

comment-installer-syncthing-sur-ubuntu-22-04

Syncthing is a secure and open source file synchronisation program for multiple platforms, such as Windows, macOS, Linux, Android, Solaris, Darwin and BSD. It is a decentralised and peer-to-peer (P2P) file synchronisation tool that allows you to synchronise files between devices on a local network or between remote devices over the internet.

In this tutorial, I will show you how to install Syncthing on a Ubuntu 22.04 servers. You will then learn how to establish a connection between Syncthing instances, set up the synchronisation of directories and check the synchronisation.

Prerequisites

To get started with this tutorial, you must have the following:

- Two Ubuntu 22.04 Machines - In this example, we'll use two Ubuntu servers **server1** and **server2**.
- A non-root user with administrator privileges.

Adding Syncthing Repository

Syncthing can be installed in multiple ways, you can install it manually via binary file, or install it via APT for Ubuntu. In this example, you will install syncthing via APT by adding the official repository to both Ubuntu machines.

To start, execute the following command to install some basic packages to your Ubuntu system.

```
sudo apt install gnupg2 curl apt-transport-https -y
```

```
root@server1:~#
root@server1:~# sudo apt install gnupg2 curl apt-transport-https -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
curl is already the newest version (7.81.0-1ubuntu1.14).
curl set to manually installed.
The following NEW packages will be installed:
  apt-transport-https gnupg2
0 upgraded, 2 newly installed, 0 to remove and 35 not upgraded.
Need to get 7058 B of archives.
After this operation, 222 kB of additional disk space will be used.
0% [Waiting for headers]
```

Now, add the syncthing GPG key and repository to your both Ubuntu systems. In this case, you will install the latest release of Syncthing via the APT repository.

```
curl -fsSL https://syncthing.net/release-key.txt | \
sudo gpg --dearmor -o /etc/apt/trusted.gpg.d/syncthing.gpg

echo "deb https://apt.syncthing.net/ syncthing release" | \
sudo tee /etc/apt/sources.list.d/syncthing.list
```

```
root@server1:~#
root@server1:~# curl -fsSL https://syncthing.net/release-key.txt | \
sudo gpg --dearmor -o /etc/apt/trusted.gpg.d/syncthing.gpg
root@server1:~#
root@server1:~# echo "deb https://apt.syncthing.net/ syncthing release" | \
sudo tee /etc/apt/sources.list.d/syncthing.list
deb https://apt.syncthing.net/ syncthing release
root@server1:~#
```

Then, update and refresh your Ubuntu repository using the following command.

```
sudo apt update
```

When you see the syncthing repository is available, you're ready to go.

```
root@server1:~#  
root@server1:~# sudo apt update  
Hit:1 https://download.mono-project.com/repo/ubuntu stable-focal InRelease  
Get:2 https://apt.syncthing.net syncthing InRelease [15.1 kB]  
Hit:3 http://archive.ubuntu.com/ubuntu jammy InRelease  
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]  
Get:5 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]  
Get:6 https://apt.syncthing.net syncthing/release amd64 Packages [1588 B]  
Hit:7 http://archive.ubuntu.com/ubuntu jammy-backports InRelease  
Fetched 246 kB in 3s (82.0 kB/s)  
Reading package lists... 38%
```

Installing Syncthing

Now that you've added a syncthing repository to both Ubuntu machines, the next step you will install it on both machines and configure it to run as a systemd service. Also, ensure you have a non-root user on both machines to run syncthing.

To install syncthing, run the following apt command. Type y and press ENTER to proceed with the installation.

```
sudo apt install syncthing
```

```
root@server1:~#  
root@server1:~# sudo apt install syncthing  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following NEW packages will be installed:  
  syncthing  
0 upgraded, 1 newly installed, 0 to remove and 35 not upgraded.  
Need to get 10.4 MB of archives.  
After this operation, 26.1 MB of additional disk space will be used.
```

After the installation is complete, verify the syncthing version like this:

```
syncthing --version  
syncthing -h
```

In the following output, you should see syncthing **1.26** is installed.

```
root@server1:~#  
root@server1:~# syncthing --version  
syncthing v1.26.1 "Gold Grasshopper" (go1.21.4 linux-amd64) debian@github.syncthing.net  
root@server1:~#  
root@server1:~# syncthing -h  
Usage: syncthing <command>  
  
Flags:  
  -h, --help    Show context-sensitive help.  
  
Commands:  
  serve  
    Run Syncthing  
  
  generate  
    Generate key and config, then exit  
  
  decrypt <path>  
    Decrypt or verify an encrypted folder  
  
  cli  
    Command line interface for Syncthing  
  
Run "syncthing <command> --help" for more information on a command.  
root@server1:~#
```

Now that you've installed syncthing, you should start it as per-user. In this example, we'll be using user alice on the server1, and user bob on the server2.

