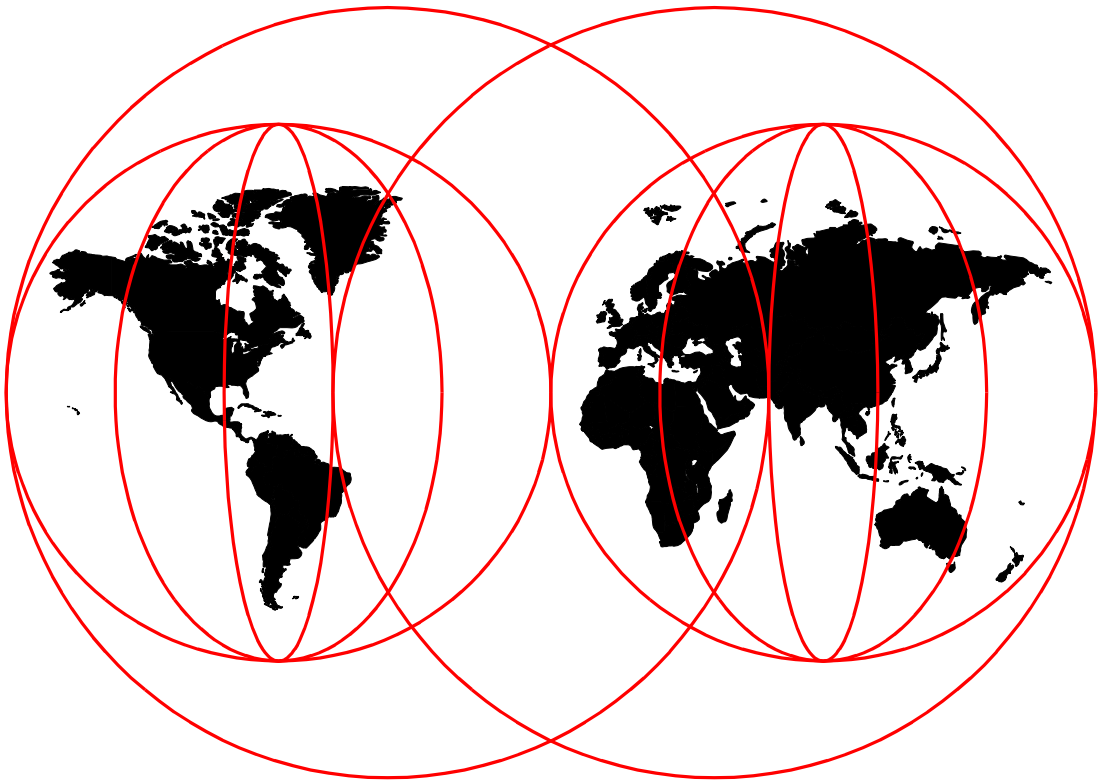
Implementing Linux in your Network using Samba

Jakob Carstensen, Ivo Gomilsek, Lenz Grimmer
Jay Haskins, Joe Kaplenk



International Technical Support Organization

http://www.redbooks.ibm.com

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix , “Special Notices” on page 97.

First Edition (November 1999)

This redpaper applies to IBM Netfinity systems preparing for or implementing Samba server for Linux.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HZ8 Building 678
P.O. Box 12195
Research Triangle Park, NC 27709-2195

Preface

This redpaper is divided into four chapters:

- Samba on SuSE Linux
- Samba on Caldera OpenLinux
- Samba on Red Hat Linux
- Samba on TurboLinux

The chapters cover how to install and manage Samba on IBM Netfinity servers. You will find the chapters very similar. However, there are differences between the four distributions and that is why we have created a chapter for each Linux version.

Assumptions about you

This redpaper assumes that you have already decided to set up your own Samba server. Chances are that you are interested in Samba because Samba can function as a file and print server in your existing Windows network. There is a slight chance that you are interested in Samba because it can function in your existing Windows network and because it is FREE. Yes, it means that you no longer have to pay for your file and print server software.

You may ask yourself: Do I have to be a Linux expert to install a Samba server in my existing network. The answer is no, you do not have to an expert. If you are a bit nervous about installing a Samba server in your existing network, just install it as a test server. Play with it and get familiar with it. You will learn that it is much easier than you think.

This document is written for all Windows NT users who are used to the safe and convenient graphical user interface.

Chapter 1. What is Samba?

If you look at any English dictionary, Samba is defined as a Brazilian dance, but Samba on Linux is something completely different. Samba is an implementation of a Server Message Block (SMB) protocol server that can be run on almost every variant of UNIX in existence. Samba is an open source project, just like Linux. The entire code is written in C so it is easily portable to all flavors of UNIX. Samba is a tool for the peaceful coexistence of UNIX and Windows on the same network on the level of File and Print sharing over the NetBIOS protocol. It allows UNIX systems to move into a Windows “Network Neighborhood” without causing a mess. With Samba, UNIX servers are acting as any other Windows server, offering its resources to the SMB clients. Recently SMB was renamed by Microsoft to Common Internet File System (CIFS).

1.1 What can you do with Samba?

- With Samba, a Linux server can act as a file/print server for Windows networks. It can replace expensive Windows NT file/print servers in this role, creating a less expensive solution.
- Samba can act as a NetBIOS name server (NBNS) in a Windows world referred to as WINS - Windows Internet Name Service.
- Samba can participate in NetBIOS browsing and master browser elections.
- Samba can provide a gateway for synchronization of UNIX and Windows NT passwords.
- With Samba client software you can access any shared directory or printer on Windows NT servers or Samba servers and allow UNIX machines to access Windows NT files.
- Using the Samba File System (SMBFS) you can mount any share from a Windows NT server or a Samba server in your directory structure (this is only available on Linux).

Chapter 2. Samba on SuSE Linux

This chapter explains in detail how to implement Samba on SuSE Linux.

2.1 Verifying if Samba is already installed

You can check if the Samba package is installed on your server by running the following `rpm` command on the command line:

```
rpm -q samba
```

If Samba is not installed, please follow the instructions for installing packages on SuSE Linux. The Samba package is located in package series n - Network-Support (TCP/IP, UUCP, Mail, News).

2.2 Configuring Samba

In this section we will explain how to configure Samba so it can participate as a file/print server in an existing Windows network or just as a stand-alone file/print server for Windows and Linux clients.

Before you can start using Samba, you need to configure the `smb.conf` file. This file is the heart of the Samba server. When the Samba package is installed on SuSE Linux, the configuration file is installed in:

```
/etc/smb.conf
```

The Samba configuration file `smb.conf` is divided into two main sections:

1. Global Settings - here you set up parameters that affect the connection parameters.
2. Share Definitions - here you define shares. A share is a directory on the server that is accessible over the network and shared among users. This section has three subsections:
    1. Homes - in this subsection you define the user's home directories.
    2. Printers - in this subsection you define the available printers.
    3. Shares - this subsection can have an entry for each share you want to define.

In the following sections we will describe how to modify `smb.conf` to efficiently and simply use Samba as a file/print server. We will cover only the basic parameters. If you need more information, see the manual entry for `smb.conf(5)` or the Samba project Web site at:

```
http://www.samba.org
```

2.2.1 Setting the NetBIOS parameters

The NetBIOS parameters are part of the Global Section. When you open your `smb.conf` file, you will see something similar to this:

```
T#================== Global Settings ======================
[global]
netbios name = NF5000
workgroup = LINUX
server string = Samba Server on Caldera OpenLinux
```

Table 1 describes parameters that define the NetBIOS naming of your Samba server.

Table 1. NetBIOS parameters

Parameter	Description
`netbios name`	This is the name by which the Samba server is known on the network. This parameter has the same meaning as the Windows NT Computer Name. If you don't specify it, it will default to the server's hostname.
`workgroup`	This parameter specifies in which Window NT Domain or Workgroup the Samba server will participate. It is equivalent to the Windows NT Domain or Workgroup name.
`server string`	This is the description string of the Samba server. It has the same role as the Windows NT Description field.

2.2.2 Global printing settings

In your `smb.conf` you will see something similar to this:

```
oad printers = yes
printcap name = /etc/printcap
printing = bsd
```

These parameters are described in Table 2.

Table 2. Printing parameters

Parameter	Description
`load printers`	This parameter controls if Samba loads all printers in the `printcap` file for browsing.
`printcap name`	With this parameter, you tell Samba the location of the `printcap` file. The default value is `/etc/printcap`.
`printing`	This parameter tells Samba what printing style to use on your server. SuSE Linux uses BSD printing style by default.

2.2.3 Global security settings

In your `smb.conf` you will see something similar to this:

```
security = user
;   password server = <NT-Server-Name>
encrypt passwords = yes
smb passwd file = /etc/smbpasswd
```

These parameters are described in Table 3.

Table 3. Security parameters

Parameter	Description
`security`	This parameter has four possible values: `share, user, server, domain`
`password server`	At the `server` or `domain` security level, the server is used for authorization. For the parameter value, you use the server NetBIOS name.
`encrypt passwords`	By setting this parameter to yes, you enable Samba to use the encrypted password protocol, which is used in Windows NT (starting with Service Pack 3) and Windows 95/98. This is needed to communicate with those clients.
`smb passwd file`	This parameter tells Samba where encrypted passwords are saved. By default, it will use `/etc/smbpasswd`.

We will briefly explain each security mode:

1. Share - for this security mode, clients need to supply only the password for the resource. This mode of security is the default for the Windows 95 file/print server. It is not recommended for use in UNIX environments, because it violates the UNIX security scheme.
2. User - the user/password validation is done on the server which is offering the resource. This mode is most widely used.
3. Server - user/password validation is done on the specified authentication server. This server can be a Windows NT server or another Samba server.
4. Domain - this security level is basically the same as server security, with the exception that the Samba server becomes a member of a Windows NT domain. In this case the Samba server can also participate in such things as trust relationships.

Because Windows NT 4.0 Service Pack 3 or later, Windows 95 with the latest patches, and Windows 98 use encrypted passwords for accessing NetBIOS resources, you need to enable your Samba server to use the encrypted passwords. Before you start the Samba server for the first time you need to create a Samba encrypted passwords file. This can be done with the `mksmbpasswd.sh` script. The recommended way is to first create the user accounts in Linux and then create the Samba password file with the command:

```
cat /etc/passwd | mksmbpasswd.sh > /etc/smbpasswd
```

This creates the Samba password file from the Linux password file.

Note

Use the same filename you specified for creating the Samba password file in the `smb.conf` configuration to tell the Samba server where the password file is.

Note

By default the passwords for the Samba users are undefined. Before any connection is made to the Samba server, users need to create their passwords.

Now you need to specify the password for all users. If you are changing or specifying the password for a user, you can do this by executing this command:

```
smbpasswd -U username
```

You will see a screen similar to the following:

```
[root@nf5000 /]# /usr/bin/smbpasswd -U user
New SMB password:
Retype new SMB password:
Password changed for user user.
[root@nf5000 /]#
```

Figure 1. Specifying the password for Samba user

Note

Anyone with access to the `/usr/bin/smbpasswd` can change passwords for the Samba users.

Another way is to have each Samba user change the password for himself, by connecting remotely to the Samba server and executing the command:

```
smbpasswd
```

The output will be similar to Figure 1. If a Samba user already has defined a password, he will need to type the old password before he can change to a new password.

If you want to add a Samba server user later, this can be done with the following command:

```
smbpasswd -a username password
```

This adds a new user to Samba password file.

Note

You have to be logged on as root if you want to manage other users. If you are logged on as a user, you can change your own password only. The `smbpasswd` utility uses the location of the password file from the `/etc/smb.conf` configuration file.

2.2.4 Global name resolution settings

In your `smb.conf` you will see something similar to this:

```
name resolve order = wins lmhosts bcast
wins support = yes
;   wins server = w.x.y.z
```

The parameters are described in Table 4.

Table 4. Name resolution parameters

Parameter	Description
`name resolve order`	With this parameter you specify how the Samba server resolves NetBIOS names into IP addresses. The preferred value is `wins lmhosts bcast`. Refer to the manual page of `smb.conf(5)` for more information.
`wins support`	If this option is enabled, Samba will also act as a WINS server.
`wins server`	With this parameter, you tell Samba which WINS server to use.

Important

Samba can act as a WINS server or a WINS client, but not both. So only one of the parameters (`wins support` or `wins server`) can be set at the same time. If you specify the IP address of WINS server then, `wins support` must be set to "no".

2.2.5 Creating shares

In the previous sections we have explained how to prepare general configuration parameters. But a Samba server is useful only when it offers resources to the users. In this section we will explain how to create a share. The simple share section in the `smb.conf` file looks similar to this:

```
[redbook]
   comment = Redbook files
   path = /redbook
   browseable = yes
   printable = no
   writable = yes
   write list = @users
```

Table 5 describes the most important parameters for creating a share.

Table 5. Share parameters

Parameter	Description
`comment`	This describes the function of this share.
`admin users`	This parameter is used to specify the users who have administrative privileges for the share. When they access the share, they perform all operations as `root`.
`path`	Defines the full path to the local directory you are sharing.
`browseable`	If this parameter is set to yes, you can see this share when you are browsing the resources on the Samba server. The value can be yes or no.
`printable`	This parameter specifies if the share is a print share. The value can be yes or no.

Parameter	Description
`write list`	Users specified in this list have write access to the share. If the name begins with @ it means a group name.
`writable`	This parameter specifies if the share is writable. The value can be yes or no.
`read list`	Users specified in this list have read access to the share. If the name begins with @ it means a group name.
`read only`	If this is set to yes, the share is read only. The value can be yes or no.
`valid users`	This parameter specifies which users can access the share.

You can easily set up a new share by using this basic set of parameters. Each share definition starts with the share name in square brackets "[]". You can specify the values for the share parameters below this name.

2.2.6 Share permissions

Although you can control the share permissions with share parameters, UNIX permissions are applied before the user can access files on the share. So you need to take care of UNIX permissions, so the user also has access to the share directory under UNIX.

When a user creates a new file on the shared directory, the default create mask for files is `0744`, and the default create mask for directories is `0755`. If you can also force the use of a certain creation mask. The parameters necessary for this are explained in Table 6.

Table 6. Create mask parameters

Parameter	Description
`create mask`	This is used for file creation to mask against UNIX mask calculated from the DOS mode requested.
`directory mask`	This is used for directory creation to mask against UNIX mask calculated from the DOS mode requested.

2.2.7 Creating shares for home directories

For handling home directories Samba has a special share section called `[homes]`. This share definition is used for all home directories, so you do not need to create separate shares for each user.

When a client requests a connection to a file share, existing file shares are scanned. If a match is found, that share is used. If no match is found, the requested share is treated as a username and validated by security. If the name exists and the password is correct, a share with that name is created by cloning the `[homes]` section. The home share definition uses the same parameters as a normal share definition. The following is an example of a home share definition in the `smb.conf` configuration file:

```
[homes]
comment = Home Directories
path = %H
valid users = %S
browseable = no
writable = yes
create mode = 0700
directory mode = 0700
```

As you can see we used some variables in this definition, which are explained in Table 7.

Table 7. Variable description

Parameter	Description
%H	This variable represents the home directory of the user.
%S	The name of the current service which, in the case of home share, is equal to the username.

As you can see in the example we used creation masks for the files and the directories, in a way so we forced all new files or directories to be accessible only by the owner of the home directory.

2.2.8 Creating a printer share

A Samba server uses the same procedure for printer shares as for the home shares. After all share definitions and usernames are tested against the requested share name and the matched definition is still not found. Samba searches for a printer with that name (if the `[printers]` section exists). If the match is found in the printer definitions that `[printers]` share section is cloned with the name of requested service, which is really a printer name. The following is an example of a printer definition in the `smb.conf` configuration file:

```
[homes]
comment = Home Directories
path = %H
valid users = %S
browseable = no
writable = yes
create mode = 0700
directory mode = 0700
```

As you can see the `[printers]` section is just another share definition, because when a user prints they basically copy the data into a spool directory; after that the data is handled by the local printing system. The only big difference between a printer share and other share definitions is that the parameter `printable` is set to "yes". This means that a user can write a spool file to the directory specified under the share definition. If the share is printable, then it is also writable by default.

2.3 Starting and stopping the Samba server

You can start the Samba server by executing the command:

```
rcsmb start
```

As you can see in the process table, two daemons are started: `smbd` and `nmbd`. `smbd` is the actual Samba server and `nmbd` is WINS server.

Samba server can be stopped by executing the command:

```
rcsmb stop
```

Whenever you make modifications to the `smb.conf` configuration file, you need to restart the Samba server. This can be done by executing the following command:

```
rcsmb restart
```

2.4 Starting Samba as startup service

You can configure your boot process so that Samba is started at bootup time.

To activate this feature, you simply have to set the variable `START_SMB` in `/etc/rc.config` to `yes`. You can either do this manually, or by using YaST.

The next time the Linux server is restarted, the Samba server will be started automatically.

2.5 Using SWAT

Samba Web Administration Tool (SWAT) allows the remote configuration of the `smb.conf` configuration file through a Web browser. That means you can configurate Samba in a GUI-like environment, which makes it much easier for administrators who are not used to a command line. SWAT itself is a small Web server and CGI scripting application designed to run from `inetd` provides access to the `smb.conf` configuration file.

Authorized users with the root password can configure the `smb.conf` configuration file via Web pages. SWAT also places help links to all configurable options on every page, which help the administrator to understand the effect of the changes.

Before using SWAT you must check the following.

1. In the file `/etc/services` you should have the following line (this is the default in SuSE Linux):

   ```
   swat 901/tcp
   ```

2. In the file `/etc/inetd.conf` you must have the following line:

   ```
   swat  stream  tcp     nowait.400      root    /usr/sbin/swat  swat
   ```

If you made any modification to those two files you need to restart `inetd`. This can be done by executing the command:

```
rcinetd reload
```

Congratulations! If you did everything without errors you are ready to use SWAT. To start SWAT point your favorite Web browser to the Internet address of your Samba server on port 901, as you can see in Figure 2.

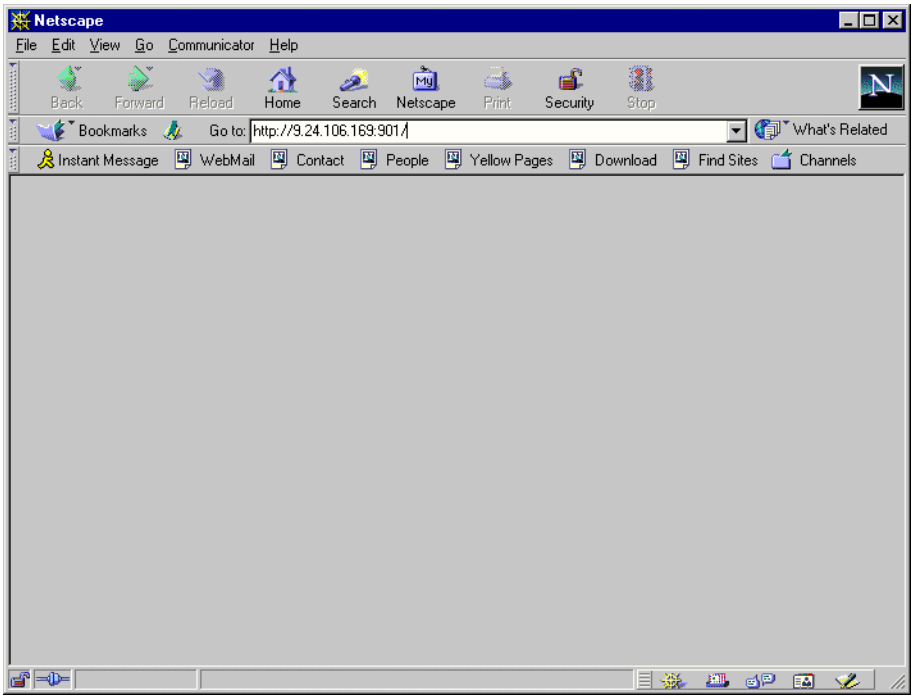


Figure 2. Starting SWAT

After you load the home page of SWAT, you will see a screen similar to Figure 3.

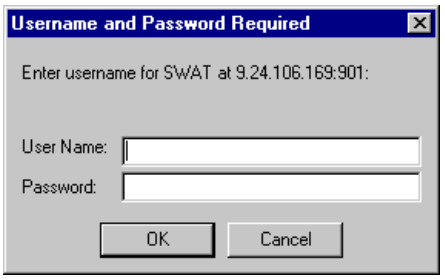


Figure 3. User authorization for SWAT

Type in the username and password of the Linux user defined on your Linux server. Click **OK** to continue. You will see a screen similar to Figure 4 on page 12.

Note

You can access SWAT with any Linux user, but you can make changes only with the root user.

Note

Remember, when you are logging on to SWAT from a remote machine you are sending passwords in plain text. This can be a security issue, so we recommend that you do SWAT administration only over a trusted network connection.

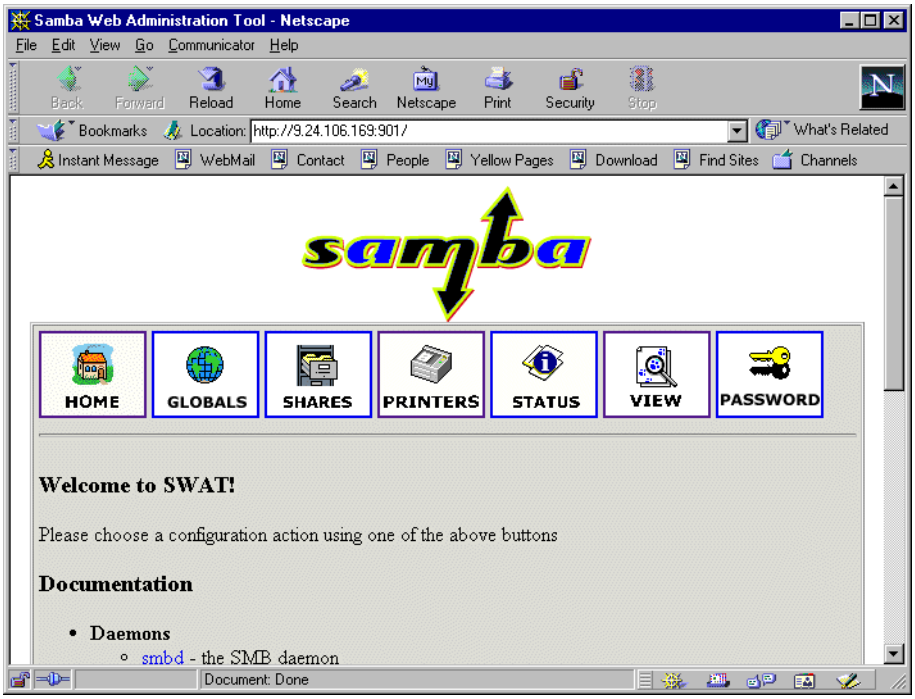


Figure 4. SWAT home page

As you can see in Figure 4, you have seven categories available:

1. Home - here you can find all the documentation you need about Samba.
2. Globals - here you can see and modify global parameters from the `smb.conf` configuration file.
3. Shares - here you can view, modify and add shares.
4. Printers - here you can view, modify and add printers.
5. Status - here you can check the current status of your Samba server.
6. View - here you can view the current configuration of the `smb.conf` configuration file.
7. Passwords - here you can manage passwords for the Samba server.

In the following sections we will briefly describe the sections available in SWAT.

Note

You can reach any of the seven sections on all SWAT Web pages. There are always icons for the sections on the top of each page.

After you make changes to `smb.conf` configuration file the Samba server must be restarted.

2.5.1 Globals

When you click on the **Globals** icon in the main SWAT window, you will see a window similar to Figure 5.

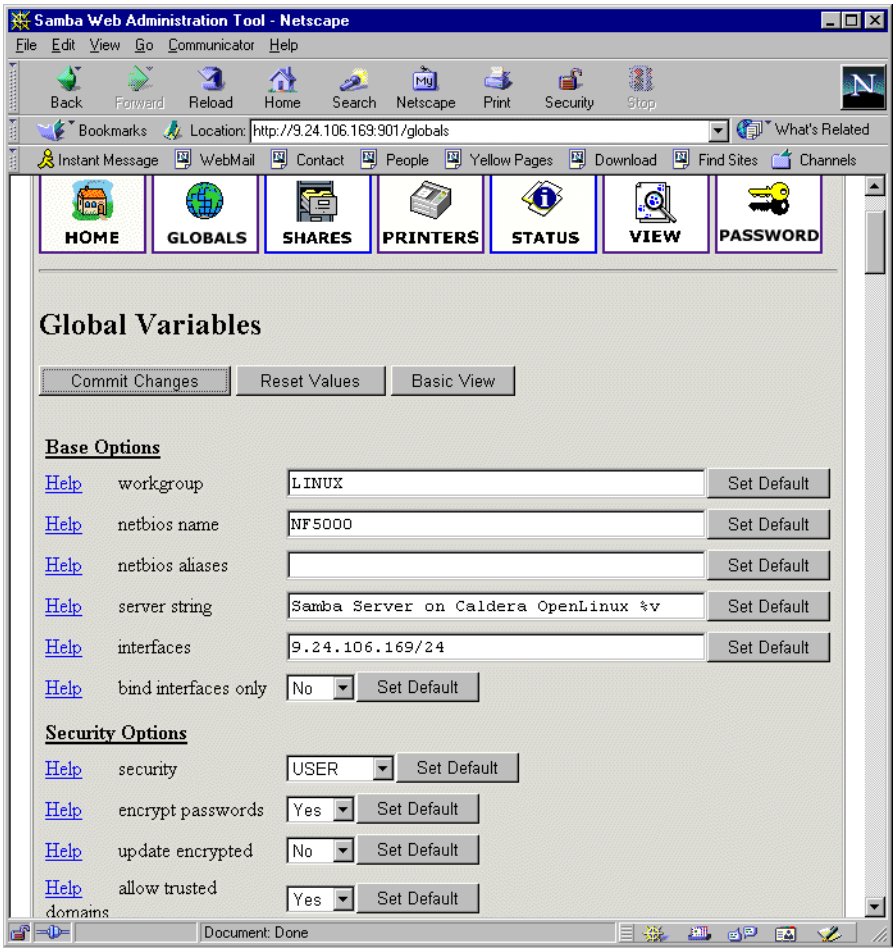


Figure 5. Global section in SWAT

In this window you can modify the global parameters for the Samba server. By default you will see the Basic View, if you want to see the Advanced View, click **Advanced View**. In the Advanced View you have all options available, while in the Basic View you can only change the basic options. To return from the Advanced View to the Basic View click on **Basic View.** After you have made your changes you can save them by clicking **Commit changes.** If you get a pop up window similar to Figure 6, which warns you that you are sending non secure information over the network, you can easily select **Continue** if you are working locally or if you know that your network is secure.

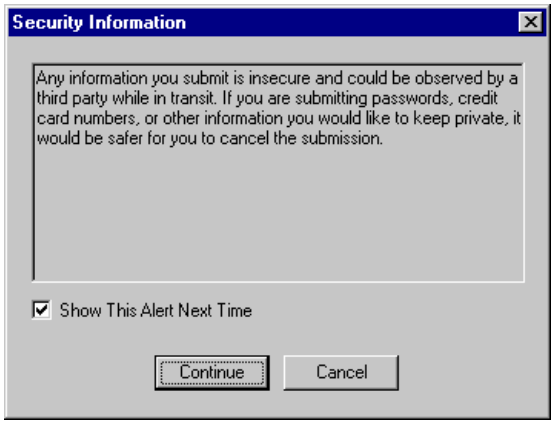


Figure 6. Security warning

2.5.2 Shares

When you click the **Shares** icon on any of the SWAT Web pages, you will see a screen similar to Figure 7.

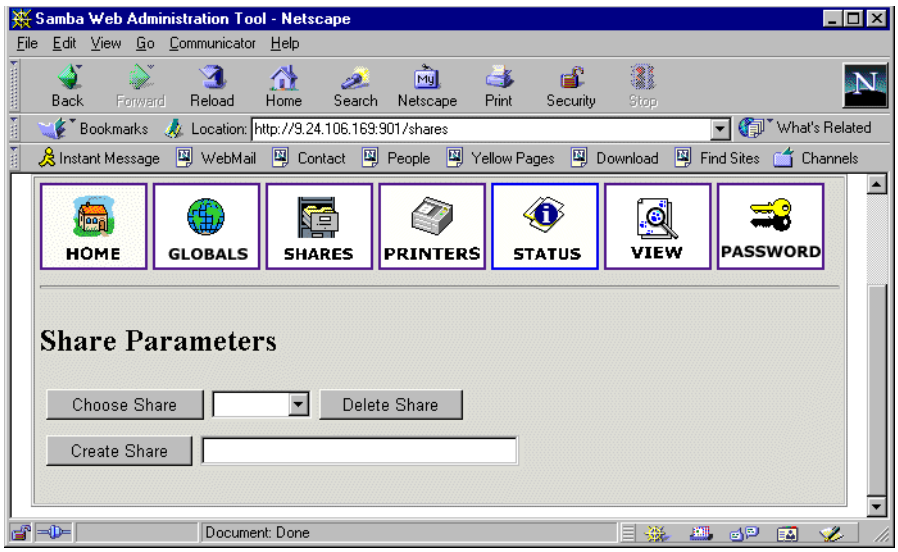


Figure 7. Shares section in SWAT

Here you can:

1. View the defined share
2. Delete share
3. Create a new share

2.5.3 Viewing or modifying an existing share

To view an already defined share, select the share from the field to the right of the **Choose Share** button, as shown in Figure 8.

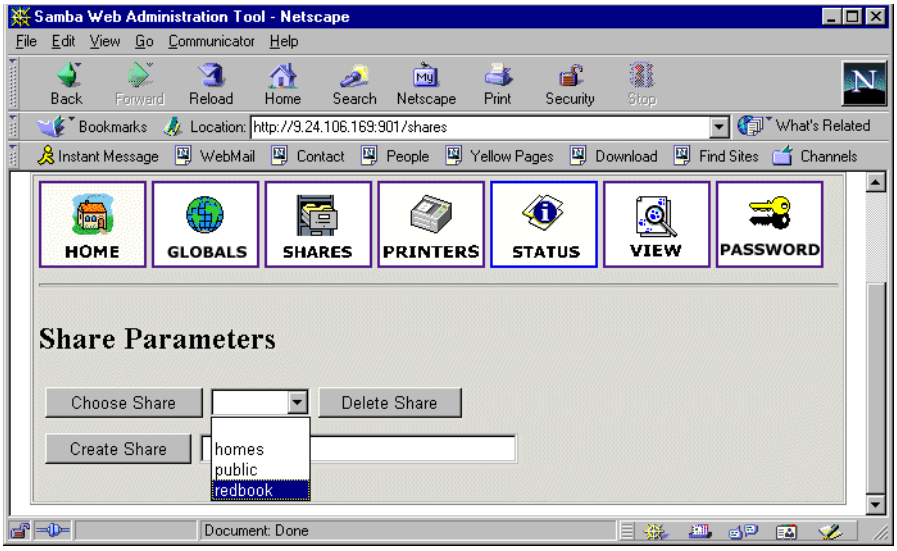


Figure 8. Choosing a share to view

After you have selected the share, click **Choose Share** to view the share properties. You will see a screen similar to Figure 9.

