

# Samba for OS/2 and eComStation

Neil Waldhauer

# Introduction

Samba for OS/2 and eComStation comes in two varieties, the client and the server. The client allows users access to a shared drive on another computer. The client is installed on eComStation by default.

Samba server allows users to share files with users on other computers on the local network. This presentation will concentrate on using Samba server.

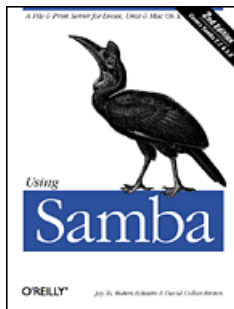
# Credits

Samba for OS/2 and eComStation is intended to replace the IBM File and Print Services that came with OS/2.

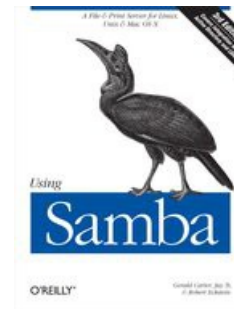
Samba is currently maintained by BitWise, Alex Taylor and Paul Smedley.

# Books about Samba

The fine points of Samba configuration are tedious to learn. The O'Reilly press book may be helpful. The full text of the book is available on-line.



Using Samba, 2nd Edition  
(free on-line)



Using Samba, 3rd Edition  
(subscription required)

# Samba Server Installation

The software can be downloaded from the Samba for eCS (OS/2) website.

<http://trac.netlabs.org/samba>

Samba Server version 1.1.4.804.

RxLib4Smb version 1.1.1

Set the TMPDIR environment variable.

# Installation Files

The installation files for Samba are

Samba Server

<ftp://ftp.netlabs.org/pub/samba/server/samba-1-1-4-804.wpi> > (17,908,342 bytes)

Rexx Libraries

<ftp://ftp.netlabs.org/pub/samba/server/rxlib4smb.wpi> (1,154,898)

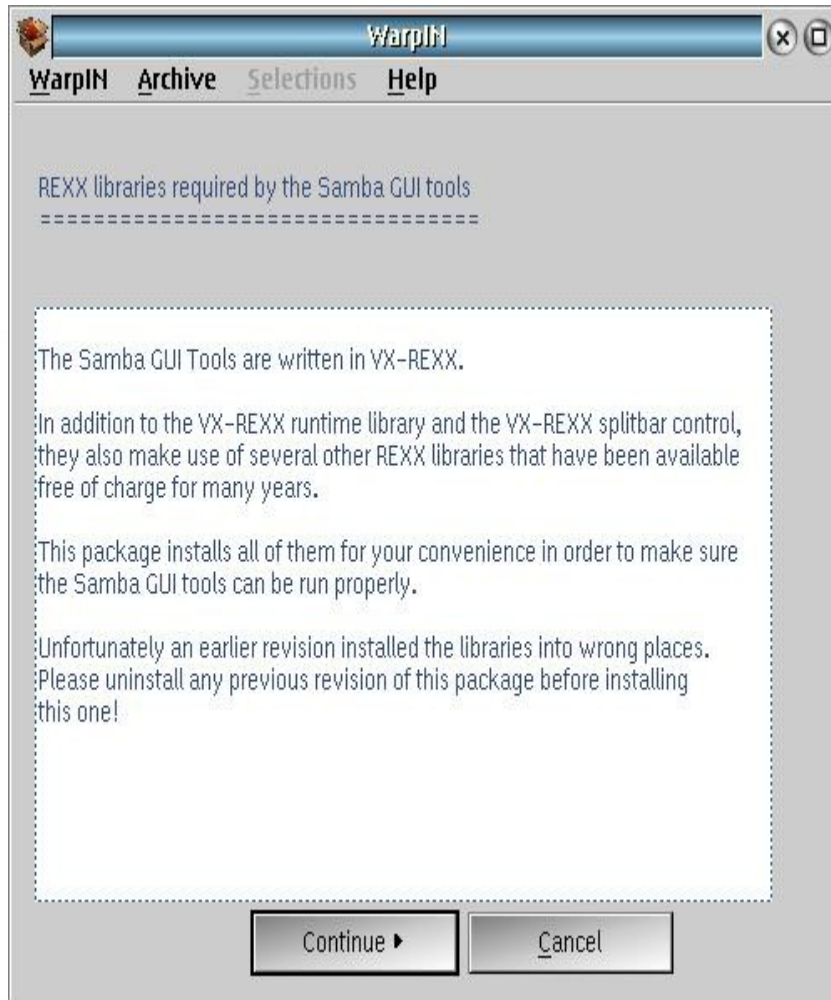
# Installation details

Install the REXX Libraries first, then the Samba Server. At the end of the Samba Server installation, you will see a configuration screen in which you will set the administrator password.

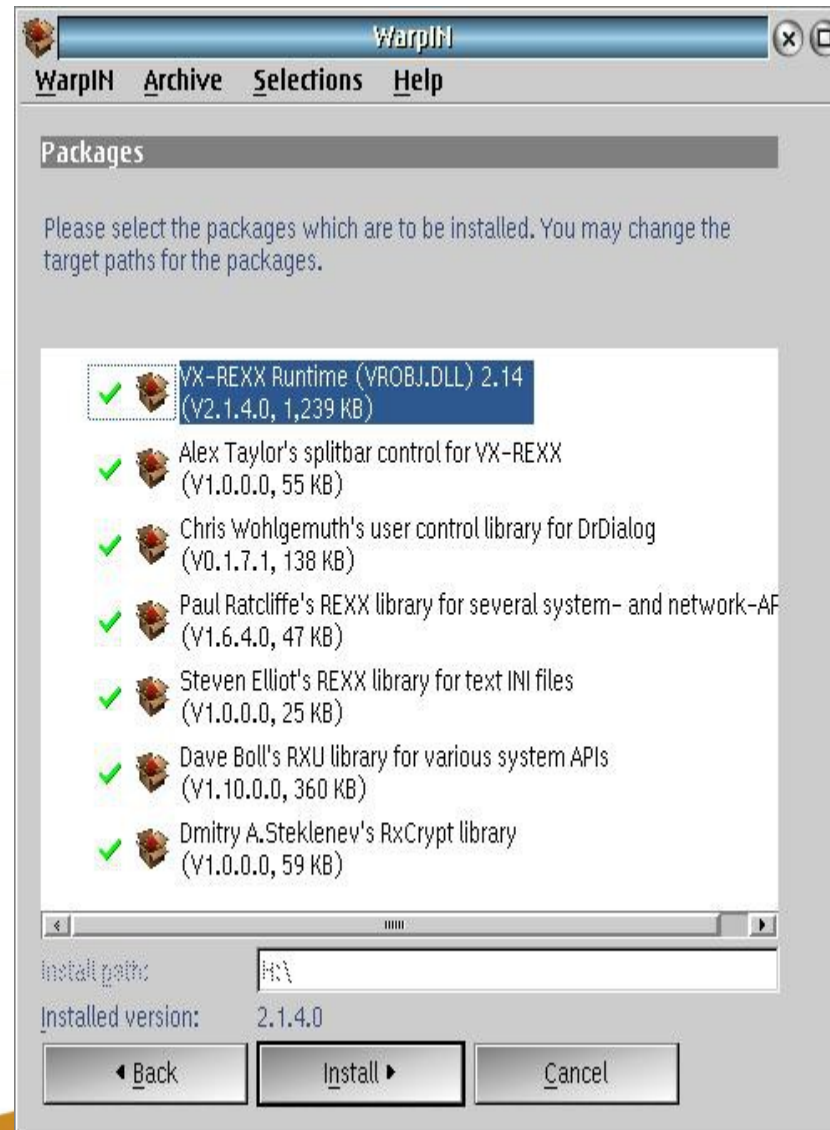
# Installation Failures

If this configuration screen does not appear, or if you do not set an administrator password, then the installation has failed. Sometimes a second attempt to install will be successful. Other times, re-installing the prerequisite REXX libraries can help.