Log in to your user using the command below.

```
su - alice
```

Then, start and enable the syncthing service for user alice using the command like this:

```
sudo systemctl start syncthing@alice.service  
sudo systemctl enable syncthing@alice.service
```

Be sure to input your password to get the root privileges when asked:

```
root@server1:~#  
root@server1:~# su - alice  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
alice@server1:~$ sudo systemctl start syncthing@alice.service  
[sudo] password for alice:  
alice@server1:~$  
alice@server1:~$ sudo systemctl enable syncthing@alice.service  
Created symlink /etc/systemd/system/multi-user.target.wants/syncthing@alice.se  
alice@server1:~$  
alice@server1:~$
```

Now verify the syncthing service to ensure that the service is running via the command below.

```
sudo systemctl status syncthing@alice.service
```

You should see the *syncthing* service for user **alice** is running. Also, ensure the syncthing service on both your servers is running.

```
alice@server1:~$  
alice@server1:~$ sudo systemctl status syncthing@alice.service  
● syncthing@alice.service - Syncthing - Open Source Continuous File Synchronization for alice  
   Loaded: loaded (/lib/systemd/system/syncthing@.service; enabled; vendor preset: enabled)  
   Active: active (running) since  
     Docs: man:syncthing(1)  
  Main PID: 6344 (syncthing)  
    Tasks: 16 (limit: 2310)  
  Memory: 17.7M  
     CPU: 2.063s  
   CGroup: /system.slice/system-syncthing.slice/syncthing@alice.service  
           └─6344 /usr/bin/syncthing serve --no-browser --no-restart --logflags=0  
           └─6352 /usr/bin/syncthing serve --no-browser --no-restart --logflags=0
```

Setting Up Firewall

In this guide, you will enable the firewall on both Ubuntu machines. On Ubuntu, you can use UFW (Uncomplicated Firewall), and by default, syncthing also provides a UFW application profile, so you can easily enable it.

Run the command below to add the **syncthing** and **syncthing-gui** profiles to UFW.

```
sudo ufw allow syncthing  
sudo ufw allow syncthing-gui
```

Now add the OpenSSH profile, then start and enable UFW using the below command. Type **y** to confirm with the operation.

```
sudo ufw allow OpenSSH  
sudo ufw enable
```

Once complete, you will see the following output.


```

alice@server1:~$
alice@server1:~$ sudo ufw allow syncthing
Rules updated
Rules updated (v6)
alice@server1:~$ sudo ufw allow syncthing-gui
Rules updated
Rules updated (v6)
alice@server1:~$
alice@server1:~$ sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
alice@server1:~$

```

Next, verify the list-enabled rules and profiles on UFW using the command below.

```
sudo ufw status verbose
```

Ensure both syncthing and syncthing-gui are added to UFW.

```

alice@server1:~$
alice@server1:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To Action From
--
22000 (syncthing) ALLOW IN Anywhere
21027/udp (syncthing) ALLOW IN Anywhere
8384/tcp (syncthing-gui) ALLOW IN Anywhere
22/tcp (OpenSSH) ALLOW IN Anywhere
22000 (syncthing (v6)) ALLOW IN Anywhere (v6)
21027/udp (syncthing (v6)) ALLOW IN Anywhere (v6)
8384/tcp (syncthing-gui (v6)) ALLOW IN Anywhere (v6)
22/tcp (OpenSSH (v6)) ALLOW IN Anywhere (v6)

```

Initializing Syncthing Instances

Now that you've installed Syncthing and configured UFW, you're ready to start initializing your installation. In this step, you will set up syncthing to run on a local IP address with HTTPS enabled, and you will set up user authentication for syncthing.

By default, syncthing is running on localhost. If you want to run syncthing on a local IP address, modify the syncthing configuration `~/.config/syncthing/conf.xml` file.

Open the `~/.config/syncthing/conf.xml` file using the nano editor command below.

```
nano ~/.config/syncthing/conf.xml file
```

Within the '`<gui ..>`' section, change the '`tls`' value to '`true`' and change the default localhost to your local IP address.

```

<gui enabled="true" tls="true" debugging="false" sendBasicAuthPrompt="false">
  <address>192.168.5.30:8384</address>
  <apikey>GENERATED-API-KEY</apikey>
  <theme>default</theme>
</gui>

```

Save and close the file when finished.

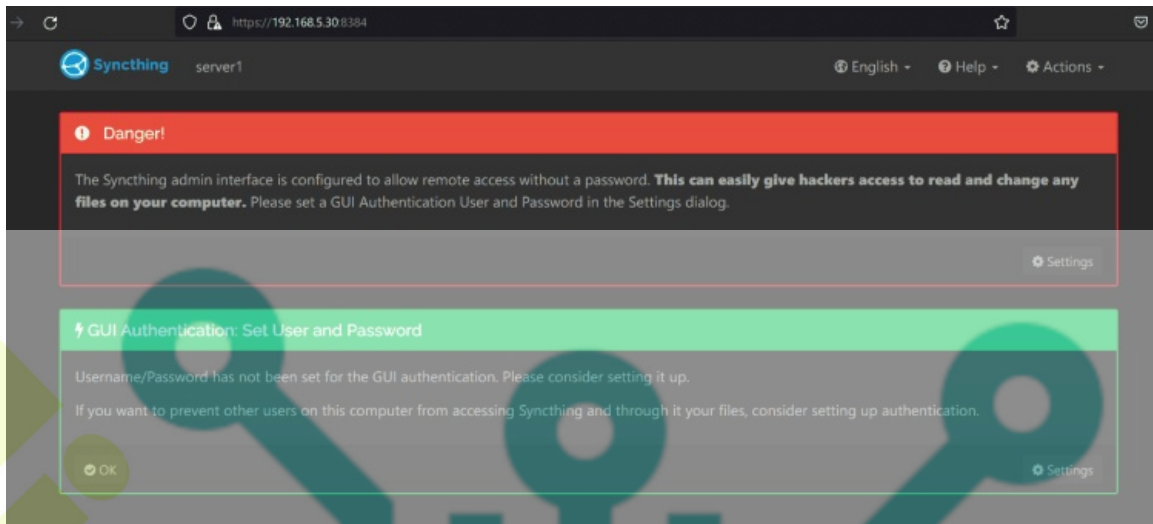
Now run the following command to restart the syncthing service.

```
sudo systemctl restart syncthing@alice.service
```