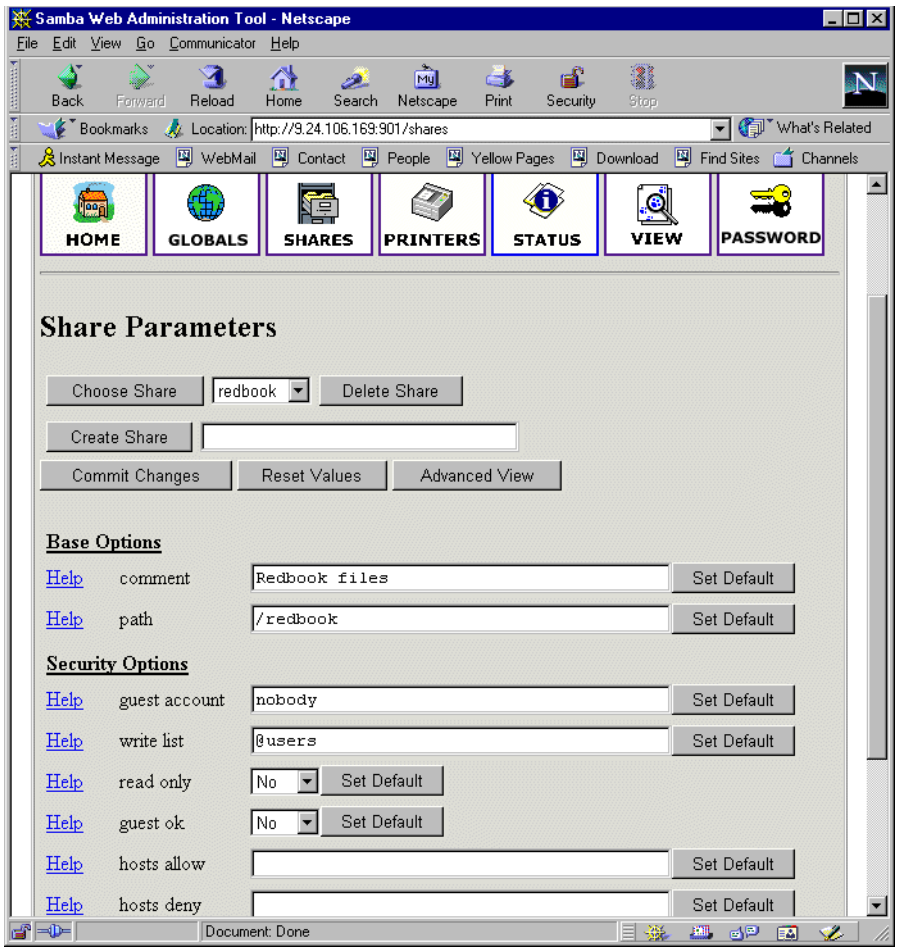


Figure 9. Share properties

If you want to see all available parameters, click **Advanced View.** In this view you can also make changes and you can save them by clicking **Commit Changes.**

2.5.4 Deleting an existing share

To delete an existing share you must first select an already defined share similar to Figure 8 on page 15. Then click **Delete Share.**

> **Note**
>
> A share is deleted immediately and without warning.

After you have deleted a share you must restart the Samba server.

2.5.5 Creating a new share

In this section we will show how to create a simple share. To accomplish this follow these steps:

1. Create a directory that will be used for the share. You can do this by executing this command from the terminal:

   ```
   mkdir /home/public
   ```

   In our example we created a “public” directory in the “home” directory.

2. Make sure that the UNIX permissions are set correctly in that directory, so that only intended users have access rights to it.

3. In the shares view on the SWAT Web pages, type in the name of the share you are creating similar to Figure 10.

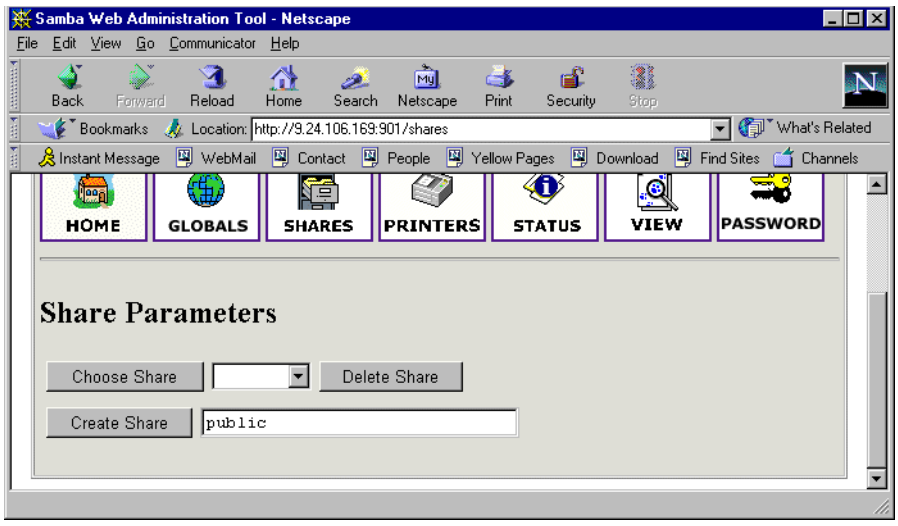


Figure 10. Entering the name for new share

4. Click **Create Share** to continue. You will see a screen similar to Figure 11.

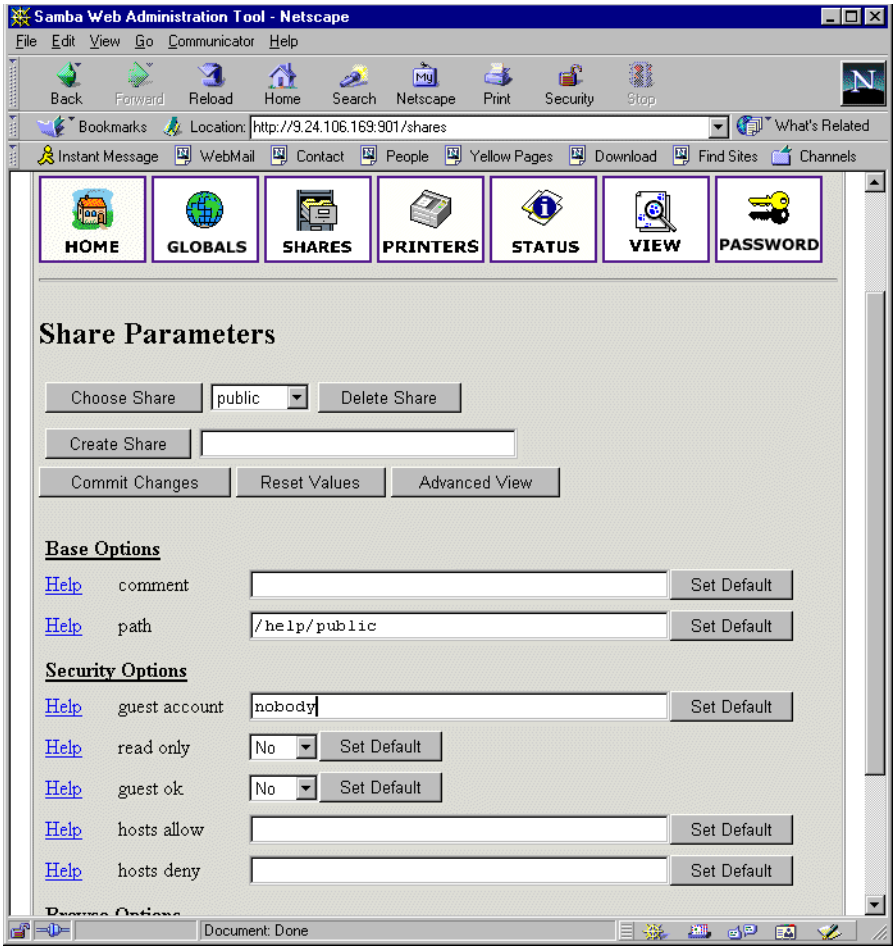


Figure 11. Entering the new share parameters

5. Fill in the needed parameters. If you need to set more advanced parameters, click **Advanced View** and you will see all available parameters. After you typed in all you want, click **Commit Changes** to save your new share.
6. You can see the changes in the `smb.conf` configuration file by selecting the **View** section from the SWAT Web page. You will see a screen similar to Figure 12.

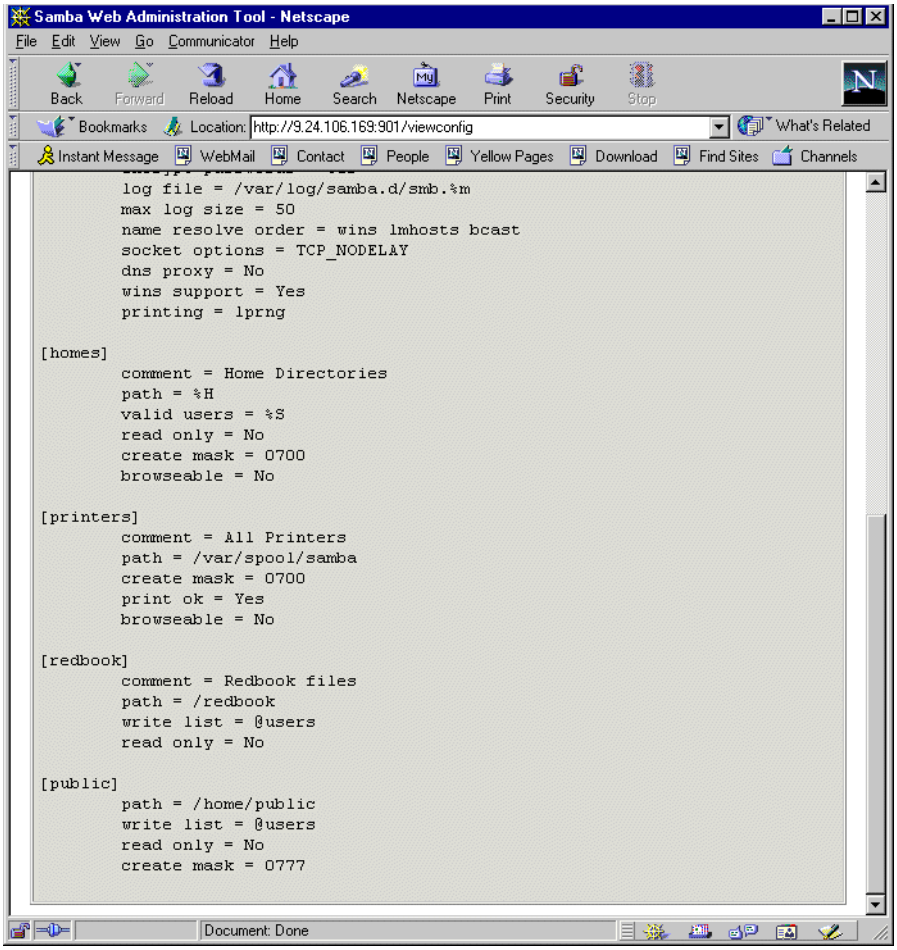


Figure 12. Viewing smb.conf configuration file

7. Restart the Samba server.

Congratulations! You have just created your first usable share on the Samba server. Be friendly and share it with other users!

2.5.6 Restarting the Samba server

The Samba server can be restarted from the Status section. To get to this section click the **Start** icon on any SWAT Web page. You will see a screen similar to Figure 13.

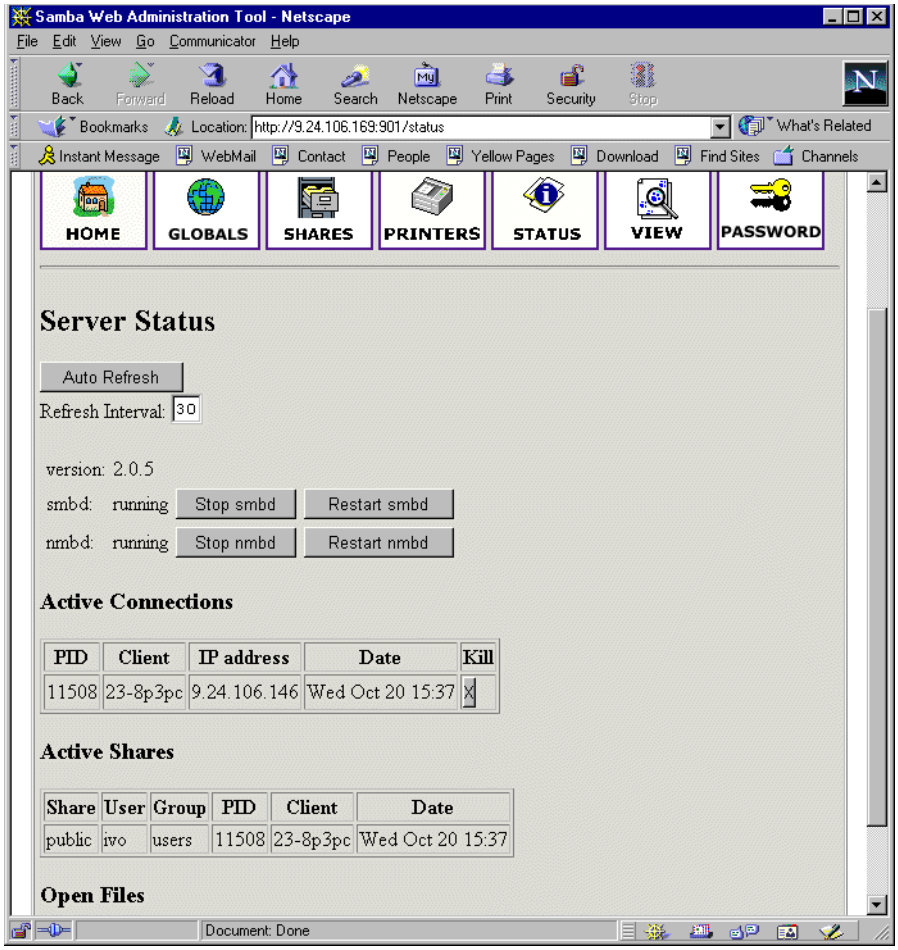


Figure 13. Restarting Samba server

To restart the Samba server, simply click **Restart smbd.** On this page you can also restart just the WINS server by clicking **Restart nmbd.**

2.5.7 Printers

In the printer section you can view, modify, or add printers. The operations for handling printers are the same as for handling shares. You can access the printer settings by clicking the **Printers** icon on the SWAT Web page similar to Figure 14.

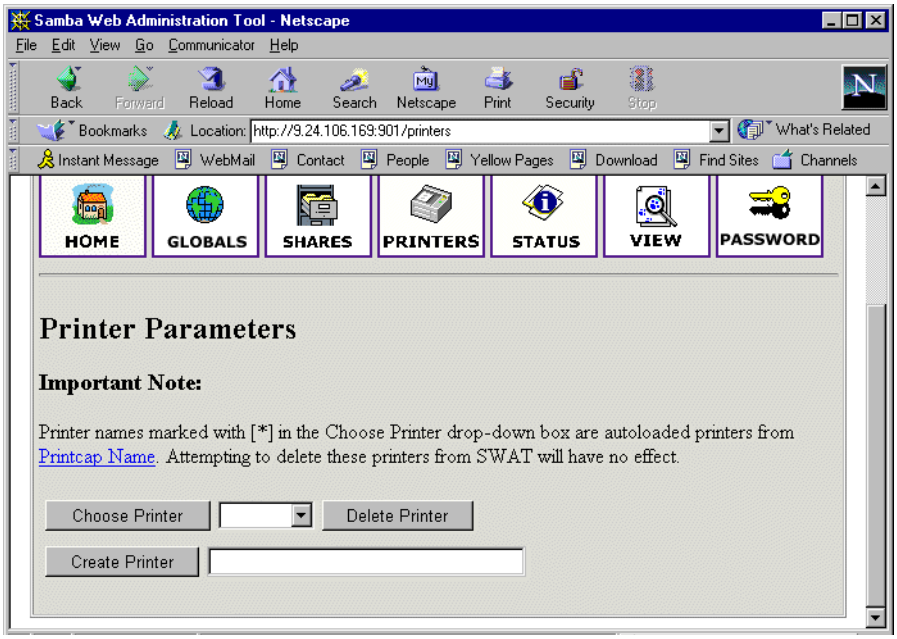


Figure 14. SWAT printers section

If you want to see the settings for a specific printer, select the printer from the list as shown in Figure 15.

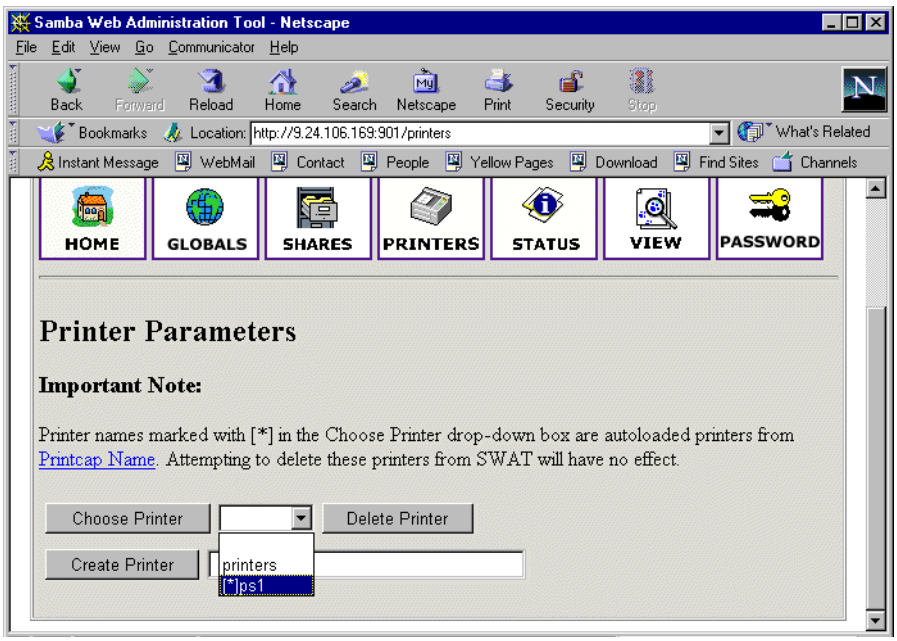


Figure 15. Selecting printer

After you have selected the printer click **Choose Printer** to view its properties. You will see a screen similar to Figure 16.

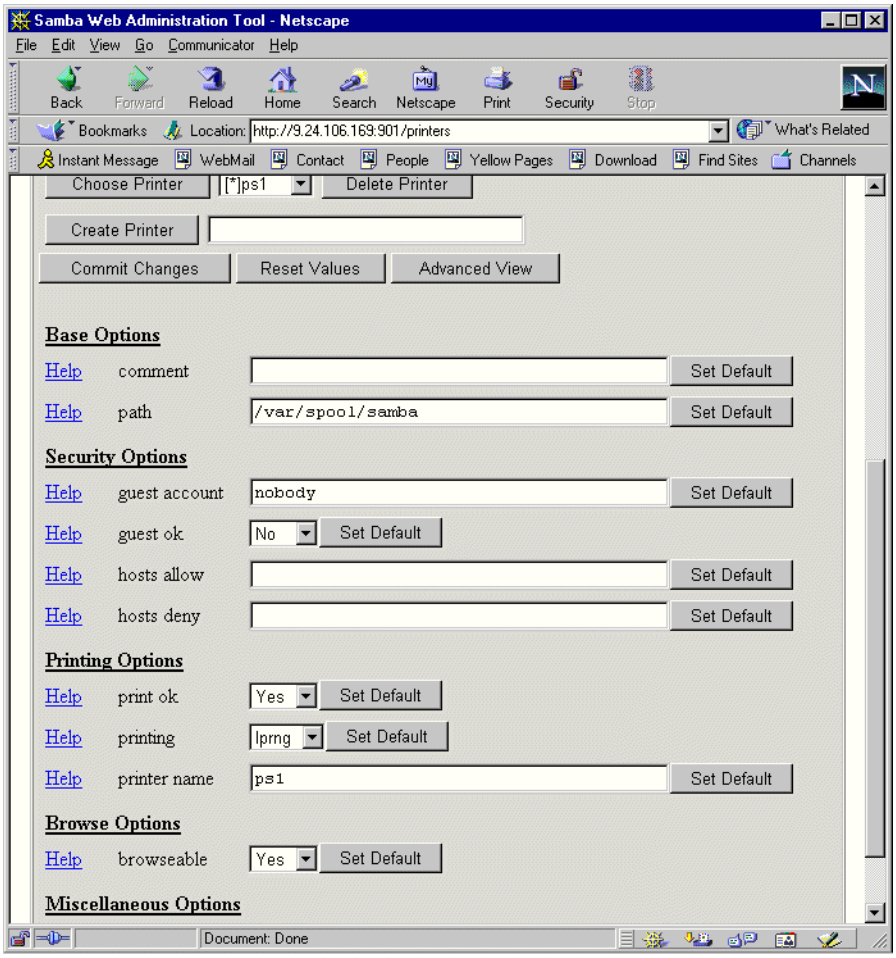


Figure 16. Printer properties

On this screen you can also modify the printer properties. When you are done, save the settings by clicking **Commit Changes.**

2.5.8 Status

In this section you can check the status of the Samba server. Here you can see all the connections and the open files. You can also start or restart the Samba server or just its components. You can access printer settings by clicking the **Status** icon on the SWAT Web page, as you can see in Figure 17.

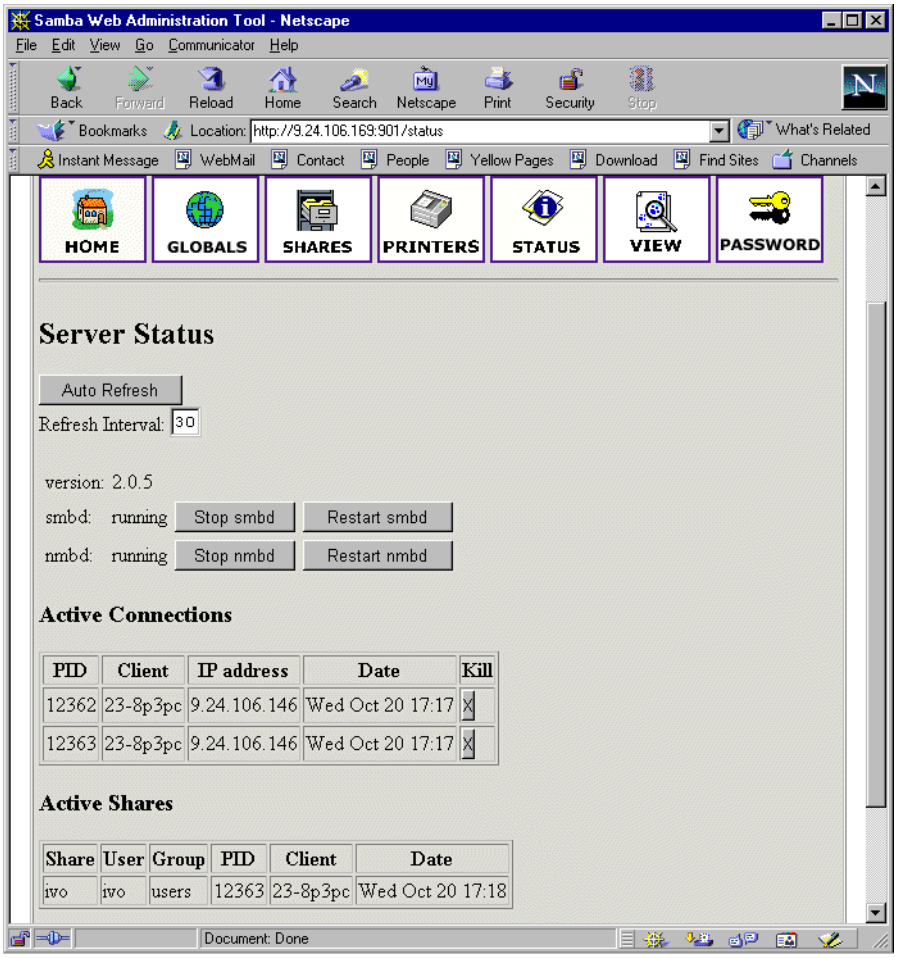


Figure 17. Status section

2.5.9 View

In this section you can see the current `smb.conf` configuration file. You can access printer settings by clicking the **View** icon on the SWAT Web page similar to Figure 18.

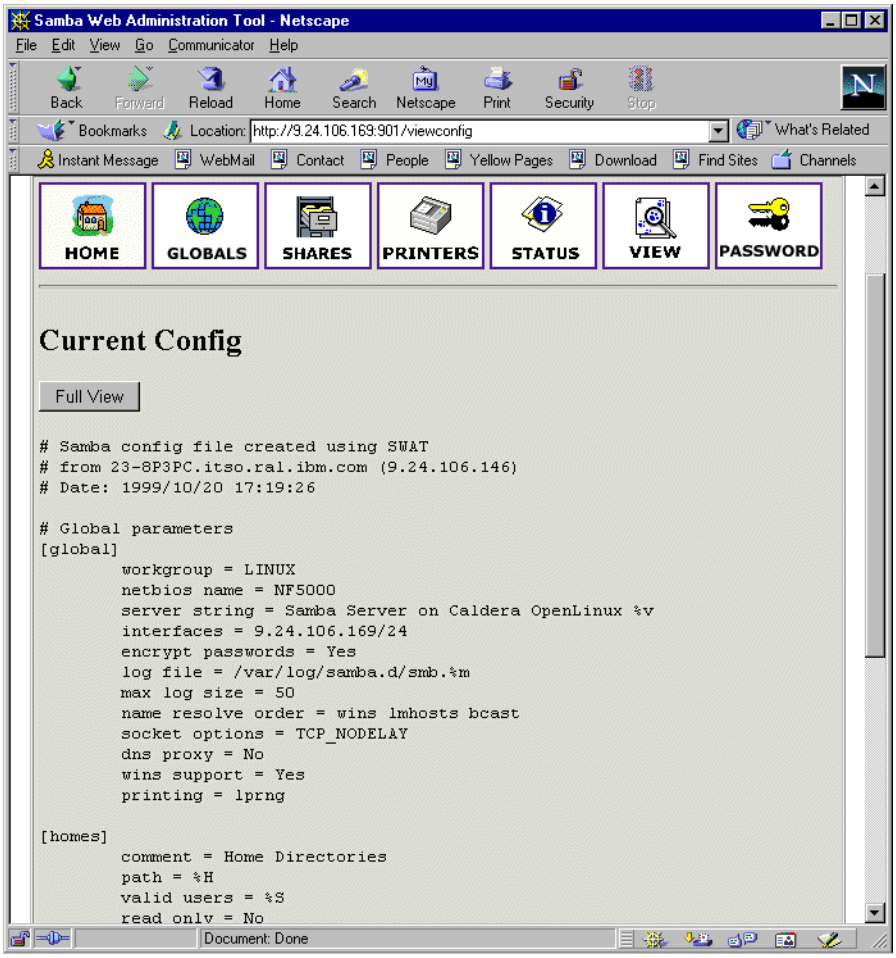


Figure 18. View section of SWAT

2.5.10 Password

In this section you can manage the passwords of all Samba users. You can access printer settings by clicking the **Password** icon on the SWAT Web page similar to Figure 19.

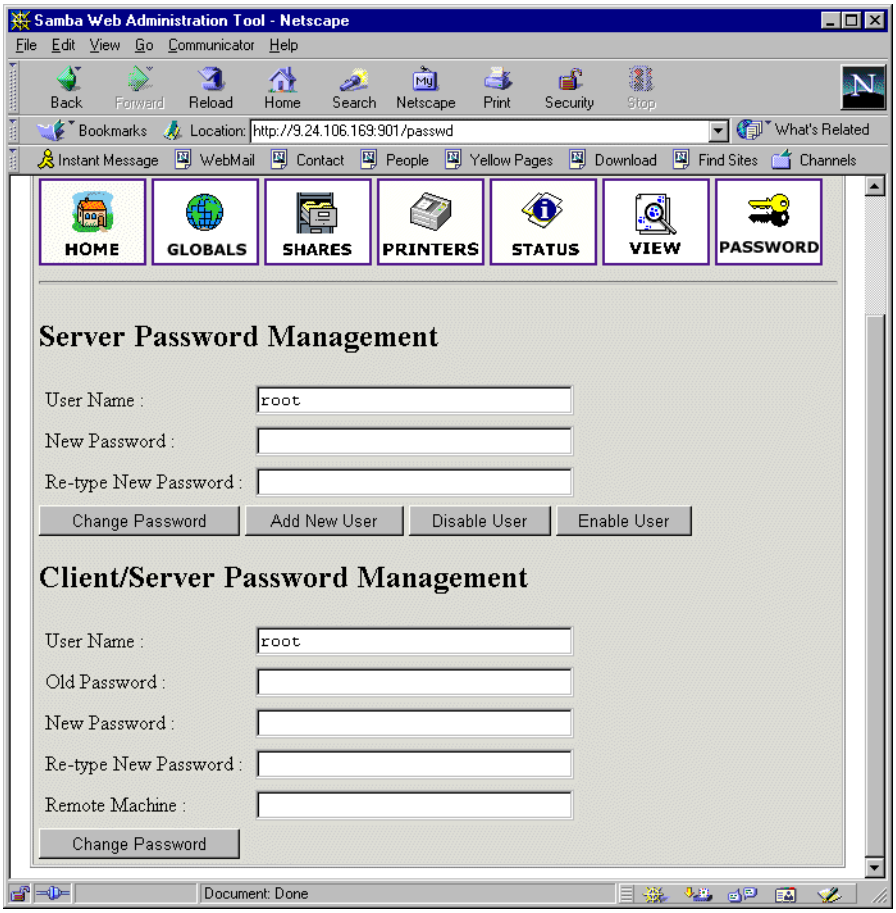


Figure 19. Managing passwords

2.6 Sources and additional information

You can find more information at the official Web site of the Samba project:

`http://www.samba.org`

And there are always good HOWTO documents at the Linux Documentation project Web site:

`http://www.linuxdoc.org/`

Chapter 3. Samba on Caldera OpenLinux

This chapter covers how to implement Samba on Caldera OpenLinux.

3.1 Verifying if Samba is already installed

You can check if the Samba package is installed on your server by running kpackage. To start kpackage click the **K** sign on the panel, select **COAS** and then **kpackage**.

