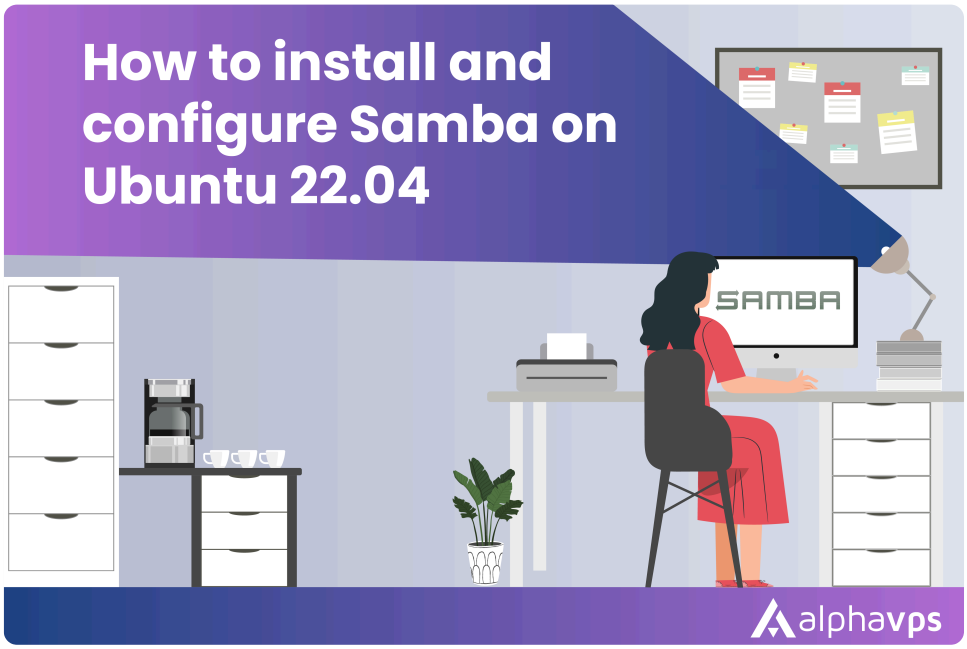
How to install and configure Samba on Ubuntu 22.04

Vladislav Peev
May 31, 2023 • 3 min read



Samba is an open-source software that enables file sharing between Linux-based systems and Windows systems. It allows Windows clients to access shared resources on Linux/Unix servers, and vice versa.

Samba implements the SMB/CIFS (Server Message Block/Common Internet File System) protocol, which is the standard protocol used by Windows for file and printer sharing. By using Samba, Linux/Unix systems can act as file servers in Windows-based networks, providing services such as file sharing.

For this tutorial, we are going to use an Ubuntu 22.04 VPS from AlphaVPS with a created sudo user.

Installing Samba

The installation process is quite simple. Just run the following commands:

```
sudo apt update
sudo apt install samba
```

We can confirm our installation by running:

```
whereis samba
```

Our output should look like this:

```
samba: /usr/sbin/samba /usr/lib/x86_64-linux-gnu/samba /etc/samba /usr/share/samba /usr/sha
```

Configuring Samba

First up, we will need to create the directory which we want to share via Samba.

We are able to create the directory wherever we wish. A common practice is to create the directory in our user's home directory.