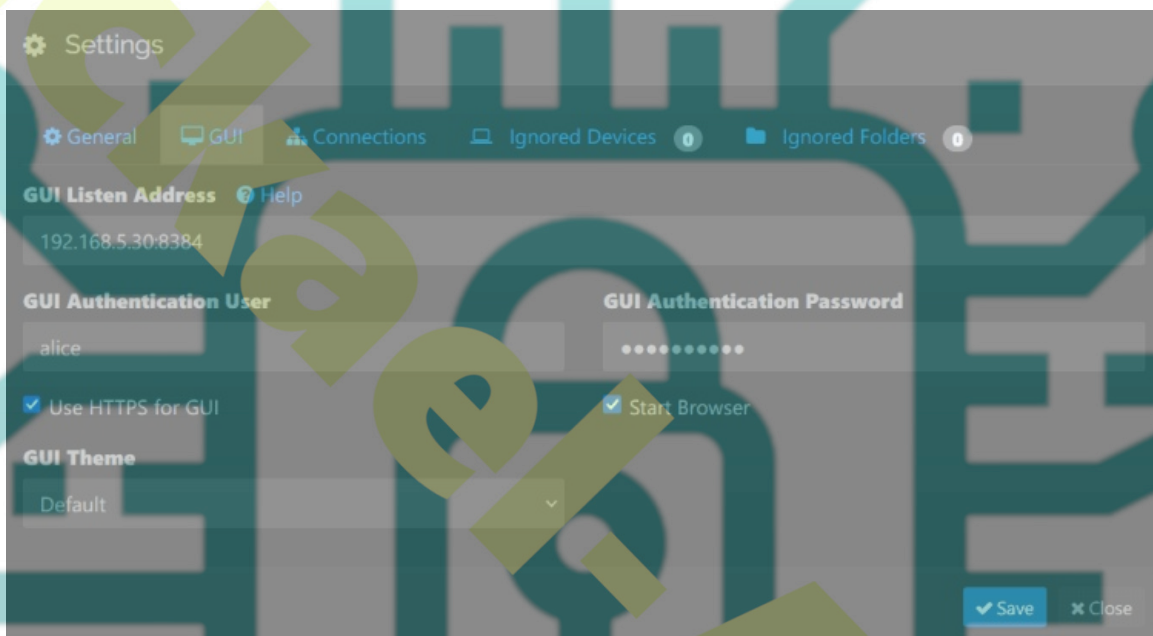
Open your web browser and visit syncthing via server IP address and port 8384 (i.e. <https://192.168.5.30:8384/>).

If your installation is successful, you should get the following page:

Click **Settings** to set up syncthing.



Within the GUI tab, change your user and password, then click **Save**.

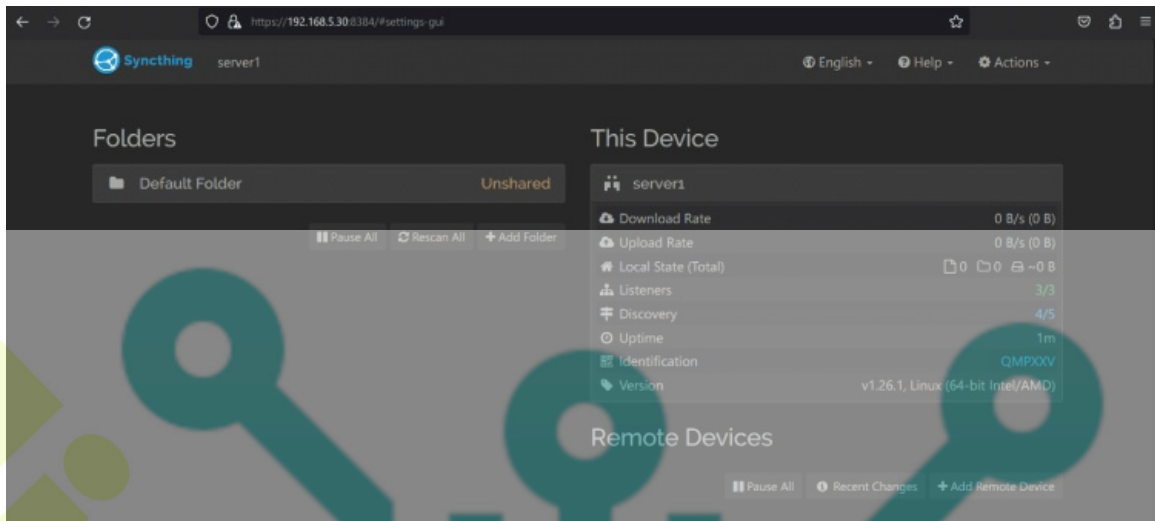


After the new password is configured, you will be prompted with syncthing authentication.