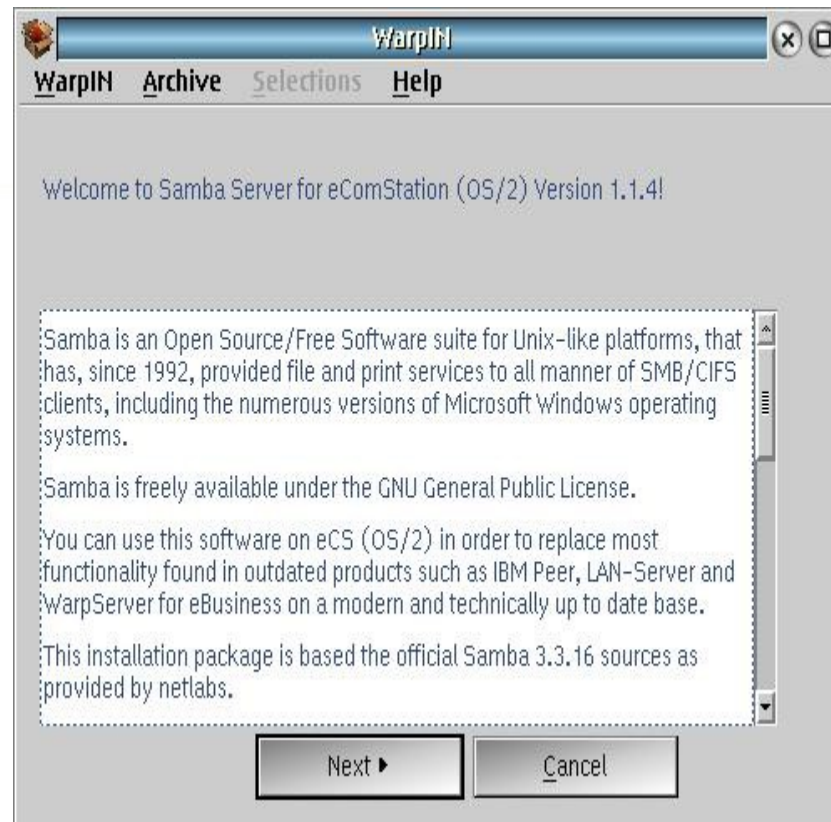
# REXX libraries required by Samba



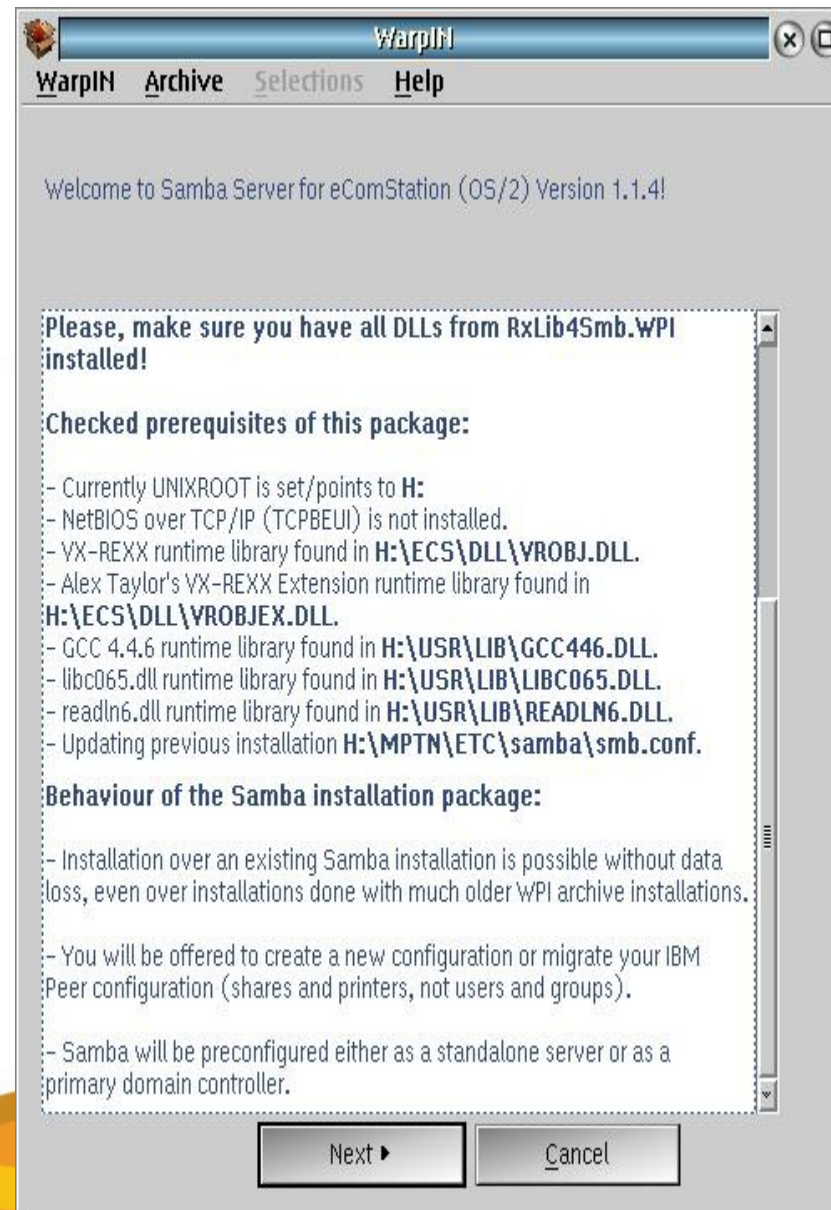
# REXX Packages



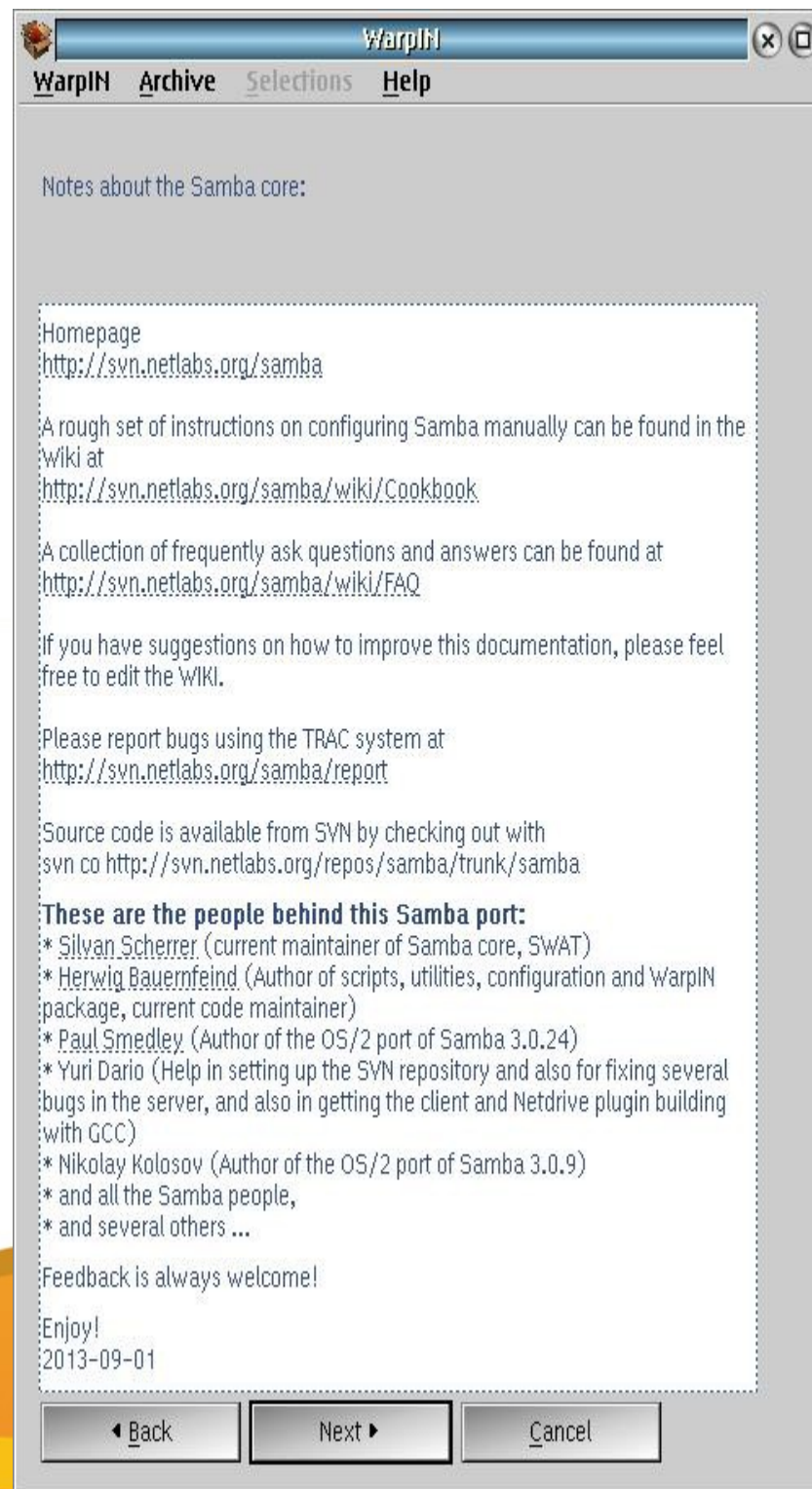
# Samba Server for eComStation (OS/2)