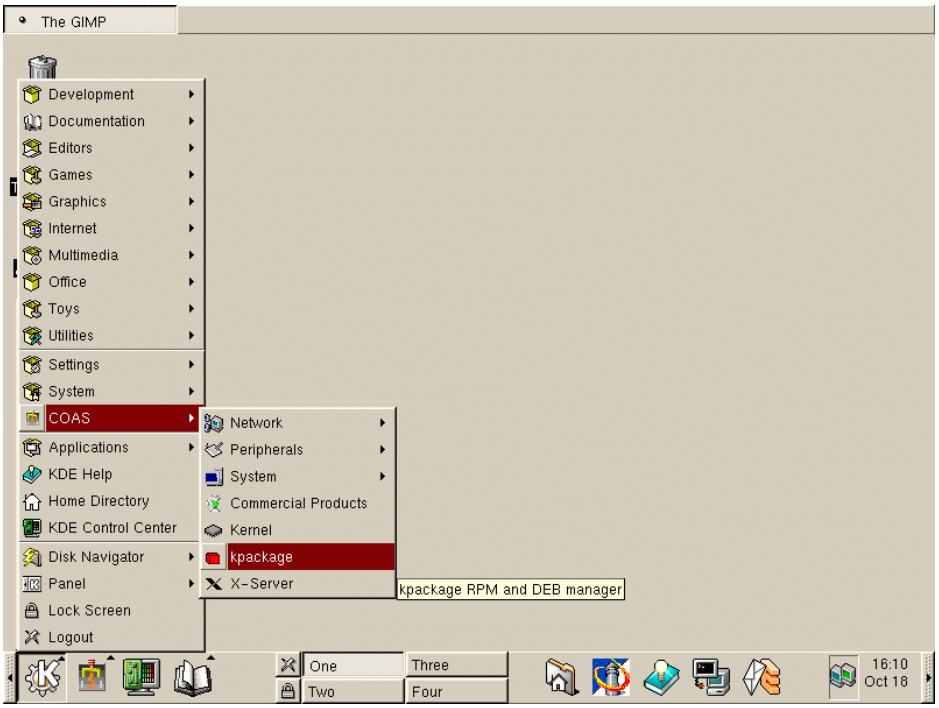


Figure 20. Starting kpackage

When kpackage is started, search for the **Server** section and then under this find the **Network** section and expand it. If the Samba package is installed you will see a screen similar to Figure 21.

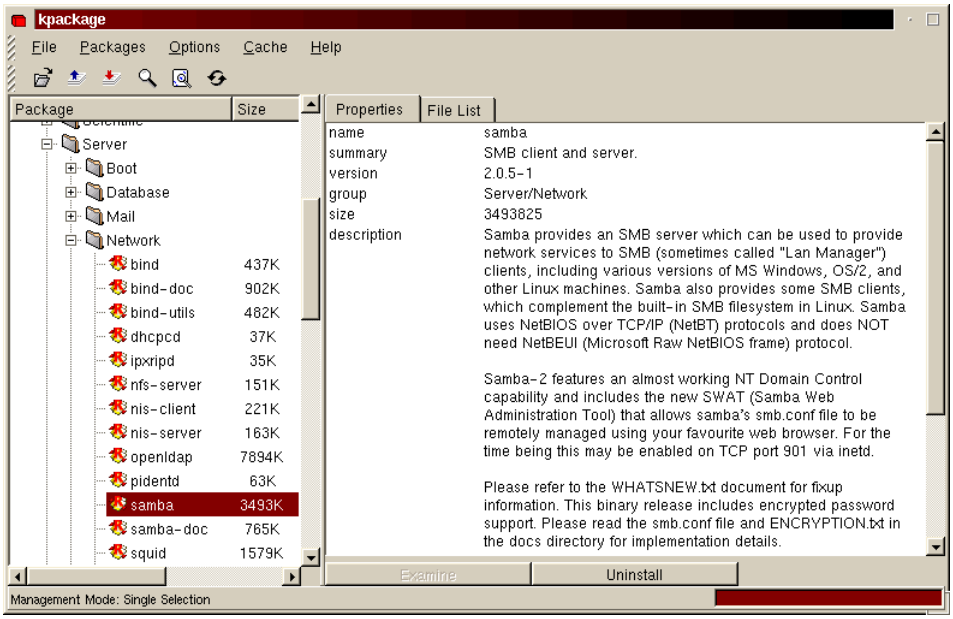


Figure 21. Checking for the Samba package

As you can see in Figure 21, the Samba package is installed.

3.2 Configuring Samba

In this section we will explain how to configure Samba so it can participate as a file/print server in an existing Window network or be a stand alone file/print server for Windows and Linux clients.

Before you can start using Samba, you need to configure the `smb.conf` file. This file is the heart of the Samba server. When the Samba package is installed on Caldera OpenLinux, the sample configuration file is installed in:

```
/etc/samba.d/smb.conf.sample
```

In Caldera OpenLinux, Samba by default uses the `smb.conf` file in the directory `/etc/samba.d`. To begin with, it is enough just to make a copy of the sample file by executing the command:

```
cp /etc/samba.d/smb.conf.sample /etc/samba.d/smb.conf
```

The Samba configuration file `smb.conf` is divided into two main sections:

1. Global Settings - here you set up parameters that affect the connection parameters.
2. Share Definitions - here you define shares. A share is a directory on the server that is accessible over the network and shared among users. This section has three subsections:
    1. Homes - in this subsection you define the user's home directories.
    2. Printers - in this subsection you define the available printers.
    3. Shares - this subsection can have an entry for each share you define.

In the following sections we will describe how to modify the `smb.conf` to efficiently and simply use Samba as a file/print server. We explain only the most necessary parameters. If you need more information, see the manual entry for `smb.conf` or the Samba project Web site at:

```
http://www.samba.org
```

3.2.1 Setting the NetBIOS parameters

The NetBIOS parameters are part of the Global Section. When you open your `smb.conf` file, you will see something similar to this:

```
#================== Global Settings =======================
[global]
      netbios name = NF5000
      workgroup = LINUX
      server string = Samba Server on Caldera OpenLinux
```

The parameters are described in Table 8.

Table 8. NetBIOS parameters

Parameter	Description
`netbios name`	The Samba server is known by this name on the network. This parameter has the same meaning as a Windows NT computer name. If you do not specify anything it defaults to the server's hostname.
`workgroup`	This parameter specifies in which Window NT domain or workgroup the Samba server will participate. It is equivalent to a Windows NT domain or a workgroup name.
`server string`	This is the description string of the Samba server. It has the same role as the Windows NT Description field.

3.2.2 Global printing settings

In your `smb.conf` you will see something similar to this:

```
load printers = yes
printcap name = /etc/printcap
printing = lprng
```

The parameters are described in Figure 9.

Table 9. Printing parameters

Parameter	Description
`load printers`	This parameter controls if Samba loads all printers in the printcap file for browsing.
`printcap name`	With this parameter, you tell Samba the location of the `printcap` file. The default value is `/etc/printcap`.
`printing`	This parameter tells Samba what printing style to use on your server. Caldera OpenLinux by default uses the LPRNG printing style.

3.2.3 Global security settings

In your `smb.conf` you will see something similar to this:

```
security = user
;   password server = <NT-Server-Name>
encrypt passwords = yes
smb passwd file = /etc/samba.d/smbpasswd
```

The parameters are described in Table 10.

Table 10. Security parameters

Parameter	Description
`security`	This parameter has four possible values: `share, user, server, domain`
`password server`	In the case of `server` or `domain` security level this server is used for authorization. For the parameter value, you use the server NetBIOS name.
`encrypt passwords`	With setting this parameter to yes, you enable Samba to use the Encrypted Password Protocol, which is used in Windows NT Service Pack 3 and in Windows 95/98. This is needed to communicate with those clients.
`smb passwd file`	This parameter tells Samba where encrypted passwords are saved.

We will briefly explain each security mode:

1. Share - for this security mode, clients need to supply only the password for the resource. This mode of security is the default for the Windows 95 file/print server. It is not recommended for use in UNIX environments, because it violates the UNIX security scheme.
2. User - user/password validation is done on the server that is offering the resource. This mode is most widely used.
3. Server - the user/password validation is done on the specified authentication server. This server can be a Windows NT server or another Samba server.
4. Domain - this security level is basically the same as server security, with the exception that the Samba server becomes a member of a Windows NT domain. In this case a Samba server can also participate in such things as trust relationships.

Because Windows NT 4.0 Service Pack 3 or later, Windows 95 with the latest patches, and Windows 98 use the encrypted passwords for accessing NetBIOS resources, you need to enable your Samba server to use the encrypted passwords. Before you start the Samba server for the first time, you need to create a Samba encrypted passwords file. This can be done with the `mksmbpasswd` utility. The recommended way is to first create the user accounts in Linux and then create the Samba password file with the command:

```
cat /etc/passwd | /usr/sbin/mksmbpasswd > /etc/samba.d/smbpasswd
```

This creates the Samba password file from the Linux password file.

Note

Use the same filename you specified for creating the Samba password file in the `smb.conf` configuration to tell the Samba server where the password file is.

By default the passwords for the Samba users are undefined. Before any connection is made to the Samba server, users need to create their passwords.

Now you need to specify passwords for all users. If you are changing or specifying a password for a user, you can do this by executing the command:

```
/usr/bin/smbpasswd -U username
```

You will see a screen similar to the following:

```
[root@nf5000 /]# /usr/bin/smbpasswd -U user
New SMB password:
Retype new SMB password:
Password changed for user user.
[root@nf5000 /]#
```

Figure 22. Specifying a password for a Samba user

Note

Anyone with access to the `/usr/bin/smbpasswd` can change passwords for the Samba users.

Another way is to have each Samba user change the password for themselves, by remotely connecting to the Samba server and executing the command:

```
/usr/bin/smbpasswd
```

The output will be similar to Figure 22. If a Samba user already has defined a password he will need to type the old password before he can change to a new password.

If you want to add a Samba server user later, this can be done with the following command:

```
/usr/sbin/smbpasswd -a username password
```

This adds a new user to the Samba password file.

Note

You have to be logged on as root if you want to manage other users. If you are logged on as a user, you can change your password only. The `smbpasswd` utility uses the location of the password file from the `smb.conf` configuration file.

3.2.4 Global name resolution settings

In your `smb.conf` you will see something similar to this:

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

```
wins support = yes
;   wins server = w.x.y.z
```

The parameters are described in Table 11.

Table 11. Name resolution parameters

Parameter	Description
`name resolve order`	With this parameter you specify how the Samba server resolves NetBIOS names into IP addresses. The preferred value is `wins lmhosts bcast`. Refer to the manual page of the `smb.conf` for more information.
`wins support`	If this option is enabled the Samba server will also act as a WINS server.
`wins server`	With this parameter, you tell Samba which WINS server to use.

Note

Samba can act as a WINS server or a WINS client, but not both. So only one of the parameters (`wins support` or `wins server`) can be set at the same time. If you specify the IP address of a WINS server, then `wins support` must be set to "no".

3.2.5 Creating shares

In the previous sections we explained how to prepare general configuration parameters. But a Samba server can be useful when you offer resources to the users. In this section we will explain how to create a share. The simple share section in the `smb.conf` file looks similar to this:

```
[redbook]
   comment = Redbook files
   path = /redbook
   browseable = yes
   printable = no
   writable = yes
   write list = @users
```

Table 12 describes the most important parameters for creating a share.

Table 12. Share parameters

Parameter	Description
`comment`	This describes the function of the share.
`admin users`	This parameter is used to specify the users who have administrative privileges for the share. When they access the share, they perform all operations as root.
`path`	Defines the full path to the directory you are sharing.

Parameter	Description
`browseable`	If this parameter is set to yes, you can see the share when you are browsing the resources on the Samba server. The value can be yes or no.
`printable`	This parameter specifies if the share is a print share. The value can be yes or no.
`write list`	Users specified in this list have write access to the share. If the name begins with @ it means a group name.
`writable`	This parameter specifies if the share is writable. The value can be yes or no.
`read list`	Users specified in this list have read access to the share. If the name begins with @ it means a group name.
`read only`	If this is set to yes, the share is read only. The value can be yes or no.
`valid users`	This parameter specifies which users can access the share.

By using these parameters you can easily set up a new share. Each share definition starts with the share name in brackets "[]". Below this name you can specify the values for the share parameters.

3.2.6 Share permissions

Although you can control the share permissions with share parameters, UNIX permissions are applied before user can access files on the share. So you need to take care of UNIX permissions, so the user also has access to the share directory under UNIX.

When a user creates a new file on the shared directory, the default create mask used is 0744. For directory creation the default create mask is 0755. If you want, you can force a different creation mask. The parameters for doing this are explained in Table 13.

Table 13. Create mask parameters

Parameter	Description
`create mask`	This is used for file creation to mask against UNIX mask calculated from the DOS mode requested.
`directory mask`	This is used for directory creation to mask against UNIX mask calculated from the DOS mode requested.

3.2.7 Creating shares for home directories

For handling home directories Samba has a special share section called [homes]. This share definition is used for all home directories, so you do not need to create separate shares for each user.

When a client requests a connection to a file share, existing file shares are scanned. If a match is found, that share is used. If no match is found, the requested share is treated as a username and validated by security. If the name exists and the password is correct, a share with that name is created by cloning the [homes] section. The home share definition uses the same parameters as a

normal share definition. The following is an example of a home share definition in the `smb.conf` configuration file:

```
[homes]
   comment = Home Directories
   path = %H
   valid users = %S
   browseable = no
   writable = yes
   create mode = 0700
   directory mode = 0700
```

As you can see, we used some variables in this definition, which are explained in Table 14.

Table 14. Variable description

Parameter	Description
%H	This variable represents the home directory of the user.
%S	The name of the current service which, in the case of home share, is equal to the username.

As you can see in the example, we used creation masks for the files and the directories, in such a way that we forced all new files or directories to be accessible only by the owner of the home directory.

3.2.8 Creating a printer share

A Samba server uses the same procedure for printer shares as for the home shares. After all share definitions and usernames are tested against the requested share name and the matched definition is still not found, Samba searches for a printer with that name (if the [printers] section exists). If the match is found in the printer definitions that [printers] share section is cloned with the name of requested service, which is really a printer name. The following is an example of the printers definition in the `smb.conf` configuration file:

```
[printers]
   comment = All Printers
   path = /var/spool/samba
   browseable = no
   # Set public = yes to allow user 'guest account' to print
   guest ok = no
   writable = no
   printable = yes
   create mask = 0700
```

As you can see the [printers] section is just another share definition, because when a user prints they basically copy the data into a spool directory; after that the data is handled by the local printing system. The only big difference between a printer share and other share definitions is that the `printable` parameter is set to "yes". This means that a user can write a spool file to the directory specified in the share definition. If the share is printable, then it is also writable by default.

3.3 Starting and stopping the Samba server

You can start the Samba server by executing the command:

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

You will see output similar to the following:

```
[root@nf5000 /root]# /etc/rc.d/init.d/samba start
Starting samba:   smbd nmbd.
```

Figure 23. Starting Samba

As you can see two daemons are started: `smbd` and `nmbd`.. `Smbd` is the actual Samba server and `nmbd` is WINS server.

Samba server can be stopped by executing the command:

```
/etc/rc.d/init.d/samba stop
```

Whenever you make modifications to the `smb.conf` configuration file you must restart the Samba server.

3.4 Starting Samba as startup service

You can configure your boot process so Samba is started when the server is booting. You can do this by using the System Daemon configuration tool. To start the System Daemon configuration tool click the **K** sign on the panel, select **COAS**, then **System** and at the end **Daemons**. You will see a screen similar to Figure 24.

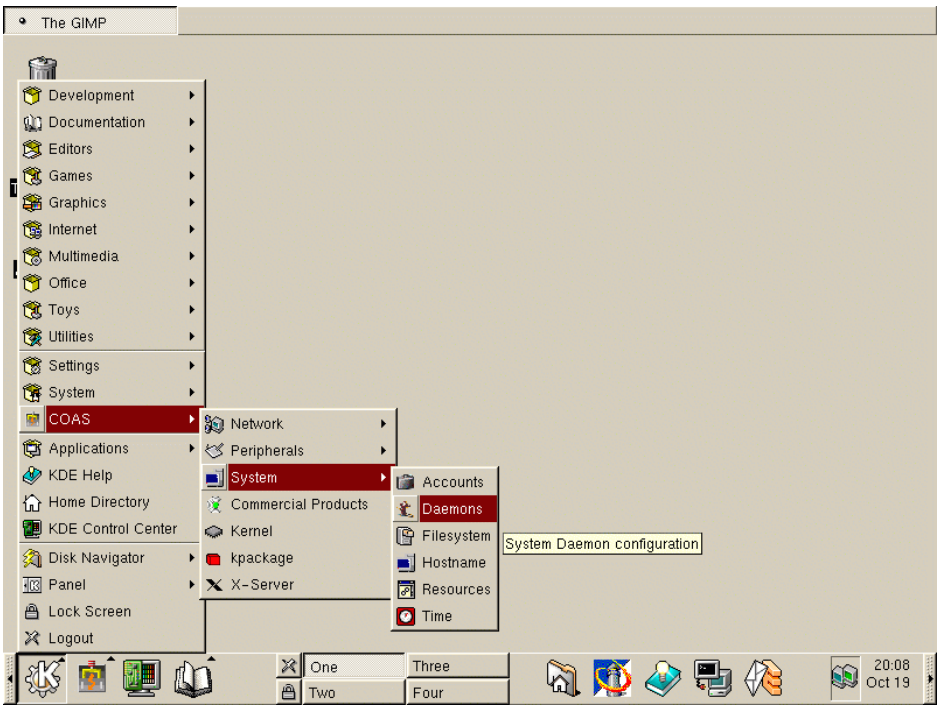


Figure 24. Starting the System Daemon configuration tool

After the System Daemon configuration tool is started you will see a welcome screen to the COAS administration tools. Click **OK** to continue. You will see a screen similar to Figure 25.

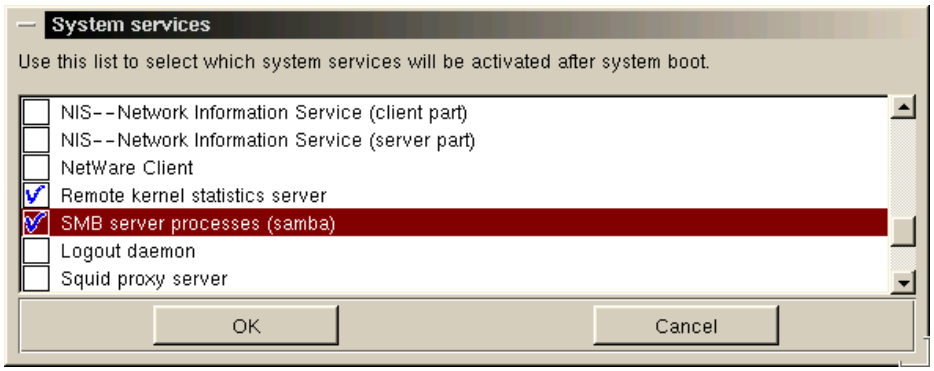


Figure 25. Selecting Samba to start as a boot process

Select **SMB server process (Samba)** on the list. Click **OK** to save your new settings.

When the Linux server is restarted, the Samba server will be started automatically.

3.5 Using SWAT

The Samba Web Administration Tool (SWAT) allows the remote configuration of the `smb.conf` configuration file through a Web browser. That means you can configurate Samba in a GUI-like environment. SWAT itself is a small Web server. A CGI scripting application, designed to run from `inetd`, provides access to the `smb.conf` configuration file.

An authorized user with the root password can configure the `smb.conf` configuration file via Web pages. SWAT also places help links to all configurable options on every page, which lets an administrator easily understand the effect of the changes.

Before using SWAT you must check the following.

1. In the file `/etc/services` you must have the following line:

   ```
   swat 901/tcp
   ```

2. In the file `/etc/inetd.conf` you must have the following line:

   ```
   swat stream tcp nowait.400 root /usr/sbin/tcpd swat
   ```

   As you can see, SWAT is started with a TCP wrapper, so you can control who can access the SWAT service with the `/etc/hosts.deny` file. For example if you want to access SWAT only localy your `/etc/hosts.deny` file should look similar to this:

   ```
   #
   # hosts.deny    This file describes the names of the hosts which are
   #               *not* allowed to use the local INET services, as decided
   #               by the '/usr/sbin/tcpd' server.
   #
   ```

```
# The portmap line is redundant, but it is left to remind you that
# the new secure portmap uses hosts.deny and hosts.allow.  In particular
# you should know that NFS uses portmap!
swat:ALL EXCEPT 127.0.0.1
```

If you made any modification to those two files you need to restart `inetd`. This can be done by executing the commands:

```
/etc/rc.d/init.d/init stop
/etc/rc.d/init.d/init start
```

If you did everything without errors you are ready to use SWAT. To start SWAT point your favorite Web browser to the Internet address of your Samba server on port 901, as you can see in Figure 26.

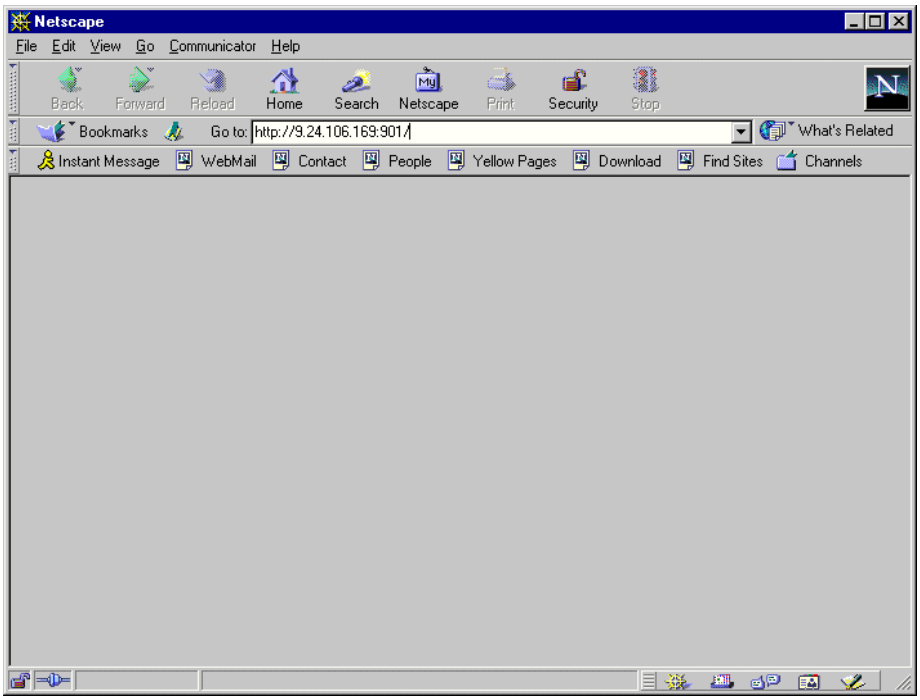


Figure 26. Starting SWAT

After you load the home page of SWAT, you will see a screen similar to Figure 27.

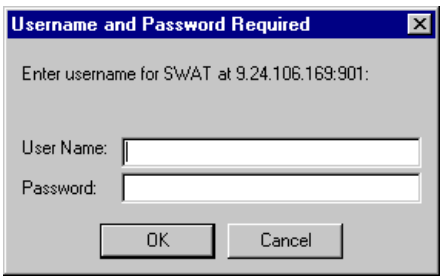


Figure 27. User authorization for SWAT

Type in the username and password of the Linux user defined on your Linux server. Click **OK** to continue. You will see a screen similar to Figure 28 on page 36.

Note

Any Linux user can access SWAT, but only a root user can make changes .

Remember, when you are logging on to SWAT from a remote machine you are sending passwords in plain text. This can be a security issue, so we recommend that you do SWAT administration locally only.

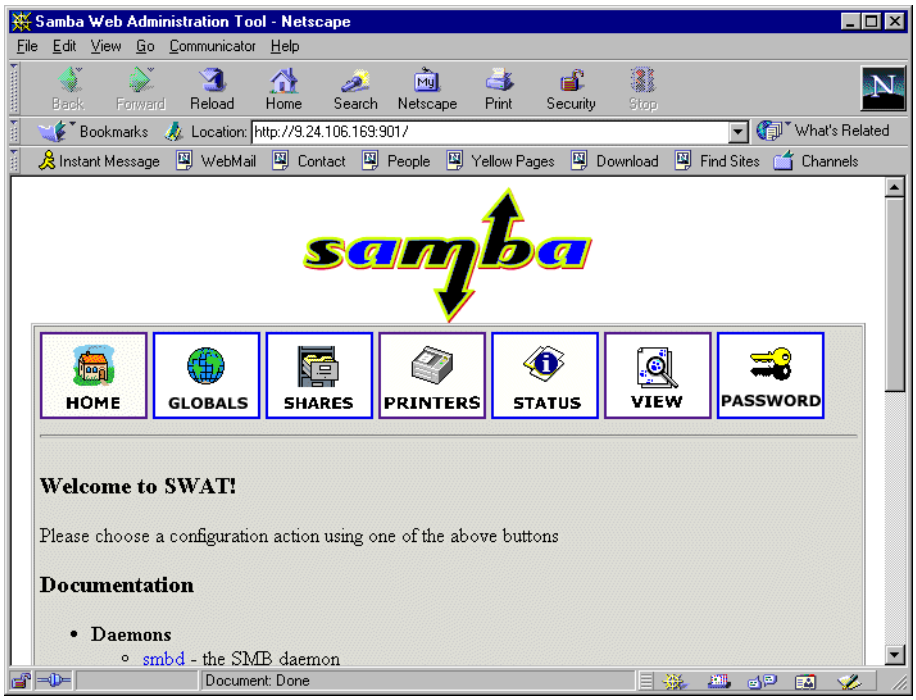


Figure 28. SWAT home page

As you can see in Figure 28, you have seven categories available:

1. Home - here you can find all the documentation you need about Samba.
2. Globals - here you can see and modify global parameters from the `smb.conf` configuration file.
3. Shares - here you can view, modify, and add shares.
4. Printers - here you can view, modify, and add printers.
5. Status - here you can check the current status of your Samba server.
6. View - here you can see the current configuration of the `smb.conf` configuration file.
7. Passwords - here you can manage passwords for the Samba server.

Now we will briefly describe the sections available in SWAT.

Note

You can reach any of the seven sections on all SWAT Web pages. There are always icons for the sections on the top of each page.

After you make changes to the `smb.conf` configuration file, the Samba server must be restarted.

3.5.0.1 Globals

When you click the **Globals** icon in the main SWAT window, you will see a window similar to Figure 29.

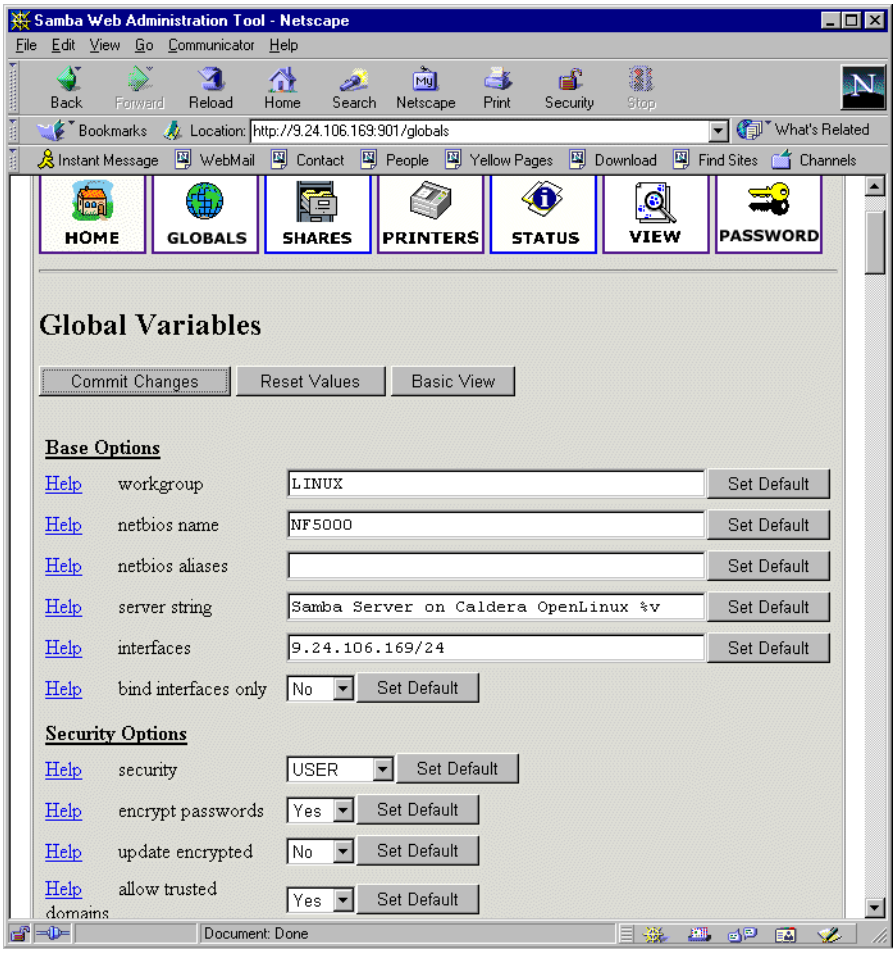


Figure 29. Global section in SWAT

In this window you can modify the global parameters for the Samba server. By default you will see the Basic View; if you want to see the Advanced View, click **Advanced View**. In the Advanced View you have all options available, while in the Basic View you can only change the basic options. To return from the Advanced View to the Basic View click **Basic View.** After you have made your changes you can save them by clicking **Commit changes.** If you get a pop-up window similar to Figure 30, which warns you that you are sending non-secure information over the network, you can easily select **Continue** if you are working locally or if you know that your network is secure.

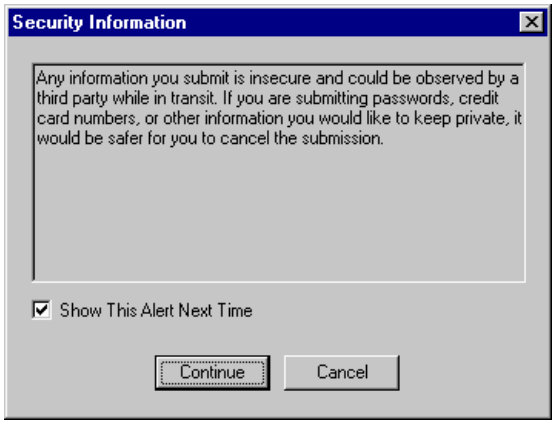


Figure 30. Security warning

3.6 Shares

When you click the **Shares** icon on any of the SWAT Web page, you will see a screen similar to Figure 31.

