigurasi dengan perintah **ip a** kembali.

```
root@tkj:~# ip a
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: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:9b:bc:5c brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.22.1/24 brd 192.168.22.255 scope global ens33
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe9b:bc5c/64 scope link
        valid_lft forever preferred_lft forever
3: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:9b:bc:66 brd ff:ff:ff:ff:ff:ff
    altname enp2s5
    inet 192.168.103.242/24 metric 100 brd 192.168.103.255 scope global dynamic ens37
        valid_lft 3529sec preferred_lft 3529sec
    inet6 2404:c0:1c61:2564:20c:29ff:fe9b:bc66/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 7132sec preferred_lft 7132sec
    inet6 fe80::20c:29ff:fe9b:bc66/64 scope link
        valid_lft forever preferred_lft forever
root@tkj:~#
```

8. Untuk menguji koneksi yang sudah dibuat silahkan uji konektivitas jaringan dengan *utili ping* ke IP google. Jika hasil menunjukkan TTL maka konfigurasi sudah bisa dilanjutkan.

```
root@tkj:~# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=54 time=62.0 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=54 time=53.8 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=54 time=40.3 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=54 time=41.0 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=54 time=40.0 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=54 time=76.9 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=54 time=50.9 ms
^C
--- 8.8.8.8 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6011ms
rtt min/avg/max/mdev = 40.049/52.139/76.920/12.673 ms
root@tkj:~# _
```

9. Lakukan perintah update “*apt update*” dan tunggu prosesnya setelah itu masukkan perintah upgrade “*apt upgrade*” tunggu prosesnya hingga selesai.

```
root@tkj:~# apt update
root@tkj:~# apt upgrade
```

10. Setelah itu lakukan instalasi web server dengan menginstall *apache2* dengan menggunakan perintah “*apt install apache2*”

```
root@tkj:~# apt install apache2
```

11. Hanya untuk melihat status apache2 lakukan perintah “*systemctl status apache2*” jika sudah terdapat kata **active running** maka apache2 sudah berjalan.

```
root@tkj:~# systemctl status apache2
• apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2025-01-12 10:31:13 UTC; 53s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2034 (apache2)
    Tasks: 55 (limit: 2182)
   Memory: 4.9M
      CPU: 117ms
   CGroup: /system.slice/apache2.service
           └─2034 /usr/sbin/apache2 -k start
             └─2036 /usr/sbin/apache2 -k start
               └─2037 /usr/sbin/apache2 -k start

Jan 12 10:31:12 tkj systemd[1]: Starting The Apache HTTP Server...
Jan 12 10:31:12 tkj apachectl[2033]: AH00558: apache2: Could not reliably determine the server's fu
Jan 12 10:31:13 tkj systemd[1]: Started The Apache HTTP Server.
lines 1-16/16 (END)
```

12. Selanjutnya tambahkan repositori ondrej untuk **php** yang akan di install “*add-apt-repository ppa:ondrej/php*”

```
root@tkj:~# add-apt-repository ppa:ondrej/php
```

13. Setelah itu lakukan update dengan perintah “*apt update*”

```
root@tkj:~# apt update
```

14. Kemudian install php dengan perintah berikut “***apt install php7.4 php7.4-{opcache,gd,curl,mysqlnd,intl,json,ldap,mbstring,mysqlnd,xml,zip}***”

```
root@tkj:~# apt install php7.4 php7.4-{opcache,gd,curl,mysqlnd,intl,json,ldap,mbstring,mysqlnd,xml,zip}
```

15. Selanjutnya install *Mariadb server* dengan perintah “***apt install mariadb-server***”

```
root@tkj:~# apt install mariadb-server
```

16. Kemudian untuk membuat instalasi yang aman masukkan perintah “***mysql_secure_installation***”

```
root@tkj:~# mysql_secure_installation
```

17. Ikuti langkah berikut pada secure instalasi yang dilakukan. Password yang dibuat disini adalah password yang akan digunakan untuk masuk ke konfigurasi MYSQL Server.

```
Enter current password for root (enter for none):
Set root password? [Y/n] Y
New password:
Re-enter new password:
Remove anonymous users? [Y/n] Y
Disallow root login remotely? [Y/n] Y
Remove test database and access to it? [Y/n] Y
Reload privilege tables now? [Y/n] Y
```

18. Login ke MYSQL server dan gunakan user **root** dengan perintah “***mysql -u root -p***”

```
root@tkj:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 41
Server version: 10.6.18-MariaDB-0ubuntu0.22.04.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

19. Kita akan diminta memasukkan password yang kita buat tadi. Selanjutnya lakukan perintah ini untuk memuat *database* owncloud yang dibuat.

