

Samba

HOWTO: Fileserver with Samba and Printserver with CUPS

Contributed by: Alexander Kurusingal

License: Linux Forums Article License

Posted: 16 February, 2006

Source: http://www.linuxforums.org/articles/howto-fileserver-with-samba-and-printserver-with-cups_52.html

Copy date: 1 March 2010

Disclaimer: I (Jason) made changes to this document. So if you want the original, hit the link.

ASSUMPTIONS	1
CH 1. SETTING UP SAMBA	3
CH 1.1 CREATING THE GLOBAL SECTION.....	3
CH 1.2 SETTING UP SHARES.....	5
1.2.1 <i>Setting up file shares</i>	5
1.2.2 <i>Setting up Print shares</i>	6
CH 1.3 TESTING THE SAMBA SERVER.....	7
CH 1.4 CLIENT CONFIGURATION	8
CH 2. SETTING UP CUPS	9
CH 2.1 EDITING CUPSD.CONF	9
CH 2.1 INSTALLING THE DRIVERS	10
CH 2.2 CLIENT CONFIGURATION FOR CUPS	12

NOTE: This has been used and tested under SUSE Linux 9.3 and Gentoo 2005.1. I make an effort to be as distro-independant as possible, but cannot promise anything.

Assumptions

1. I assume you the reader will have some linux knowledge and are capable of using the CLI and editing various system files with root permission.
2. I assume you know how to install various packages either from source i.e. `./configure && make && make install` or by using various package managers for available with your distro e.g. YaST for SUSE, apt for Debian, YUM for Fedora, URPMI for Mandriva, Portage for Gentoo etc..
3. I assume your client computers are running either some version of Linux or Windows 2000 or above. Samba will work with Macs and versions of Windows prior to 2K. However since I don't have a Mac or a "Copy" of 98 lying around, I can't test out to see if this howto works.

4. I assume your network works, i.e. you can ping each and every single computer at any given time. For more information on how to ping, read the ping man page i.e. `man ping`

5. I assume your server will have a static ip, trust me, you will always want your server to have the same IP so that you don't always have to change settings when its dhcp lease expires etc..

6. I assume you are setting this up for a home network, and hence won't require Samba to act as a PDC. (Primary Domain Controller)

7. I assume the printer will be directly connected to the server by a usb cable or parallel port cable.

I suggest you read through this guide and understand the steps before attempting to perform it on your systems.

Note: anything placed inside `//` is a comment and not a command

Ch 1. Setting up Samba

Using your distro's package manager install the latest version of samba available for it. As of 30/01/2006 the latest version of samba is 3.0.21. You can get binary i.e. precompiled packages of Samba from here for various linux distros.

Install samba on your system before continuing.

Once samba has installed successfully, it is now time to edit the main samba configuration file. So in the terminal, become root and using your favourite editor, edit the `/etc/samba/smb.conf` file.

Code:

```
# nano -w /etc/samba/smb.conf
```

You may already see some things in this file, but for the purpose of this tutorial I am going to start from a clean `smb.conf` file. If you feel the need, backup your present `smb.conf` file with:

Code:

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

Now the `smb.conf` file is divided into 2 sections, the global section and the shares section. Firstly we will create the global section. As you may have guessed, the global section contains settings which will define the server.

Ch 1.1 Creating the global section

Quote:

```
[global]
netbios name = The_Server
server string= Samba Server
workgroup= My_Windows_Work_group
security = user
encrypt passwords = yes
smb passwd file = /var/lib/samba/private/smbpasswd
log file= /var/log/samba/%m.log
socket options= TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
wins support= yes
hostname lookups = yes
hosts equiv= /etc/hosts
hosts allow = 192.168.0.0/255.255.255.0 localhost
hosts deny = All
interfaces = lo eth0
bind interfaces only = yes
guest ok = yes
browse list = yes
```

printcap name = cups
printing = cups
load printers = yes

These fields will be explained below:

netbios name: This field should be filled in with the name of your server, this is what your windows computer will refer to the server as.

server string: This field is not really required, but it is handy incase you don't remember what you called your computer e.g. like me :P

workgroup: This field tells the server which Workgroup it belongs to. This field is absolutely essential.

security: This field refers to the level of security which should be present. There are two levels of security which I feel is within the scope of this howto to explain:

1. Share level security: Anyone can access any share without entering a username and password, i.e. no security.

2. User level security. If someone wants to access a share on the server, they need to enter a valid username and password. A valid username is any user who has an account on your server. To add a user, use the useradd command. A valid user needs a valid password, but to be able to use the share, they need a valid samba password. To add a samba password for a user, just execute this command:

Code:

```
# smbpasswd -a user_name
```

Personally I recommend you use user level security. However, if you feel the need to use share level security, by all means do so, but for making sure everything works, set it to user level security. This is because the command which tests whether the Samba server works requires user level security to work. Once the server is up and running, you can set it back to share level security.

encrypt passwords: Sounds pretty obvious doesn't it, I don't want other people knowing my passwords easily, so they should be encrypted.

smb passwd file: The file where the samba passwords should be stored. You can use a whereis or a find to find the proper location of your smbpasswd file.

log file: The file where the server should create the logs for each machine the %m you see there will actually be replaced with the name of the machine.

socket options: These are supposed to improve the sending and receiving of data.

wins support: One of my W2K box refuses to see the Samba server without this option enabled. This option just causes the Samba server to act as a WINS server. Because this option is enabled, you should set the WINS server option in your Windows clients to the IP address of the Samba server.

hostname lookups: This field just asks whether the server should perform lookups based on the hostname of the client computers. If you set this field, you need a hosts equiv field to tell the server the equivalent ip's of the other computers.

hosts equiv: This field just tells the server the location of the file which translates a IP address to a hostname.

hosts allow / hosts deny: These fields set which hosts can access the server based on their IP address or hostnames. In the example hosts allow field, I have told it to allow any computer with an IP address in the range 192.168.0.1 to 192.168.0.255 on the subnet 255.255.255.0 to have access to the server, localhost should also be included so that we can test to see whether the server works.

The hosts deny entry is set to All as I don't want anyone else accessing the shares.

interfaces: This field is only required if you have multiple network cards/connections. (Yes the Internet is a type of connection). This field should be set with the names of the interface/s that the Samba server should listen to for requests. Valid interface names on your system can be found by using the /sbin/ifconfig command.

bind interfaces only: This field tells the server only to listen to the interfaces listed in the "interfaces" field.

guest ok: allow guests to see the server, and some limited browsing. Usually should be set to no.

printcap name / printing: The type of printing system we are going to use.

load printers: Well we do want the system to load the printers automatically so that clients can use it, don't we?

Ch 1.2 Setting up shares

Shares on the server are what other people can see and access.

1.2.1 Setting up file shares

A basic file share should have the name of the share and any other relevant details, below is an example of a public file share which everybody can access.

[public] <----- This is the name of the share

path = /home/samba/public

comment= Shared folders

guest ok = yes

create mode = 0766

```
browseable = yes
public = yes
read only = no
```

this share allows everybody to access the shared folder on the server under /home/samba/server.

To create such a share with relevant permissions, execute these commands:

Code:

```
# mkdir /home/samba/public
# chmod -R 777 /home/samba/public
```

Now some people may want to access their own files from any computer connected to the network and prevent others from accessing it. In such a case, create a share like the following:

```
[home_directories]
comment= User's home directory
path= /home/%U
read only= no
valid users= %U root <--- We only want the legitimate user and root to be able to access the share
```

Note for sharing home directories with password protections, you obviously have to set the security level to user in the global section.

File shares can be set to things like removable media such as cd/dvd drives and usb drives, all you have to do is tell the server the correct path to the resource e.g. /mnt/usb

1.2.2 Setting up Print shares

We are now going to setup the Print share

Quote:

```
[printers]
comment = All Printers
browseable = no
printable = yes
writable = no
public = yes
guest ok = yes
path = /var/spool/samba
printer admin= root
```

This section is like a "global" setting for printer, we are allowing guests to be able to print from it and only allow root to administer it.

The next section just defines which printer to share, if you have multiple printers connected to your computer, just create more of the printer shares outlined below:

Quote:

```
[HP5160] <--- The name I will be referring to my printer from now on  
comment = HP Deskjet 5160  
printable = yes  
path = /var/spool/samba  
public= yes  
guest ok= yes  
printer admin= root
```

Once all that is done, save the file and perhaps create a backup of it. Now we will test the server.

Ch 1.3 Testing the Samba Server

There is a command called testparm which will parse the smb.conf file and see if you have made any errors. To run it, just do:

Code:

```
# testparm  
// if /usr isn't in your path, you may have to run the following command:  
# /usr/bin/testparm
```

Now we have to see if the samba service is running, to do that, use the following commands:

Code:

```
# ps -e |Samba by Example guide| grep nmbd  
If samba has not started, run the following command:
```

Code:

```
# /etc/init.d/samba start
```

Now we see if the services are running and the shares can be accessed by using the smbclient command.

Code:

```
# smbclient -L localhost
```

It will prompt you for the present user's samba password.

If the present user doesn't have a valid samba password, simply execute this command:

Code:

```
# smbpasswd -a user_name
```

Note, the execution of the smbclient command results in the most errors, most notably a NT_STATUS_LOGON_FAILURE error. A few things you can do to fix this:

1. Check that the smb.conf file has you in the correct workgroup
2. Check your samba passwords and which password file it uses.