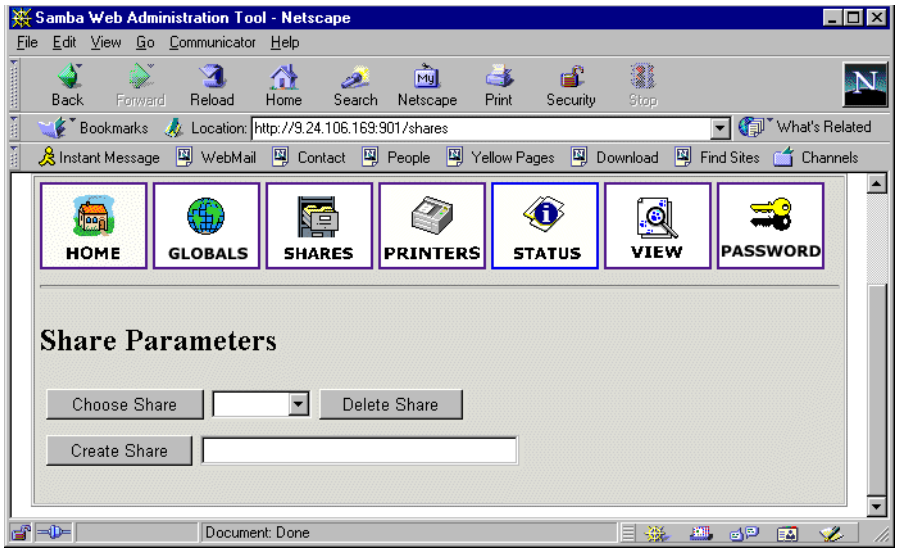


Figure 31. Shares section in SWAT

Here you can:

1. View the defined share
2. Delete share
3. Create a new share

3.7 Viewing or modifying an existing share

To view an already defined share select the share from the field to the right of the **Choose Share** button, as shown in Figure 32.

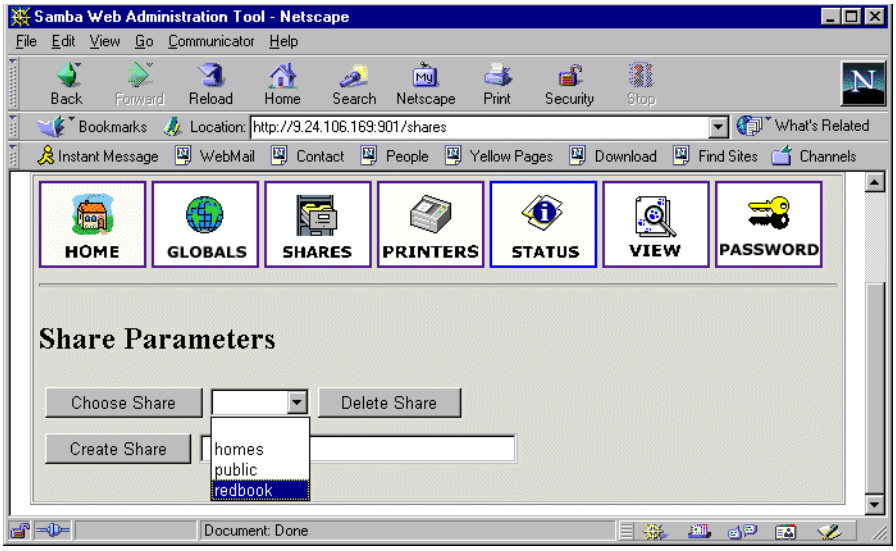


Figure 32. Choosing a share to view

After you have selected the share, click **Choose Share** to view the share properties. You will see a screen similar to Figure 33.

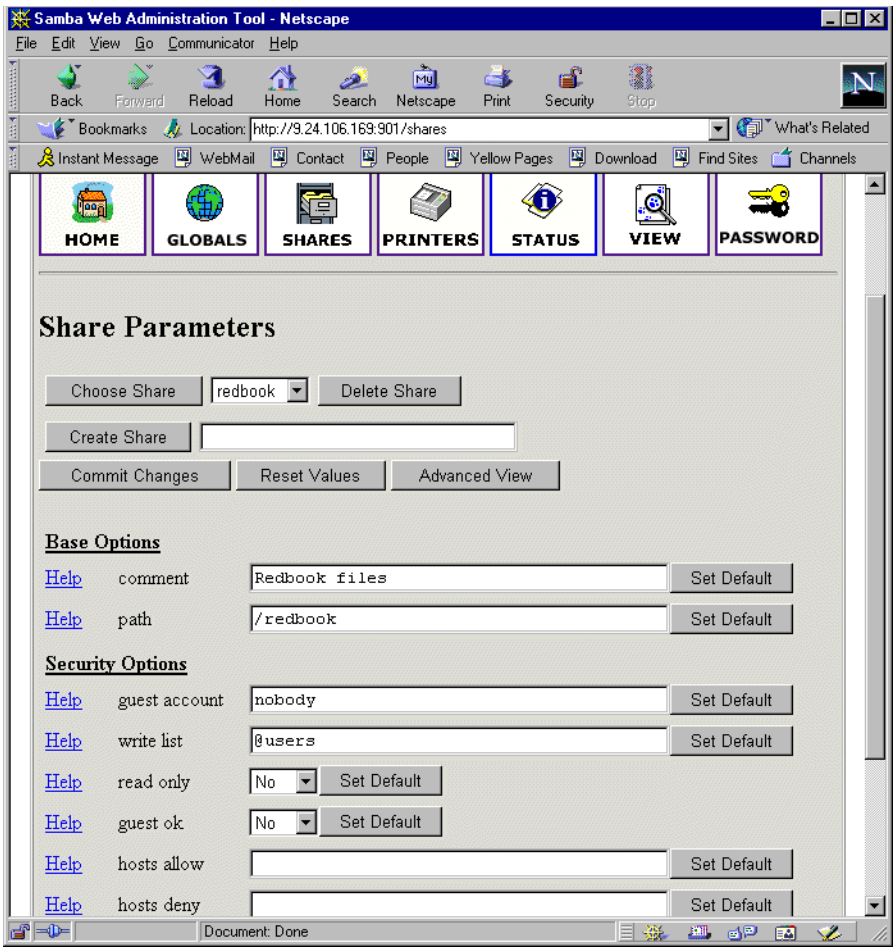


Figure 33. Share properties

If you want to see all available parameters, click the **Advanced View.** In this view you can also make changes and save them by clicking **Commit Changes.**

3.7.1 Deleting the existing share

To delete the existing share you must first select an already defined share similar to Figure 32. Then click **Delete Share.**

> **Attention**
>
> A share is deleted immediately and without warning.

After you have deleted the share, the Samba server must be restarted.

3.7.2 Creating a new share

In this section we will show how to create a simple share. To accomplish this follow these steps:

1. Create a directory that will be used for the share. You can do this by executing this command from the terminal:

   ```
   mkdir /home/public
   ```

   In our example we created a “public” directory in the “home” directory.

2. Make sure that the UNIX permissions are set correctly in that directory, so that only intended users have access rights to it.

3. In the shares view of the SWAT Web pages, type in the name of the share you are creating similar to Figure 34.

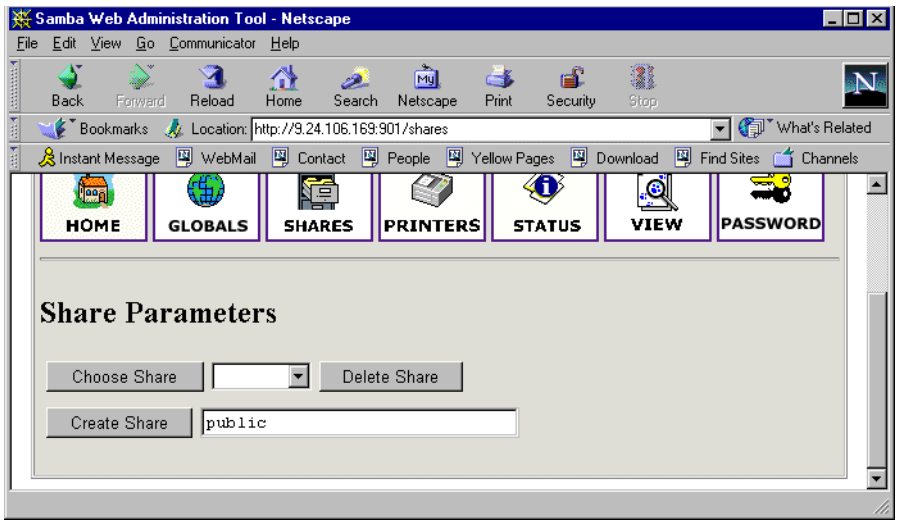


Figure 34. Entering the name for a new share

4. Click **Create Share** to continue. You will see a screen similar to Figure 35.

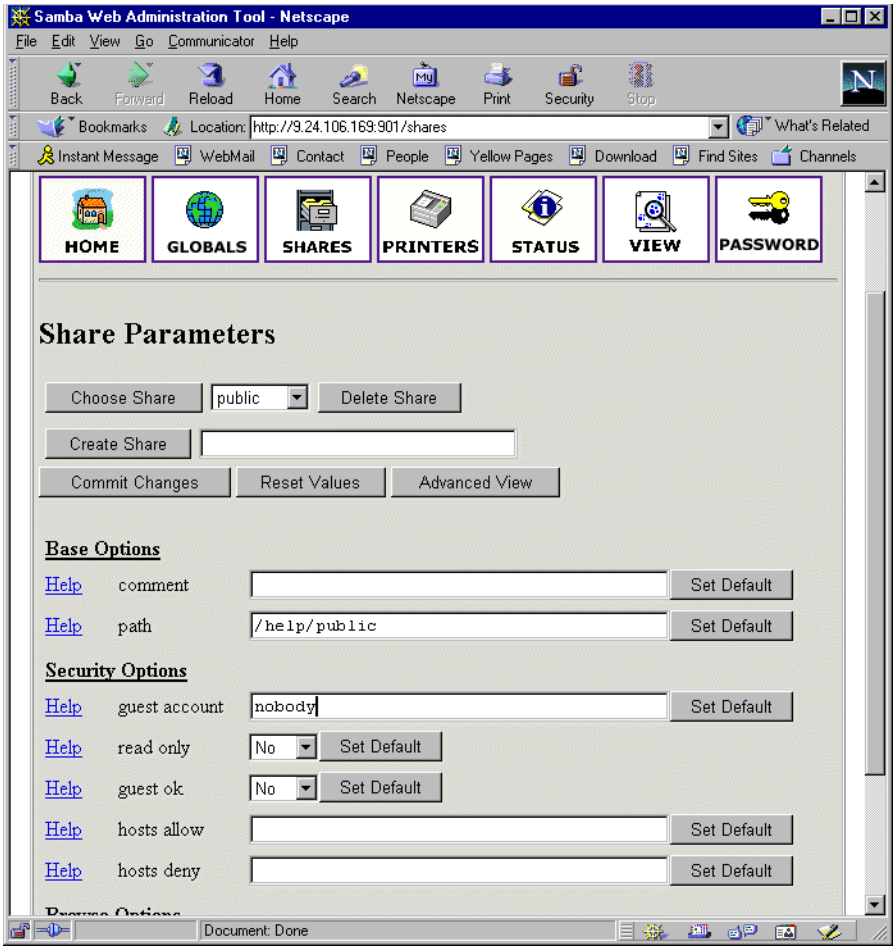


Figure 35. Entering the new share parameters

5. Fill in the needed parameters. If you need to set more advanced parameters, click **Advanced View** and you will see all available parameters. After you typed in all you want, click **Commit Changes** to save your new share.
6. You can see the changed `smb.conf` configuration file by selecting the **View** icon from the SWAT Web page. You will see a screen similar to Figure 36.

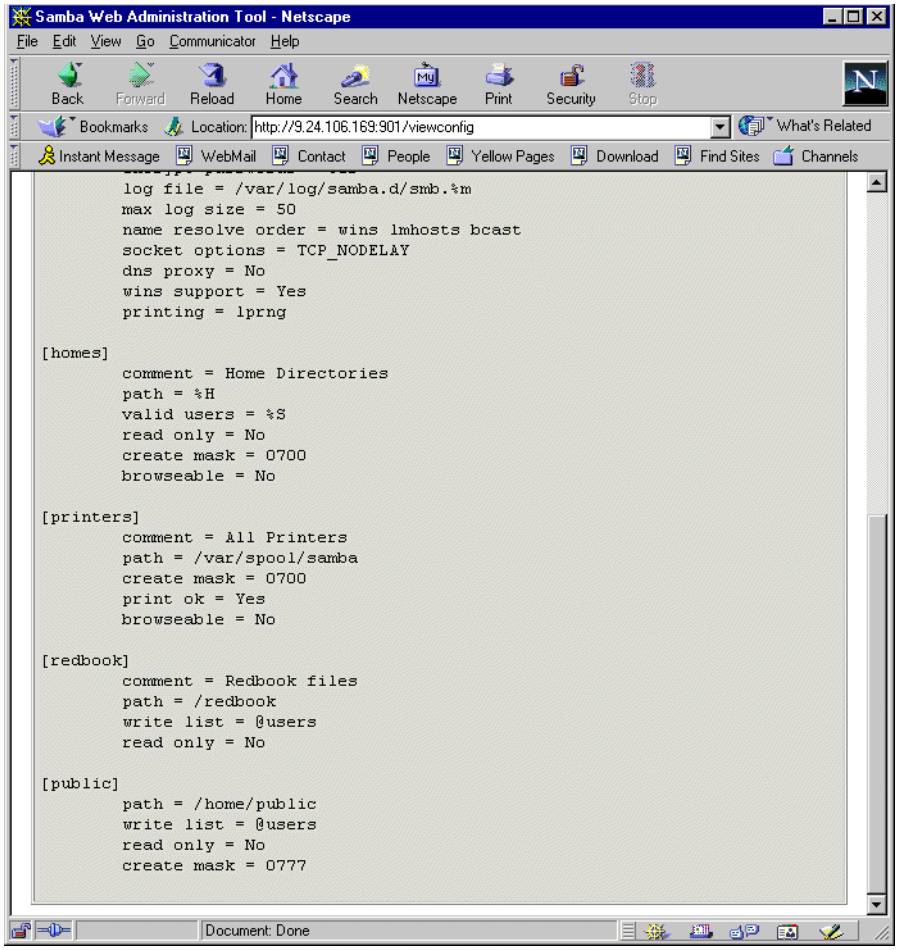


Figure 36. Viewing the smb.conf configuration file

7. Restart the Samba server.

Congratulations! You have just created your first usable share on the Samba server. Be friendly and share it with other users!

3.7.3 Restarting the Samba server

The Samba server can be restarted by clicking the **Start** icon on any of the SWAT Web pages. You will see a screen similar to Figure 37.

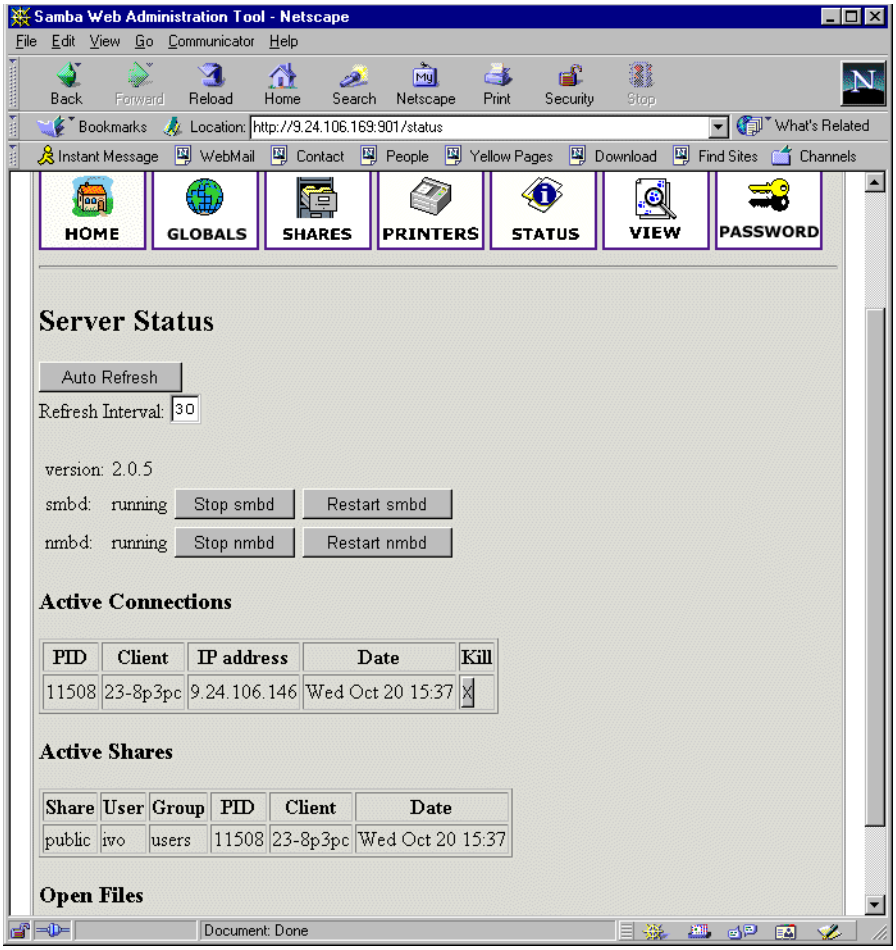


Figure 37. Restarting the Samba server

To restart the Samba server, simply click **Restart smbd.** On this page you can also restart the WINS server by clicking **Restart nmbd.**

3.7.4 Printers

In the printer section you can view, modify, or add printers. The operations for handling printers are the same as for handling shares. You can access the printer settings by clicking the **Printers** icon on the SWAT Web page similar to Figure 38.

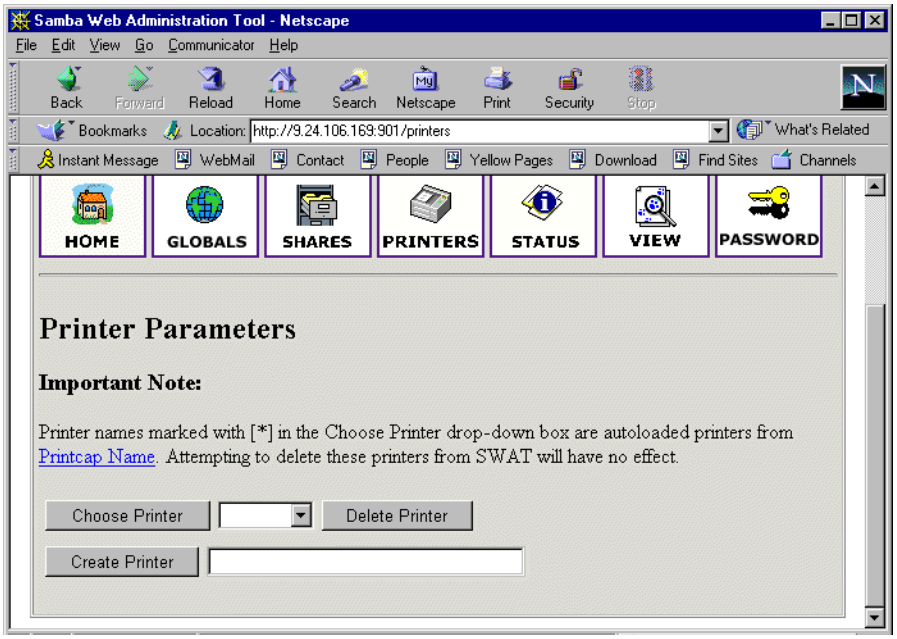


Figure 38. SWAT printers section

If you want to see the settings for a specific printer, select the printer from the list as shown in Figure 39.

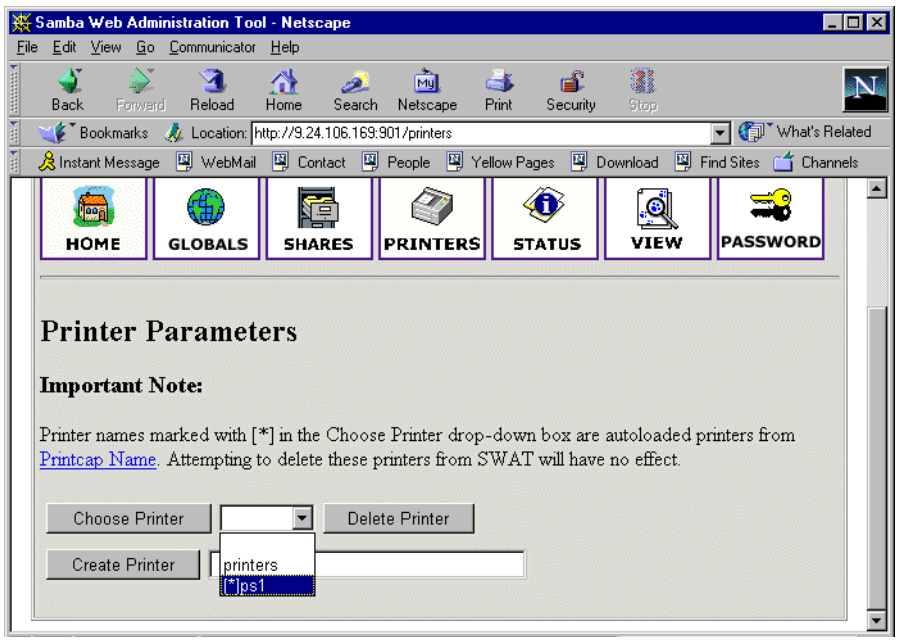


Figure 39. Selecting a printer

After you have selected the printer click **Choose Printer** to view its properties. You will see a screen similar to Figure 40.

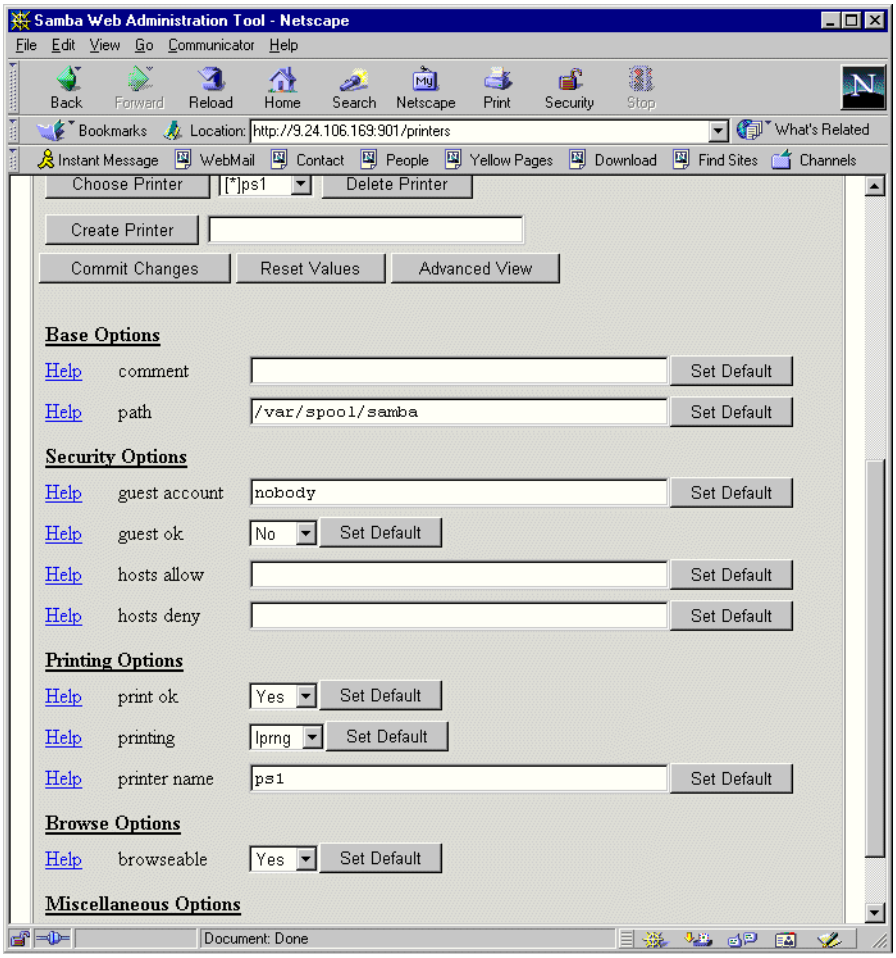


Figure 40. Printer properties

In this window you can also modify the printer properties. When you are done, save the settings by clicking **Commit Changes.**

3.7.5 Status

In this section you can check the status of the Samba server. Here you can see all the connections and open files. You can also start or restart the Samba server or just its components. You can access the printer settings by clicking the **Status** icon on the SWAT web page, as you can see in Figure 41.

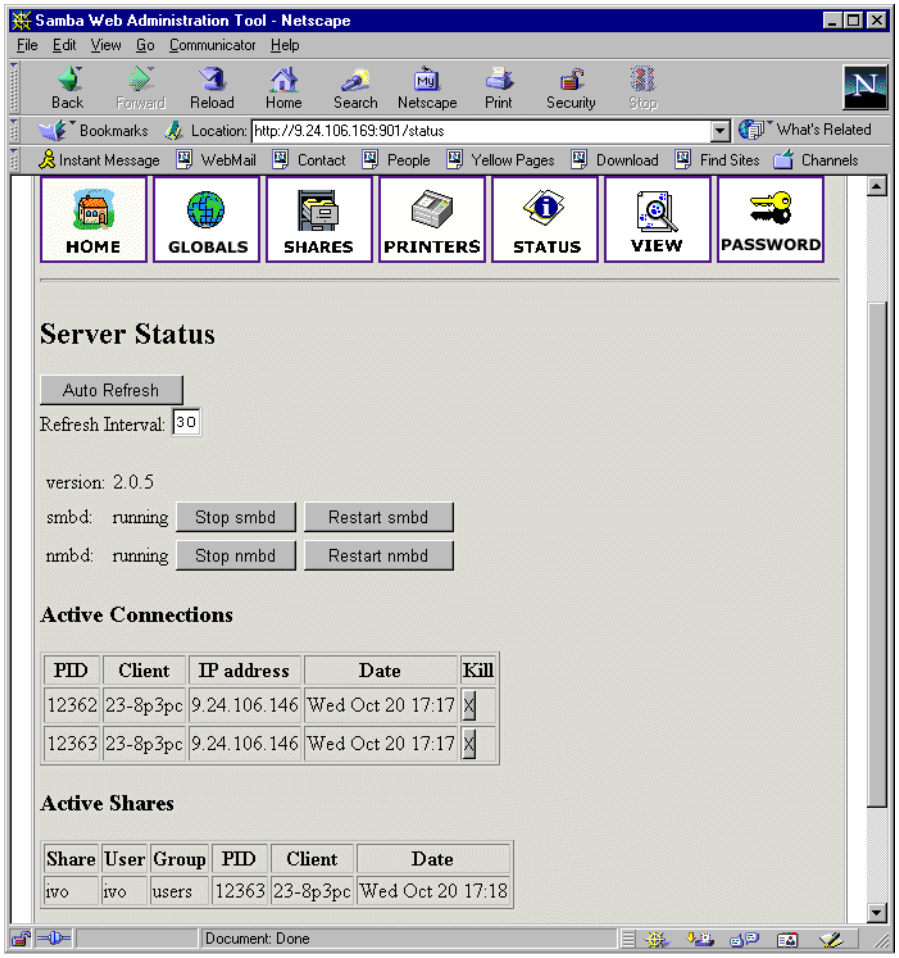


Figure 41. Status section

3.7.6 View

In this section you can see the current `smb.conf` configuration file. You can access printer settings by clicking the **View** icon on the SWAT Web page similar to Figure 42.

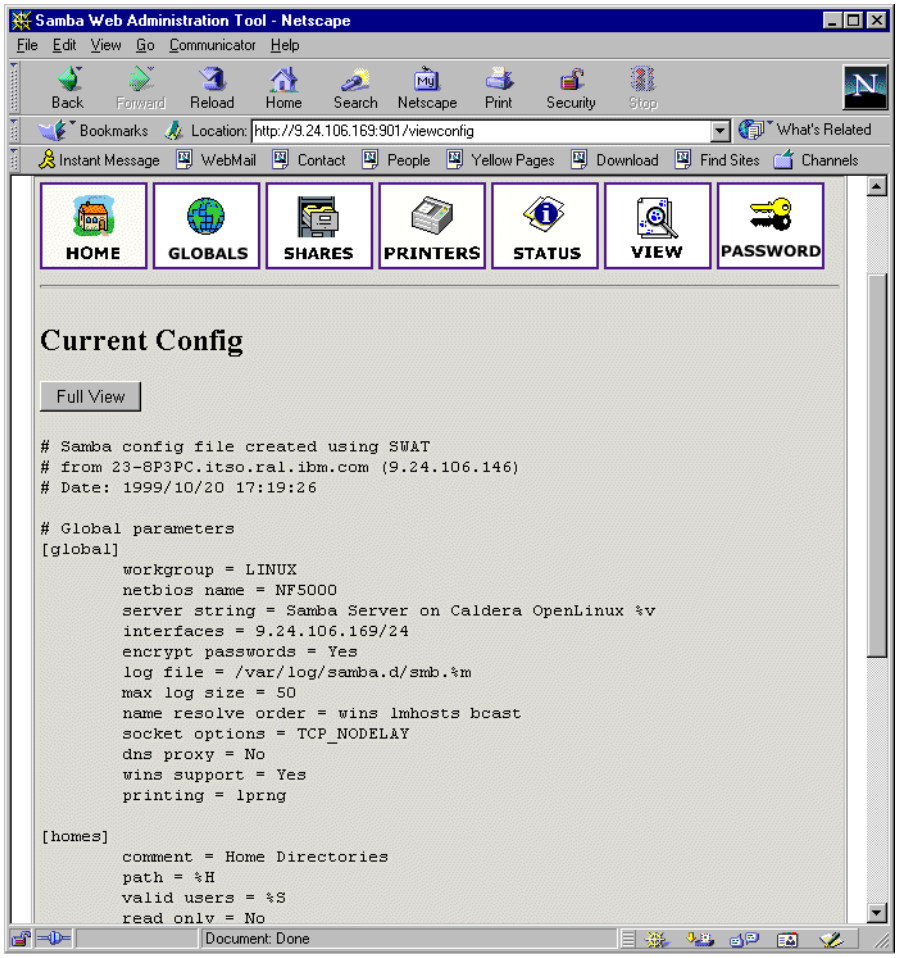


Figure 42. View section of SWAT

3.7.7 Password

In this section you can manage the passwords of all Samba users. You can access printer settings by clicking the **Password** icon on the SWAT Web page similar to Figure 43.