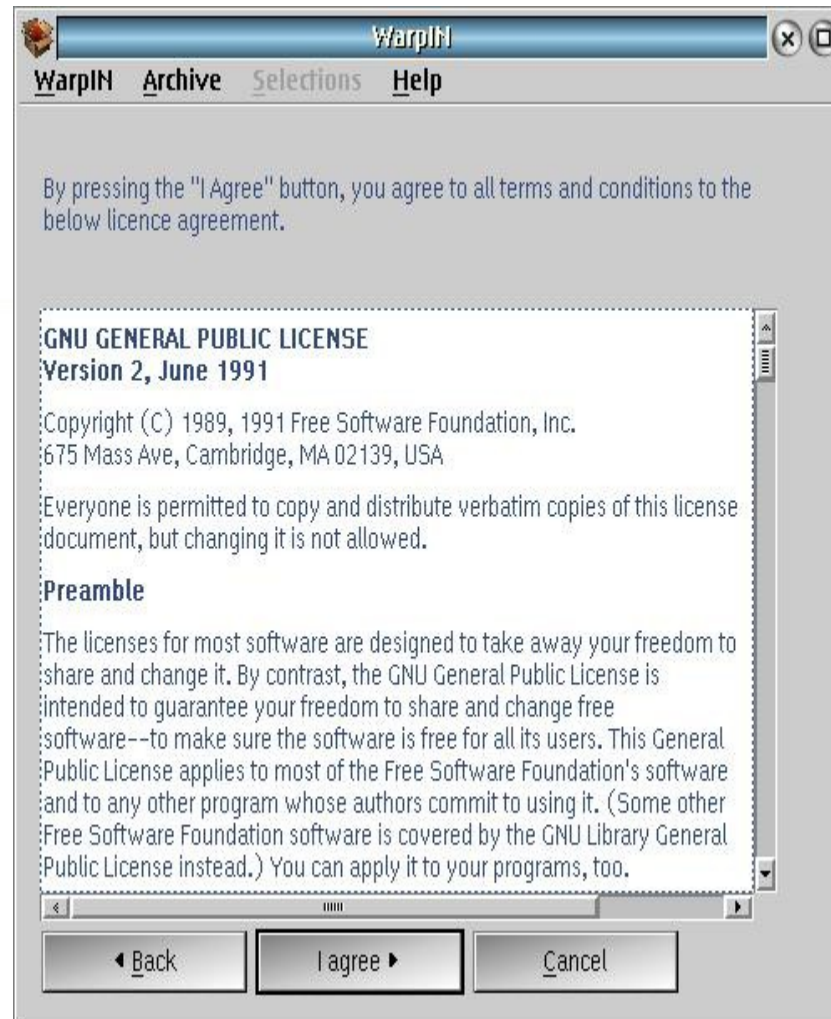
# Prerequisites



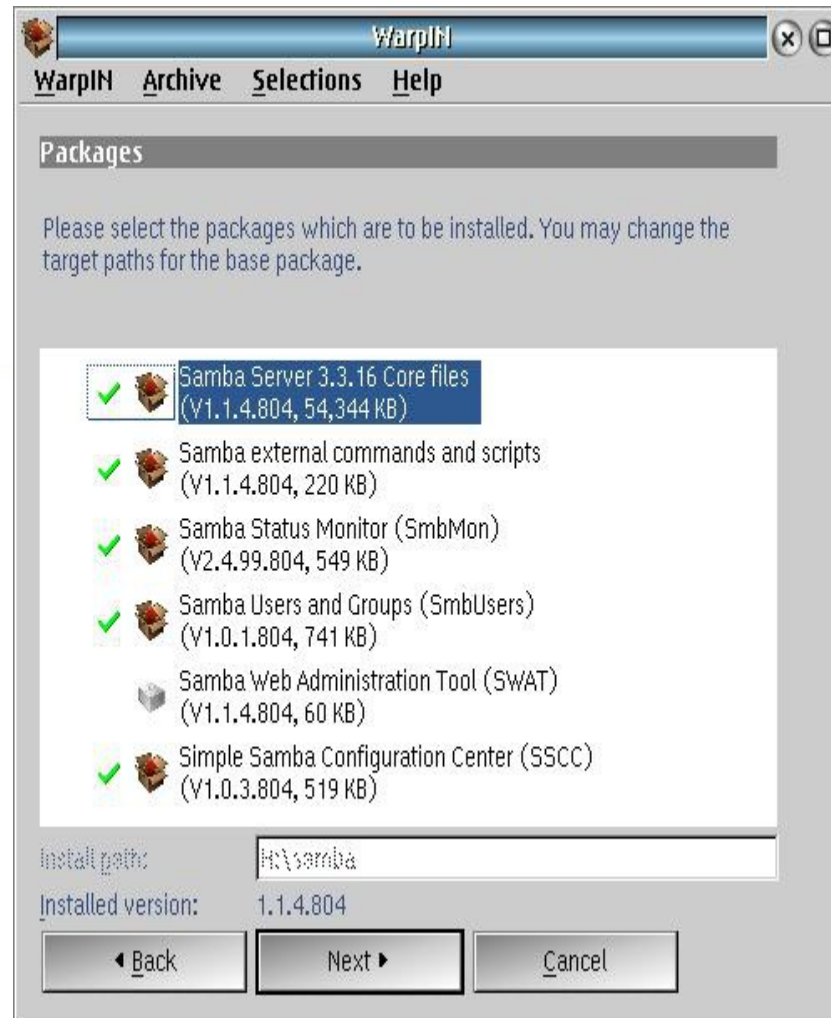
# Notes on Samba



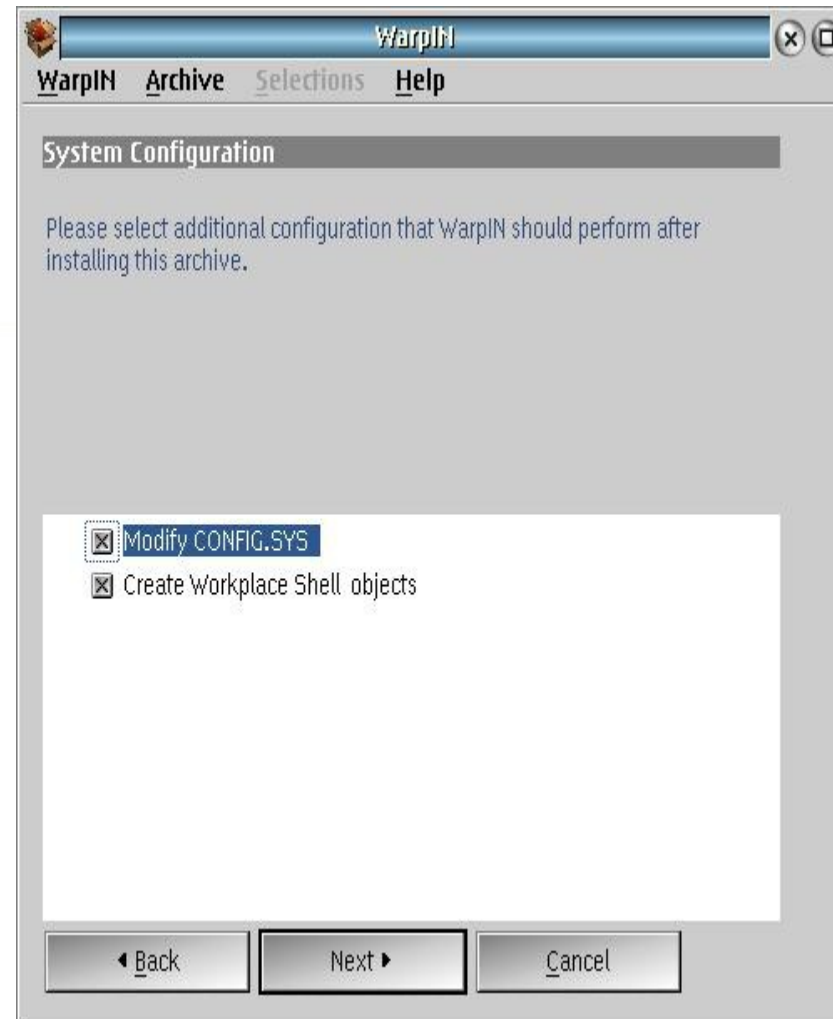
# License



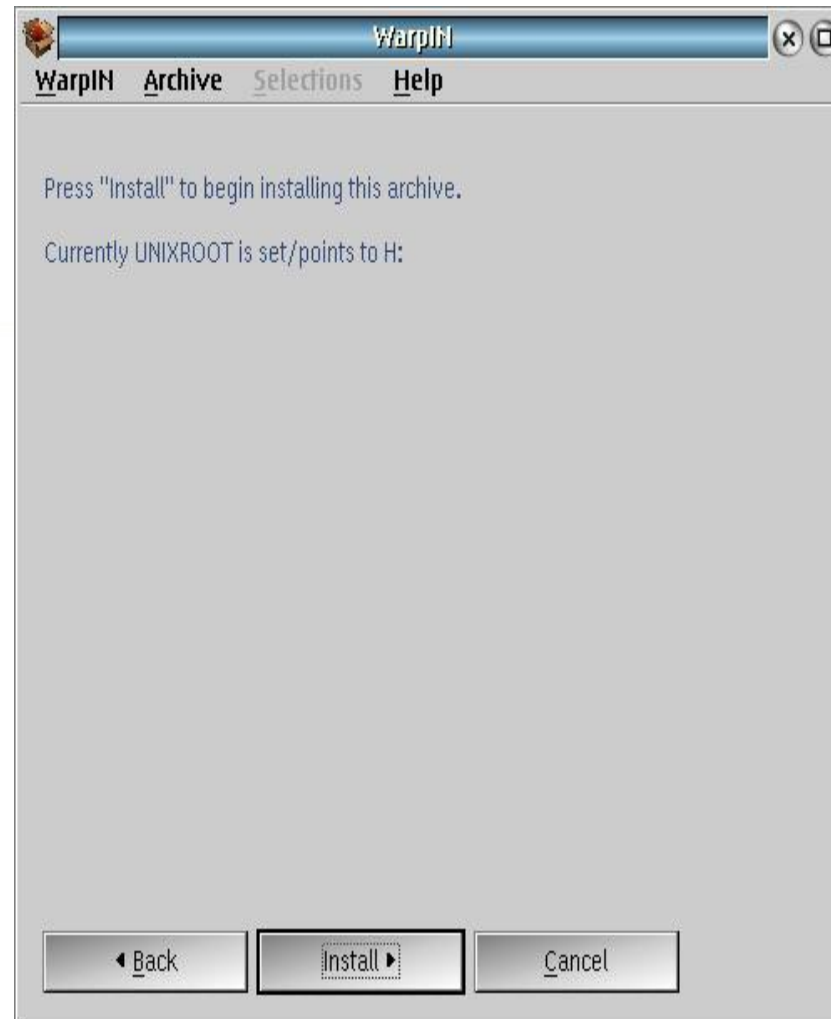
# Packages



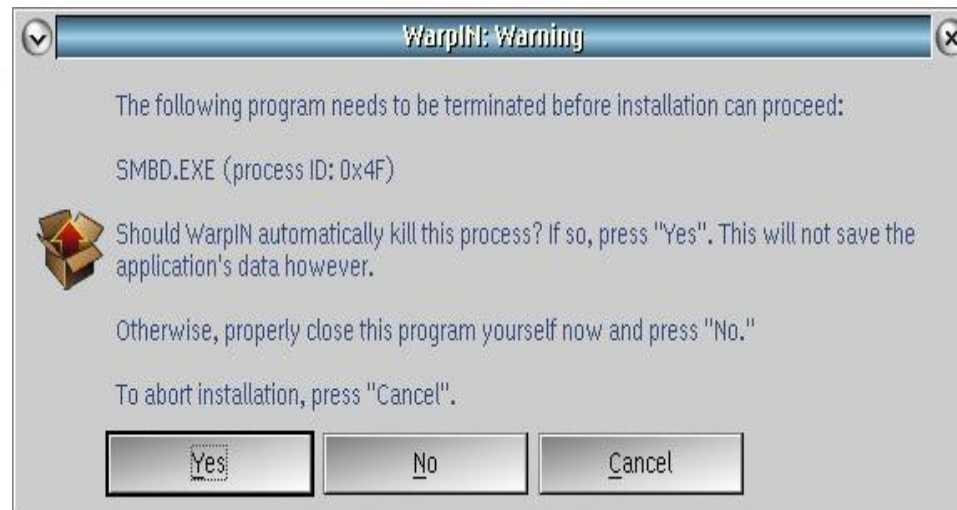
# System Configuration



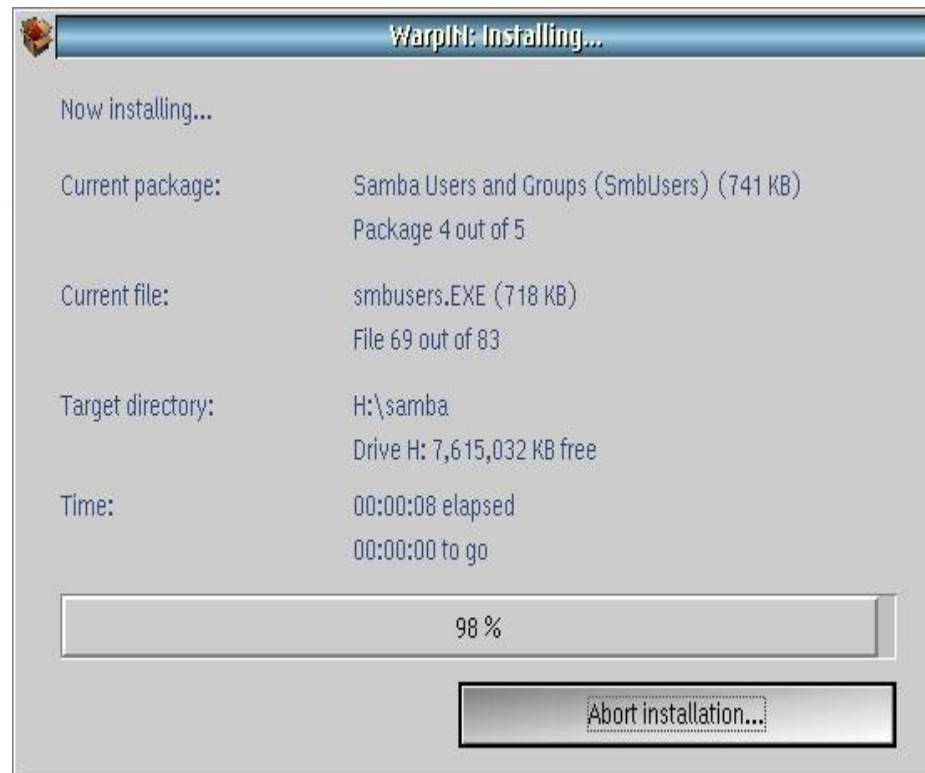
# Environment variables



# Terminating existing Samba Server before installation



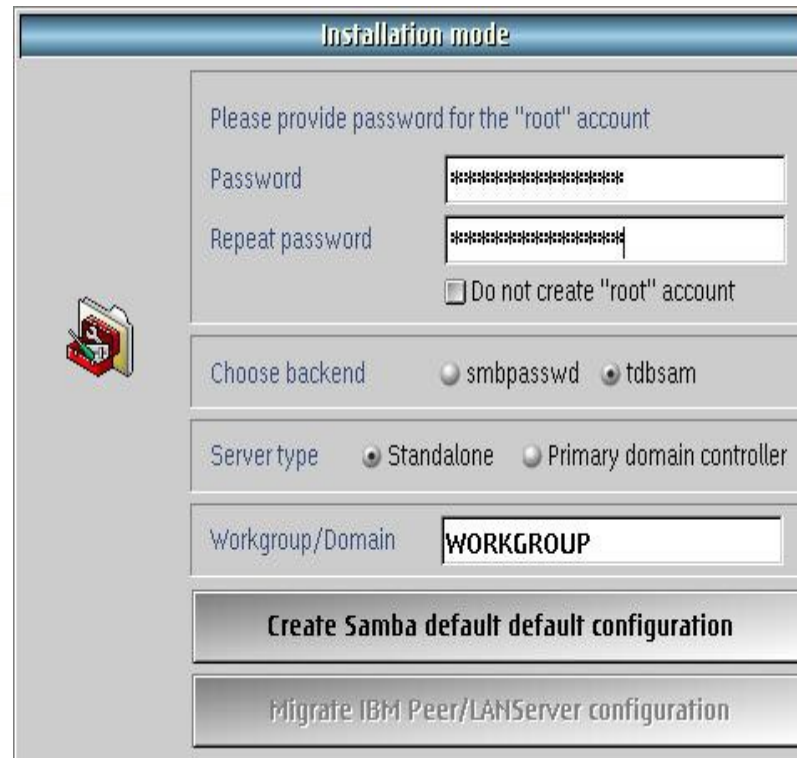
# Installing...



# Installation mode



# root password

A screenshot of a Samba installation window titled "Installation mode". On the left is a folder icon. The main area contains several sections: a password prompt "Please provide password for the 'root' account" with "Password" and "Repeat password" fields (both masked with asterisks) and a checkbox "Do not create 'root' account" which is unchecked; a "Choose backend" section with radio buttons for "smbpasswd" and "tdbsam" (selected); a "Server type" section with radio buttons for "Standalone" (selected) and "Primary domain controller"; a "Workgroup/Domain" text field containing "WORKGROUP"; and two buttons at the bottom: "Create Samba default default configuration" and "Migrate IBM Peer/LANServer configuration".

Installation mode

Please provide password for the "root" account

Password

Repeat password

☐ Do not create "root" account

Choose backend ☐ smbpasswd ☒ tdbsam

Server type ☒ Standalone ☐ Primary domain controller

Workgroup/Domain

Create Samba default default configuration

Migrate IBM Peer/LANServer configuration

# Setting up Samba to start every time

I prefer to set up Samba to start from the Workplace Shell Startup Folder. The script I use is shown below. I create a program object of the script, with the Samba installation directory as the working directory, and place it in the Startup Folder.

```
/* Script to start Samba on Blonde Guy server */  
say "wait 10 seconds before starting Samba"  
'sleep 10'  
say "starting Samba"  
call smb 'start'  
say "Samba started"  
return
```

# Installation Complete

Restart the machine to let everything take effect, and Samba server is ready to configure.

# Installing the client

Samba client software is installed by eCS 2.1 and newer. Since then, newer versions of the client have been released.

Samba client software can be found on the same website as the server files.

For OS/2, NetDrive is required. For eCS, the EVFS program supplied with the operating system will work.

# Installation Files

Samba client plugin

[ftp://ftp.netlabs.org/pub/samba/client/ndpsmb-2\\_2\\_0-Beta4-20140226.zip](ftp://ftp.netlabs.org/pub/samba/client/ndpsmb-2_2_0-Beta4-20140226.zip) (1,754,798 bytes)

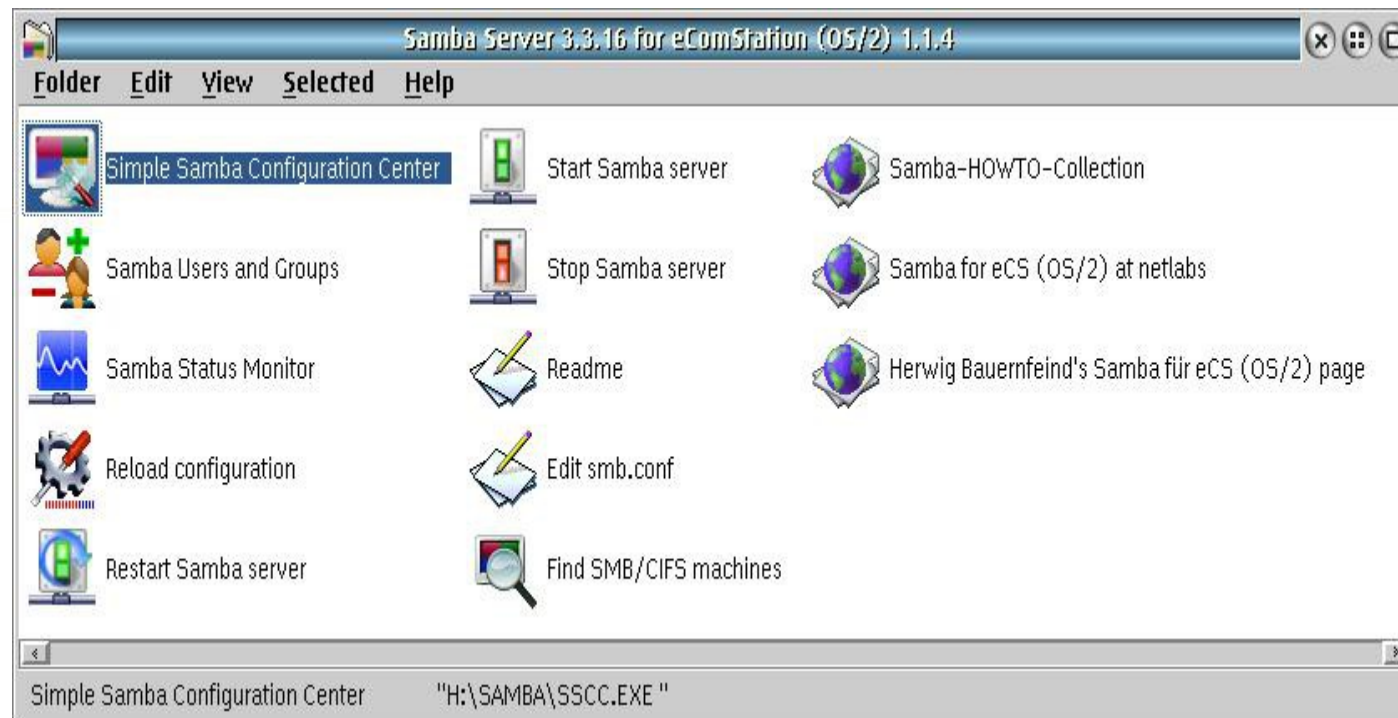
Samba client GUI

<ftp://ftp.netlabs.org/pub/samba/client/evfsgui/evfsgui-25RC7-20141213.zip> (1,154,898)

