

# Samba

From ArchWiki

**Samba** is a re-implementation of the SMB/CIFS networking protocol, it facilitates file and printer sharing among Linux and Windows systems as an alternative to NFS. Some users say that Samba is easily configured and that operation is very straight-forward. However, many new users run into problems with its complexity and non-intuitive mechanism. It is strongly suggested that the user sticks close to the following directions.

## Related articles

Samba/Tips and tricks

Samba/Troubleshooting

Samba/Advanced file sharing with KDE4

Samba domain controller

Active Directory Integration

Samba 4 Active Directory Domain Controller

OpenChange Server NFS

## Contents

- 1 Server configuration
  - 1.1 Creating a share
  - 1.2 Starting services
  - 1.3 Creating usershare path
  - 1.4 Adding a user
  - 1.5 Changing Samba user's password
  - 1.6 Required ports
- 2 Client configuration
  - 2.1 Manual mounting
    - 2.1.1 Add Share to /etc/fstab
    - 2.1.2 User mounting
  - 2.2 WINS host names
  - 2.3 Automatic mounting
    - 2.3.1 smbnetfs
      - 2.3.1.1 Daemon
    - 2.3.2 autofs
  - 2.4 File manager configuration
    - 2.4.1 GNOME Files, Nemo, Thunar and PCManFM
    - 2.4.2 KDE
    - 2.4.3 Other graphical environments
- 3 See also

## Server configuration

To share files with Samba, install `samba` (<https://www.archlinux.org/packages/?name=samba>), from the official repositories.

The Samba server is configured in `/etc/samba/smb.conf`. Copy the default Samba configuration file to `/etc/samba/smb.conf`:

```
# cp /etc/samba/smb.conf.default /etc/samba/smb.conf
```

**Tip:** Run `testparm` to check the validity of *samba* configuration file.

## Creating a share

Edit `/etc/samba/smb.conf`, scroll down to the **Share Definitions** section. The default configuration automatically creates a share for each user's home directory. It also creates a share for printers by default. There are a number of commented sample configurations included. More information about available options for shared resources can be found in `man smb.conf`. Here (<http://www.samba.org/samba/docs/man/manpages-3/smb.conf.5.html>) is the on-line version.

On Windows side, be sure to change `smb.conf` to the Windows Workgroup. (Windows default: WORKGROUP)

## Starting services

To provide basic file sharing through SMB start/enable `smbd.service` and `nmbd.service` services. See `smbd` (<http://www.samba.org/samba/docs/man/manpages-3/smbd.8.html>) and `nmbd` (<http://www.samba.org/samba/docs/man/manpages-3/nmbd.8.html>) manpages for details.

**Tip:** Instead of having the service running since boot, you can enable `smbd.socket` so the daemon is started on the first incoming connection. Don't forget to disable `smbd.service`.

## Creating usershare path

**Note:** This is an optional feature. Skip this section if you don't need it.

"Usershare" is a feature that gives non-root users the capability to add, modify, and delete their own share definitions.

This creates the usershares directory in `/var/lib/samba`:

```
# mkdir -p /var/lib/samba/usershare
```

This makes the group `sambashare`:

```
# groupadd sambashare
```

This changes the owner of the directory and group you just created to root:

```
# chown root:sambashare /var/lib/samba/usershare
```

This changes the permissions of the usershares directory so that users in the group sambashare can read, write and execute files:

```
# chmod 1770 /var/lib/samba/usershare
```

Set the following variables in `smb.conf` configuration file:

```
/etc/samba/smb.conf
...
[global]
usershare path = /var/lib/samba/usershare
usershare max shares = 100
usershare allow guests = yes
usershare owner only = yes
...
```

Add your user to the *sambashare* group. Replace *your\_username* with the name of your user:

```
# usermod -a -G sambashare your_username
```

Restart `smbd` and `nmbd` services.

Log out and log back in. You should now be able to configure your samba share using GUI. For example, in Thunar you can right click on any directory and share it on the network. If you want to share pathes inside your home directory you must make it listable for the group others.