```
MariaDB [(none)]> CREATE DATABASE owncloud;
MariaDB [(none)]> GRANT ALL PRIVILEGES ON owncloud.* TO 'owncloud'@'localhost' IDENTIFIED BY 'Pass123';
MariaDB [(none)]> FLUSH PRIVILEGES;
MariaDB [(none)]> EXIT;
```

Pada bagian ini tulisan yang di bold italic boleh diganti sesuai dengan yang diinginkan yakni untuk nama database, nama user dan password usernya boleh disesuaikan.

```

MariaDB [(none)]> CREATE DATABASE owncloud;
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON owncloud.* TO 'owncloud'@'localhost' IDENTIFIED BY 'Pass123';
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> exit;
Bye
root@tkj:~#

```

20. Setelah itu lakukan download OwnCloud dari website resminya dengan perintah berikut ***“wget https://download.owncloud.com/server/stable/owncloud-complete-latest.zip”***

```
root@tkj:~# wget https://download.owncloud.com/server/stable/owncloud-complete-latest.zip
```

21. Silahkan install unzip untuk dapat mengekstrak file *owncloud* yang sudah di download sebelumnya dengan menggunakan perintah ***“apt install unzip”***.

```
root@tkj:~# apt install unzip
```

22. Kemudian lakukan perintah ini untuk unzip hasil download dari OwnCloud ***“unzip owncloud-complete-latest.zip -d /var/www/html/”***.

```
root@tkj:~# unzip owncloud-complete-latest.zip -d /var/www/html/
```

23. Kemudian jalankan perintah berikut untuk mengubah hak akses ownCloud menjadi www-data ***“chown -R www-data:www-data /var/www/html/owncloud/”***

```
root@tkj:~# chown -R www-data:www-data /var/www/html/owncloud/
```

24. Setelah itu silahkan copy file **000-default.conf** ke **owncloud.conf** dengan perintah berikut ***“cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/owncloud.conf”***

```
root@tkj:~# cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/owncloud.conf
```

25. Kemudian edit file **owncloud.conf** dengan perintah berikut ***“vi /etc/apache2/sites-available/owncloud.conf”***

```
root@tkj:~# vi /etc/apache2/sites-available/owncloud.conf
```

26. Pada file ini hanya dibaris *DocumentRoot* buat tambahkan *owncloud* hingga menjadi

DocumentRoot /var/www/html/owncloud

```

<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html/owncloud

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

```

27. Setelah itu kita akan lakukan perintah *disable* dan *enable site* pada *sites-available* yang terpada pada apache2. Pertama *disable* terlebih dahulu **000-default.conf** dengan perintah **"a2dissite 000-default.conf"** setelah itu *enable site* **owncloud** **"a2ensite owncloud.conf"**

```

root@tkj:~# a2dissite 000-default.conf
Site 000-default disabled.
To activate the new configuration, you need to run:
systemctl reload apache2
root@tkj:~# a2ensite owncloud.conf
Enabling site owncloud.
To activate the new configuration, you need to run:
systemctl reload apache2
root@tkj:~#

```

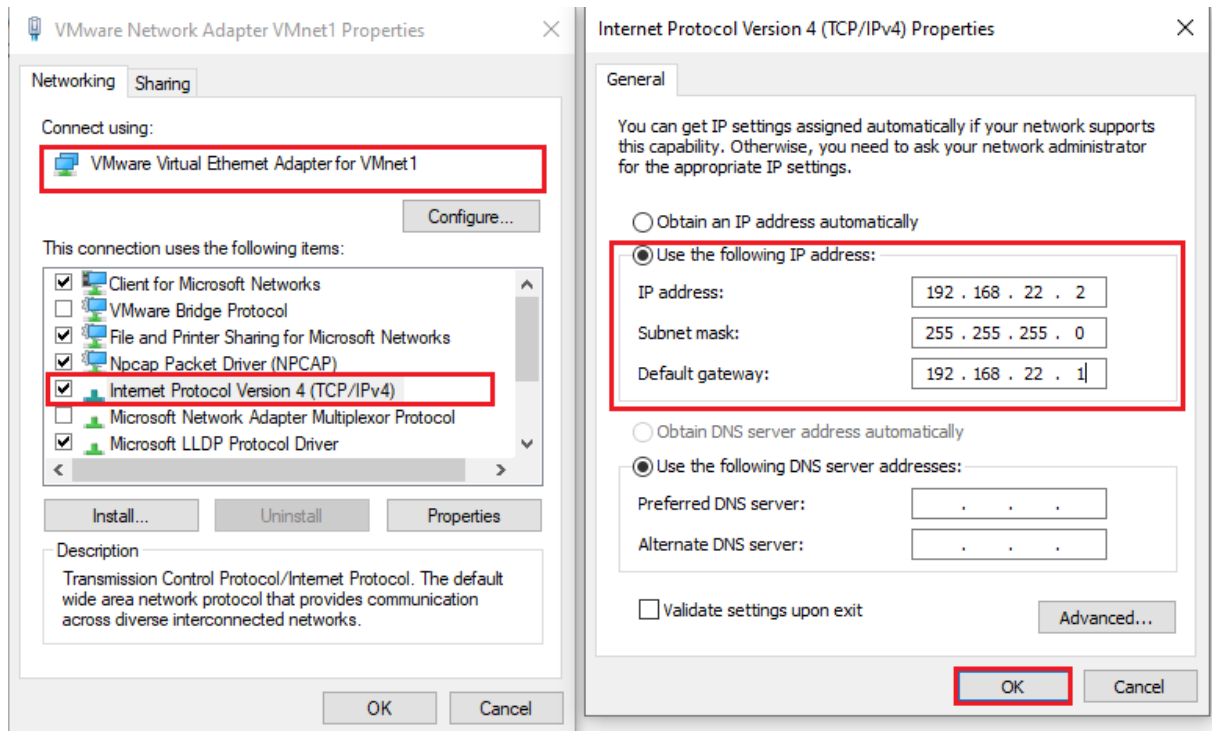
28. Jika sudah dilakukan selanjutnya *restart* apache2 dengan perintah **"systemctl restart apache2"**