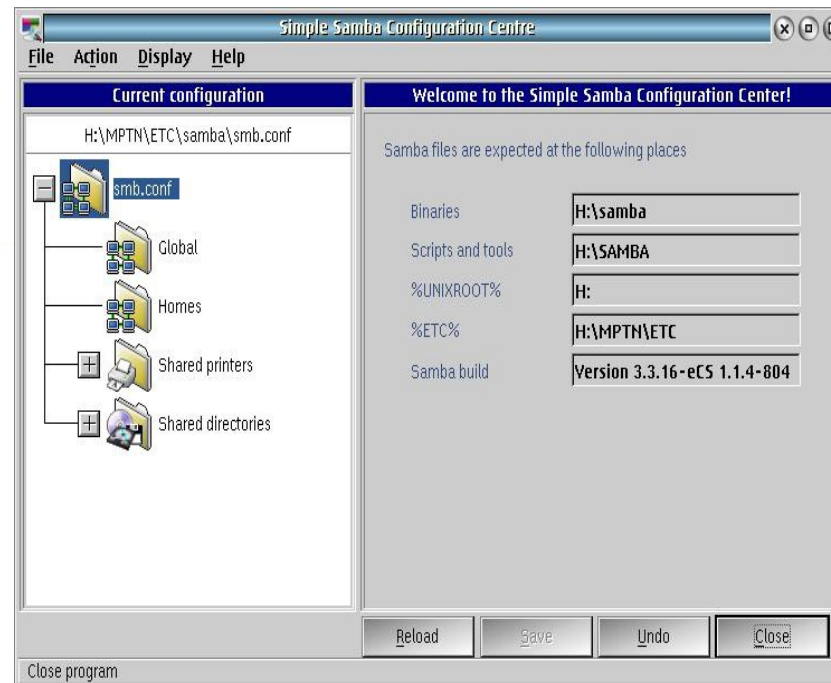
Samba client CLI

[ftp://ftp.netlabs.org/pub/samba/client/smbclut/smbclut-1\\_1\\_2-760.zip](ftp://ftp.netlabs.org/pub/samba/client/smbclut/smbclut-1_1_2-760.zip) (1,154,898)

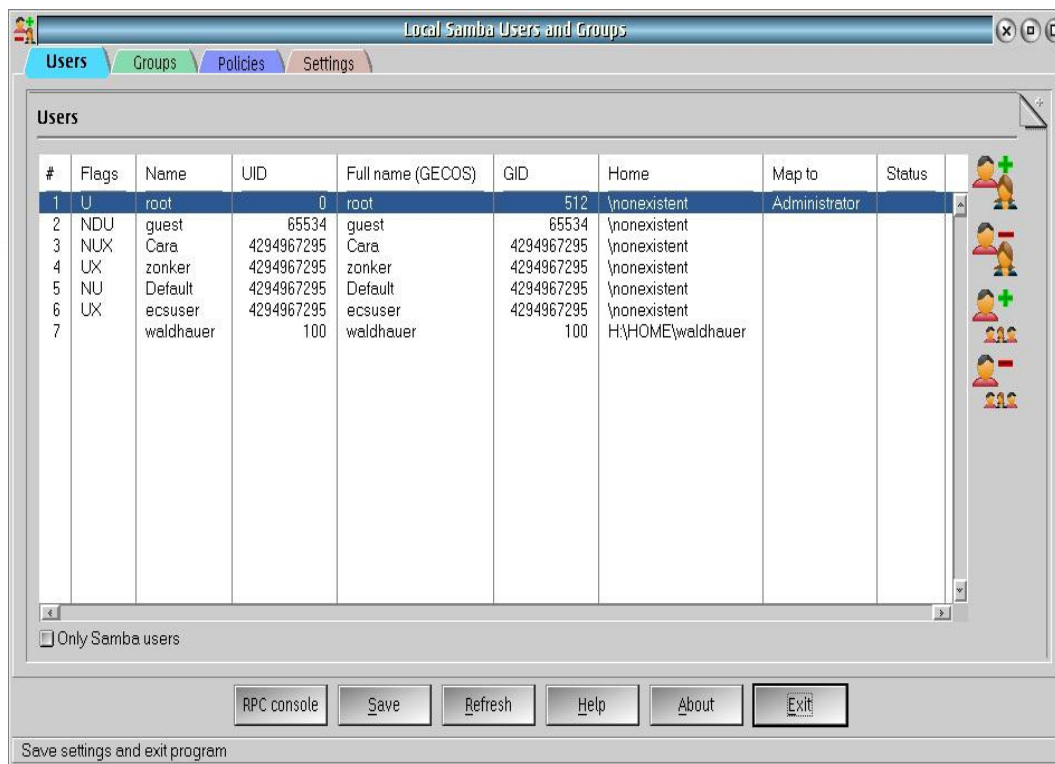
# Server Programs



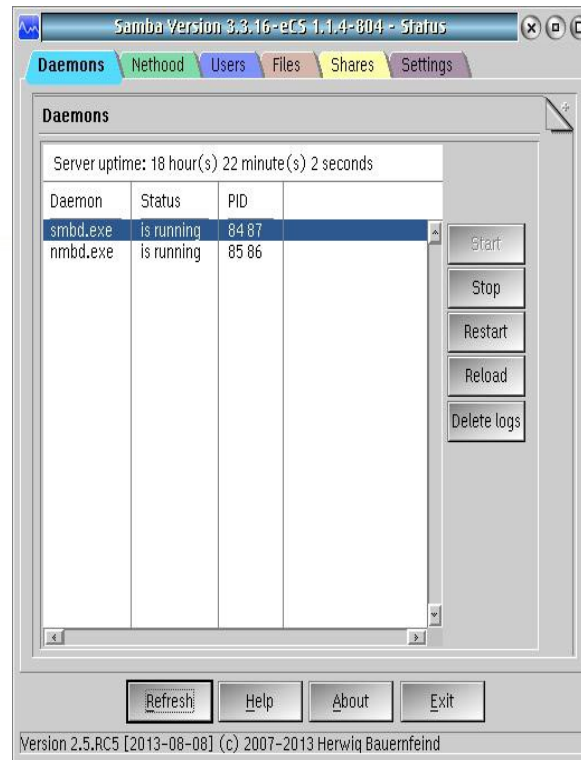
# Simple Samba Configuration Center



# Samba Users and Groups



# Samba Status Monitor



# Reload Configuration

After making configuration changes, Reload Configuration invokes a script to notify the running Samba server to use the new configuration.

# Controlling Samba Server

## **Restart Samba Server**

Uses the command line tool smb to stop then start the Samba Server.

## **Start Samba Server**

Uses the command line tool smb to start the Samba Server.

## **Stop Samba Server**

Uses the command line tool smb to stop the Samba Server.

# Other Samba functions

## **Edit smb.conf**

Directly edit the Samba configuration file smb.conf, which is in plain text.

## **Find SMB/CIFS machines**

Uses the command line tool FINDSMB to find other machines that are visible on the local network.

# Resources

Here are the resources for Samba networking on eCS and OS/2 that I have found on the web.

- OS/2 Release Notes has a nicely done article.

<http://os2notes.duckdns.org/os2win.html>

- OS/2 World forum discussion on installing Samba

<http://www.os2world.com/forum/index.php/topic,720.0.html>

# EVFS command line

eComStation Virtual File System Command Line Utility, Version 1.0.5.5

(C) Copyright Blueprint Software Works, 2000-2006.

Syntax:

EVFSCLI command [parameters]

commands are: ATTACH, CHECK, CREATEMP, DELETEMP, DETACH, MOUNT, QUERY, REFRESH, UNMOUNT, WAIT.

# EVFSCLI ATTACH x:

where x is the drive letter to be created.

Example: evfscli attach s:

# EVFSCLI CHECK [period retries]

default values are period=10 (in seconds),  
retries=infinity

Example: `evfscli check 1 5`

# EVFSCLI CREATEEMP path

where path is the mount point to be created.

Example: `evfscli createmp s:\os2`

# EVFSCLI DELETEMP path

where path is the mount point to be created.

Example: `evfscli deletemp s:\os2`

# EVFSCLI DETACH x:

where x is the drive letter to be deleted.

Example: evfscli detach s:

# EVFSCLI MOUNT

EVFSCLI MOUNT type mountpoint resource {R|W} [F]

where

type        type of the resource;

mountpoint the mount point that the resource has to be mounted to (the mount point must exist already)

resource    resource description;

{R|W}       R for read only access, W for read/write;

[F]        force mount and do not check whether the resource already exists in the mount point.

# EVFSCLI QUERY

EVFSCLI QUERY {PATH | IFS | CTL | MOUNT  
mountpoint}

Example: evfscli query mount s:\

# EVFSCLI REFRESH path [S]

where

path the directory to be refreshed

[S] refresh subdirectories

# EVFSCLI UNMOUNT

EVFSCLI UNMOUNT mountpoint index  
where

mountpoint the mount point that the  
resource is to be unmounted from;

index index of the resource to be  
unmounted (0 - all resources).

# EVFSCLI WAIT [wait\_time retries]

default values are wait\_time=10 (in seconds),  
retries=3.