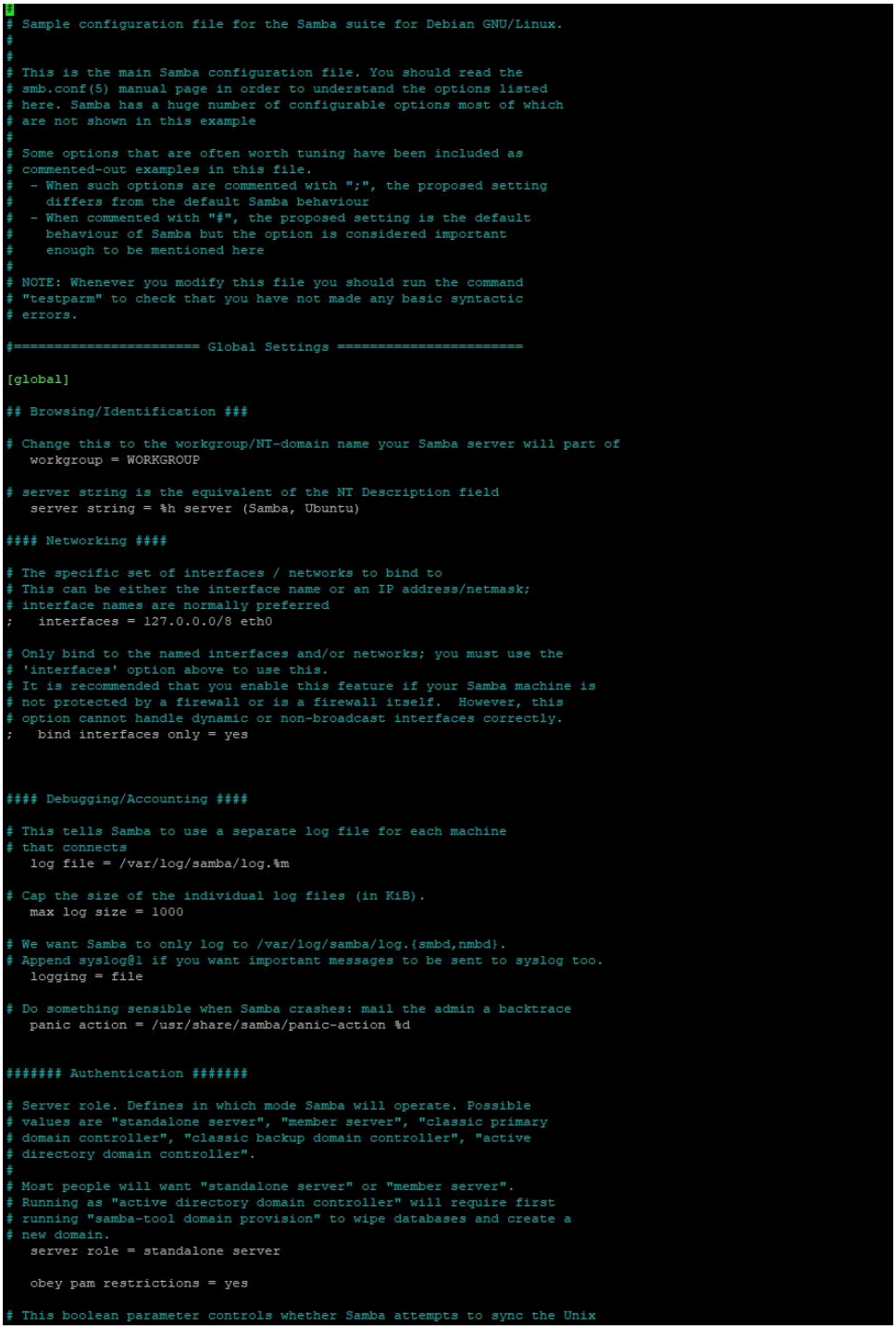
To do so, run:

```
sudo mkdir /home/sambashare/
```

Samba's configuration file is located at `/etc/samba/smb.conf`. To add the new directory as a share, we will need to edit the file.
We can do so by running:

```
sudo nano /etc/samba/smb.conf
```

The file should look like this:



```
# Sample configuration file for the Samba suite for Debian GNU/Linux.
#
#
# This is the main Samba configuration file. You should read the
# smb.conf(5) manual page in order to understand the options listed
# here. Samba has a huge number of configurable options most of which
# are not shown in this example
#
# Some options that are often worth tuning have been included as
# commented-out examples in this file.
#  - When such options are commented with ";", the proposed setting
#    differs from the default Samba behaviour
#  - When commented with "#", the proposed setting is the default
#    behaviour of Samba but the option is considered important
#    enough to be mentioned here
#
# NOTE: Whenever you modify this file you should run the command
# "testparm" to check that you have not made any basic syntactic
# errors.

#======================= Global Settings =======================

[global]

## Browsing/Identification ###

# Change this to the workgroup/NT-domain name your Samba server will part of
   workgroup = WORKGROUP

# server string is the equivalent of the NT Description field
   server string = %h server (Samba, Ubuntu)

#### Networking ####

# The specific set of interfaces / networks to bind to
# This can be either the interface name or an IP address/netmask;
# interface names are normally preferred
;   interfaces = 127.0.0.0/8 eth0

# Only bind to the named interfaces and/or networks; you must use the
# 'interfaces' option above to use this.
# It is recommended that you enable this feature if your Samba machine is
# not protected by a firewall or is a firewall itself.  However, this
# option cannot handle dynamic or non-broadcast interfaces correctly.
;   bind interfaces only = yes



#### Debugging/Accounting ####

# This tells Samba to use a separate log file for each machine
# that connects
   log file = /var/log/samba/log.%m

# Cap the size of the individual log files (in KiB).
   max log size = 1000

# We want Samba to only log to /var/log/samba/log.{smbd,nmbd}.
# Append syslog@1 if you want important messages to be sent to syslog too.
   logging = file

# Do something sensible when Samba crashes: mail the admin a backtrace
   panic action = /usr/share/samba/panic-action %d


####### Authentication #######

# Server role. Defines in which mode Samba will operate. Possible
# values are "standalone server", "member server", "classic primary
# domain controller", "classic backup domain controller", "active
# directory domain controller".
#
# Most people will want "standalone server" or "member server".
# Running as "active directory domain controller" will require first
# running "samba-tool domain provision" to wipe databases and create a
# new domain.
   server role = standalone server

   obey pam restrictions = yes

# This boolean parameter controls whether Samba attempts to sync the Unix
```