```

root@tkj:~# systemctl restart apache2

```

29. Setelah itu lakukan uji dengan masukkan IP address di *web browser*. Namun, sebelum itu silahkan atur alamat IP pada Vmnet *Host-only* di *Control Panel Network* di komputer menjadi satu jaringan dengan alamat IP yang dikonfigurasi pada virtual server ini. Atau dapat dengan menerapkan fitur dhcp server di Ubuntu dengan konfigurasi dhcp. Pada hal ini menggunakan alamat IP static dengan konfigurasi seperti ini.



30. Langkah berikutnya adalah uji koneksi antar server *virtual* dengan komputer yang digunakan dengan *utili ping IP Server* menggunakan **command prompt**.

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.19045.5247]
(c) Microsoft Corporation. All rights reserved.

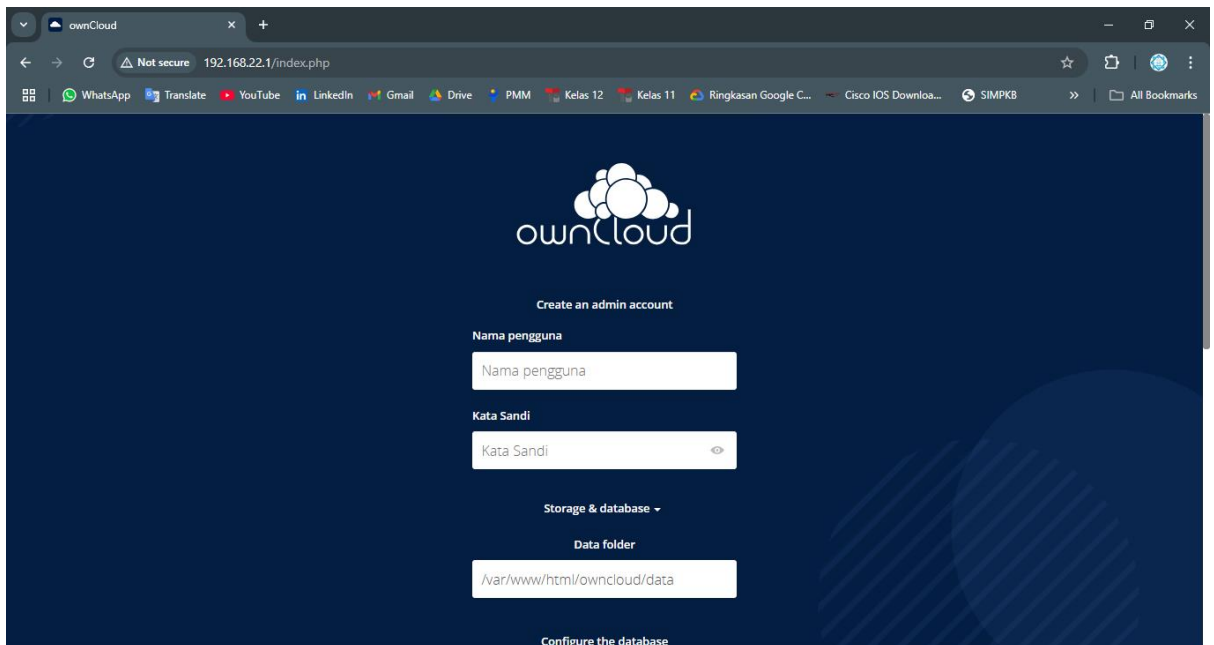
C:\Users\Khairunnisa Alwita>ping 192.168.22.1

Pinging 192.168.22.1 with 32 bytes of data:
Reply from 192.168.22.1: bytes=32 time<1ms TTL=64
Reply from 192.168.22.1: bytes=32 time<1ms TTL=64
Reply from 192.168.22.1: bytes=32 time<1ms TTL=64
Reply from 192.168.22.1: bytes=32 time<1ms TTL=64

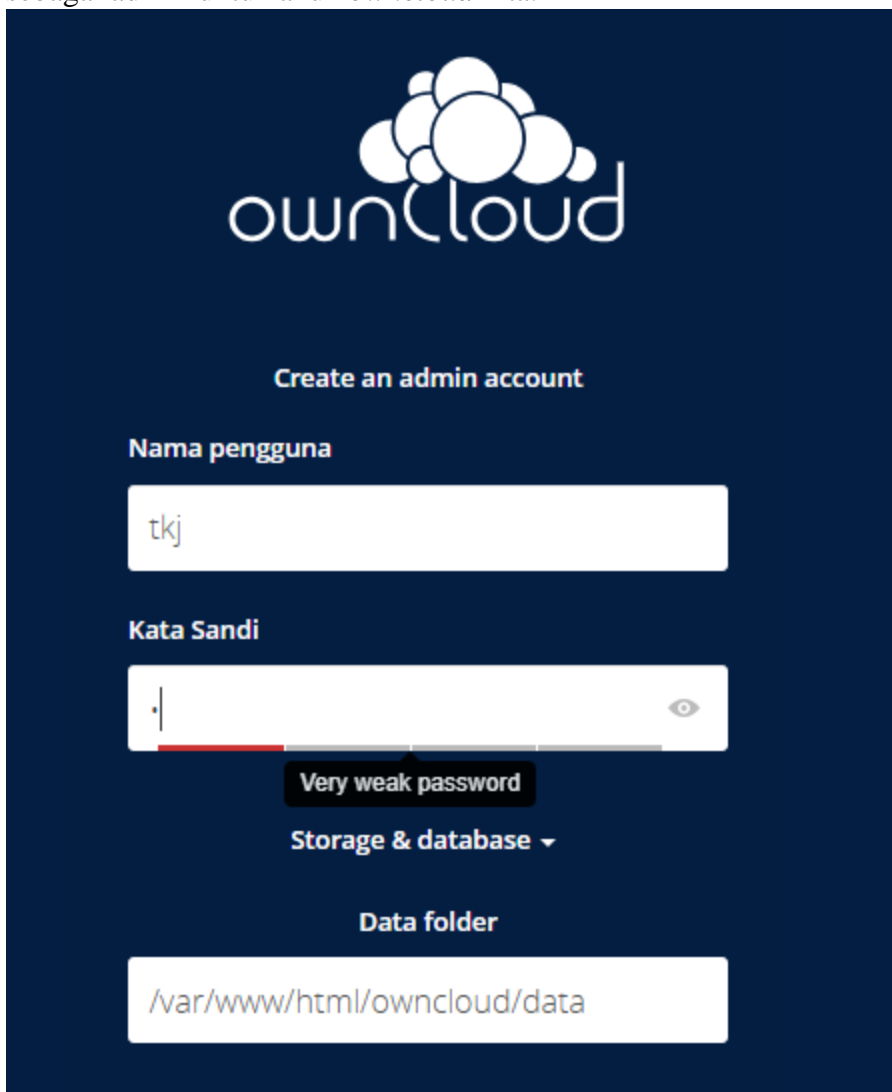
Ping statistics for 192.168.22.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Khairunnisa Alwita>
```