Example: `evfscli wait 1 5`

# Samba command line example

```
/*mount a remote resource from command line.  
For the resource description, type out the evp  
file for the resource. */
```

```
evfscli attach r:
```

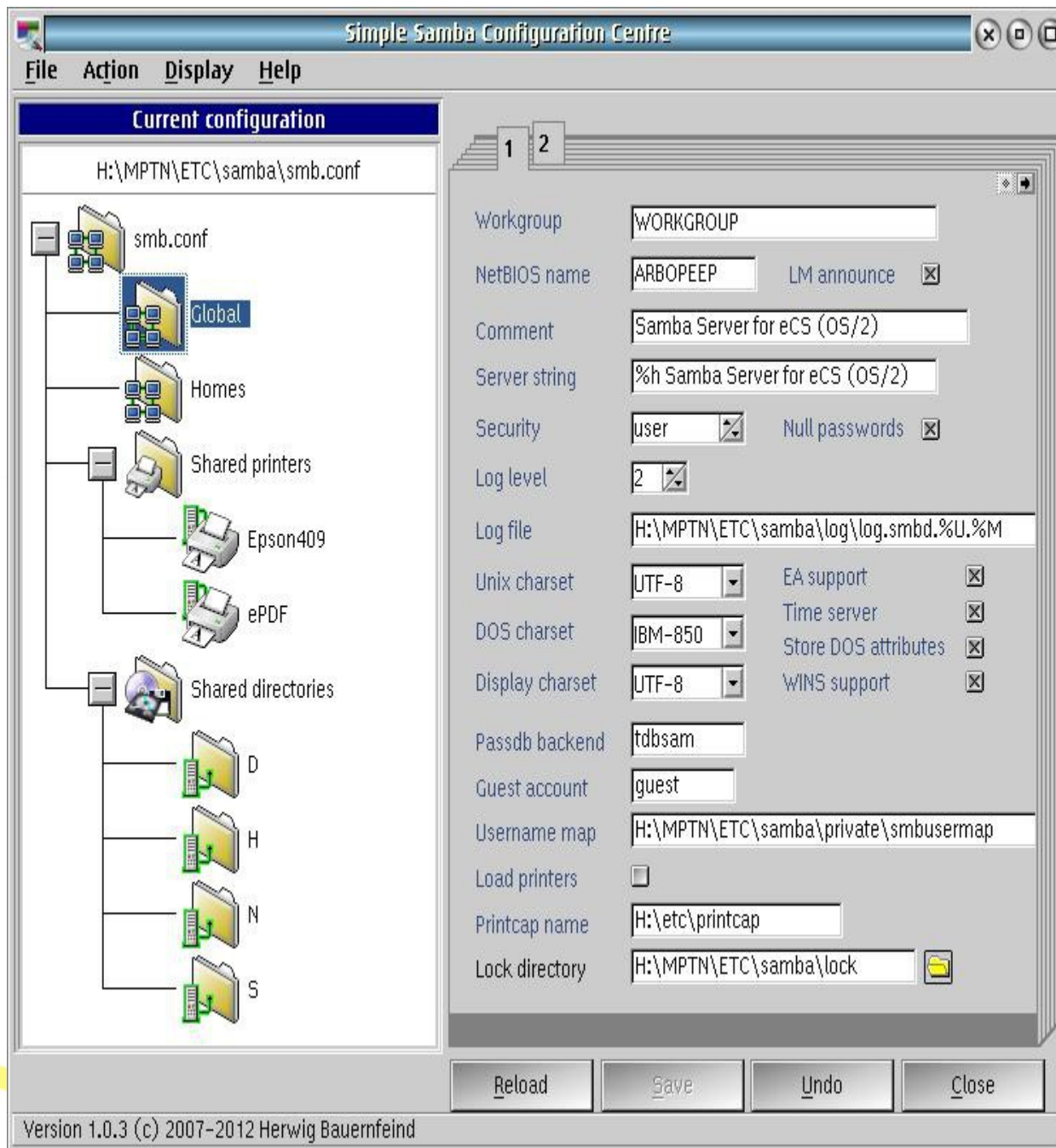
```
evfscli mount smbfs R:\  
\BEACH2:SERBOPEEP\JFS@zonker;WORKGROUP=BEACH2;  
SERVRBOPEEP;SHARE=JFS;USER=zonker;PASSWORD=;SPA  
SSWORD=666666666666664D6F726E696E6TER=WORKGROUP;M  
ASTERTYPE=1;CTO=10;CLD=32;EASUPPORT=1 W
```

# Configuration file

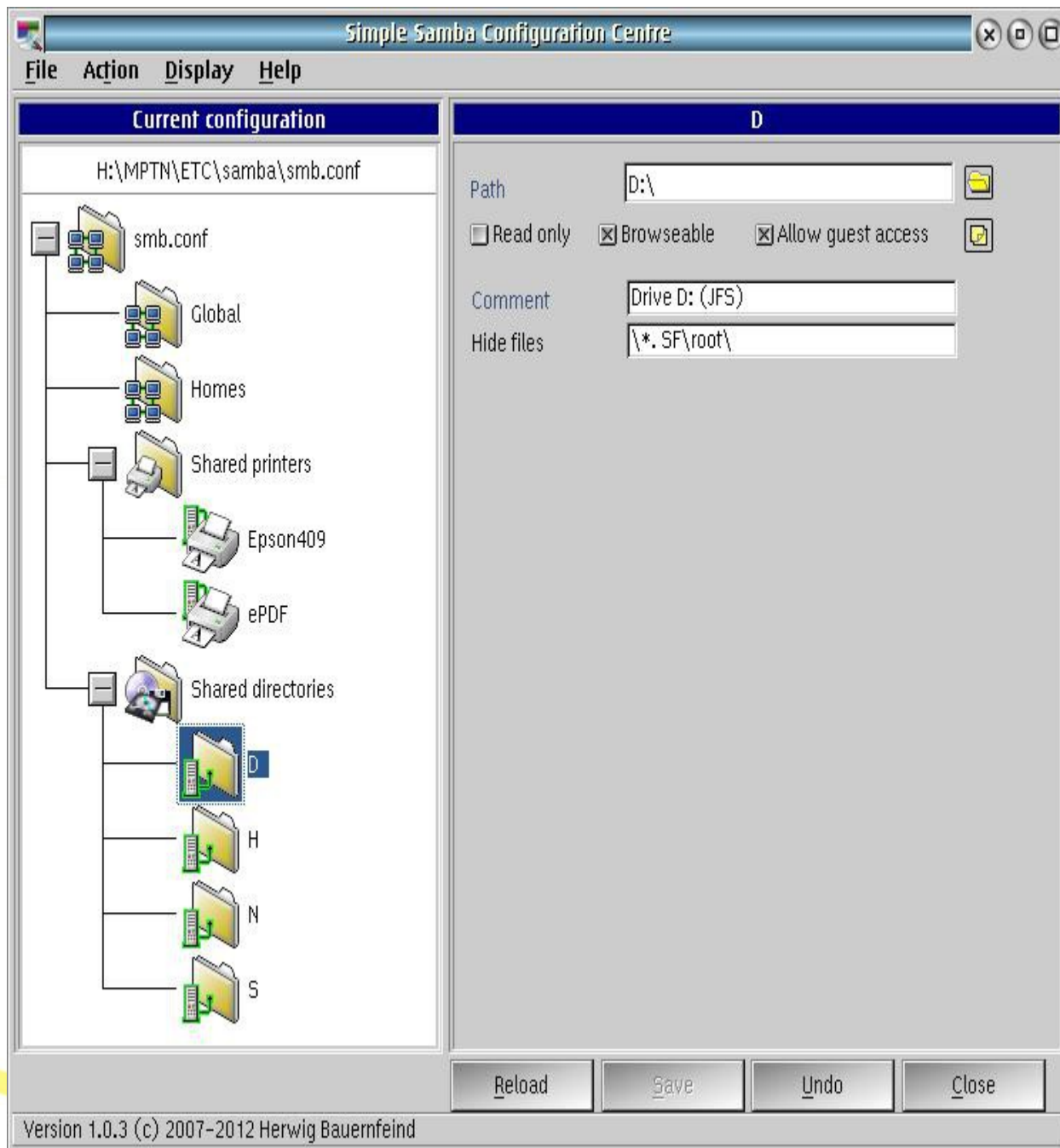
When Samba server starts, it reads configuration files that define what network resources are to be shared and which users and groups have access to those network resources.