Also, if you make any changes to the smb.conf file, you need to restart the samba service by executing this command:

Code:

```
# /etc/init.d/samba restart
// alternatively, you can do this
# /etc/init.d/samba stop
# /etc/init.d/samba start
```

Ch 1.4 Client Configuration

For all Windows clients you need to do the following:

1. Set the workgroup to the correct workgroup set in the smb.conf file
2. Under the Advanced TCP/IP settings for the adapter, there will be an entry for WINS server, set this to the IP address of the Samba server.
3. If you have a firewall e.g. Norton or Zone Alarm running, tell it to allow communication from the Samba server, normally, you just tell it the IP address of the server and set it to "Allow".
4. Sometimes you may need a restart for things to work :P

Linux clients:

To access Windows shares or SMB shares on the Samba server, you need to execute following command as root:

Code:

```
# mount -t smbfs -o username=YOUR_USERNAME,password=YOUR_SHARE_PASSWORD
//Server_name/share /mount_point
```

Alternatively you can write up the entry in your /etc/fstab file like so:

Quote:

```
//Server_name/Share /mount_point smbfs
username=YOUR_USERNAME,password=YOUR_SHARE_PASSWOR
D,rw,users,umask=000 0 0
```

YOUR_SHARE_PASSWORD is the password you have assigned to that share, if it is a windows share, use your windows password.

Since you have mounted windows shares in linux through Samba, you can now write to those partitions even NTFS based ones.

Note: for the above commands to work, you need the correct entries in the /etc/hosts file as shown below:

Quote:

```
// Server's IP address Hostname
192.168.0.1 My_Samba_Server
```

Ch 2. Setting up CUPS

CUPS is the Common Unix Printing System, we shall use this on our server to share the printer with clients. Note, here I am assuming that the printer will be connected to the Samba Server by means of either a USB or Parallel cable and not through the network.

Ch 2.1 Editing cupsd.conf

Now, like samba, cups has a configuration file, namely cupsd.conf. We shall edit the default cupsd.conf file as it would be difficult to start with a clean file, also there is plenty of documentation in the cupsd.conf file and if you feel brave enough, by all means enable some settings and whatnot. Below, I will show you how to edit the cupsd.conf file to get the basics running.

Firstly open the cupsd.conf file with your favourite editor:

Code:

```
# nano -w /etc/cups/cupsd.conf
```

the cupsd.conf file is fairly large, I'd suggest you edit the sections below first to get the server working, then play around with it.

Quote:

```
ServerName Name_of_Print_Server
```

```
ServerAdminroot@Name_of_Print_Server
```

```
.
```

```
.
```

```
MaxCopies 10 // I don't want someone accidentally wasting paper and ink on a job
```

```
.
```

```
.
```

```
MaxClients5 // Set this to whatever you like I don't want more than 5 connections to my server
```

```
.
```

```
.
```

```
BrowseAddress @IF(eth0) // change eth0 to your lan connection, just tells where to send printing updates to
```

```
.
```

```
BrowseAllow@IF(eth0) // only allow printing from LAN.
```

```
BrowseDenyAll // I don't want people on internet to try print using my printer [Razz]
```

```
BrowseOrder deny,allow // We first stop everyone from printing, then allow only local printing.
```

```
.
```

```
.
```

```
<Location />
```

```
Order Deny,Allow
```

```
Deny From All
```

```
Allow From 127.0.0.1 192.168.0.* // Change 192.168.0.* to address of internal network
```

```
</Location>
.
.
<Location /admin>
AuthType Basic
AuthClass System
Order Deny,Allow
Deny From All
Allow From 127.0.0.1 // Only the users sitting at the print server can perform admin
</Location>
```

Now, I know that at some point people are going to print MS Office Documents, if the following lines aren't uncommented, then you are going to get some screwed up prints. Trust me, I learned the hard way...

In `/etc/cups/mime.convs` file, uncomment the following line, it is towards the end.

Quote:

```
# application/octet-stream application/vnd.cups-raw 0
i.e. remove the # sign at beginning of line.
```

Similarly, uncomment the following line in `/etc/cups/mime.types`:

Quote:

```
# application/octet-stream
```

Now, before proceeding further, we need to start cups with:

Code:

```
# /etc/init.d/cupsd start
```

Ch 2.1 Installing the Drivers

Installing the Linux drivers:

Firstly, goto the linuxprinting.org site and get the correct CUPS driver for your printer. (link). Download the ppd file and place it in `/usr/share/cups/model`.

