

Setting Up Samba in Xubuntu 7.10

Written by Roberto J. Dohnert

Copyright © 2008

WARRANTY: There is no warranty either expressed or implied for this content. You agree to hold blameless for the content Roberto J Dohnert, RJD LLC, Microsoft Corporation and their respective partners as well as Canonical and their respective partners for any loss of data or damage.

Microsoft Windows and Microsoft are trademarks of the Microsoft Corporation. Ubuntu, Xubuntu and Canonical are trademarks of Canonical Inc.

Since Xubuntu comes stripped down there are a few tools you will need before you get started. There are two ways of achieving smb browsing in Xubuntu and we will discuss them both here. First we will go over setting up the Samba server then through client configuration and fuse configuration.

Configuring your computer as a server

Unlike KDE and GNOME, Xubuntu has very few tools with which to set up a Samba server. One unique tool you can get through Synaptic called `system-config-samba`.

Otherwise we must configure the server manually. This tutorial expects that the server is running Xubuntu as well.

First open a terminal: Accessories > Terminal and prep the `smb.conf` file. You can do this two ways; `sudo nano -w /etc/samba/smb.conf` or if you want a graphical way to do it, `sudo mousepad /etc/samba/smb.conf`. You must use `sudo` as you have to have admin rights in order to change the `smb.conf` file. To save in nano use "CTRL-O", then "CTRL-X". To save in mousepad its File > Save

The file `*smb.conf*` is divided in several sections:

Global Settings

Debugging/Accounting

Authentication

Printing

File sharing

Misc

Share Definitions

Let's start with Global Settings. Here you will see several lines, which you can also see in the graphical network tool like workgroup and wins server. If you changed everything to your liking already then you can skip this section, if not change to what you need. If you do not know what items mean, leave them alone and read the relevant part instead of changing them. It will save you trouble-shooting later.

The important part for us is File sharing. We need to change:

```
[homes]
```

```
comment = Home Directories
```

```
browseable = no
```

By default, the home directories are exported read-only. Change next parameter to 'yes' if you want to be able to write to them.

```
writable = no
```

This describes your `/home` folder. Usually you want to share this folder in a home-environment, because the directory has the files you want to share. To share this directory make the following changes.

```
[homes]
```

```
comment = Home Directories
```

```
browseable = yes
```

By default, the home directories are exported read-only. Change the next parameter to 'yes' if you want to be able to write to them.

```
writable = yes
```

This finishes sharing your /home folder. The last thing we need to do is designate a user. Add users who can access your shares with the 'smbpasswd' command.

```
sudo smbpasswd -a username
```

New SMB password:

Retype new SMB password:

Added user username.

NOTE: the username used here should be a real user setup on your PC/Server. Reload Samba for every change to users/passwords or 'smb.conf'

```
sudo /etc/init.d/samba reload
```

Private and public shares in same configuration

First you'll want to set this up in the [global] section of your smb.conf

```
[global]
```

```
security = user
```

```
encrypt passwords = true
```

```
map to guest = bad user
```

```
guest account = nobody
```

security = user restricts logins to users on your server. encrypt passwords = true is necessary for most modern versions of Windows to login to your shares. map to guest = bad user will map login attempts with bad user names to the guest account you specify with guest account = nobody. That is, if you attempt to login to the share with a user name not set up with smbpasswd then you will be logged in as the user nobody.

Next the private share

[private]

```
comment = Private Share
path = /path/to/share/point
browseable = no
read only = no
```

path is the path to the directory that you want to share out. browseable = no will have the share not show up when users browse the network. read only = no will let you, as an authenticated user, write to the share.

Finally, the public share

[public]

```
comment = Public Share
path = /path/to/share/point
read only = no
guest only = yes
guest ok = yes
```

Again, path is the path to the directory that you want to share out. read only = no will allow users to write to this share. guest only = yes and guest ok = yes will allow guest logins and also force users to login as guests. The user you specified with guest account in the [global] section must have write permissions on /path/to/share/point in order to write files to the share.

When Windows attempts to access a Samba share it will use the current Windows user name and password. The map to guest = bad user trick above allows access to the public share only if you give Samba an incorrect user name. If you give it a valid user name, but a bad password, the login will fail and Windows will give you a password prompt when you try to access the share. If you have the same user name for your Windows machine and your Ubuntu machine, you could be unwittingly giving the Samba server a valid user name, but invalid password. To resolve this you will either have to change the Windows user name, or to remove that user name from the Samba password file with `sudo smbpasswd -x [username]`.