## Adding a user

Create a Linux user account for *samba* user. Substitute *samba\_user* with preferred name if desired:

```
# useradd samba_user
```

Then create a *Samba* user account with the same name:

```
# pdbedit -a -u samba_user
```

## Changing Samba user's password

To change a user's password, use `smbpasswd` :

```
# smbpasswd samba_user
```

## Required ports

If running a firewall, don't forget to open required Samba ports ([https://wiki.samba.org/index.php/Samba\\_port\\_usage](https://wiki.samba.org/index.php/Samba_port_usage)).

## Client configuration

Only `smbclient` (<https://www.archlinux.org/packages/?name=smbclient>) is required to access files from a Samba/SMB/CIFS server. It is available from the official repositories.

Shared resources from other computers on the LAN may be accessed and mounted locally by GUI or CLI methods. Depending on the desktop environment, GUI methods may not be available. See also `#File_manager_configuration` for use with a file manager.

There are two parts in sharing access. The first is the underlying file system mechanism, which some environments have built in. The second is the interface which allows the user to mount shared resources.

## Manual mounting

For a lighter approach without support for listing public shares, only install `cifs-utils` (<https://www.archlinux.org/packages/?name=cifs-utils>) to provide `/usr/bin/mount.cifs`.

To list public shares on a server:

```
$ smbclient -L hostname -U%
```

Create a mount point for the share:

```
# mkdir /mnt/mountpoint
```

Mount the share using the `mount.cifs` type. Not all the options listed below are needed or desirable (ie. `password`).

```
# mount -t cifs //SERVER/sharename /mnt/mountpoint -o user=username,password=password,workgroup=workgroup
```

*SERVER*

The Windows system name.

*sharename*

The shared directory.

*mountpoint*

The local directory where the share will be mounted.

-o [options]

See `man mount.cifs` for more information.

#### Note:

- Abstain from using a trailing `/`. `//SERVER/sharename/` will not work.
- If your mount does not work stable, stutters or freezes, try to enable different SMB protocol version with `vers=` option. For example, `vers=2.0` for Windows Vista mount.

## Add Share to `/etc/fstab`

The simplest way to add an `fstab` entry is something like this:

```
/etc/fstab
//SERVER/sharename /mnt/mountpoint cifs username=username,password=password 0 0
```

However, storing passwords in a world readable file is not recommended! A safer method would be to use a credentials file. As an example, create a file and `chmod 600 filename` so only the owning user can read and write to it. It should contain the following information:

```
/path/to/credentials/sambacreds
username=username
password=password
```

and the line in your `fstab` should look something like this:

```
/etc/fstab
//SERVER/SHARENAME /mnt/mountpoint cifs credentials=/path/to/credentials/sambacreds 0 0
```

If using `systemd` (modern installations), one can utilize the `comment=systemd.automount` option, which speeds up service boot by a few seconds. Also, one can map current user and group to make life a bit easier, utilizing `uid` and `gid` options.

**Warning:** Using the `uid` and `gid` options may cause input output errors in programs

that try to fetch data from network drives.

```
/etc/fstab
```

```
//SERVER/SHARENAME /mnt/mountpoint cifs credentials=/path/to/smbcredentials,comment=systemd.automount
```

**Note:** Space in sharename should be replaced by `\040` (ASCII code for space in octal). For example, `//SERVER/share name` on the command line should be `//SERVER/share\040name` in `/etc/fstab`.

## User mounting

```
/etc/fstab
```

```
//SERVER/SHARENAME /mnt/mountpoint cifs users,credentials=/path/to/smbcredentials,workgroup=workgroup
```

**Note:** The option is `users` (plural). For other filesystem types handled by `mount`, this option is usually `user`; sans the "s".

This will allow users to mount it as long as the mount point resides in a directory controllable by the user; i.e. the user's home. For users to be allowed to mount and unmount the Samba shares with mount points that they do not own, use `smbnetfs`, or grant privileges using `sudo`.

## WINS host names

The `smbclient` (<https://www.archlinux.org/packages/?name=smbclient>) package provides a driver to resolve host names using WINS. To enable it, add "wins" to the "hosts" line in `/etc/nsswitch.conf`.