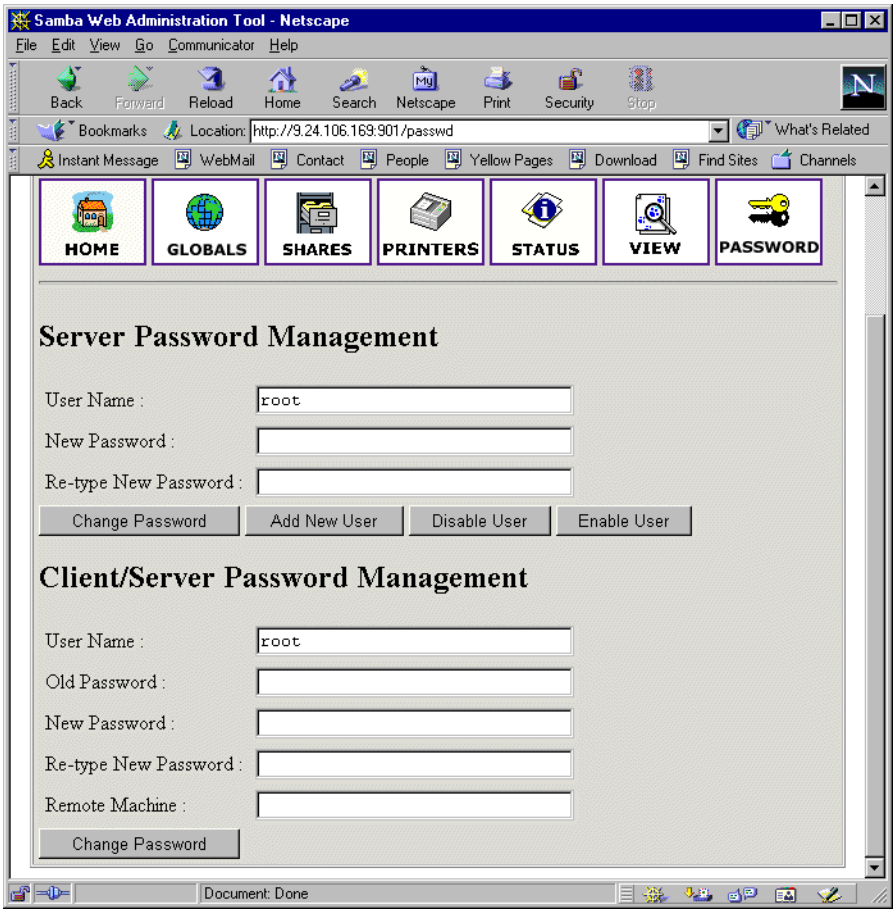


Figure 43. Managing passwords

3.8 Sources and additional information

You can find more information at the official Web site of the Samba project:

`http://www.samba.org`

And there are always good How-to documents at the Linux Documentation project Web site:

`http://www.linuxdoc.org/`

Chapter 4. Samba on Red Hat Linux

This chapter covers how to implement Samba on Red Hat Linux.

4.1 Installing Samba

To install the Samba packages:

Insert the Red Hat Linux CD into the CD-ROM drive.

```
mount /mnt/cdrom
rpm -ivh /mnt/cdrom/RedHat/RPMS/samba*
```

This will install the Samba server, client, and common packages.

4.2 Configuring Samba

In this section we will explain how to configure Samba so it can participate as a file and print server in an existing Windows network or be a stand-alone file and print server for Windows and Linux clients.

Before you can start using Samba, you will need to configure the `smb.conf` file. This file is the heart of the Samba server. When the Samba package is installed, a default configuration file is installed in:

```
/etc/smb.conf
```

The `smb.conf` file is divided into two main sections:

1. Global Settings - Used to define connection parameters.
2. Share Definitions - here you define shares. A share is a directory on the server that is accessible over the network and shared among users. This section has three subsections:
    1. Homes - in this subsection you define the user's home directories.
    2. Printers - in this subsection you define the available printers.
    3. Shares - this subsection you can have an entry for each share you would like to define.

In the following sections we will describe how to modify `smb.conf` to efficiently and simply use Samba as a file and print server. We will explain only the most basic parameters. If you need more information, look at the manual page for `smb.conf` or at the Web site for the Samba project:

```
http://www.samba.org
```

4.2.1 Setting the NetBIOS parameters

The `smb.conf` file begins with global settings for setting the NetBIOS parameters of the Samba server:

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

[global]
   netbios name = NF5000
```

```
workgroup = LINUX
server string = Samba Server
```

The parameters are described in Table 15.

Table 15. NetBIOS parameters

Parameter	Description
`netbios name`	This is the name the Samba server is known by on the network. This parameter has the same meaning as a Windows NT computer name. If you do not specify, it defaults to the server's hostname.
`workgroup`	This parameter specifies in which Windows NT workgroup the Samba server will participate. It is equivalent to Windows NT domain or workgroup name.
`server string`	This is the description string of the Samba server. It has the same role as the Windows NT description field.

4.2.2 Global printing settings

After the global settings for the NetBIOS parameters of the Samba server, the global printer parameters follows.

```
load printers = yes
printcap name = /etc/printcap
printing = lprng
```

The parameters are described in Figure 16.

Table 16. Printing parameters

Parameter	Description
`load printers`	This parameter controls if Samba loads all printers in the printcap file for browsing.
`printcap name`	With this parameter, you tell Samba the location of `printcap` file. The default value is `/etc/printcap`.
`printing`	This parameter tells Samba what printing style to use on your server. By default, Samba uses the LPRNG printing style.

4.2.3 Global security settings

Following the global settings for the printer parameters, you will find the global security parameters.:

```
security = user
;   password server = <NT-Server-Name>
encrypt passwords = yes
smb passwd file = /etc/samba.d/smbpasswd
```

The parameters are described in Table 17.

Table 17. Security parameters

Parameter	Description
`security`	This parameter has four possible values: `share, user, server, domain`
`password server`	At the `server` or `domain` security level this server is used for authorization. For the parameter value, you use the server's NetBIOS name.
`encrypt passwords`	Setting this parameter to yes will enable Samba to use the encrypted password protocol. This is used in the Windows NT Service Pack 3 and in Windows 98 and is required to communicate with those clients.
`smb passwd file`	This parameter tells Samba where the encrypted password file is.

We will briefly explain each security mode:

1. Share - for this security mode, clients need to supply only the password for the resource. This mode of security is the default for the Windows 95 file and print servers. It is not recommended for use in a UNIX environment as it violates the UNIX security model.
2. User - user/password validation is done on the server that is offering the resource. This mode is the most widely used.
3. Server - the user/password validation is done on the specified authentication server. This server can be a Windows NT server or another Samba server.
4. Domain - this security level is basically the same as server security, with the exception that the Samba server becomes a member of a Windows NT domain. In this case, the Samba server can also participate in such things as trust relationships.

Windows NT 4 Service Pack 3 or later, Windows 95 with the latest patches, and Windows 98 use encrypted passwords for accessing NetBIOS resources. We will need to enable the Samba server to use encrypted passwords. Before you start the Samba server for the first time, you need to create a Samba encrypted passwords file. To add all the current users to the Samba server, run the `mksmbpasswd.sh` utility:

```
cat /etc/passwd | /usr/bin/mksmbpasswd.sh > /etc/smbpasswd
```

We have now created the Samba password file with all of the current users. Although the users have been added, you still need to update the passwords individually by:

```
/usr/bin/smbpasswd -U username
```

To add a user after running the `mksmbpasswd` utility, use:

```
/usr/bin/smbpasswd -a username password
```

Note

Use the same filename you specified for creating the Samba password file in the `smb.conf` configuration to tell the Samba server where the password file is.

You have to be logged in as root if you want to manage other users.

4.2.4 Global name resolution settings

Following the global settings for security parameters, you will finde the global name resolution parameters.

```
name resolve order = wins lmhosts bcast
wins support = yes
;   wins server = w.x.y.z
```

The parameters are described in Table 18.

Table 18. Name resolution parameters

Parameter	Description
`name resolve order`	With this parameter you specify how the Samba server will resolve NetBIOS names into IP addresses. The preferred value is `wins lmhosts bcast`.Refer to the manual page of `smb.conf` for more information.
`wins support`	If this option is enabled the Samba server will also act as a WINS server.
`wins server`	With this parameter, you tell Samba which WINS server to use.

Note

Samba can be a WINS server or a WINS client, but not both. So only one of the `WINS support` and `WINS server` parameters can be set at a time. If you specify the IP address of a WINS server then `wins support` must be set to “no”.

4.2.5 Creating shares

In the previous sections we explained how to prepare global configuration parameters. In this section we will explain how to create shares. A simple share can be defined in the `smb.conf` file as follows:

```
[redbook]
   comment = Redbook files
   path = /redbook
   browseable = yes
   printable = no
   writable = yes
   write list = @users
```

In Table 19 we explain some of the parameters for creating shares.

Table 19. Share parameters

Parameter	Description
`comment`	This describes the function of a share.
`admin users`	This parameter is used to specify the users who have administrative privileges for the share. When they access the share, they perform all operations as `root`.
`path`	Defines the full path to the directory you are sharing.
`browseable`	If this parameter is set to yes, you can see the share when you are browsing the resources on the Samba server. The value can be yes or no.
`printable`	This parameter is used to specify if the share is a print share. The value can be yes or no.
`write list`	Users specified in this list have write access to the share. If the name begins with @ it indicates a group name.
`writable`	This parameter specifies if the share is writable. The value can be yes or no.
`read list`	Users specified in this list have read access to the share. If the name begins with @ it indicates a group name.
`read only`	If this is set to yes, the share is read only. The value can be yes or no.
`valid users`	This parameter specifies which users can access the share.

Using these parameters will allow you to easily set up a new share. Each share definition starts with the share name in brackets "[]". Following the share name you specify the share parameters and values.

4.2.6 Share permissions

Although you can control the share permissions with share parameters, UNIX permissions are applied before share permissions. Make sure the UNIX permissions let the users access the share directory in the UNIX environment.

When a user creates a new file on the shared directory, the default create mask used is 0744. For directory creation, the default create mask is 0755. You can force a different creation mask by using the parameters explained in Table 20.

Table 20. Create mask parameters

Parameter	Description
`create mask`	This is used for file creation to mask against the UNIX mask calculated from the DOS mode.
`directory mask`	This is used for directory creation to mask against the UNIX mask calculated from the DOS mode.

4.2.7 Creating shares for home directories

For handling home directories `smb.conf` has a special share section called [homes]. This share definition is used for all home directories, so we don't need to create separate shares for each user.

When a client requests a connection to a share, existing shares are scanned. If a match is found, that share is used. If no match is found, the requested share is treated as the username and validated by security. If the name exists and the password is correct, a share with that name is created by cloning the [homes] section. Home share definitions use the same parameters as normal shares. The following is an example of a [homes] share definition in the `smb.conf` file:

```
[homes]
   comment = Home Directories
   path = %H
   valid users = %S
   browseable = no
   writable = yes
   create mode = 0700
   directory mode = 0700
```

Table 21 explains the use of certain variables in the [homes] share definition.

Table 21. Variable description

Parameter	Description
%H	This variable represents the home directory of the user.
%S	The name of the current service, which is in the case of [home] shares equal to username.

In this example we used creation masks for files and directories so that all new files and directories are accessible only by the owner of that home directory.

4.2.8 Creating a printer share

A Samba server uses the same procedure for [printer] shares as for [home] shares. The share definitions and user names are tested against the requested share name. If a match is found in the [printers] share section, a share with that name is cloned with the name of the requested service. The following is an example of a [printers] definition in the `smb.conf` file:

```
[printers]
   comment = All Printers
   path = /var/spool/samba
   browseable = no
   # Set public = yes to allow user 'guest account' to print
   guest ok = no
   writable = no
   printable = yes
   create mask = 0700
```

The [printers] section is just like the other share definitions.When a user prints, Samba copies the data into the spool directory, after which it is handled by the

local printing system. The only big difference between a printer share and other share definitions is that the `printable` parameter is set to “yes”. This means that a user can write a spool file to the directory specified under the share definition. If the share is printable, then it is also writable by default.

4.2.9 Starting and stopping the Samba server

You can start the Samba server by executing the following command:

```
/etc/rc.d/init.d/smb start
```

You will see:

```
Starting SMB services:
Starting NMB services:
```

Two daemons are started: `smbd` and `nmbd`. `smbd` is the Samba server and `nmbd` is the WINS server.

The Samba server can be stopped by executing the command:

```
/etc/rc.d/init.d/smb stop
```

Whenever you modify the `smb.conf` configuration file you must restart the Samba server.

4.3 Using SWAT

the Samba Web Administration Tool (SWAT) allows configuration of the `smb.conf` configuration file through a Web interface. This means you can make configuration changes in a GUI-like environment. SWAT itself is a small Web server and CGI scripting application designed to run from `inetd`, which provides access to the `smb.conf` configuration file.

Authorized users can configure the `smb.conf` file via a Web interface. SWAT also has help links to the options on every page.

Before using SWAT you must check for the following.

1. In the file `/etc/services` you must have the following line:

   ```
   swat 901/tcp
   ```

2. In the file `/etc/inetd.conf` you must have the following line:

   ```
   swat stream tcp nowait.400 root /usr/sbin/tcpd swat
   ```

   If you modify either file you will need to restart inetd. You can do this with the following command:

   ```
   /etc/rc.d/init.d/inet reload
   ```

   SWAT is started via a TCP wrapper so you can control who can access SWAT with the `/etc/hosts.deny` file. For example, if you want access to SWAT granted locally only, your `/etc/hosts.deny` file would be:

   ```
   #
   # hosts.deny    This file describes the names of the hosts which are
   #               *not* allowed to use the local INET services, as decided
   ```