There are 2 ways of installing the Linux driver, firstly using the command line, as root do the following:

Code:

```
# lpadmin -p Printer_name_used_in_Samba -E -v usb:/dev/usb/ltp0 -m
Some_printer_name.ppd
```

The field `Printer_name_used_in_Samba` should be replaced with whatever you have shared your printer as. In the example `smb.conf` file given in Ch 1, I shared my printer as HP5160.

The field `usb:/dev/usb/ltp0` is what the system refers to as the location of your printer, note this field will vary across different systems. On some systems, when using usb printers, it could be at `/dev/ulpt0`. If you have a parallel printer, replace `usb` with `parallel:/dev/lpt0` or similar.

The field `Some_printer_name.ppd` is the name of the printer driver you have downloaded. For example, the HP Deskjet 5160 printer has a ppd file with the name `HP-DeskJet_5160-hpijs.ppd`.

If that method doesn't work, you can use the CUPS web interface to setup the printer. Simply launch your favourite web browser and point it to `http://Name_of_Print_Server:631` or `http://localhost:631`. Simply point it to the location of the printer, setup its share name and tell it the correct driver to use. Note, you would need to login to this admin webpage with username as root and with your root password. Note this is your root system password and not the samba password.

Installing Windows Drivers:

You can install the drivers in one of two ways. You can either have the driver files installed on to the CUPS server, then when you add a printer on the client, it will go to that directory and fetch the drivers. Or you can install the driver as normal on each client and point it to the shared printer on the CUPS server (Note, with this method, I couldn't get it to work using HP's own drivers and had to use Adobe's drivers).

Firstly, I will explain how to set it up so that the drivers reside on the server.

At the time of writing of this howto, the CUPS Windows drivers are still under development and hence won't be used here. Instead you have two options, either to use the Windows or Adobe Postscript drivers. Note If you have Windows clients which are pre Win 2K, you will need to use the Adobe Drivers.

Using Windows Postscript drivers

1. Make a directory in `/usr/local/share/cups` called "drivers"
2. Now on your windows machine, Navigate to the `C:WindowsSystem32SpoolDriversW32X863` folder. Copy whatever files in this folder to a flash drive, or if your samba server is working, copy it to a share on the server.
3. Now copy whatever files which are in this directory to `/usr/local/share/cups/drivers`

Using Adobe Postscript drivers

1. Make a directory in `/usr/share/cups` called "drivers"
2. Grab yourself a copy of the Adobe postscript drivers for your language from here. Also, get a ppd file for your printer.
3. Launch the adobe Installer and tell it to use the ppd for your printer, now the drivers will be extracted to `C:WindowsSystem32SpoolDrivers` folder. Copy these files to usb or a samba share on your server.
4. Copy the extracted driver files from a usb drive or samba server to `/usr/local/share/cups/drivers`.

Now, because we are setting it up so that the server will contain the drivers, we need add some things to the `smb.conf` file regarding the location of the drivers. Note the configuration below must be used for all printers

Quote:

```
[print$]
```

```
comment = Printer Drivers
```

```
path = /etc/samba/printer # this path holds the driver structure after cupsaddsmb command
```

```
guest ok = yes
```

```
browseable = yes
```

```
read only = yes
```

```
write list = root
```

Once that has been added, restart your samba service i.e:

Code:

```
/etc/init.d/samba restart
```

Now, to add the drivers to samba to be shared to all clients, we execute this command:

Code:

```
cupsaddsmb -H Name_of_Samba_Server -U root -h Name_of_Print_Server -a
```

In most cases, Name_of_Samba_Server and Name_of_Print_Server are the same.

The 2nd method of installing the drivers would i.e so that drivers are on client systems is like so:

Using the printer driver which came with your printer, tell it to install as a network printer and point it to the printer which resides on the Samba server. Note, for some reason this method didn't work for me and I had to use the Adobe method outlined below:

Grab yourself a copy of the Adobe postscript drivers for your language from here. Also, get a ppd file for your printer. Run the Adobe installer, point it to the location of the printer on the samba server. Now Under printing in the Control Panel, tell it to use this as default printer.

Ch 2.2 Client Configuration for CUPS

Windows Client configuration:

If you set it up so that driver files reside on the server, in explorer, simply navigate to the shared printer, right click on it and say "Connect", the drivers will be downloaded and you can start using it.

Otherwise simply go to the Add printer wizard in Control Panel and point it to the location of the Printer on the server. You may also want to set it as your default printer.

Linux Client Configuration (Other than the server):

Install a CUPS client on your system, usually by installing the CUPS server package, a CUPS client will also be installed. Now edit the /etc/cups/client.conf file and add the following:

Quote:

```
ServerName Name_of_Print_Server
```

And that is all there is to it. Now you should have a working file and print server.