The above uses security = user. To access the private shares you will have to make sure the user exists in smbpasswd. These users must also already exist as normal users on your machine. You add users to smbpasswd simply by running `sudo smbpasswd -a [username]` and giving a password.

Browsing Samba shares

There are a couple of tools you can get to browse Samba shares in Synaptic there is xSMBrowser, PyNeighborhood, TkSMB, for the next section make sure you have fuse and fusesmb installed.

Thunar and Fusesmb

Thunar by default doesn't have native network browsing, Thunar is the filemanager included with XFCE. Since Thunar doesn't have native network Browsing we have to use fusesmb to achieve the goal.

Install fusesmb from Synaptic. Edit /etc/modules and make sure the word 'fuse' is in the modules list as modules to be automatically started. In 7.10 and 8.04 Alpha its already there, I am not sure about 7.04. If you had to add fuse reboot your client PC so the fuse module loads, and the proper workgroup is read for samba. Next, in Applications -> System -> Users and Groups..Properties of your username... User Priveleges Tab... check "Allow use of fuse file systems..." Create a directory that you are going to mount your network browse to. I used /media/network. Change permissions to read / write for group and others (777). In a terminal, type: sudo chown <username>:fuse /media/network username will be your user account that you use. Make sure that the permission to use fuse took. Applications -> System -> Users and Groups... Manage Groups... find fuse and choose properties. Make sure your user name account is in that group and check-marked. Reboot the client PC in applications -> Settings -> Autostarted Applications... Add an application... name and describe it as you want for command line, put: fusesmb /media/network (Or whatever mountoint you created). Open Thunar, and navigate to the parent folder of your mountpoint... then drag the 'mounpoint folder' to the places (shortcut) pane of thunar. Logout and log back in this will invoke the user privilege and the fusesmb autostart will take affect

Mounting a Samba share

Mounting a share on the local filesystem allows you to work around programs that do not yet use a Virtual Filesystem like fusesmb or GnomeVFS to browse remote shares transparently. To mount a Samba share, first install smbfs from Synaptic.

To allow non root accounts to mount shares, change the permissions on the smbmnt program thus:

```
sudo chmod u+s /usr/bin/sbmbmnt /usr/bin/smbumount
```

The following will mount the share folder on your server to /media/network or in your home directory if you wish. Make sure you designate a directory as your mount point and NOT your home directory. Make the directory `mkdir ~/data`

```
smbmount //your server/share ~/data
```

To umount,

```
smbumount ~/data
```

In order to have a share mounted automatically every time you reboot, you need to do the following:

Open a shell as root `sudo -s` create a file containing your Windows/Samba user account details:

```
mousepad or nano /etc/samba/user
```

it should contain two lines as follows:

```
username = user
```

```
password = password
```

Change the permissions on the file for security:

```
chmod 0600 /etc/samba/user
```

Now create a directory where you want to mount your share:

```
mkdir /media/my_data
```

Now edit the file system table (/etc/fstab) and add a line as follows:

```
//server/share /media/my_data smbfs credentials=/etc/samba/user,rw,uid=user 0 0
```

where `user` is the non-root user you log into Xubuntu with, 'server' is the name or address of the server and `share` is the name of the share.

To mount the share now, just use the following command as root. It will mount automatically on subsequent reboots.

```
mount /media/my_data
```

Sharing Printers

If you want to share your printers make the following changes to Samba. If not already done create the Samba-user You want the share to be used by.

In `smb.conf` uncomment and change the lines ending up with the Printing configuration:

If you want to automatically load your printer list rather than setting them up individually then you'll need this

```
load printers = yes
```

CUPS printing. See also the `cupsaddsmb(8)` manpage in the

```
cupsys-client package.
```

```
printing = cups
```

```
printcap name = cups
```

and in the Share Definitions section modify the `[printers]` part ending up like this:

```
[printers]
```

```
comment = All Printers
```

```
browseable = no
```

```
path = /tmp
```

```
printable = yes
```

```
public = yes
```

```
writable = no
```

```
create mode = 0700
```

```
printcap name = /etc/printcap
print command = /usr/bin/lpr -P%p -r %s
printing = cups
```

Some explanation what is done:

the printers part defines the behavior for all the printers that are mentioned in "printcap name". A sort of template how to create shares for these printers. This template is applied if "load printers" is set to true. For more detailed explanation refer to the Samba documentation.

And do not forget to reload Samba:

```
sudo /etc/init.d/samba reload
```

If you run into problems check the web forums and the Samba documentation for help. This how-to contains the very basic steps in configuring Samba. If you need a more advanced howto you can scour the web for it or try a retailer such as Barnes and Noble or Borders books for a more in-depth literature. These basic steps should only be used in a home or small business office network.