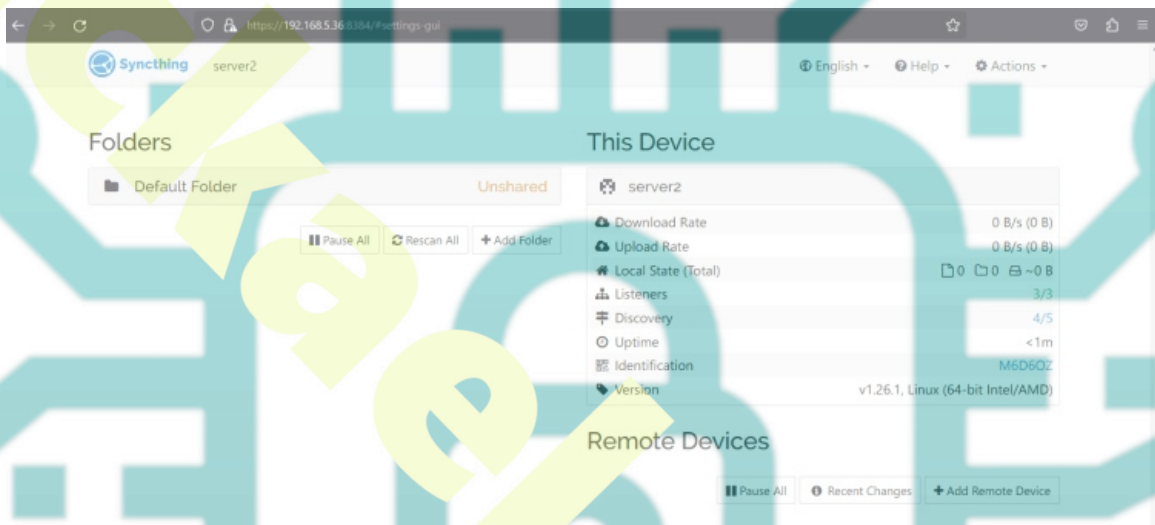
Input your username and password, then click **Log In**.



If you have the proper user and password, you should get the syncthing dashboard like the following:



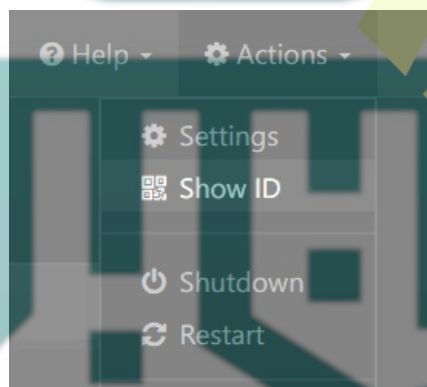
Below is the synthing dashboard of **server2** after the user and password are configured.



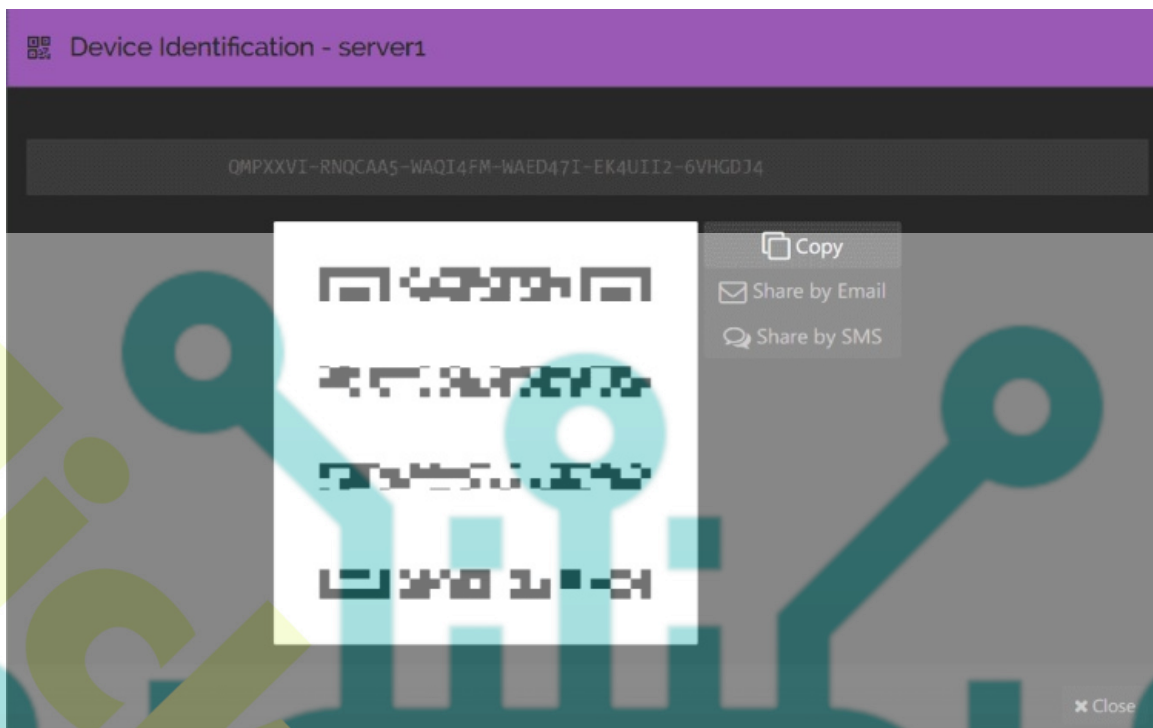
Connecting Two Synthing Instances

After configuring Synthing on both Ubuntu machines, you will connect, so you can sync files between those machines. To do that, you must verify both machines via GUI.