```
#               by the '/usr/sbin/tcpd' server.
#
# The portmap line is redundant, but it is left to remind you that
# the new secure portmap uses hosts.deny and hosts.allow.  In particular
# you should know that NFS uses portmap!
swat:ALL EXCEPT 127.0.0.1
```

If you did everything without errors you are ready to use SWAT. To start SWAT point your favorite Web browser to the Internet address of your Samba server on port 901, as you can see in Figure 44.

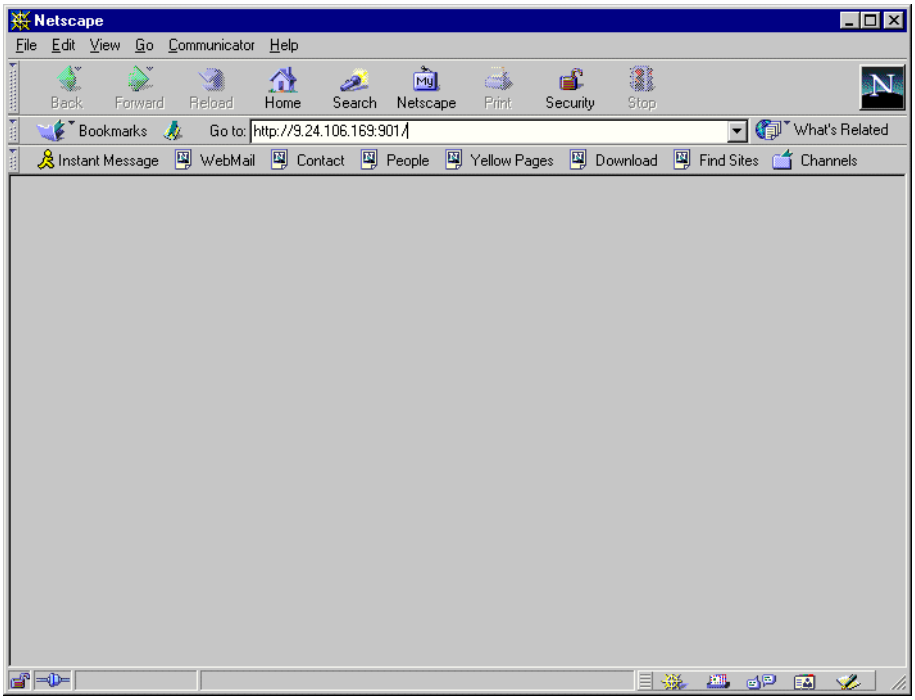


Figure 44. Starting SWAT

After entering the IP address for SWAT, you will see a screen similar to Figure 45.

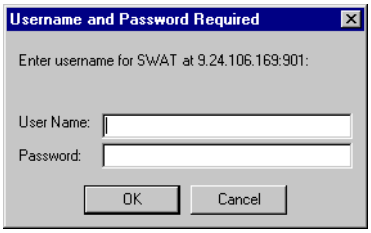


Figure 45. User authorization for SWAT

Type in the username and password of the Linux user defined on your Linux server. Click **OK** to continue.

Note

You can access SWAT with any Linux user, but you can make changes only via root.

Remember, when you are logging on to SWAT from a remote machine you are sending passwords in plain text. This can be a security issue, so we recommend that you do SWAT administration locally only.

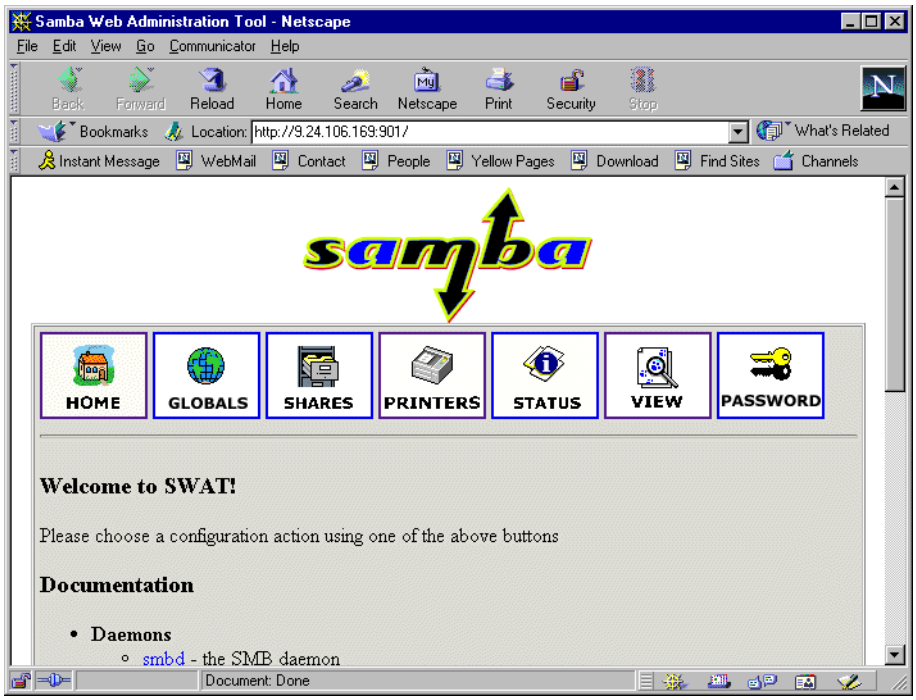


Figure 46. SWAT home page

In Figure 46 we have seven categories available:

1. Home - here you can find all the documentation you will need about Samba.
2. Globals - here you can view and modify global parameters from the `smb.conf` file.
3. Shares - here you can view, modify, and add shares.
4. Printers - here you can view, modify, and add printers.
5. Status - here you can check the current status of your Samba server.
6. View - here you can view the current configuration of the `smb.conf` file.
7. Passwords - here you can manage passwords for the Samba server.

In the following we briefly describe the sections available in SWAT.

Note

If you make any changes to the `smb.conf` file, the Samba server must be restarted.

4.3.1 Globals

When you click the **Globals** icon in the main SWAT window, you will see a window similar to Figure 47.

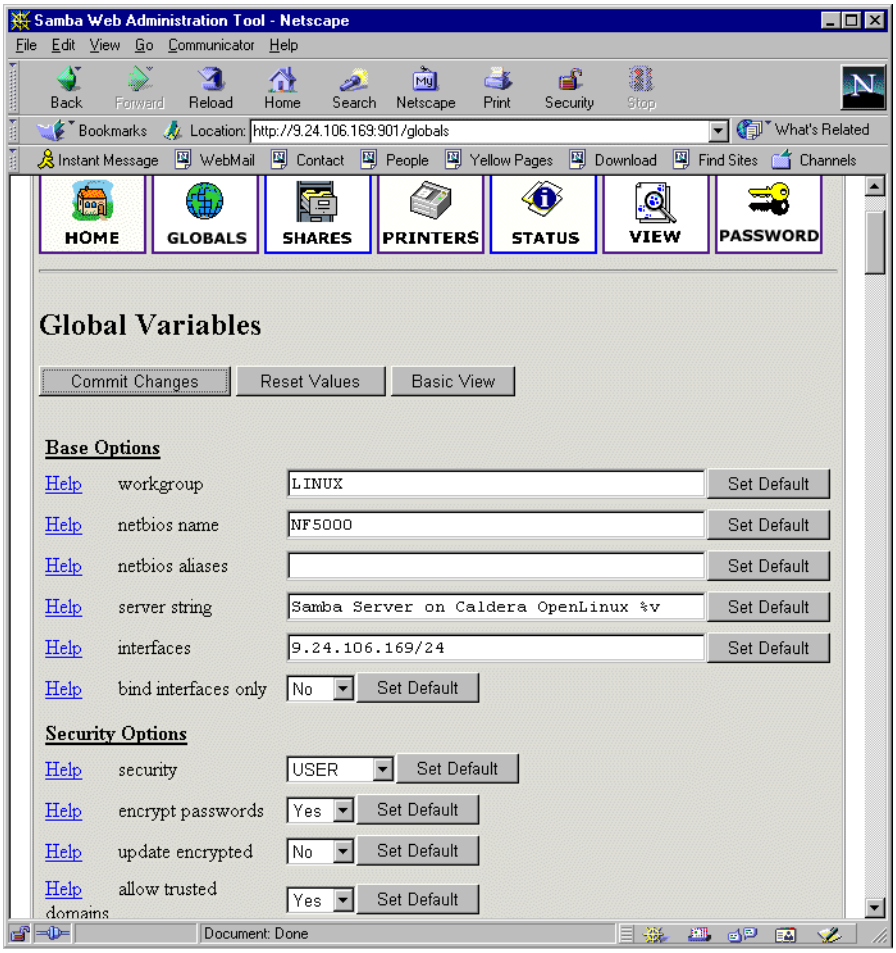


Figure 47. Global section in SWAT

In this window you can modify global parameters for the Samba server. By default you will see the Basic View; if you want to see the Advanced View click **Advanced View**. In the Advanced View you have all of the options available, while in the Basic View you can only change the basic options. To return from the Advanced View to Basic View, click **Basic View.** After you made your changes you can save them by clicking **Commit changes.** If you get a pop-up window like Figure 48, which warns you that you are sending non-secure information over the network, select **Continue** *only* if you are working locally or if you know that your network is secure.

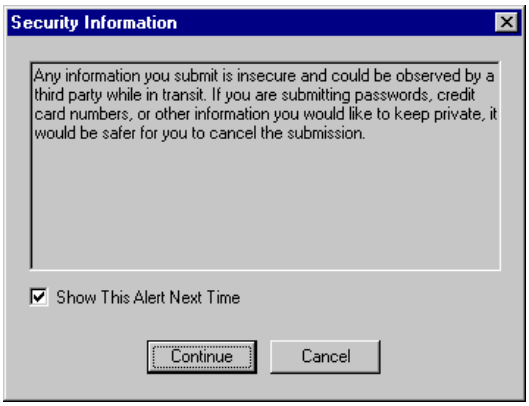


Figure 48. Security warning

4.3.2 Shares

When you click the **Shares** icon on any SWAT Web page, you will see the screen in Figure 49.

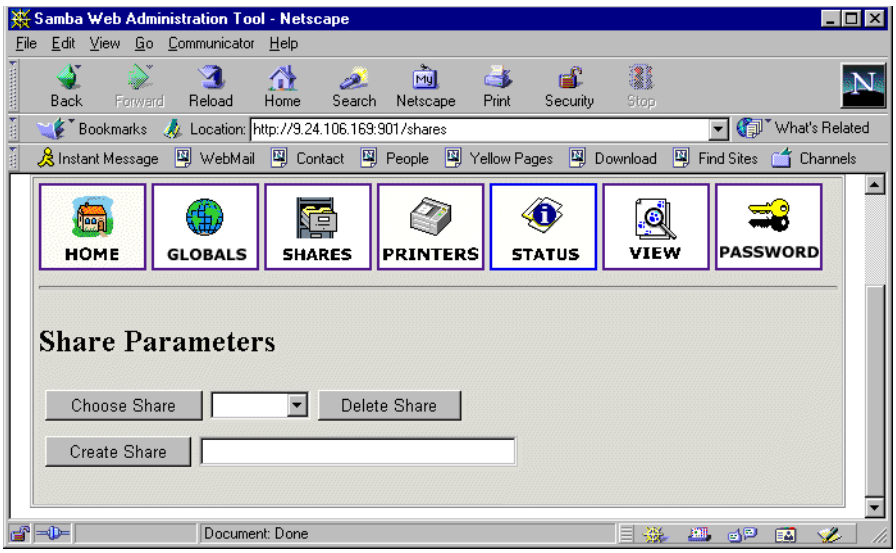


Figure 49. Shares section in SWAT

Here you can:

1. View the defined share
2. Delete share
3. Create a new share

4.3.3 Viewing or modifying an existing share

To view a share, select the share from the drop-down menu (Figure 50) and click the **Choose Share** button (Figure 51).

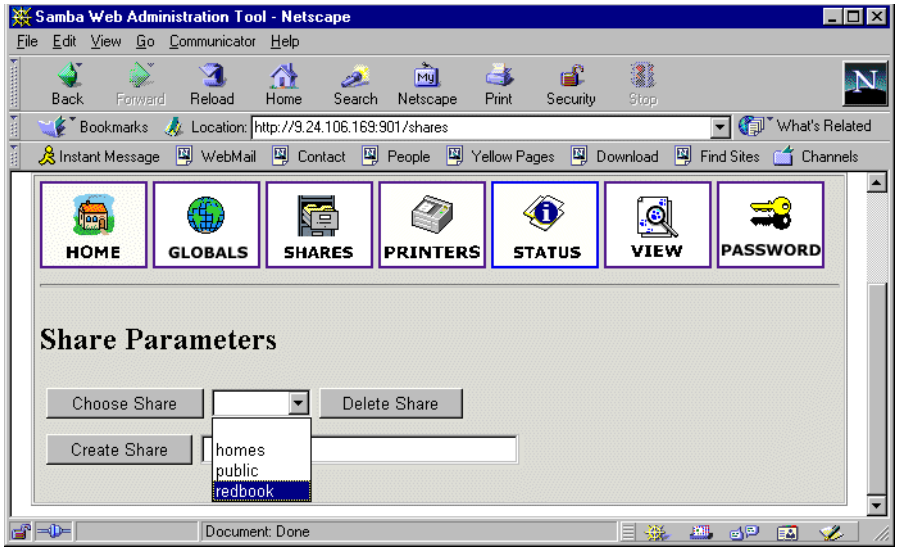


Figure 50. Choosing a share to view

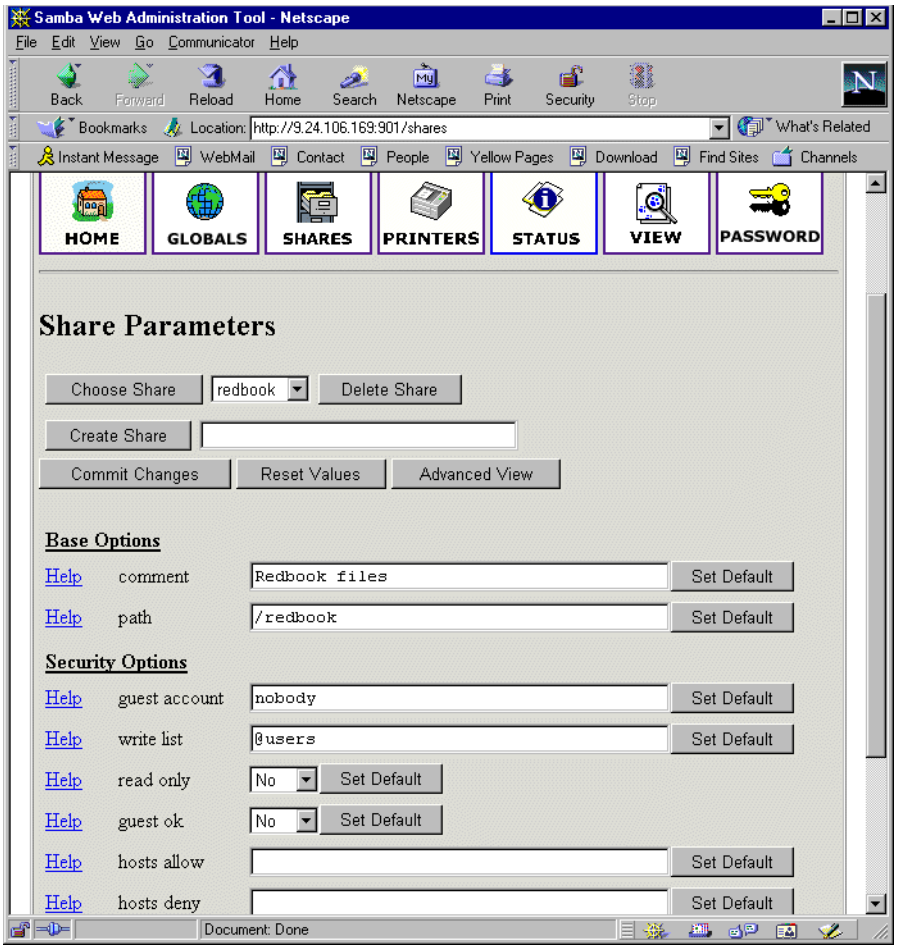


Figure 51. Share properties

If you want to see all available parameters click **Advanced View.** In this view you can also make changes and you can save them by clicking the **Commit Changes** button.

4.3.4 Deleting an existing share

To delete an existing share you must first select the share similar to Figure 50 on page 60. Then you click on **Delete Share.**

Attention

The share is deleted immediately and without warning.

After you have deleted a share, the Samba server must be restarted.

4.3.5 Creating a new share

In this section we will show how to create a new share:

1. Create the directory that will be used for the share. You can do this by executing the command:

   ```
   mkdir /home/public
   ```

   In our example we created the “public” directory in the “home” directory.

2. Make sure that UNIX permissions are set correctly in the directory so that only intended users have access rights to it.

3. In the shares view of the SWAT Web pages, type in the name of the share you are creating (Figure 52).

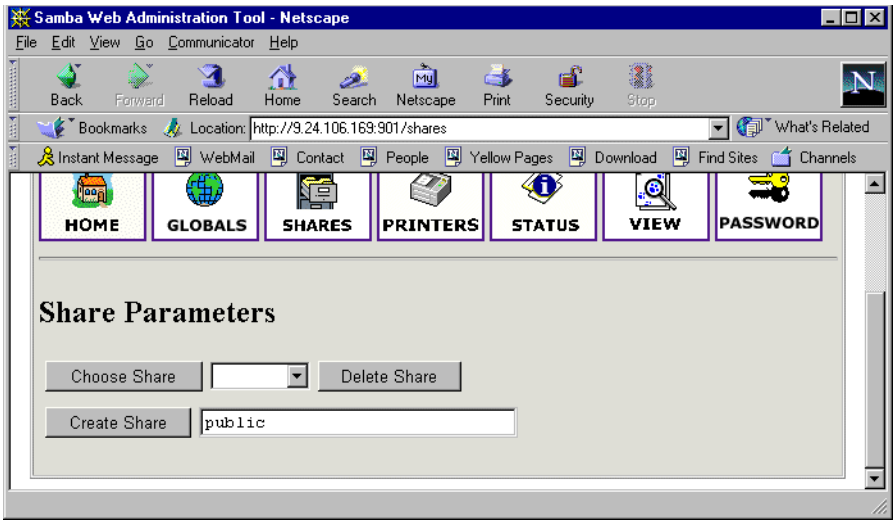


Figure 52. Entering the name for a new share

4. Click **Create Share** to continue, you will see a screen similar to Figure 53.

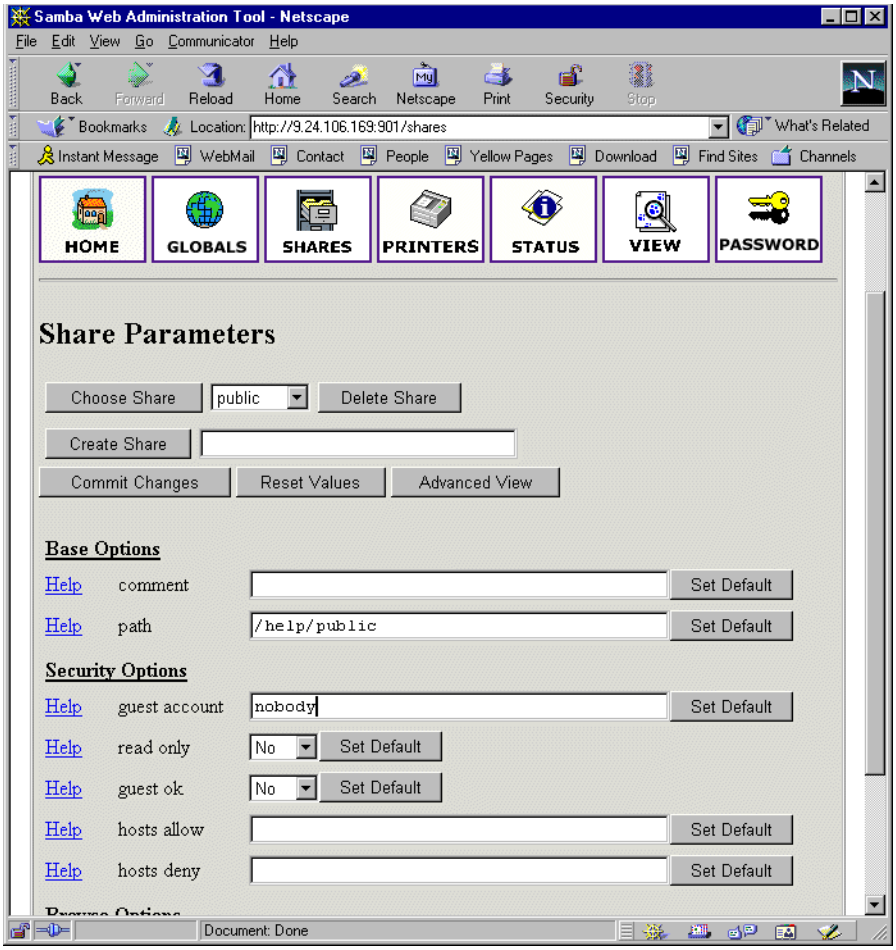


Figure 53. Entering the new share parameters

5. Fill in the needed parameters. If you need to set more advanced parameters, click the **Advanced View** button. Click **Commit Changes** when you are done to save the new share.
6. You can see the updated `smb.conf` file by selecting **View** from SWAT web pages (Figure 54).

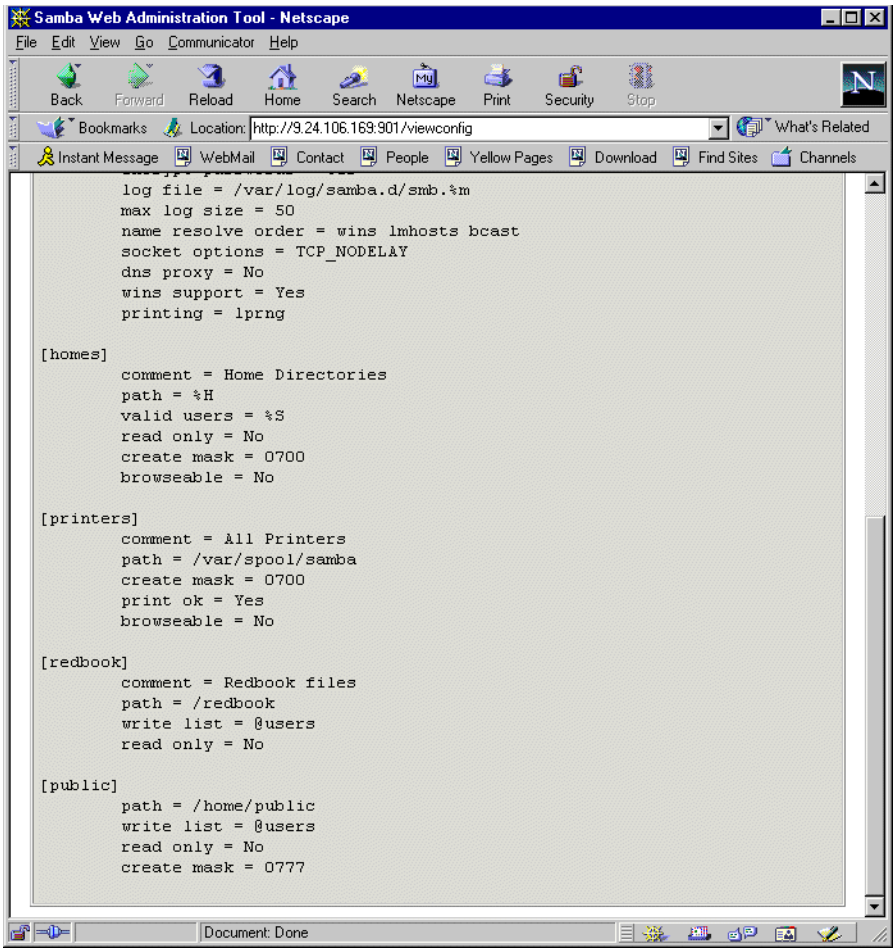


Figure 54. Viewing the smb.conf configuration file

7. Restart the Samba server.

Congratulations! You have just created your first share on the Samba server. Be friendly and share it with others!

4.3.6 Restarting the Samba server

To restart the Samba server, click the **Status icon** (Figure 55). Click **Restart smbd.** On this page you can also restart the WINS server by clicking **Restart nmbd.**

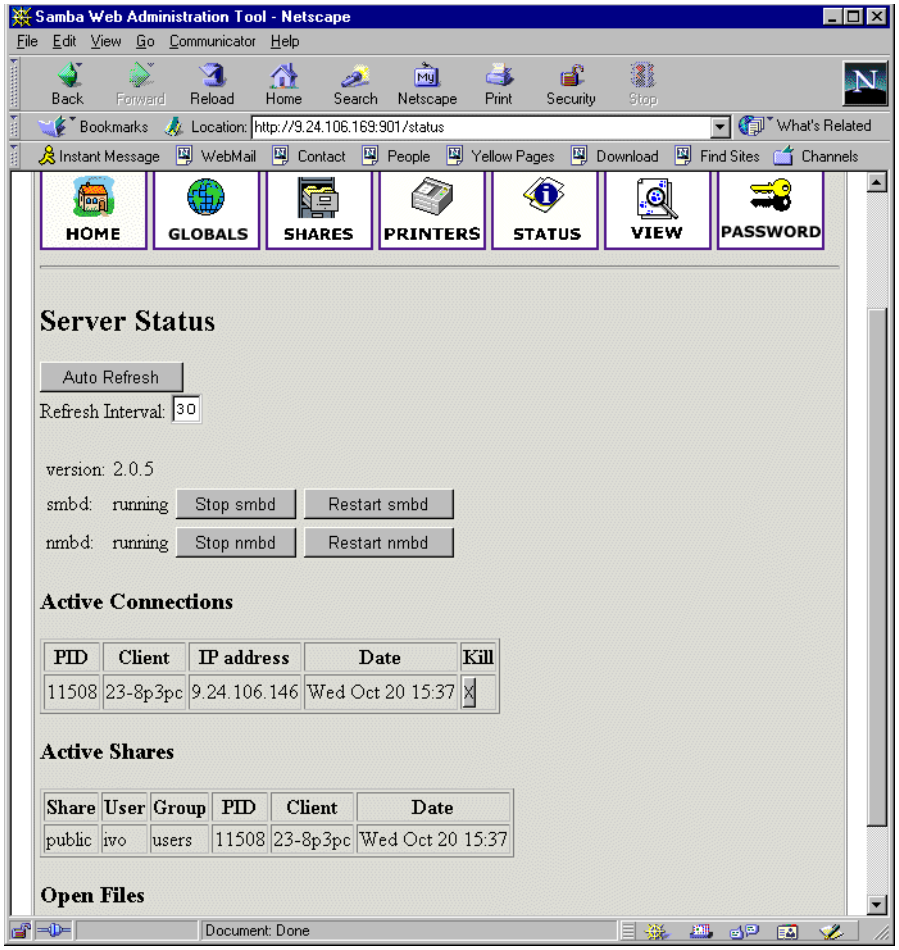


Figure 55. Restarting the Samba server

4.3.6.1 Printers

In the printer section you can view, modify, or add printers. The operations for handling printers are the same as for handling shares. You can access printer settings by clicking the **Printers** icon on the SWAT Web page (Figure 56).

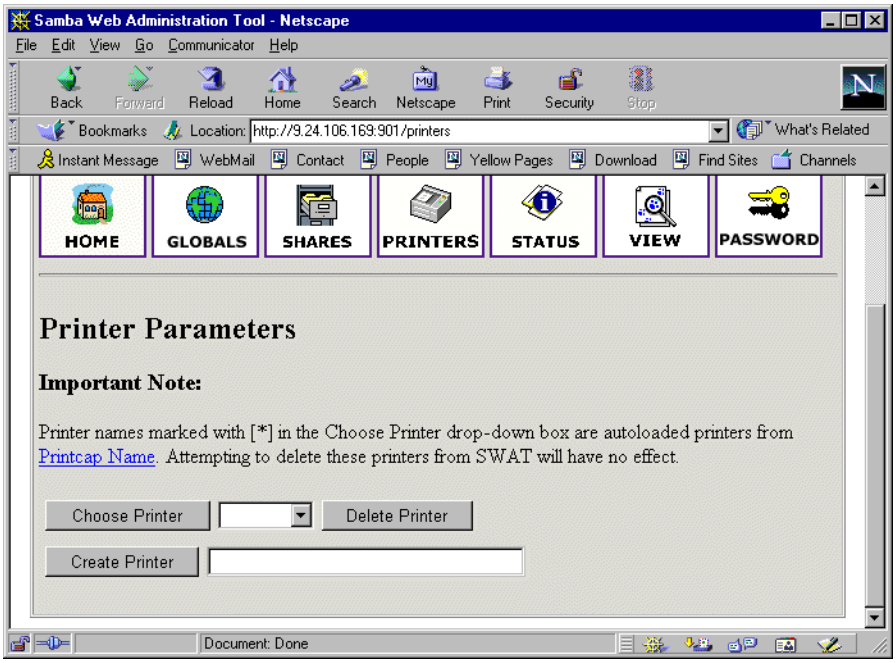


Figure 56. SWAT printers section

If you wish to see the settings for a specific printer, select the printer from the drop-down menu as shown in Figure 57.

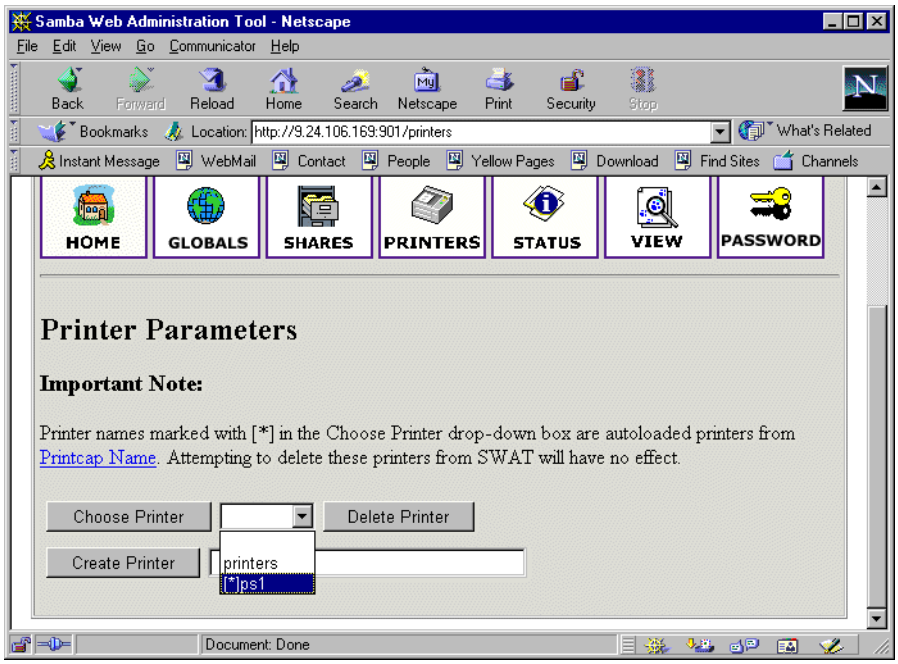


Figure 57. Selecting a printer

After selecting the printer, click the **Choose Printer** button to view the printer's properties (Figure 58).

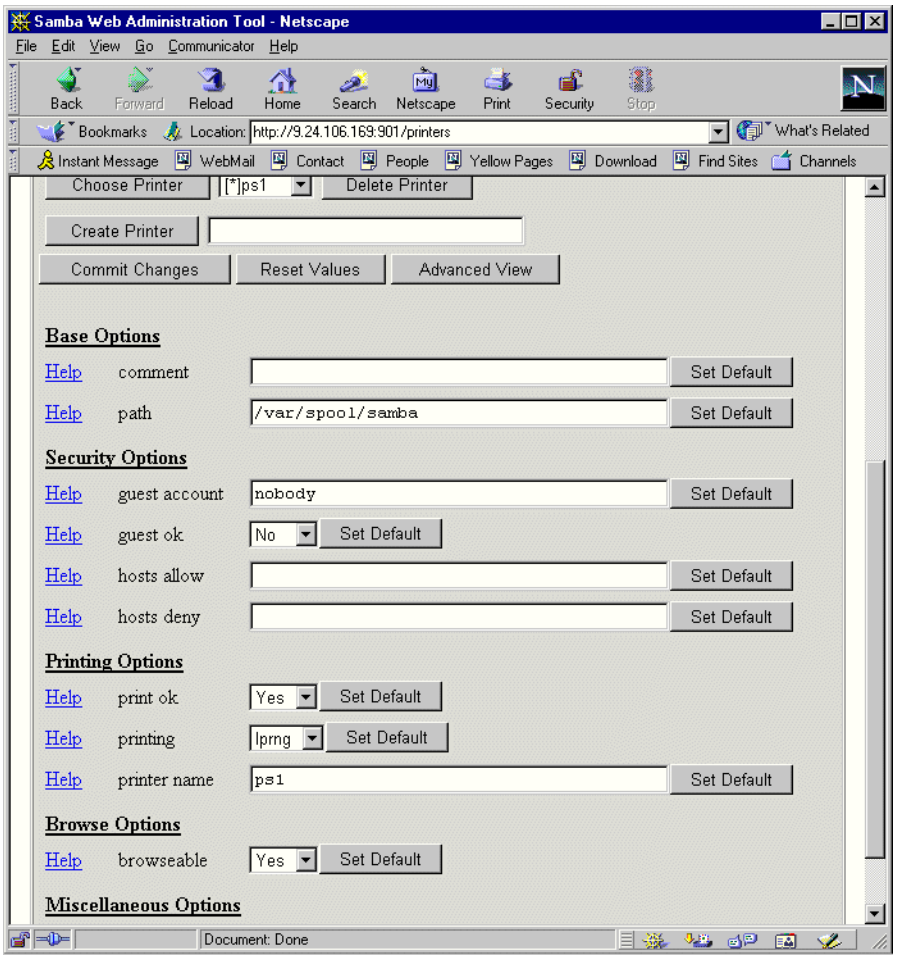


Figure 58. Printer properties

In this window you can also modify printer properties. When you are done, save the settings by clicking the **Commit Changes** button.

4.3.7 Status

In this section you can check the status of the Samba server. Here you can view all of the current connections and open files. You can also start or restart the Samba server.

.

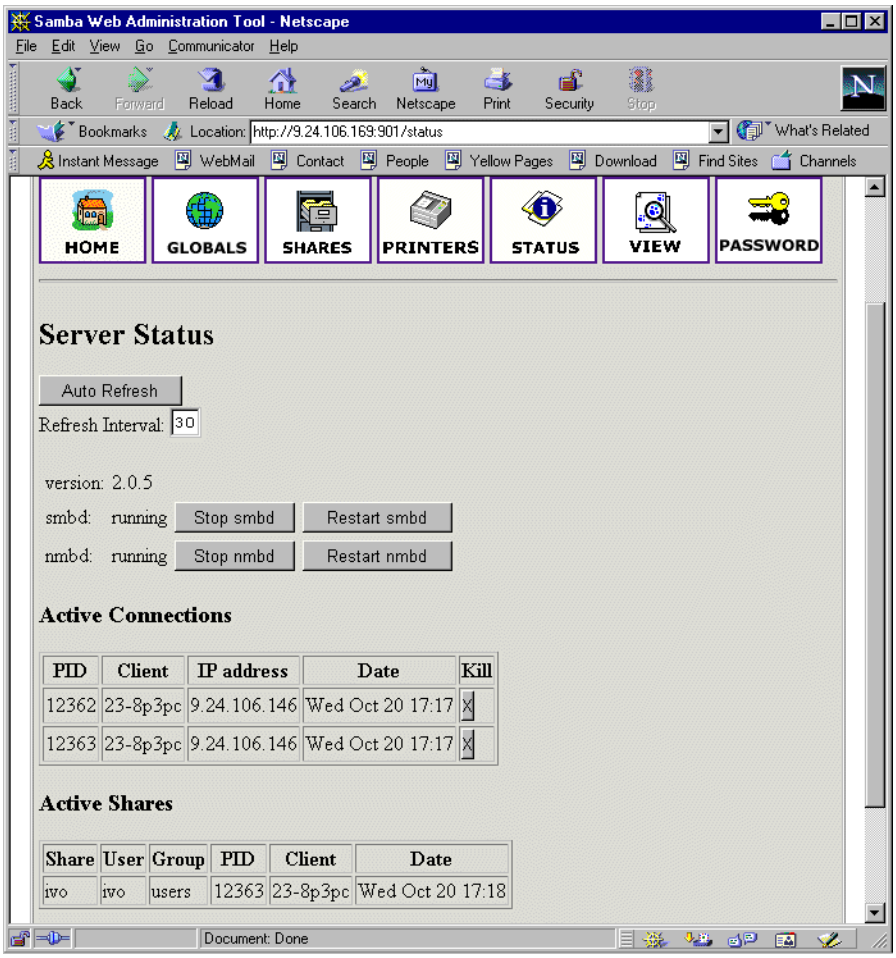


Figure 59. Status section

4.3.7.1 View

In this section you can see the current `smb.conf` configuration file. You can view detailed options by clicking the **Full View** button.

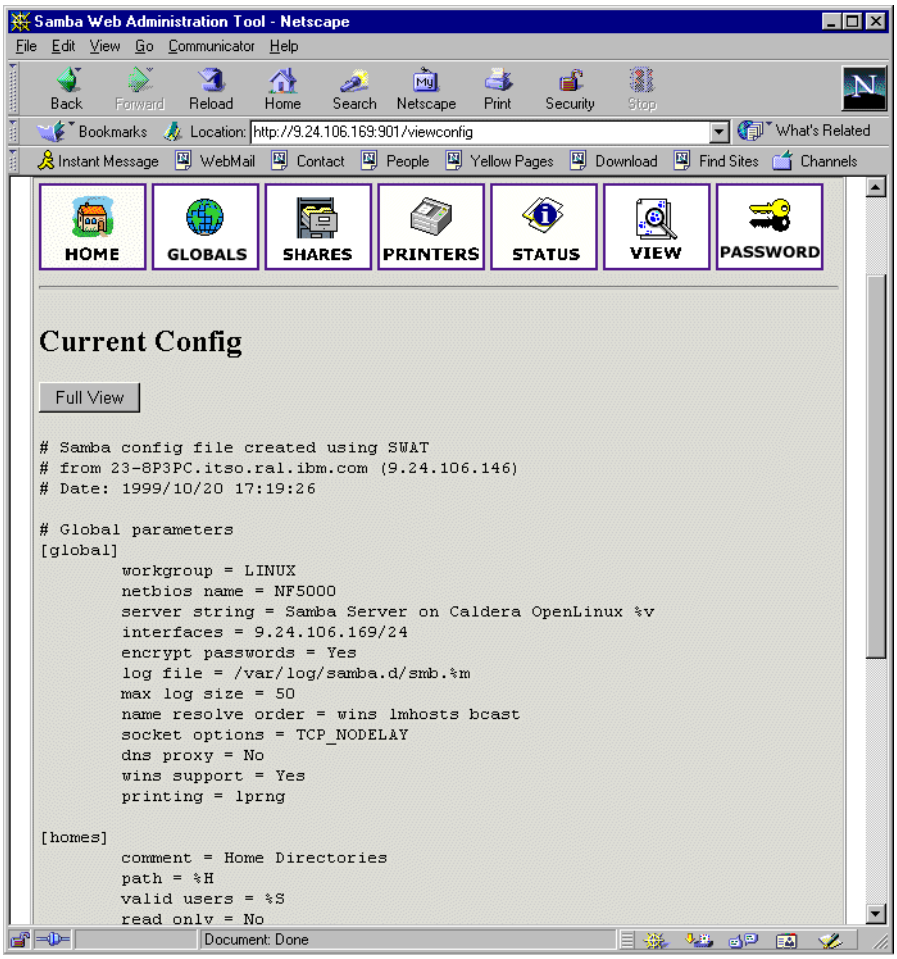


Figure 60. View section of SWAT

4.3.7.2 Password

In this section you can manage the passwords for all of your Samba users as shown in Figure 61.

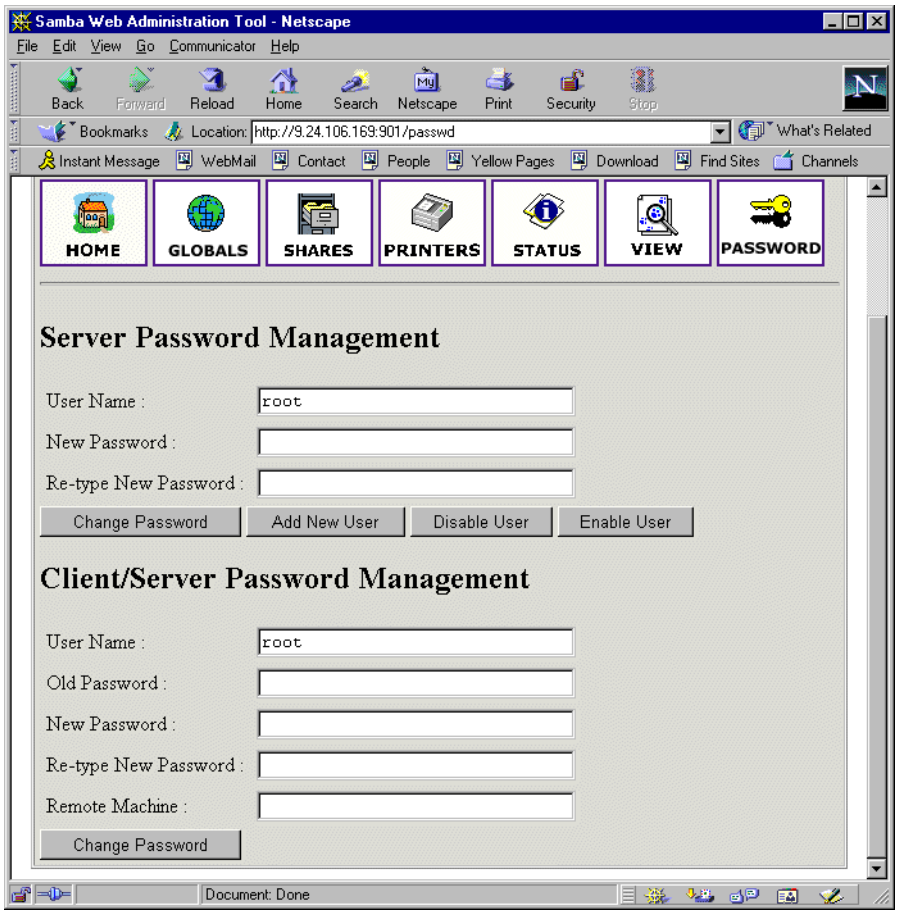


Figure 61. Managing passwords

For more information on Samba we encourage you to take a look at the Samba Web site:

`http://www.samba.org`

Chapter 5. Samba on TurboLinux

This chapter covers how to implement Samba on TurboLinux.

5.1 Installing Samba

You can verify whether Samba is installed by using the `rpm -q` command as you see in Figure 62

```
# rpm -q samba
samba-2.0.5a-19990721
#
```

Figure 62. Vefifying that Samba is installed

If Samba is not installed on your system you would get a message to that effect.

In order to install the Samba packages, insert the TurboLinux CD into the CD-ROM drive.Then type the command:

```
mount /mnt/cdrom
```

Then enter the command:

```
rpm -ivh /mnt/cdrom/TurboLinux/RPMS/samba*
```

This will install the Samba server, client, and common packages.

5.2 Configuring the Samba server

In this section we will explain how to configure Samba so it can participate as a file/print server in an existing Windows network or be a stand-alone file/print server for Windows and Linux clients.

5.2.1 The smb.conf file

Before you can start using Samba you need to configure the `smb.conf` file. This file is the heart of the Samba server. When the Samba package is installed in TurboLinux the sample configuration file is installed in:

```
/etc/smb.conf
```

In TurboLinux, Samba by default uses the `smb.conf` file in the directory `/etc`. Before you make any changes it is good to make a backup copy of the original file by doing the following:

```
cp /etc/smb.conf /etc/samba.d/smb.conf.bak
```

The Samba `smb.conf` configuration file is divided into two main sections:

1. Global Settings - here you set up parameters that affect the connection parameters.
2. Share Definitions - here you define shares. A share is a directory on the server that is accessible over the network and shared among users. This section has three parts:
    1. Homes - in this subsection you define the user's home directories.

2. Printers - in this subsection you define the available printers.
3. Shares - this subsection can add an entry for each share you want to define.

In the following sections we will describe how to modify `smb.conf` to efficiently and simply use Samba as a file/print server. We will explain only the most necessary parameters. If you need more information, look in the manual entry for `smb.conf` or on the home page of the Samba project:

```
http://www.samba.org
```

In `smb.conf` the semi-colon ";" is used to indicate a comment line. To activate a line, remove the semi-colon and make whatever changes you think are appropriate. Otherwise, Samba will use default settings or no settings, depending on the parameter.

5.2.1.1 Setting the NetBIOS parameters

The NetBIOS parameter is a part of the Global Section as seen in Figure 63.

```
T#====================== Global Settings ==================================
[global]

# workgroup = NT-Domain-Name or Workgroup-Name
   workgroup = MYGROUP

# server string is the equivalent of the NT Description field
   server string = Samba Server
```

Figure 63. Setting global NetBIOS values in /etc/smb.conf

Some important parameters to consider when you set up your Samba server are described in Table 22.

Table 22. Some Important NetBIOS parameters

Parameter	Description
`netbios name`	By this name the Samba server is known on the network. This parameter has the same meaning as the Windows NT computer name. If you do not specify it defaults to the server's hostname. This is an optional line that does not appear in the smb.conf file in TurboLinux.
`workgroup`	This parameter specifies in which Window NT domain or workgroup the Samba server will participate. It is equivalent to a Windows NT domain or workgroup name. The default workgroup for Samba is MYGROUP.
`server string`	This is the description string of the Samba server. It has the same role as the Windows NT Description field. This can be anything you want and does not have any effect on Samba.

You can use these and other parameters to set up your NetBIOS.

5.2.1.2 Global printing settings

In your `smb.conf` file you will see a section on defining your printers.

```
T..
# if you want to automatically load your printer list rather
# than setting them up individually then you'll need this
   printcap name = /etc/printcap
   load printers = yes
# It should not be necessary to spell out the print system type unless
# yours is non-standard. Currently supported print systems include:
# bsd, sysv, plp, lprng, aix, hpux, qnx
;   printing = bsd

{remainder of file is not displayed here}
```

Figure 64. /etc/smb.conf printer definitions

The parameters are described in Table 23.

Table 23. Printing parameters

Parameter	Description
`printcap name`	With this parameter you tell Samba the location of the `printcap` file. The default value is `/etc/printcap`.
`load printers`	This parameter controls if Samba loads all printers in the printcap file for browsing. The default value is yes.
`printing`	This parameter tells Samba what printing style to use on your server. TurboLinux by default uses the LPRNG printing style. You should not need to change this unless you want to run a different printing style. Otherwise you specify in `printing = lprng`

5.2.1.3 Global security settings

The global security parameters are seen in Figure 65.

```
 Security mode. Most people will want user level security. See
# security_level.txt for details.
   security = user
# Use password server option only with security = server
;   password server = <NT-Server-Name>

# Password Level allows matching of _n_ characters of the password for
# all combinations of upper and lower case.
;  password level = 8
;  username level = 8

# You may wish to use password encryption. Please read
# ENCRYPTION.txt, Win95.txt and WinNT.txt in the Samba documentation.
# Do not enable this option unless you have read those documents
;  encrypt passwords = yes
;  smb passwd file = /etc/smbpasswd
```

Figure 65. /etc/smb.conf security section

For a Windows 95/98 system with user security, enable the following:

```
encrypt passwords = yes
smb passwd file = /etc/smbpasswd
```

The important parameters that are needed here are described in Table 24.

Table 24. Security parameters

Parameter	Description
`security`	This parameter has four possible values: `share, user, server, domain.` The user setting is the most used setting
`password server`	At the `server` or `domain` security level, this server is used for authorization. For the parameter value you use the server NetBIOS name.
`encrypt passwords`	By setting this parameter to yes, you enable Samba to use the encrypted password protocol, which is used in the Windows NT Service Pack 3 and in Windows 98. This is needed to communicate with those clients.
`smb passwd file`	This parameter tells Samba where encrypted passwords are saved.

We will briefly explain each security mode:

1. Share - for this security mode, clients need to supply only the password for the resource. This mode of security is the default for the Windows 95 File/Print server. It is not recommended for use in UNIX environments, because it violates the UNIX security scheme.
2. User - user/password validation is done on the server that is offering the resource. This mode is most widely used.
3. Server - the user/password validation is done on the specified authentication server. This server can be a Windows NT server or another Samba server.
4. Domain - this security level is basically the same as server security, with the exception that the Samba server becomes a member of a Windows NT domain. In this case the Samba server can also participate in such things as trust relationships.

5.2.2 Enabling Samba access on the server

Because Windows NT 4.0 Service Pack 3 or later, Windows 95 with the latest patches, and Windows 98 use the encrypted passwords for accessing NetBIOS resources, you need to enable your Samba server to use the encrypted passwords. Before you start the Samba server for the first time you need to create a Samba encrypted passwords file. This can be done with the `mksmbpasswd` utility. The recommended way is to first create the user accounts in Linux and then create the Samba password file with the command:

```
cat /etc/passwd | /usr/bin/mksmbpasswd.sh > /etc/smbpasswd
```

This creates the Samba password file from the Linux password file.

Note

Use the same filename you specified for creating the Samba password file in the `smb.conf` configuration to tell the Samba server where the password file is.

By default the passwords for the Samba users are undefined. Before any connection is made to the Samba server, users need to create their passwords.

Now you need to specify the password for all users. If you are changing or specifying the password for the user, you can do this by executing command:

```
/usr/bin/smbpasswd -U username
```

You will see a screen similar to Figure 66.