We will need to add the following lines to the bottom of our configuration file.

```
[sambashare]
    comment = Samba on Ubuntu
    path = `path-to-sambashare-directory`
    read only = no
    browsable = yes
```

An example pathname would be `home/sambashare`

To apply the changes to our configuration file we need to run:

```
sudo service smbd restart
```

If you have an active UFW, you can update your firewall rules by running:

```
sudo ufw allow samba
```

Setting up a User account

We need to set up a Samba password for our user account. We can do so by running:

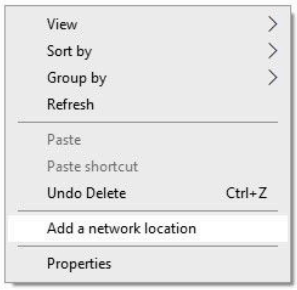
```
sudo smbpasswd -a username
```

Provide the required username and you will be prompted to set the user's Samba password.

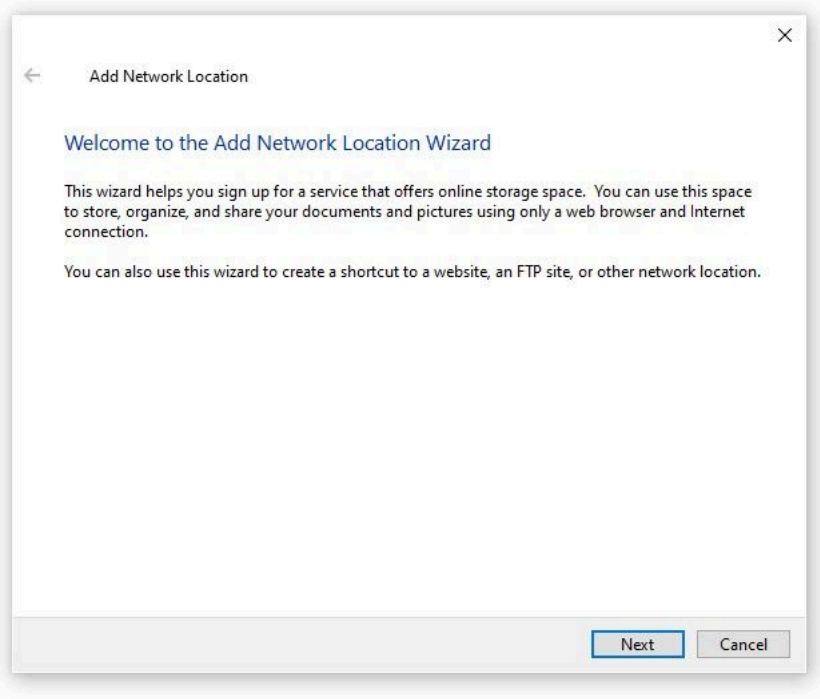
Connecting to Share

Windows

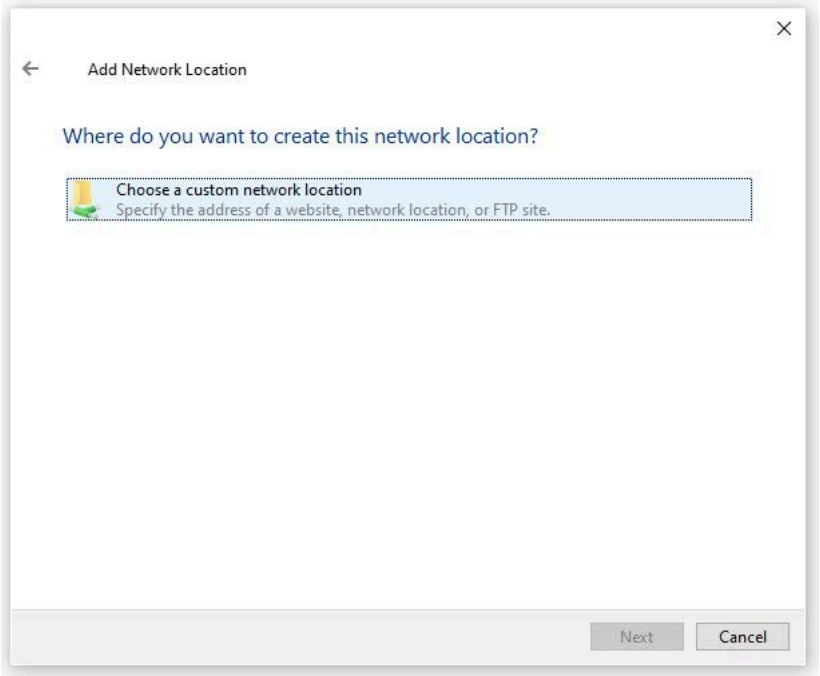
You can easily navigate to **This PC**. Right-click in the folder and select the `Add a network location` option.



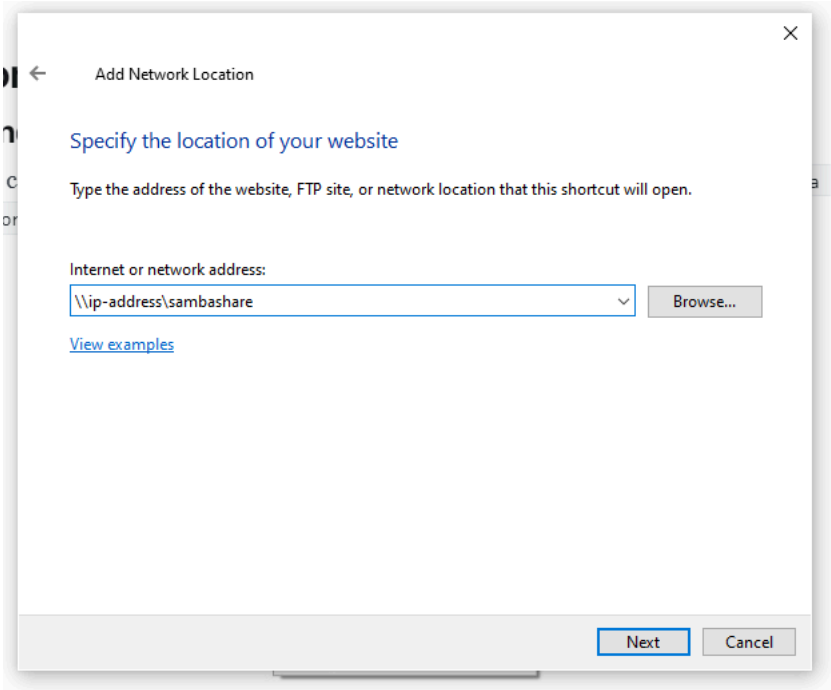
You will be presented with a prompt. Click on `Next` to proceed further.



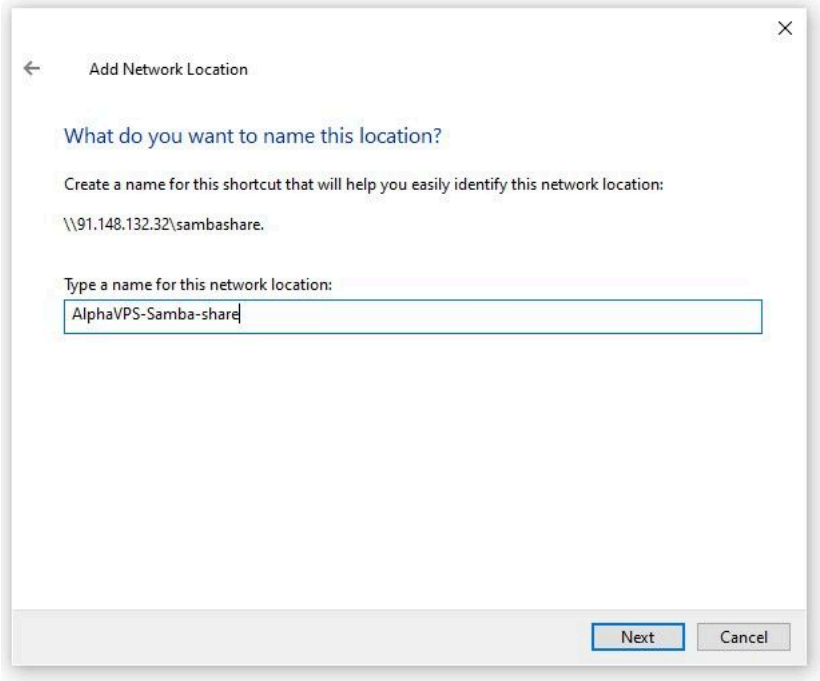
Click on the first and only available option. Proceed further by clicking `Next` again.