On the **server1**, click the **Action** button at the top right, then select **Show ID**.

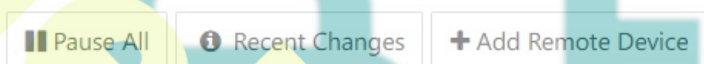


Copy the device ID of the **server1**. The synthing device ID is generated automatically during stat. Furthermore, you can also use QRcode to connect between synthing instances, for example between computers and Android phones.



Move to **server2**, and click **Add Remote Device** to add a new synching instance.

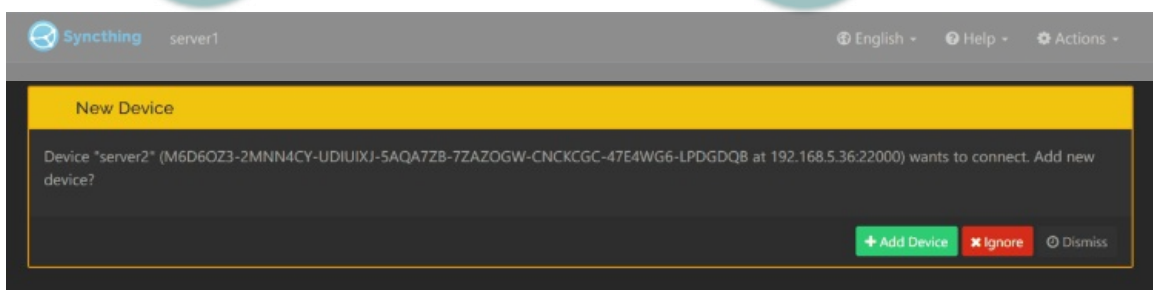
Remote Devices



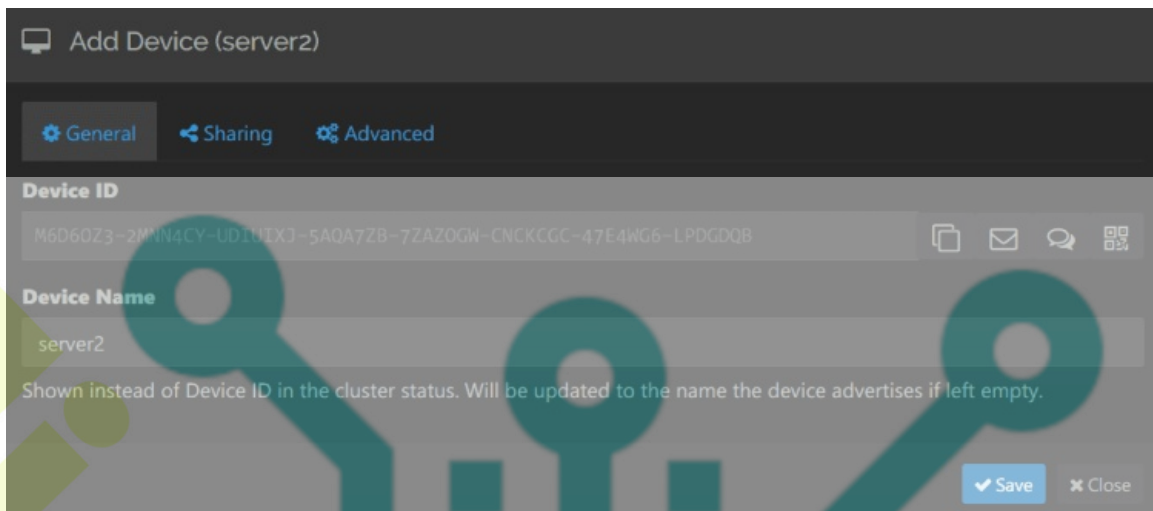
Paste the device ID of **server1**, input the name as **server1**, and then click **Save** to confirm.



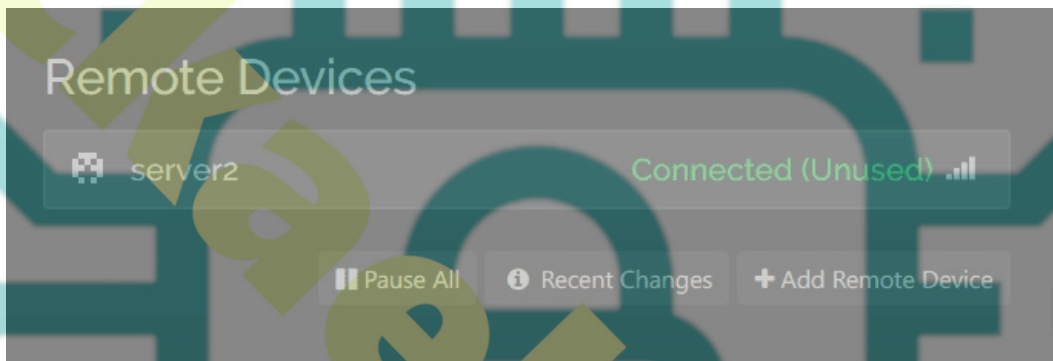
Next, move back to the **server1** dashboard and the information about **server2** wants to connect to. Click **Add Device** to add the **server1**.