```
# smbpasswd -U newuser
New SMB password:
Retype new SMB password:
Password changed for user newuser.
#
```

Figure 66. Specifying the password for a Samba user

Note

Anyone with access to the `/usr/bin/smbpasswd` can change passwords for the Samba users.

Another way is to have each Samba user change the password for himself, by connecting remotely to the Samba server and executing the command:

```
/usr/bin/smbpasswd
```

The output will be similar to Figure 66. If a Samba user already has defined a password he will need to type the old password before he can change the password.

If you want to add a Samba server user later, this can be done with the following command:

```
/usr/sbin/smbpasswd -a username password
```

This adds a new user to the Samba password file.

Note

You have to be logged in as root if you want to manage other users. If you are logged in as a user, you can change your password only. The `smbpasswd` utility uses the location of the password file from the `smb.conf` configuration file.

5.2.2.1 Additional smb.conf settings

While the above steps will be sufficient to enable a user to get initial access to the server, you may want to consider additional settings to take advantage of the power of Samba.

5.2.3 Global name resolution settings

In your `smb.conf` file you will see something similar to:

```
# All NetBIOS names must be resolved to IP Addresses
# 'Name Resolve Order' allows the named resolution mechanism to be specified
# the default order is "host lmhosts wins bcast". "host" means use the unix
# system gethostbyname() function call that will use either /etc/hosts OR
# DNS or NIS depending on the settings of /etc/host.config, /etc/nsswitch.conf
# and the /etc/resolv.conf file. "host" therefore is system configuration
# dependant. This parameter is most often of use to prevent DNS lookups
# in order to resolve NetBIOS names to IP Addresses. Use with care!
# The example below excludes use of name resolution for machines that are NOT
# on the local network segment
# - OR - are not deliberately to be known via lmhosts or via WINS.
; name resolve order = wins lmhosts bcast

# Windows Internet Name Serving Support Section:
# WINS Support - Tells the NMBD component of Samba to enable it's WINS Server
;   wins support = yes

# WINS Server - Tells the NMBD components of Samba to be a WINS Client
#       Note: Samba can be either a WINS Server, or a WINS Client, but NOT both
;   wins server = w.x.y.z
{remainder of file is not shown here}
```

Figure 67. Name and server resolution settings in /etc/smb.conf

We recommend that in /etc/smb.conf the following lines be uncommented:

```
name resolve order = wins lmhosts bcast
wins support = yes
```

If the above line is uncommented then the next line must be commented and vice versa:

```
;   wins server = w.x.y.z
```

The parameters are described in Table 25.

Table 25. Name resolution parameters

Parameter	Description
`name resolve order`	With this parameter you specify how the Samba server will resolve NetBIOS names into IP addresses. The preferred value is `wins lmhosts bcast`. Refer to Figure 67 and the manual page of `smb.conf` for more information.
`wins support`	If this option is enabled the Samba server will also act as a WINS server.
`wins server`	With this parameter, you tell Samba which WINS server to use.

Note

Samba can act as a WINS server or a WINS client, but not both. So only one of the parameters (`wins support` or `wins server`) can be set at the same time. If you specify the IP address of WINS server then `wins support` must be set to "no".

5.2.4 Creating shares

In the previous sections we have explained how to prepare general configuration parameters. But a Samba server can become useful when you offer resources to

the users. In this section we will explain how to create a share. The simple share section in the `smb.conf` file is seen in Figure 68.

```
[redbook]
   comment = Redbook files
   path = /redbook
   browseable = yes
   printable = no
   writable = yes
   write list = @users
```

Figure 68. Adding a directory share entry to /etc/smb.conf

In Table 26 we explain the most important parameters for creating a share.

Table 26. Share parameters

Parameter	Description
`comment`	This describes the function of share.
`admin users`	This parameter is used to specify the users who have administrative privileges for the share. When they access the share, they perform all operations as `root`.
`path`	Defines the full path to the directory you are sharing.
`browseable`	If this parameter is set to yes, you can see a share when you are browsing the resources on the Samba server. The value can be yes or no.
`printable`	This parameter specifies if the share is a print share. The value can be yes or no.
`write list`	Users specified in this list have write access to the share. If the name begins with @ it means a group name.
`writable`	This parameter specifies if the share is writable. The value can be yes or no.
`read list`	Users specified in this list have read access to the share. If the name begins with @ it means a group name.
`read only`	If this is set to yes, the share is read only. The value can be yes or no.
`valid users`	This parameter specifies which users can access the share.

By using these parameters you can easily set up a new share. Each share definition starts with the share name in brackets "[]". Below this name you can specify the values for the share parameters.

5.2.5 Share permissions

Although you can control the share permissions with share parameters, UNIX permissions are applied before a user can access files on the share. So you need to take care of UNIX permissions, to allow the user has also access to the share directory under UNIX.

When a user creates a new file on the shared directory, the default create mask used is 0744. For directory creation, the default create mask is 0755. If you want

you can force a different creation mask. The parameters for doing this are explained in Table 27.

Table 27. Create mask parameters

Parameter	Description
`create mask`	This is used for file creation to mask against UNIX mask calculated from the DOS mode requested.
`directory mask`	This is used for directory creation to mask against UNIX mask calculated from the DOS mode requested.

5.2.6 Creating shares for home directories

For handling home directories Samba has a special share section called [homes]. This share definition is used for all home directories, so you do not need to create separate shares for each user.

When a client requests a connection to a file share, the existing file shares are scanned. If a match is found, that share is used. If no match is found, the requested share is treated as a username and validated by security. If the name exists and the password is correct, a share with that name is created by cloning the [homes] section. The home share definition uses the same parameters as a normal share definition. The following is an example of a home share definition in `smb.conf` configuration file:

By default TurboLinux has the [homes] section as defined in `etc/smb.conf`.

```
[homes]
   comment = Home Directories
   browseable = no
   writable = yes
```

Figure 69. /etc/smb.conf default [homes] section

You may want to consider adding the additional parameters as shown in Figure 70.

```
[Thomes]
   comment = Home Directories
   path = %H
   valid users = %S
   browseable = no
   writable = yes
   create mode = 0700
directory mode = 0700
```

Figure 70. Modified [homes] section in /etc/smb.conf

The variables we used in this definition are explained in Table 28.

Table 28. Variable description

Parameter	Description
%H	This variable represents the home directory of the user. Samba will use the user's home directory by default if this is not defined.
%S	The name of the current service, which is in the case of a home share equal to username. This defaults to the home share equal to the username if this is not defined.

As you can see in the example we used creation masks for the files and the directories, in such a way that we forced all new files or directories to be accessible only by the owner of the home directory.

5.2.7 Creating a printer share

A Samba server uses the same procedure for printer shares as for the home shares. After all share definitions and usernames are tested against the requested share name and the matched definition is still not found, Samba searches for a printer with that name (if the [printers] section exists). If the match is found in the printer definitions, that [printers] share section is cloned with the name of the requested service, which is really a printer name. The following is an example of a printers definition in the `smb.conf` configuration file:

```
# NOTE: If you have a BSD-style print system there is no need to
# specifically define each individual printer
[printers]
   comment = All Printers
   path = /var/spool/samba
   browseable = no
# Set public = yes to allow user 'guest account' to print
   guest ok = no
   writable = no
   printable = yes
#
{remainder of file is not displayed here}
```

As you can see the [printers] section is just another share definition, because when users print they basically copy the data into a spool directory; after that the data is handled by the local printing system. The only big difference between a printer share and other share definitions is that the `printable` parameter is set to "yes". This means that a user can write a spool file to the directory specified under the share definition. If the share is printable, then it is also writable by default.

5.3 Starting and stopping the Samba server

You can start the Samba server by executing the command:

```
/etc/rc.d/init.d/smb start
```

You will see output similar to this:

```
# /etc/rc.d/init.d/smb start
Starting SMB services: smbd nmbd
#
```

As you can see two daemons are started: `smbd` and `nmbd`. `Smbd` is the actual Samba server and `nmbd` is the WINS server.

the Samba server can be stopped by executing the command:

```
/etc/rc.d/init.d/smb stop
```

Whenever you make modifications to the `smb.conf` configuration file you must restart the Samba server.You can do this with the command:

```
/etc/rc.d/init.d/smb stop
```

You can check the status of Samba with the command:

```
/etc/rc.d/init.d/smb status
```

5.4 Starting Samba on bootup

If you install Samba when you install your TurboLinux it should start up automatically. If it does not, you can start it up with the program:

```
turboservice
```

Figure 71 shows the turboservice starting up after entering the `turboservice` command.

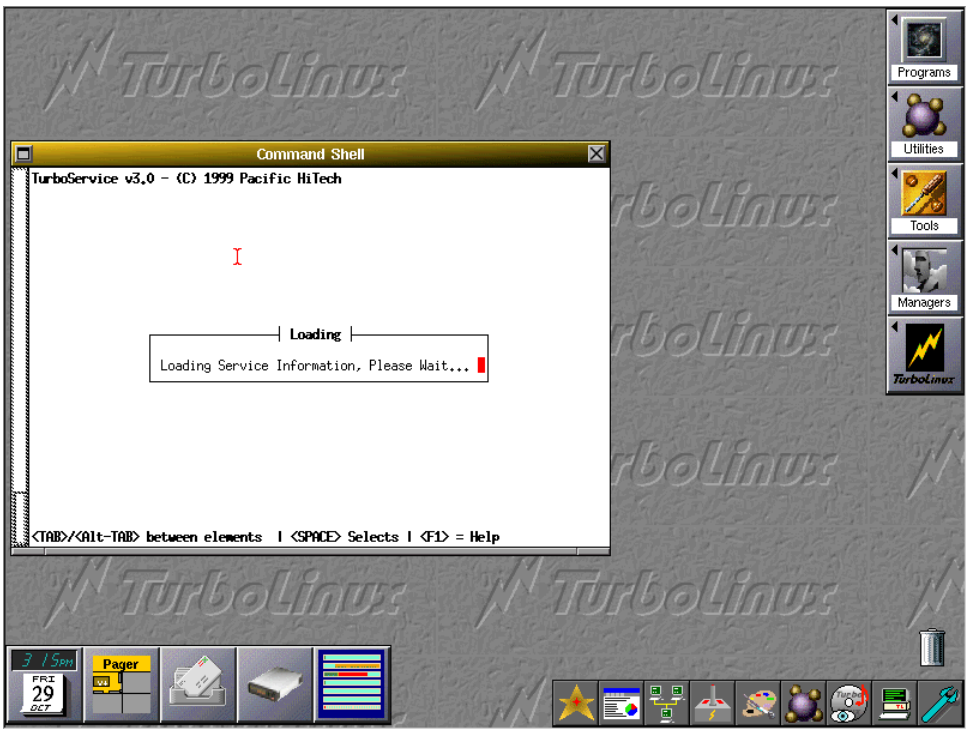


Figure 71. Starting turboservice

In Figure 72 you see the Turboservice status board.

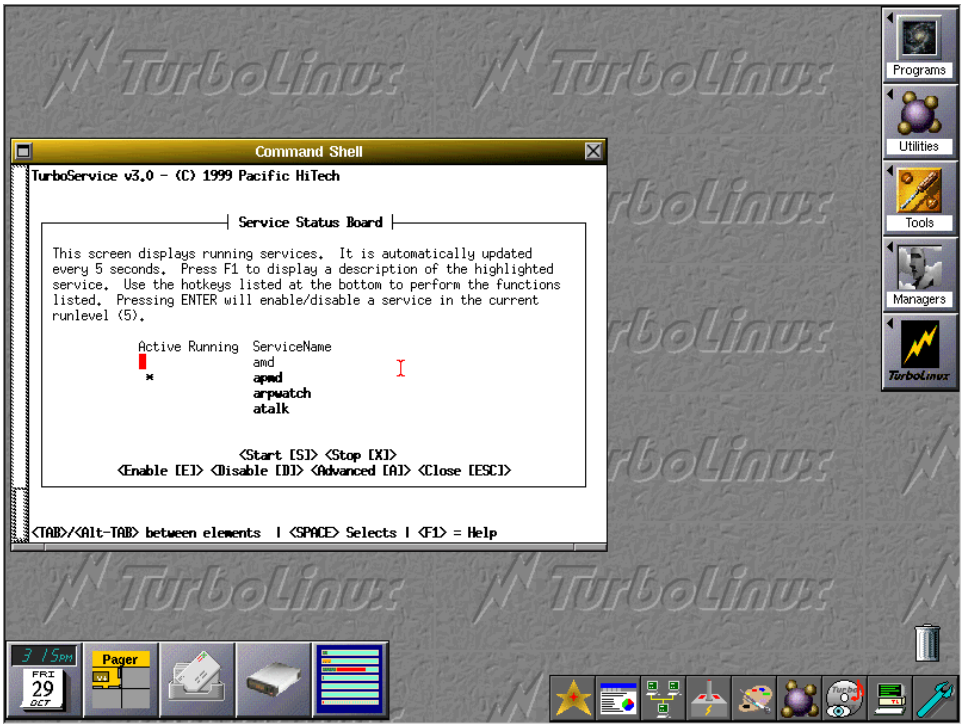


Figure 72. turboservice Service status board

You can start it up manually as explained in 5.3, “Starting and stopping the Samba server” on page 79. To manually set up TurboLinux to start Samba on bootup, you need to verify that the /etc/smb.conf file exists. You also need to verify that the following file has been installed in the startup directory. You should see the file:

`/etc/rc.d/rc3.d/S91smb` (or some file starting with S and ending with smb)

If you do not see this file, but see a file with `Kxxsmb` (were `xx` is a number) it means that Samba is disabled. The K means the process is killed when the system enters run state 3 (rc3 means run level 3). Because the system sometimes depends on certain programs starting and ending before others you cannot just change the K to an S. If all else fails, stick the following entry in /etc/rc.d/rc3.d/S99local:

```
/etc/rc.d/init.d/smb start
```

S99local is the file where you can stick whatever you want after everything else starts, or for those programs you do not know where to put.

When the Linux server is restarted the Samba server will be started automatically. You can also verify that Samba is running with the command:

```
smbstatus
```

If it is running properly you will see a display as Figure 73. If you do not get any errors then you know that Samba is running okay.

```
T# smbstatus

Samba version 2.0.5a
Service      uid      gid      pid     machine
----------------------------------------------

No locked files

Share mode memory usage (bytes):
   1048464(99%) free + 56(0%) used + 56(0%) overhead = 1048576(100%) total
#
```

Figure 73. Using smbstatus to verify that Samba is running

5.5 Using SWAT

the Samba Web Administration Tool (SWAT) allows configuration of the `smb.conf` configuration file through a Web browser. That means you can configure Samba in a GUI-like environment. SWAT itself is a small Web server. A CGI scripting application, designed to run from `inetd`, provides access to the `smb.conf` configuration file.

An authorized user with the root password can configure the `smb.conf` configuration file via Web pages. SWAT also places help links to all configurable options on every page, which lets an administrator easily understand the effect of the changes.

Before using SWAT you must check the following.

1. In the file `/etc/services` you must have the following line:

   ```
   swat 901/tcp
   ```

2. In the file `/etc/inetd.conf` you must have the following line:

   ```
   swat stream tcp nowait.400 root /usr/sbin/tcpd swat
   ```

   As you can see SWAT is started with a TCP wrapper. This is done by the expression:

   ```
   /usr/sbin/tcpd swat
   ```

   in the entry in /etc/inetd.conf. This allows you to control who can access the SWAT service with `/etc/hosts.allow` and `/etc/hosts.deny` file. For example if you want to access SWAT locally, only your `/etc/hosts.deny` file should look similar to this:

```
#
# hosts.deny    This file describes the names of the hosts which are
#               *not* allowed to use the local INET services, as decided
#               by the '/usr/sbin/tcpd' server.
#
# See man hosts_access(5) for more information.

# ALL: ALL
swat:ALL EXCEPT 127.0.0.1
```

Figure 74. /etc/hosts.deny with swat entry

If you modify either file you will need to restart inetd. You can do this with the command:

```
/etc/rc.d/init.d/inet reload
```

If you did everything without errors you are ready to use SWAT. To start SWAT point your favorite Web browser to the Internet address of your Samba server on port 901, as you can see in Figure 75.

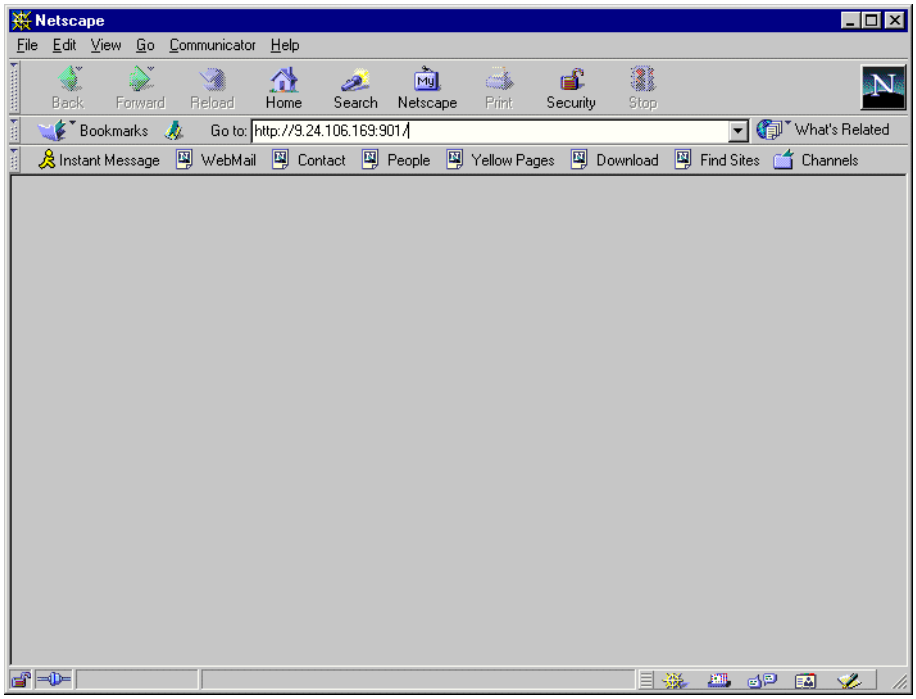


Figure 75. Starting SWAT

After you load the home page of SWAT, you will see a screen similar to Figure 76.

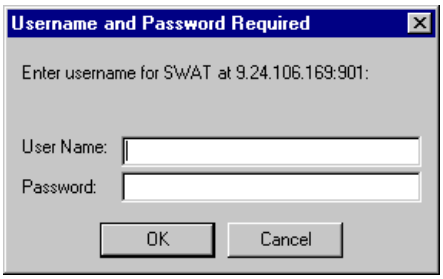


Figure 76. User authorization for SWAT

Type in the username and password of the Linux user defined on your Linux server. Click **OK** to continue. You will see a screen similar to Figure 77.

Note

You can access SWAT with any Linux user, but you can make changes only with the root user.

Remember, when you are logging on to SWAT from remote machine, you are sending passwords in plain text. This can be a security issue, so we recommend that you do SWAT administration only locally.

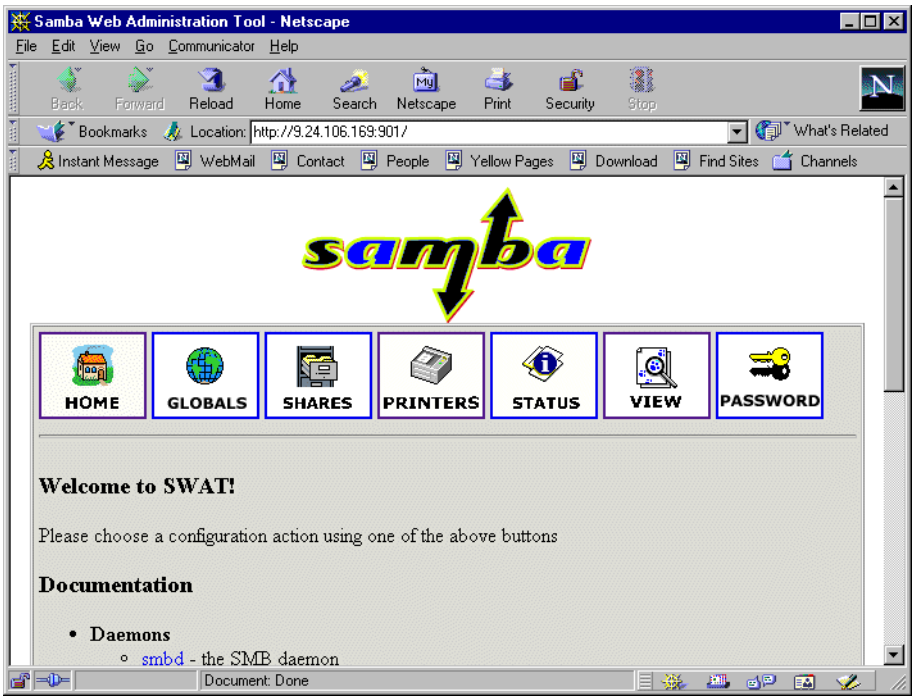


Figure 77. SWAT home page

As you can see in Figure 77, you have seven categories available:

1. Home - here you can find all the documentation you need about Samba.
2. Globals - here you can see and modify global parameters from the `smb.conf` configuration file.
3. Shares - here you can view, modify, and add shares.
4. Printers - here you can view, modify, and add printers.
5. Status - here you can check the current status of your Samba server.
6. View - here you can see the current configuration of `smb.conf` configuration file.
7. Passwords - here you can manage passwords for the Samba server.

In the following sections we briefly describe the sections available in SWAT.

Note

You can reach any of the seven sections on all SWAT Web pages. There are always icons for the sections on the top of each page.

After you make changes to the `smb.conf` configuration file, the Samba server must be restarted.

5.5.1 Globals

When you click the **Globals** icon in the main SWAT window, you will see a window similar to Figure 78.

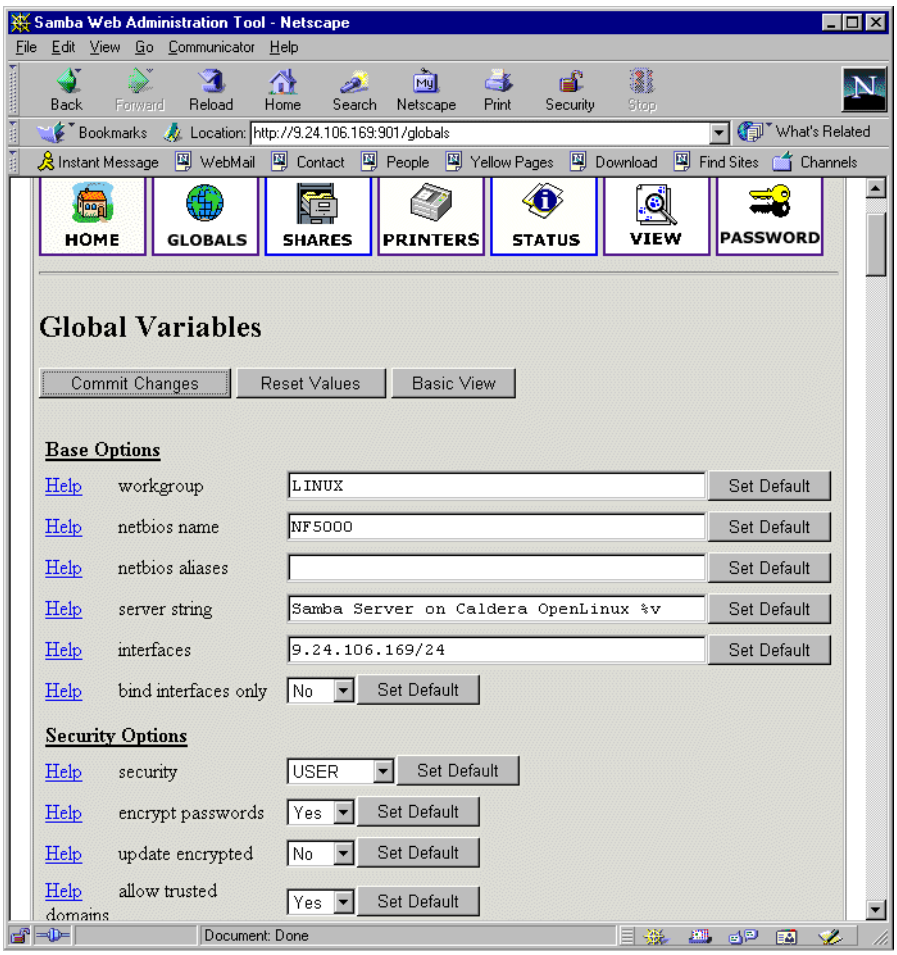


Figure 78. Global section in SWAT

In this window you can modify the global parameters for the Samba server. By default you will see the Basic View; if you want to see the Advanced View, click **Advanced View**. In the Advanced View you have all options available, while in Basic View you can only change the basic options. To return from the Advanced View to the Basic View click **Basic View.** After you have made your changes you can save them by clicking **Commit changes.** If you get a pop-up window similar to Figure 79, which warns you that you are sending non-secure information over

the network, you can easily select **Continue** if you are working locally or if you know that your network is secure.

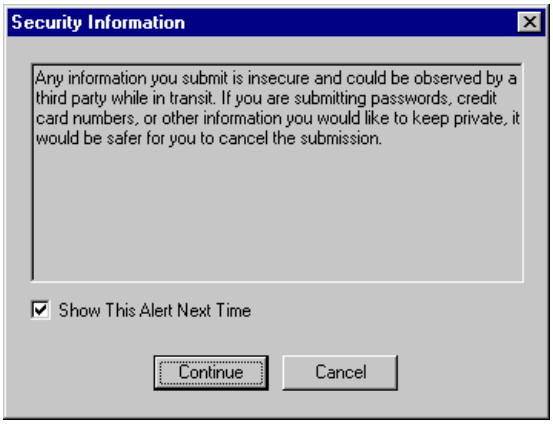


Figure 79. Security warning

5.5.2 Shares

When you click the **Shares** icon on any of the SWAT Web pages, you will see a screen similar to Figure 80.

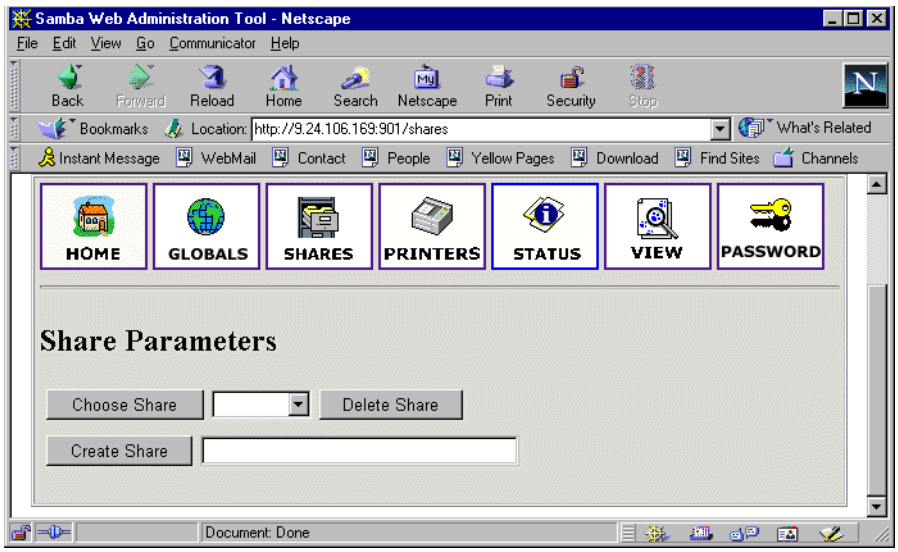


Figure 80. Shares section in SWAT

Here you can:

1. View the defined share
2. Delete share
3. Create a new share

5.5.3 Viewing or modifying an existing share

To view an already defined share select the share from the field to the right of the **Choose Share** button, similar to Figure 81.

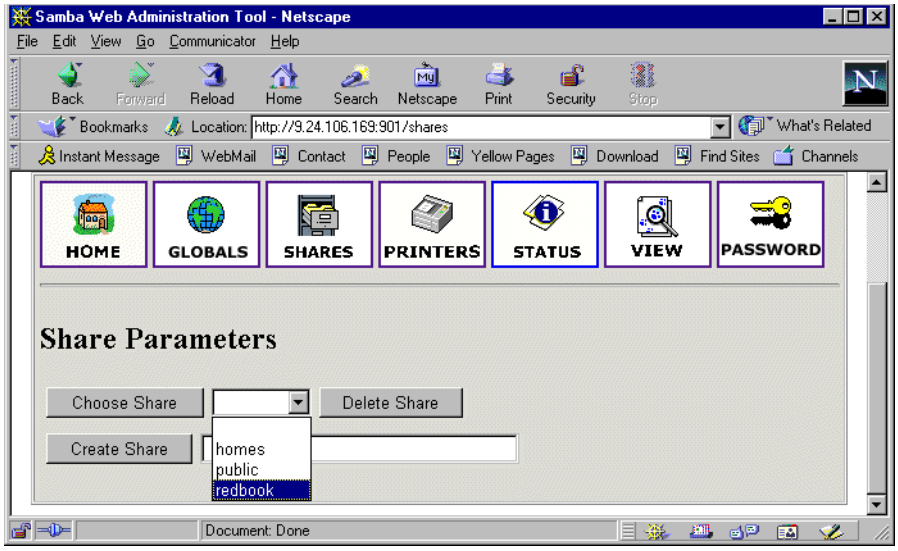


Figure 81. Choosing a share to view

After you have selected the share, click **Choose Share** to view the share properties. You will see a screen similar to Figure 82.

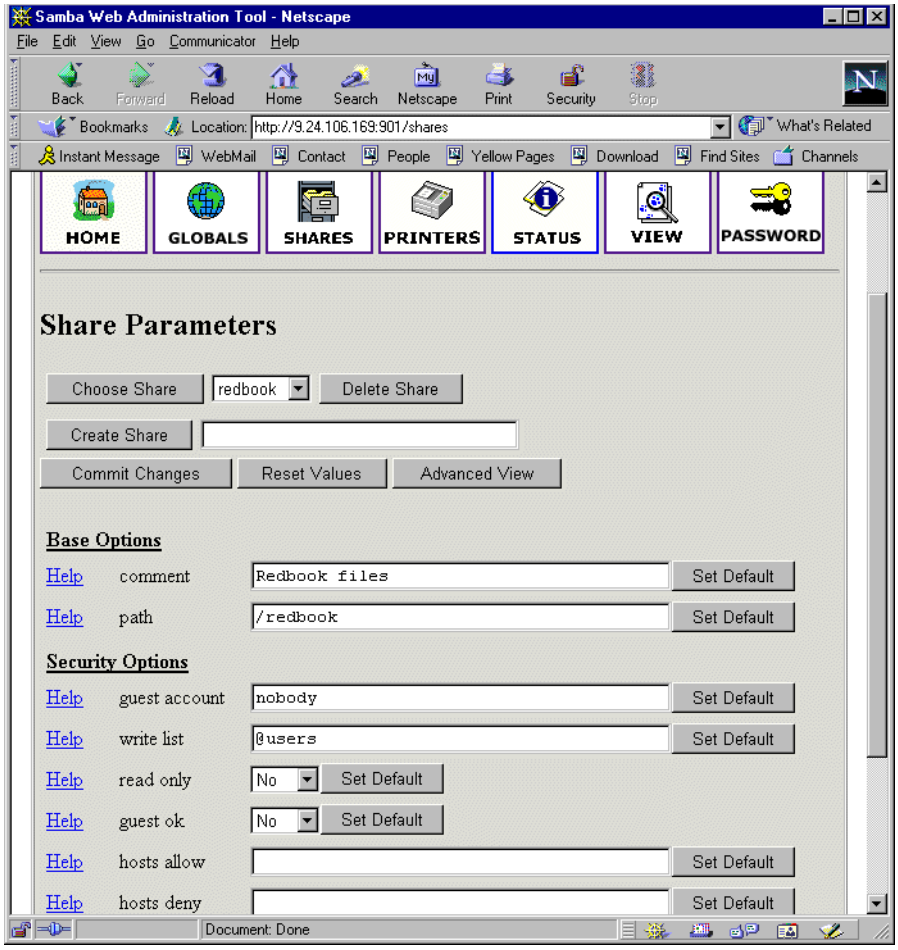


Figure 82. Share properties

If you want to see all available parameters, click **Advanced View.** In this view you can also make changes and you can save them by clicking **Commit Changes.**

5.5.4 Deleting the existing share

To delete an existing share you must first select an already defined share similar to Figure 81. Then you click **Delete Share.**

> **Attention**
>
> a share is deleted immediately and without warning.

After you have deleted the share the Samba server must be restarted.

5.5.5 Creating a new share

In this section we show how to create a simple share. To accomplish this follow these steps:

1. Create a directory that will be used for the share. You can do this by executing this command from the terminal:

   ```
   mkdir /home/public
   ```

   In our example we created a “public” directory in the “home” directory.

2. Make sure that the UNIX permissions are set correctly in that directory, so that only intended users have access rights to it.

3. In the shares view of the SWAT Web pages, type in the name of the share you are creating similar to Figure 83.

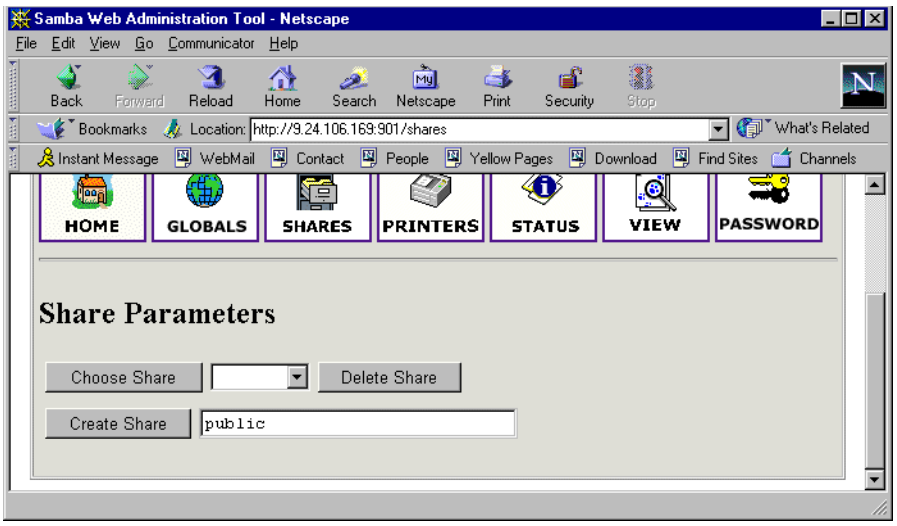


Figure 83. Entering a name for the new share

4. Click **Create Share** to continue. You will see a screen similar to Figure 84.

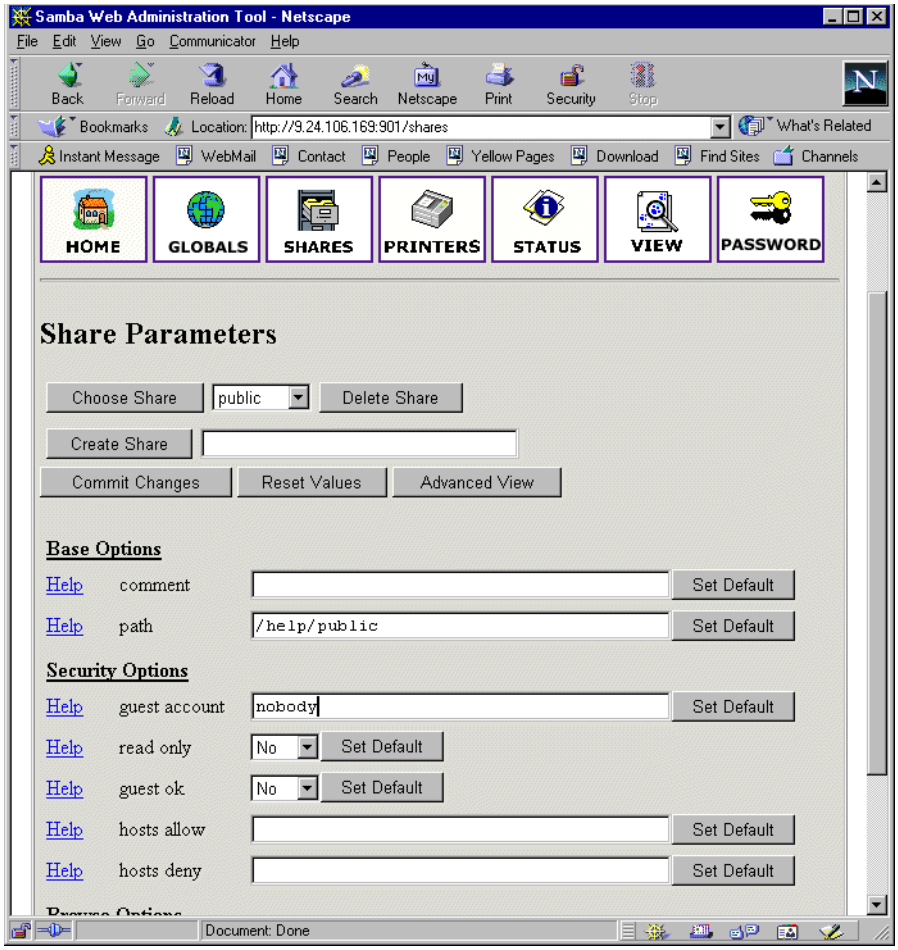


Figure 84. Entering the new share parameters

5. Fill in the needed parameters. If you need to set more advanced parameters also, click on **Advanced View** and you will see all available parameters. After you typed in all you want, click **Commit Changes** to save your new share.
6. You can see the changed `smb.conf` configuration file by selecting the **View** section from the SWAT html page. You will see a screen similar to Figure 85.

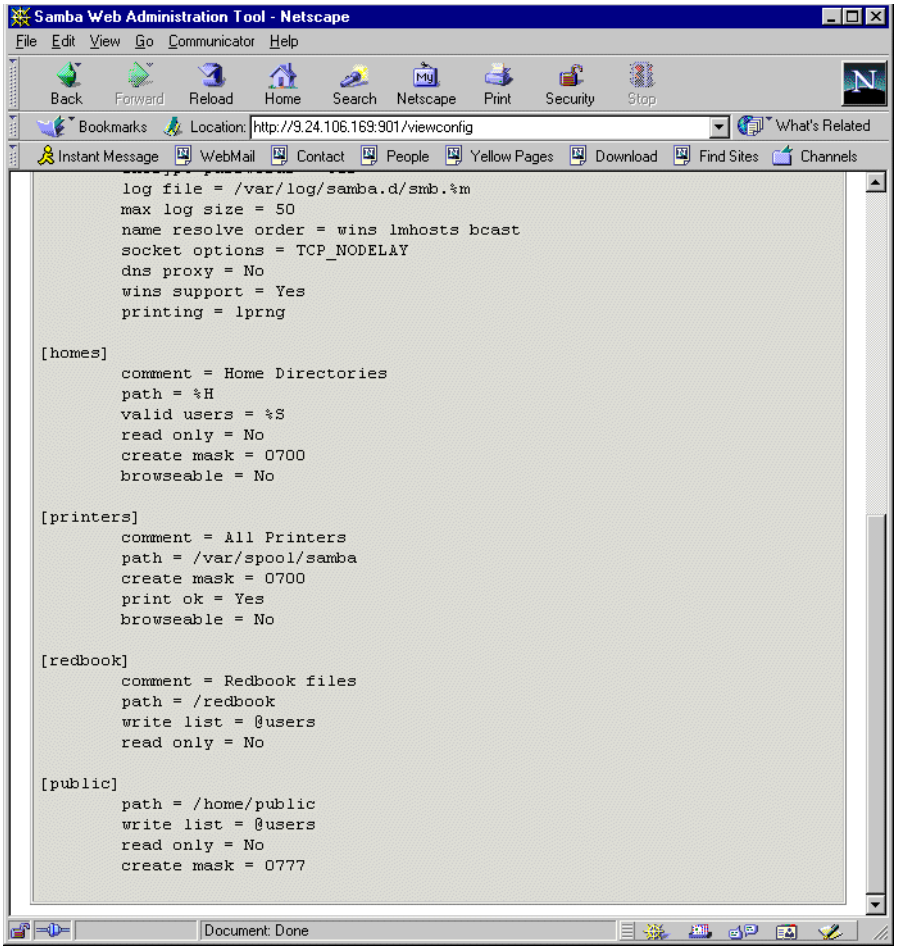


Figure 85. Viewing the smb.conf configuration file

7. Restart the Samba server.

Congratulations! You have just created your first usable share on the Samba server. Be friendly and share it with other users!

5.5.6 Restarting the Samba server

The Samba server can be restarted from the Status section. To get to this section click the **Start** icon on any SWAT Web page. You will see a screen similar to **Figure 86.**

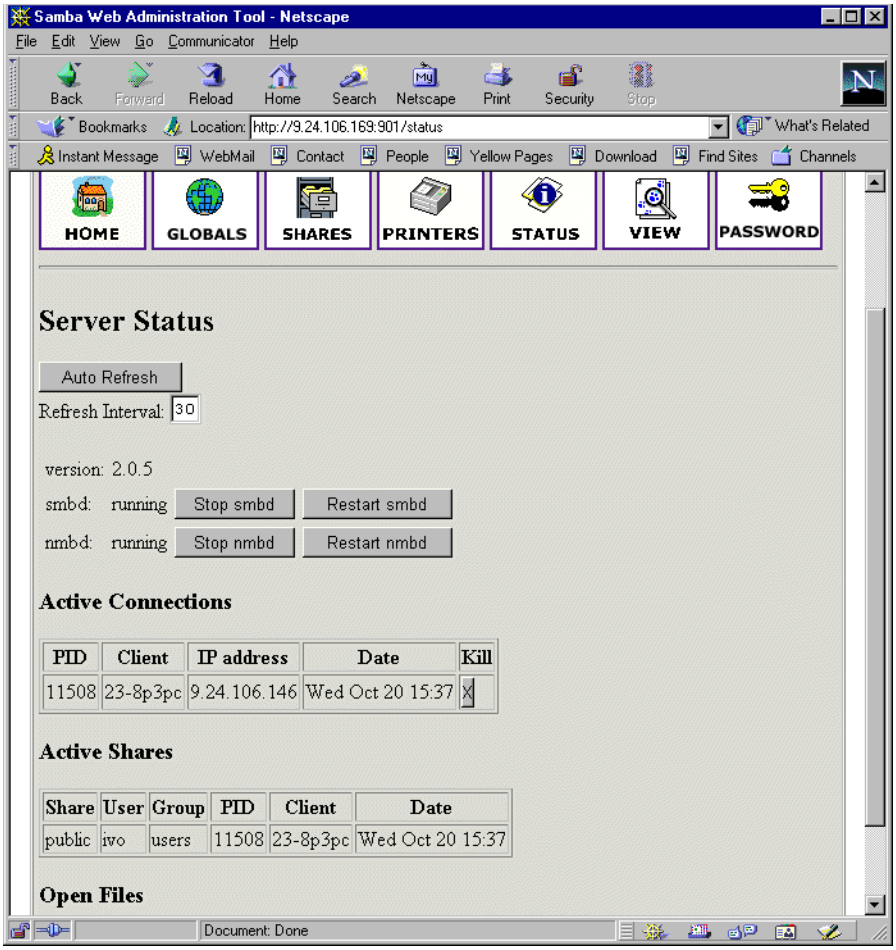


Figure 86. Restarting the Samba server

To restart the Samba server, simply click **Restart smbd.** On this page you can also restart just the WINS server by clicking **Restart nmbd.**

5.5.7 Printers

In the printer section you can view, modify, or add printers. The operations for handling printers are the same as for handling shares. You can access the printer settings by clicking the **Printers** icon on the SWAT Web page similar to Figure 87.

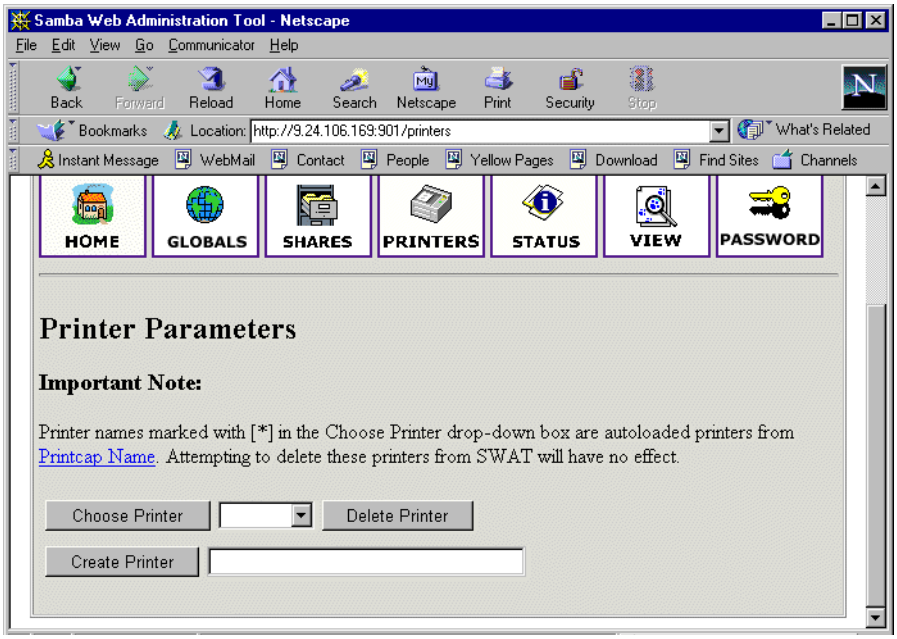


Figure 87. SWAT printers section

If you want to see the settings for a specific printer, select the printer from the list as shown in Figure 88.

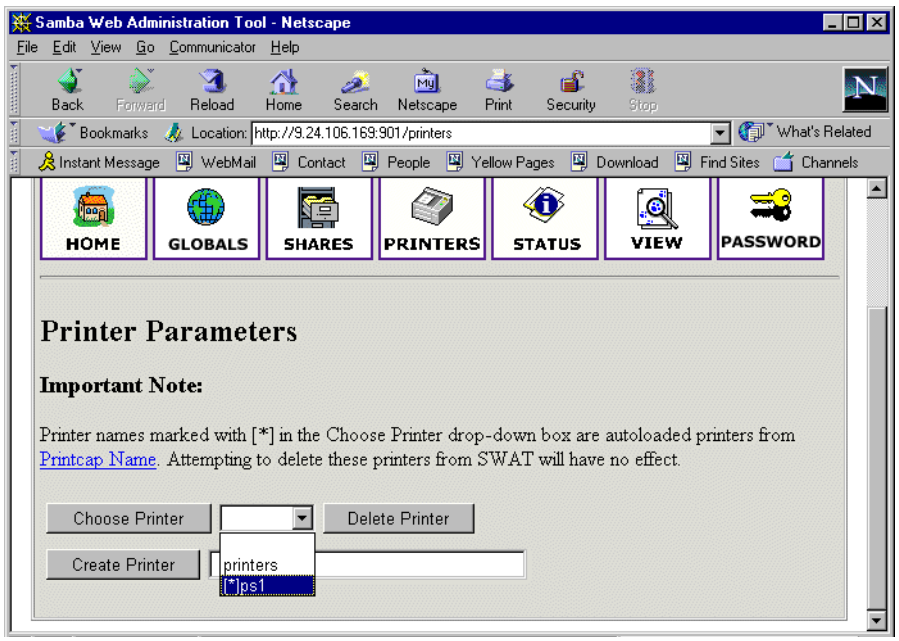


Figure 88. Selecting a printer

After you have selected the printer, click **Choose Printer** to view its properties. You will see a screen similar to Figure 89.

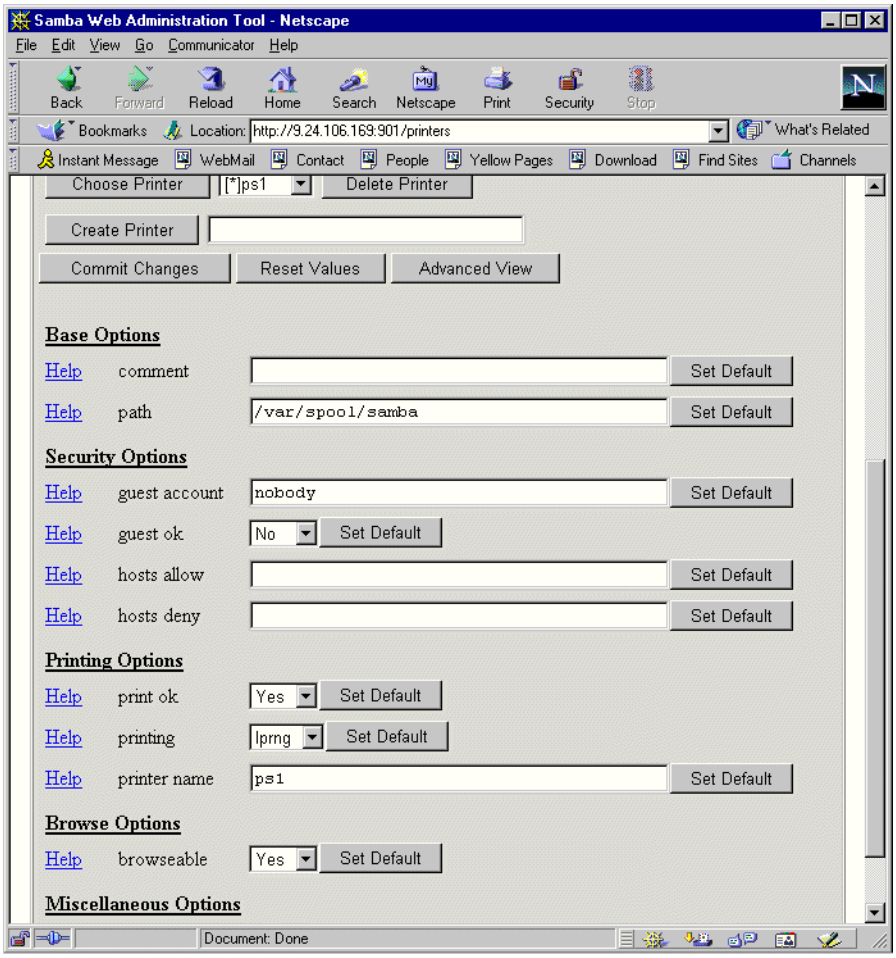


Figure 89. Printer properties

In this view you can also modify the printer properties. When you are done, save the settings by clicking **Commit Changes.**

5.5.8 Status

In this section you can check the status of the Samba server. Here you can see all the connections and open files. You can also start or restart the Samba server or just its components. You can access printer settings by clicking the **Status** icon on the SWAT Web page, as you can see in Figure 90.

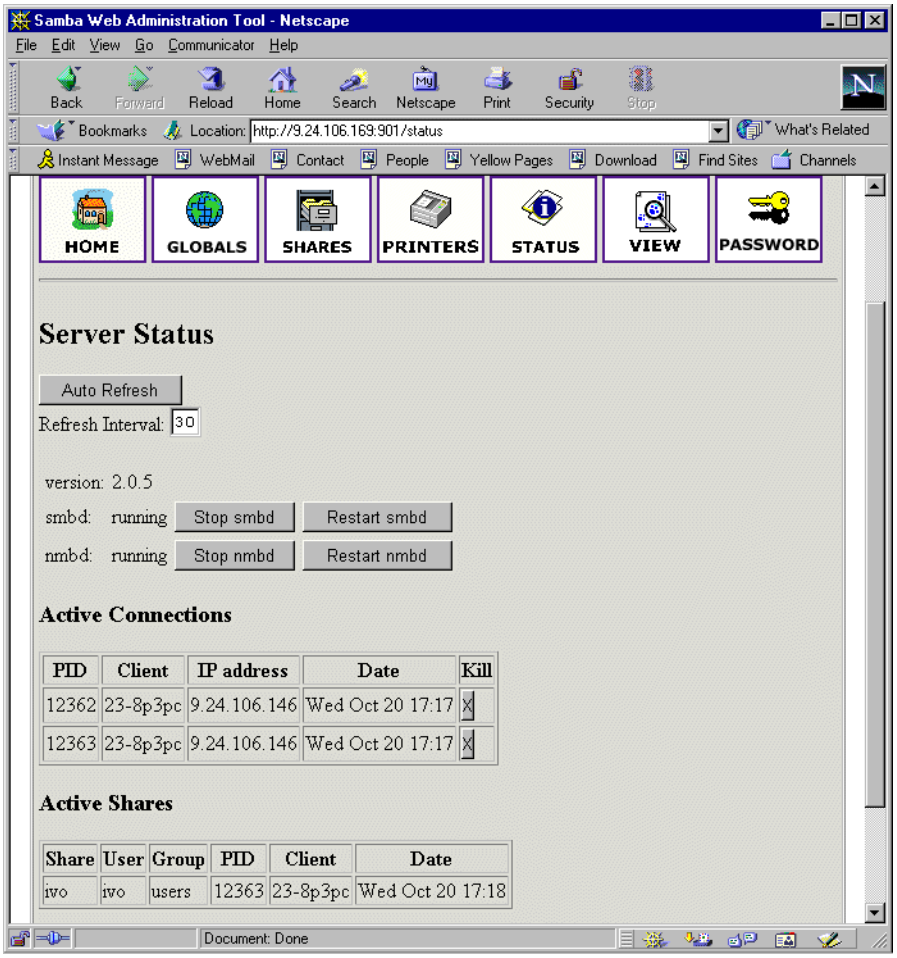


Figure 90. Status section

5.5.9 View

In this section you can see the current `smb.conf` configuration file. You can access printer settings by clicking the **View** icon on the SWAT Web page similar to Figure 91.

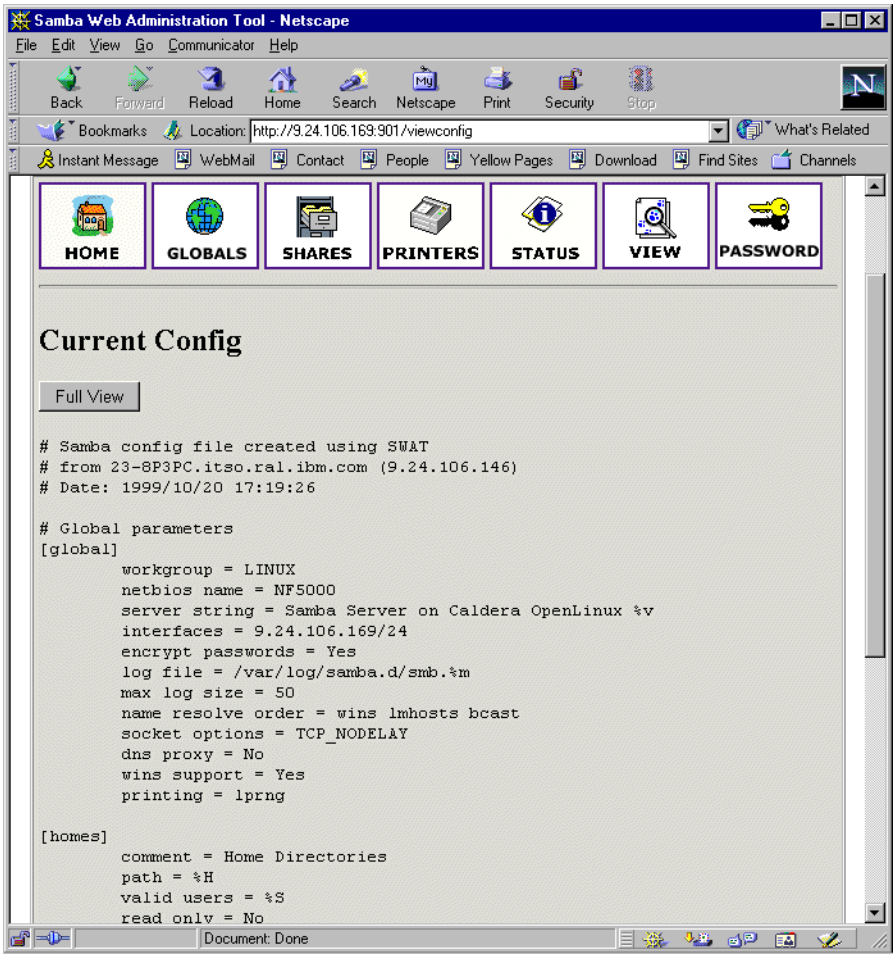


Figure 91. View section of SWAT

5.5.9.1 Password

In this section you can manage the passwords of all Samba users. You can access printer settings by clicking the **Password** icon on the SWAT Web page similar to Figure 92.

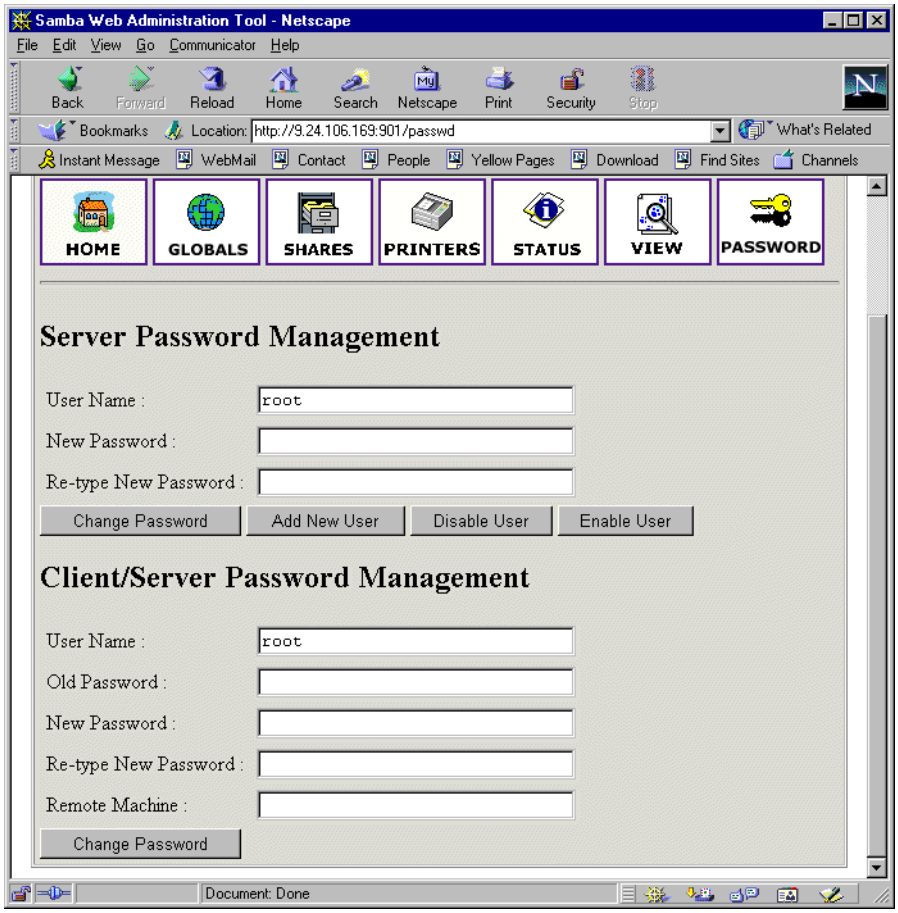


Figure 92. Managing passwords

5.6 Sources and additional information

You can find more information on the official home page of the Samba project:

`http://www.samba.org`

And there are always good How-to documents on the Linux Documentation project home page:

`http://www.linuxdoc.org/`

Special Notices

This publication is intended to help customers, business partners and IBM employees implement Samba on the Linux operating system. The information in this publication is not intended as the specification of any programming interfaces that are provided by the IBM products mentioned in this publication. See the PUBLICATIONS section of the IBM Programming Announcement for these products for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM ("vendor") products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AS/400	Home Director
IBM	Netfinity
RS/6000	System/390

The following terms are trademarks of other companies:

Linux is a trademark of Linus Torvalds.

C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and/or other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names may be trademarks or service marks of others.

Redbooks on CD-ROMs

Redbooks are available on the following CD-ROMs. Click the CD-ROMs button at http://www.redbooks.ibm.com/ for information about all the CD-ROMs offered, updates and formats.

CD-ROM Title	Collection Kit Number
System/390 Redbooks Collection	SK2T-2177
Networking and Systems Management Redbooks Collection	SK2T-6022
Transaction Processing and Data Management Redbooks Collection	SK2T-8038
Lotus Redbooks Collection	SK2T-8039
Tivoli Redbooks Collection	SK2T-8044
AS/400 Redbooks Collection	SK2T-2849
Netfinity Hardware and Software Redbooks Collection	SK2T-8046
RS/6000 Redbooks Collection (BkMgr Format)	SK2T-8040
RS/6000 Redbooks Collection (PDF Format)	SK2T-8043
Application Development Redbooks Collection	SK2T-8037
IBM Enterprise Storage and Systems Management Solutions	SK3T-3694

Referenced Web sites

The following World Wide Web sites may provide more information about the topics discussed in this redpaper:

- http://www.samba.org
- http://www.linuxdoc.org/