## Automatic mounting

There are several ways to easily browse shared resources:

### smbnetfs

**Note:** `smbnetfs` needs an intact Samba server setup. See above on how to do that.

First, check if you can see all the shares you are interested in mounting:

```
$ smbtree -U remote_user
```

If that does not work, find and modify the following line in `/etc/samba/smb.conf` accordingly:

```
domain master = auto
```

Now restart `smbd.service` and `nmbd.service`.

If everything works as expected, install `smbnetfs` (<https://www.archlinux.org/packages/?name=smbnetfs>) from the official repositories.

Then, add the following line to `/etc/fuse.conf`:

```
user_allow_other
```

and load the `fuse` kernel module:

```
# modprobe fuse
```

Now copy the directory `/etc/smbnetfs/.smb` to your home directory:

```
$ cp -a /etc/smbnetfs/.smb ~
```

Then create a link to `smb.conf`:

```
$ ln -sf /etc/samba/smb.conf ~/.smb/smb.conf
```

If a username and a password are required to access some of the shared folders, edit `~/.smb/smbnetfs.auth` to include one or more entries like this:

```
~/.smb/smbnetfs.auth
```

```
auth                "hostname" "username" "password"
```

It is also possible to add entries for specific hosts to be mounted by `smbnetfs`, if necessary. More details can be found in `~/.smb/smbnetfs.conf`.

If you are using the Dolphin or Nautilus file managers, you may want to the following to `~/.smb/smbnetfs.conf` to avoid "Disk full" errors as `smbnetfs` by default will report 0 bytes of free space:

```
~/.smb/smbnetfs.conf
```

```
free_space_size 1073741824
```

When you are done with the configuration, you need to run

```
$ chmod 600 ~/.smb/smbnetfs.*
```

Otherwise, `smbnetfs` complains about 'insecure config file permissions'.

Finally, to mount your Samba network neighbourhood to a directory of your choice, call

```
$ smbnetfs mount_point
```

### Daemon

The Arch Linux package also maintains an additional system-wide operation mode for `smbnetfs`. To enable it, you need to make the said modifications in the directoy `/etc/smbnetfs/.smb` .

Then, you can start and/or enable the `smbnetfs` daemon as usual. The system-wide mount point is at `/mnt/smbnet/` .

### autofs

See `Autofs` for information on the kernel-based automounter for Linux.

## File manager configuration

### GNOME Files, Nemo, Thunar and PCManFM

In order to access samba shares through GNOME Files, Nemo, Thunar or PCManFM, install the `gvfs-smb` (<https://www.archlinux.org/packages/?name=gvfs-smb>) package, available in the official repositories.

Press `Ctrl+l` and enter `smb://servername/share` in the location bar to access your share.

The mounted share is likely to be present at `/run/user/your_UID/gvfs` in the filesystem.

### KDE

KDE, has the ability to browse Samba shares built in. Therefore do not need any additional packages. However, for a GUI in the KDE System Settings, install the `kdenetwork-filesharing` (<https://www.archlinux.org/packages/?name=kdenetwork-filesharing>) package from the official repositories.

If when navigating with Dolphin you get a "Time Out" Error, you should uncomment and edit this line in `smb.conf`:

```
name resolve order = lmhosts bcast host wins
```

as shown in this page (<http://ubuntuforums.org/showthread.php?t=1605499>).



## Other graphical environments

There are a number of useful programs, but they may need to have packages created for them. This can be done with the Arch package build system. The good thing about these others is that they do not require a particular environment to be installed to support them, and so they bring along less baggage.

- `pyneighborhood` (<https://www.archlinux.org/packages/?name=pyneighborhood>) is available in the official repositories.
- `LinNeighborhood`, `RUmba`, `xffm-samba` plugin for `Xffm` are not available in the official repositories or the AUR. As they are not officially (or even unofficially supported), they may be obsolete and may not work at all.

## See also

- `Samba: An Introduction` (<http://www.samba.org/samba/docs/SambaIntro.html>)
- `Official Samba site` (<http://www.samba.org/>)

Retrieved from "<https://wiki.archlinux.org/index.php?title=Samba&oldid=352351>"

Category: Networking

- 
- This page was last modified on 24 December 2014, at 11:27.
  - Content is available under GNU Free Documentation License 1.3 or later unless otherwise noted.