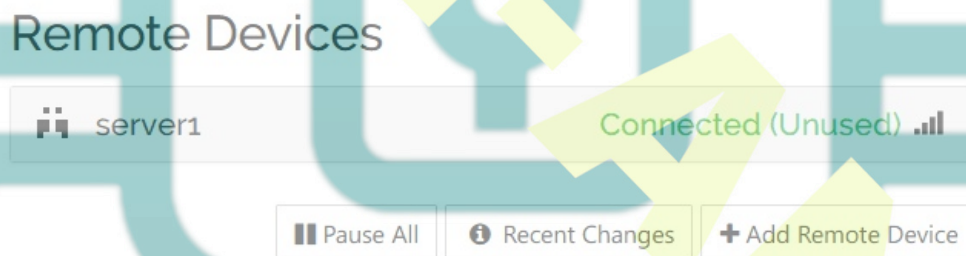
Now, the new pop-up will come up, verify the device ID of **server2**, then click **Save**.



Lastly, go to the synching dashboard and move to the **Remote Devices** section. If the process is successful, you should see the **server2** with status **connected**.



On the other hand, when you check on the **server2**, you should see that **server1** is **connected**.

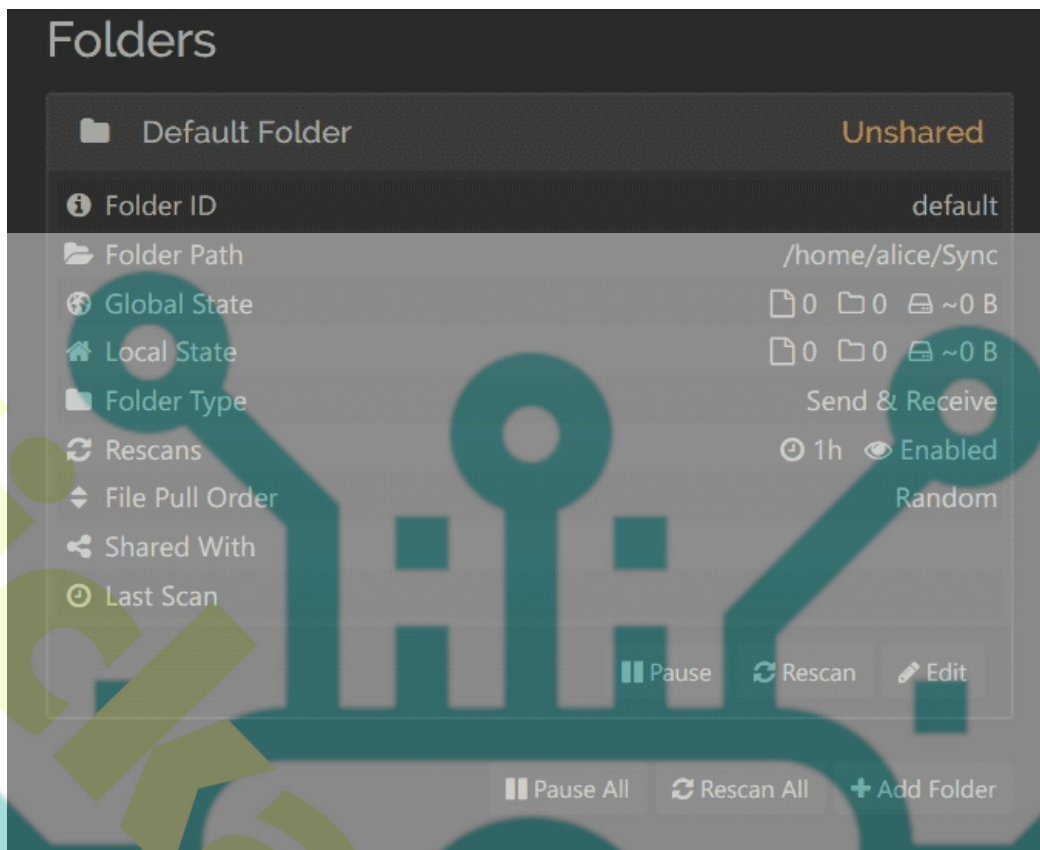


Synchronizing Directory with Synching

At this point, both server1 and server2 are connected via synching. Now, you will learn how to share a directory/folder and sync files between synching instances.