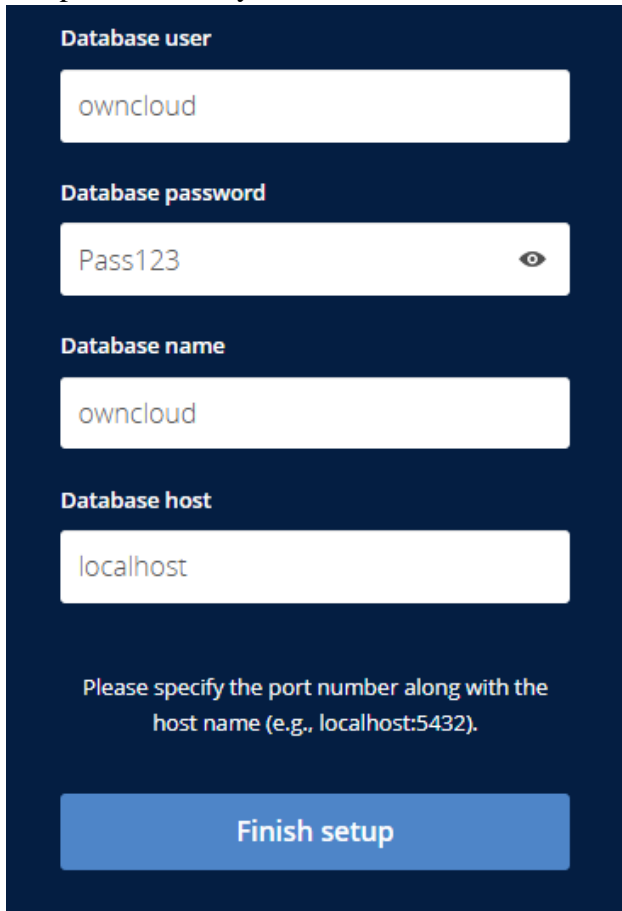
31. Setelah itu silahkan akses alamat IP server tersebut di *web browser*



32. Setelah itu silahkan isi nama pengguna dan kata sandi baru yang akan digunakan sebagai admin untuk akun *owncloud* kita.

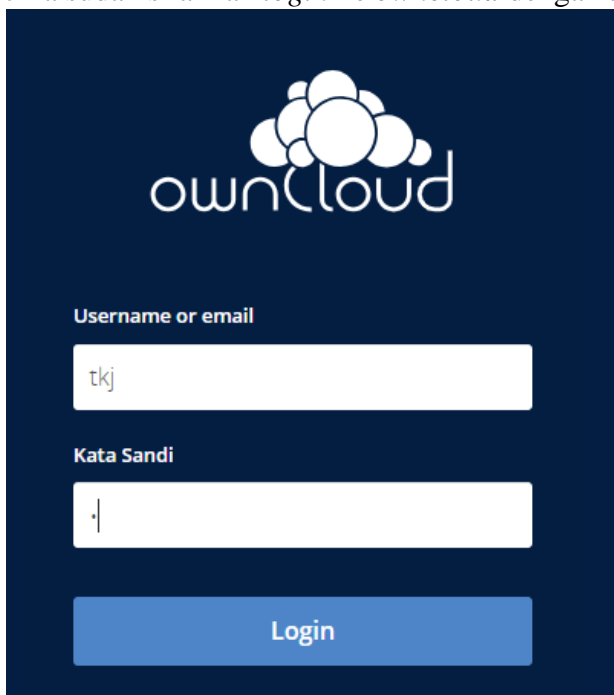


33. Kemudian silahkan *scroll* kebagian bawah lalu isikan untuk nama *database*, nama user *database* serta kata sandi *database* sesuai dengan yang sudah dibuat pada mysql di tahapan sebelumnya. Jika sudah silahkan klik **Finish Setup**.



The screenshot shows the 'Finish Setup' screen for OwnCloud database configuration. It has a dark blue background with white text and input fields. The fields are labeled 'Database user', 'Database password', 'Database name', and 'Database host'. The 'Database user' field contains 'owncloud'. The 'Database password' field contains 'Pass123' and has an eye icon to toggle visibility. The 'Database name' field contains 'owncloud'. The 'Database host' field contains 'localhost'. Below these fields is a note: 'Please specify the port number along with the host name (e.g., localhost:5432)'. At the bottom is a blue button labeled 'Finish setup'.

34. Jika sudah silahkan *login* ke *owncloud* dengan akun yang sudah dibuat sebelumnya.



The screenshot shows the OwnCloud login screen. It has a dark blue background with the OwnCloud logo at the top. Below the logo are two input fields: 'Username or email' and 'Kata Sandi'. The 'Username or email' field contains 'tkj'. The 'Kata Sandi' field contains a single dot. At the bottom is a blue button labeled 'Login'.

35. Hasil akhirnya adalah tampilan *dashboard* dari *owncloud* seperti dibawah ini. Jika sudah seperti ini maka instalasi dan konfigurasi kamu sudah berhasil. Silahkan lanjutkan untuk tahapan lain seperti pengaturan *user* dan penyimpanan (*storage*) untuk bisa digunakan oleh pengguna lain.