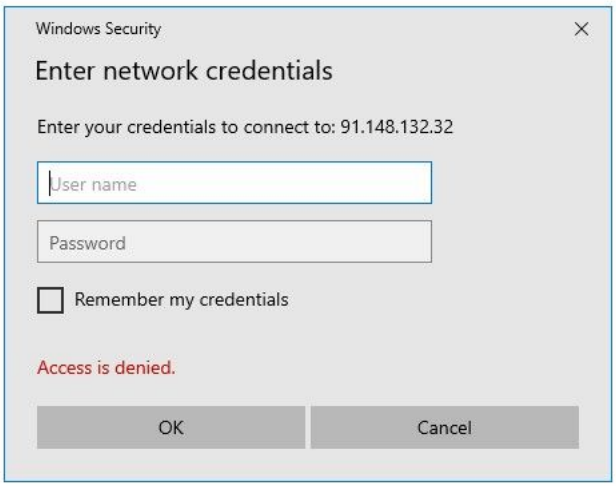
Specify the location of the Samba share by pointing out the IP address and shared directory name.



Once the connection is established, you will be able to name your Samba share.



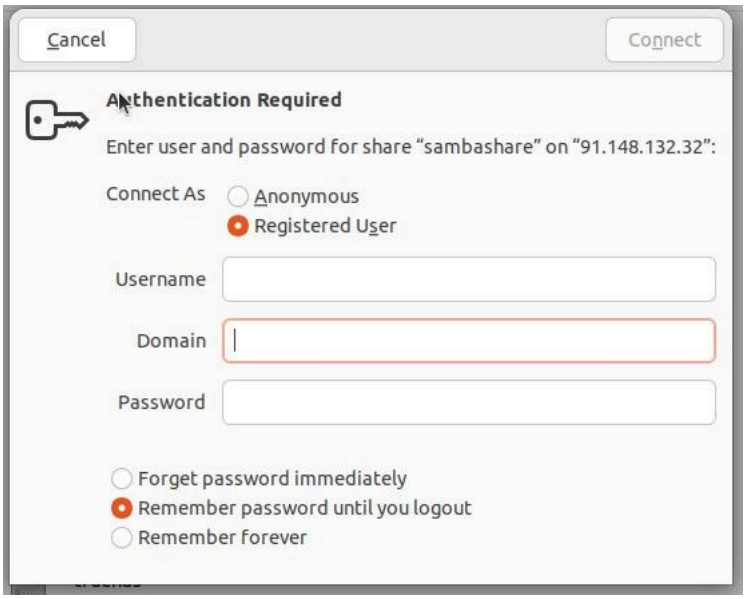
Confirm and proceed.
The only thing that is left is to provide your user credentials. Once you try to access the Samba share, you will be immediately prompted to do this.



Ubuntu

Open up the file manager and click the `Connect to Server` field and enter `smb://ip-address/sambashare`, click on `Connect`.

Provide your credentials in the same way as in Windows and proceed.



Vladislav Peev
May 31, 2023 • 3 min read

AlphaVPS is a leading hosting provider, specializing in low-cost servers and cheap VPS. Our award winning support, amazing network and high performance servers are loved by thousands of customers.

Get in touch

Support:
support@alphavps.com
Sales:
sales@alphavps.com
NOC: noc@dagroup.bg
Abuse:
abuse@alphavps.com

Office in Sofia:

8 Asen Yordanov blvd.
Sofia, Bulgaria

Navigation

Knowledgebase
Blog
Clients area
About
Service Status

Legal

Terms of Service
Privacy Policy

Services

Dedicated servers in Europe
AMD Ryzen VPS
High-Performance VPS
Cheap VPS
Powerful KVM VPS
Storage VPS
Colocation