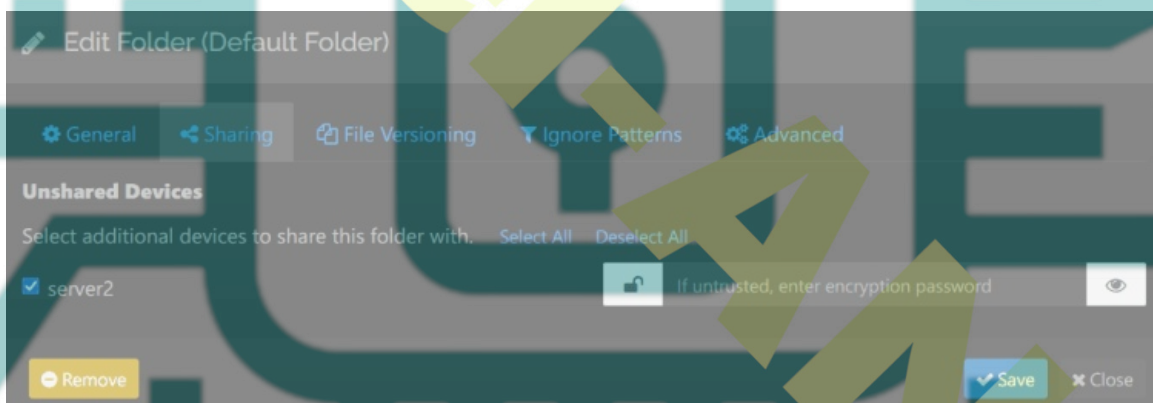
On the **server1** dashboard, take a look at the **Folders** section, and you will find the **Default Folder** with path directory `/home/alice/Sync` and status **Unshared**.

Click **Edit** to share the **Default Folder**.

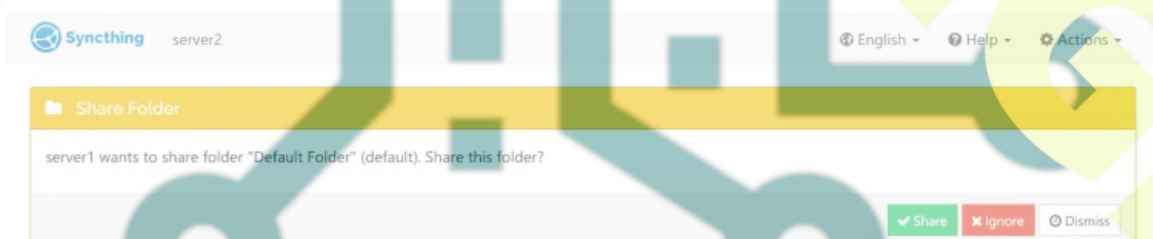


- Move to the **Sharing** tab, and you will see the list of connected synching instances. Select **server2** to share the Default Folder with the synching instance server2.
- Move to the **File Versioning** tab, select the **File Versioning** method that you want to use, and then input how much versioning you want to keep.

Click **Save** to apply your modification.

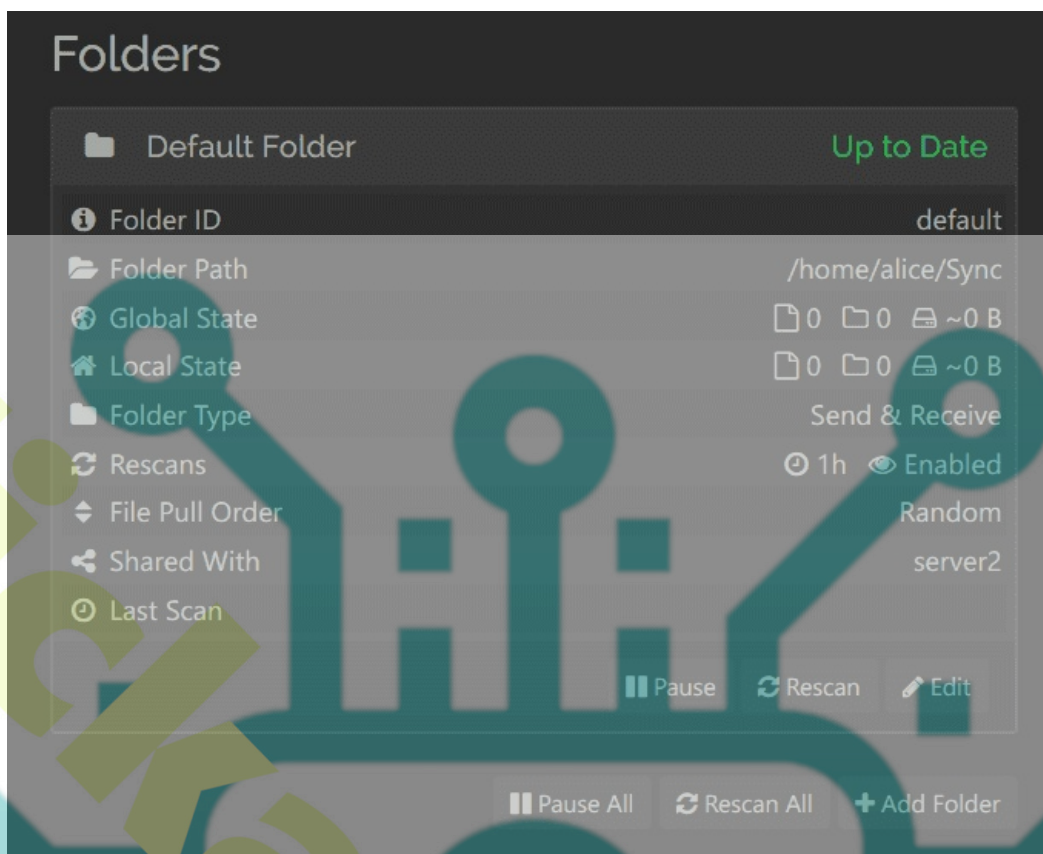


Lastly, move to the **server2** dashboard, and you will get the notification message that **server1** wants to share the Default Folder. Click **Share** to confirm.

