.10. File and Print Services for Microsoft® Windows® Clients (Samba)

Samba is a popular open source software package that provides file and print services using the SMB/CIFSprotocol. This protocol is built into Microsoft® Windows® systems. It can be added to non-Microsoft® Windows® systems by installing the Samba client libraries. The protocol allows clients to access shared data and printers. These shares can be mapped as a local disk drive and shared printers can be used as if they were local printers.

On FreeBSD, the Samba client libraries can be installed using the *net/samba-smbclient* port or package. The client provides the ability for a FreeBSD system to access SMB/CIFS shares in a Microsoft® Windows® network.

A FreeBSD system can also be configured to act as a Samba server. This allows the administrator to createSMB/CIFS shares on the FreeBSD system which can be accessed by clients running Microsoft® Windows® or theSamba client libraries. In order to configure a Samba server on FreeBSD, the *net/samba36* port or package must first be installed. The rest of this section provides an overview of how to configure a Samba server on FreeBSD.

.10.1. Configuration

A default Samba configuration file is installed as`/usr/local/share/examples/samba36/smb.conf.default`. This file must be copied to`/usr/local/etc/smb.conf` and customized before Samba can be used.

Runtime configuration information for Samba is found in `smb.conf`, such as definitions of the printers and “file system shares” that will be shared with Windows® clients. The Samba package includes a web based tool calledswat which provides a simple way for configuring `smb.conf`.

.10.1.1. Using the Samba Web Administration Tool (SWAT)

The Samba Web Administration Tool (SWAT) runs as a daemon from inetd. Therefore, inetd must be enabled as shown in Section 29.2, “The inetd Super-Server”. To enable swat, uncomment the following line in`/etc/inetd.conf`:

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

As explained in Example 29.1, “Reloading the inetd Configuration File”, the inetd configuration must be reloaded after this configuration file is changed.

Once swat has been enabled, use a web browser to connect to `http://localhost:901`. At first login, enter the credentials for `root`.

Once logged in, the main Samba configuration page and the system documentation will be available. Begin configuration by clicking on the **Globals** tab. The **Globals** section corresponds to the variables that are set in the `[global]` section of `/usr/local/etc/smb.conf`.

.10.1.2. Global Settings

Whether swat is used or `/usr/local/etc/smb.conf` is edited directly, the first directives encountered when configuring Samba are:

`workgroup`

The domain name or workgroup name for the computers that will be accessing this server.

`netbios name`

The NetBIOS name by which a Samba server is known. By default it is the same as the first component of the host's DNS name.

`server string`

The string that will be displayed in the output of `net view` and some other networking tools that seek to display descriptive text about the server.

.10.1.3. Security Settings

Two of the most important settings in `/usr/local/etc/smb.conf` are the security model and the backend password format for client users. The following directives control these options:

`security`

The two most common options are `security = share` and `security = user`. If the clients use usernames that are the same as their usernames on the FreeBSD machine, user level security should be used. This is the default security policy and it requires clients to first log on before they can access shared resources.

In share level security, clients do not need to log onto the server with a valid username and password before attempting to connect to a shared resource. This was the default security model for older versions of Samba.

`passdb backend`

Samba has several different backend authentication models. Clients may be authenticated with LDAP, NIS+, an SQL database, or a modified password file. The default authentication method is `smbpasswd`, and that is all that will be covered here.

Assuming that the default `smbpasswd` backend is used, `/usr/local/etc/samba/smbpasswd` must be created to allow Samba to authenticate clients. To provide UNIX® user accounts access from Windows® clients, use the following command to add each required user to that file:

```
# smbpasswd -a username
```

Note:

The recommended backend is now `tdbsam`. If this backend is selected, use the following command to add user

accounts:

```
# pdbedit -a -u username
```

This section has only mentioned the most commonly used settings. Refer to the Official Samba HOWTO for additional information about the available configuration options.

10.2. Starting Samba

To enable Samba at boot time, add the following line to `/etc/rc.conf`:

```
samba_enable="YES"
```

Alternately, its services can be started separately:

```
nmbd_enable="YES"
```

```
smbd_enable="YES"
```

To start Samba now:

```
# service samba start

Starting SAMBA: removing stale tdbs :

Starting nmbd.

Starting smbd.
```

Samba consists of three separate daemons. Both the nmbd and smbd daemons are started by `samba_enable`. If winbind name resolution services are enabled in `smb.conf`, the winbindd daemon is started as well.

Samba may be stopped at any time by typing:

```
# service samba stop
```

Samba is a complex software suite with functionality that allows broad integration with Microsoft® Windows® networks. For more information about functionality beyond the basic configuration described here, refer to`http://www.samba.org`.