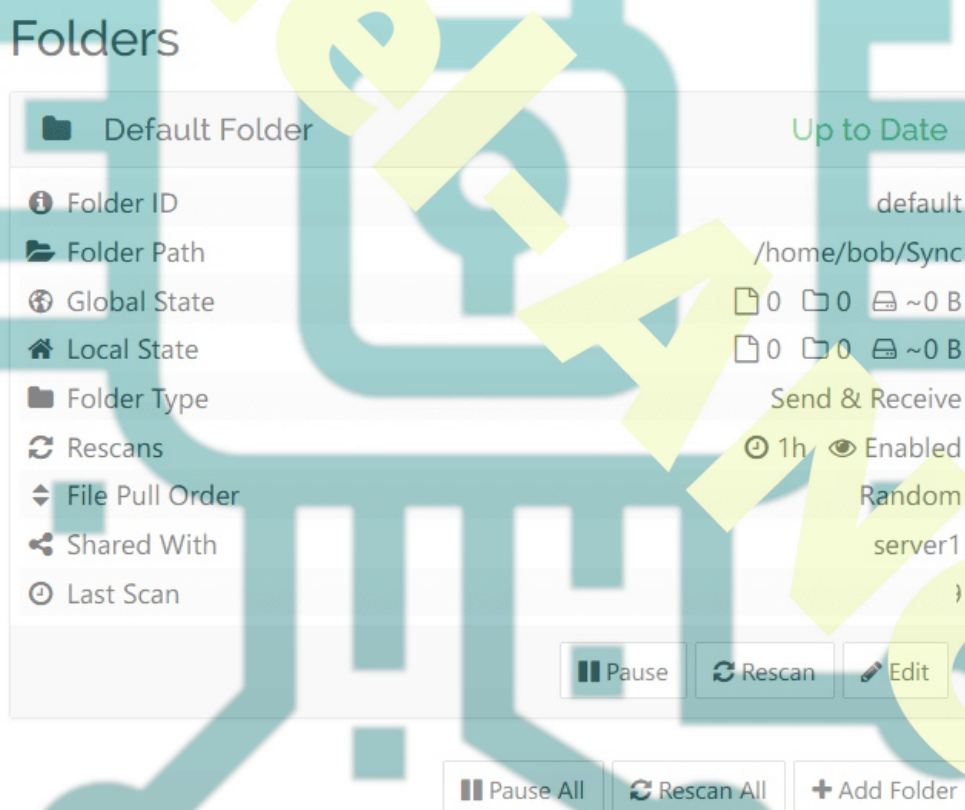


Once the process is finished, you should see the **Default Folder** is shared.

On the **server1 Default Folder** section, you should see the **Shared With** section with the value **server2**.



On the **server2**, you should see the **Default Folder** and **Shared With** section to **server1**.



File Synchronization between Syncthing Instances

Now that you've configured the shared directory via Syncthing, let's verify it by creating new files from server1 and verifying the list files on server2 to ensure the synchronization is working.

On the **server1**, go to the Sync directory and create new files using the following command.

```
cd ~/Sync/  
touch {1..20}.txt
```

```
alice@server1:~$
alice@server1:~$ cd ~/Sync/
alice@server1:~/Sync$
alice@server1:~/Sync$ touch {1..20}.txt
alice@server1:~/Sync$
alice@server1:~/Sync$ ls -ah
.  .stfolder  10.txt  12.txt  14.txt  16.txt  18.txt  2.txt  3.txt  5.txt  7.txt  9.txt
.. 1.txt     11.txt  13.txt  15.txt  17.txt  19.txt  20.txt 4.txt  6.txt  8.txt
alice@server1:~/Sync$
alice@server1:~/Sync$
```

Move to **server2**, go to the Sync directory, and check the list of files in that directory.

```
cd ~/Sync/
ls -ah
```

If you see multiple .txt files on the Sync directory, you've accomplished the synching installation without any problem.

```
bob@server2:~$
bob@server2:~$ cd ~/Sync/
bob@server2:~/Sync$
bob@server2:~/Sync$ ls -ah
.  .stfolder  1.txt  10.txt  11.txt  12.txt  13.txt  14.txt  15.txt  16.txt  17.txt  18.txt  19.txt  2.txt  20.txt  3.txt  4.txt  5.txt  6.txt  7.txt  8.txt  9.txt
bob@server2:~/Sync$
bob@server2:~/Sync$
```

From here, you can create and share a new directory between synching instances. , you can also add new instances.

Conclusion

Congratulations! You've completed the installation of Synching on Ubuntu 22.04 servers. You've also learned how to secure synching with UFW (Uncomplicated Firewall), connect between synching instances, set up shared folders, and verify the synchronization between shared directories.

Furthermore, you can new devices to your current synching installation and set up shared folders, so you can access files from multiple devices.