The configuration file, `samba\smb.conf`, is located in the `%ETC%` directory.

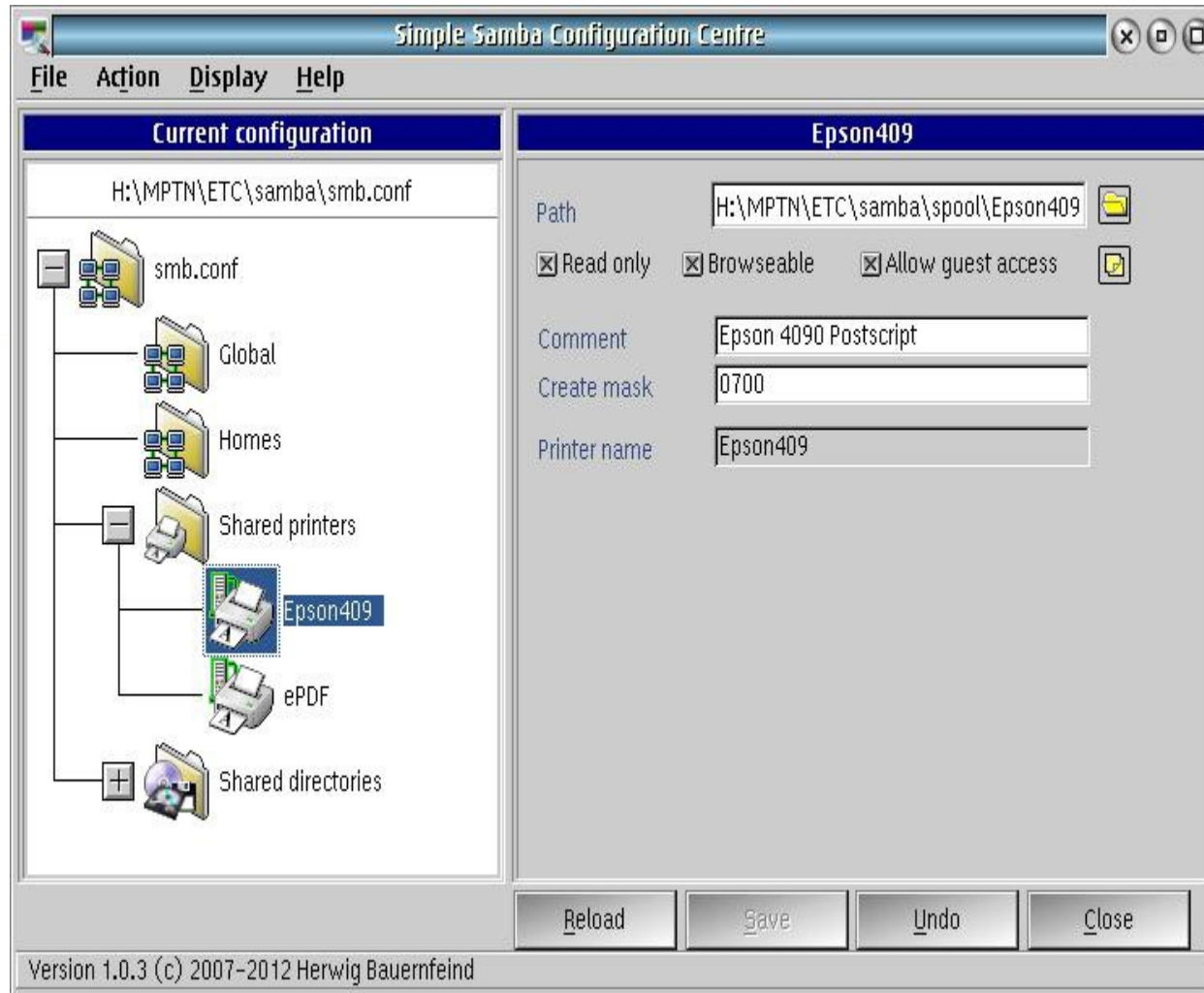
# Shared Directories



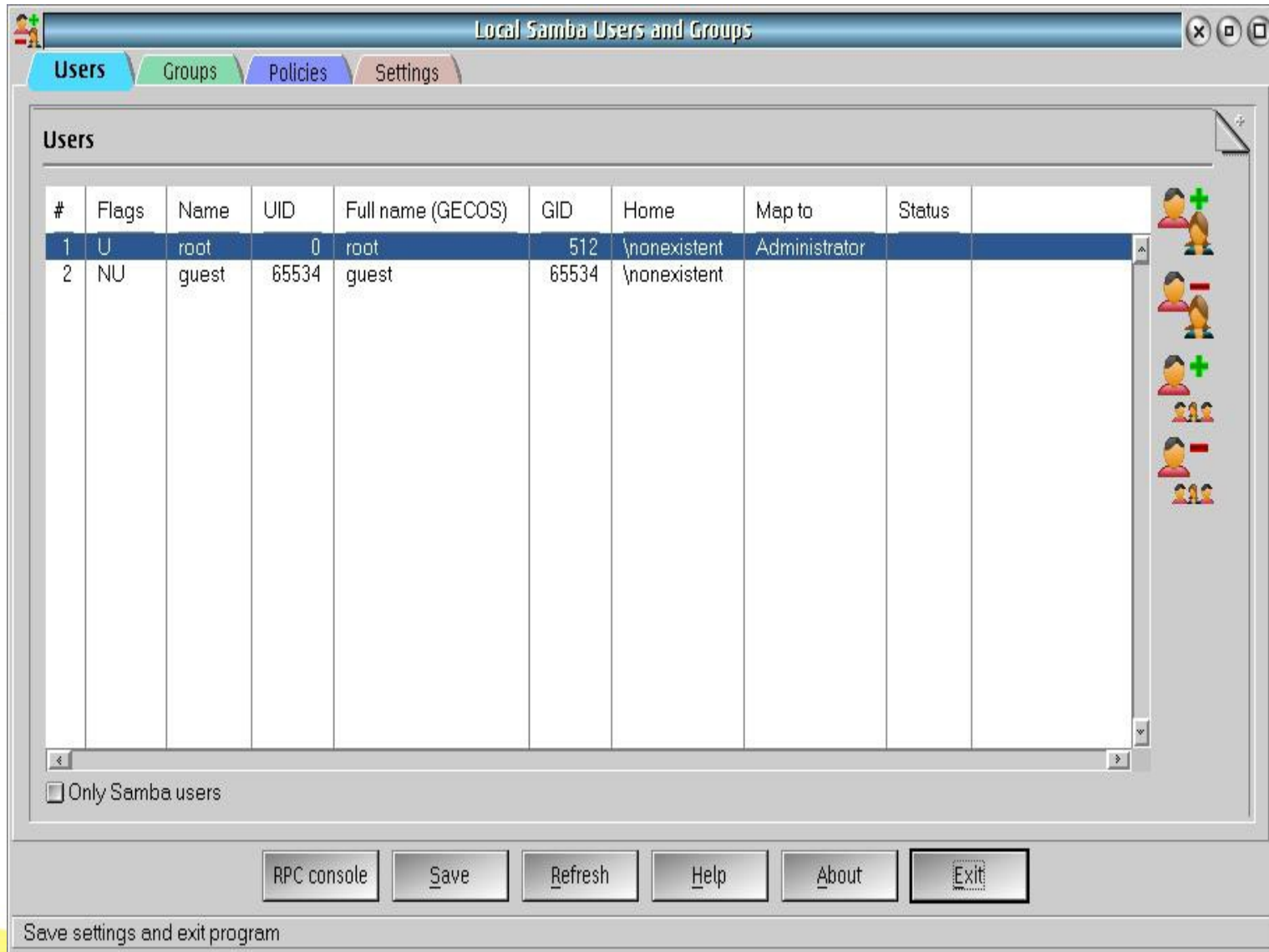
# Configuring a shared directory



# Shared Printers



# Users and Groups



# Add an administrator

**Add user**

New user:  UID: 100 ☐ Only kLIBC user

Full name:

Account description:

Password:  ☐ No Password

Repeat Password:  ☐ Machine account

☒ Allowed to use SWAT

☒ Password does not expire

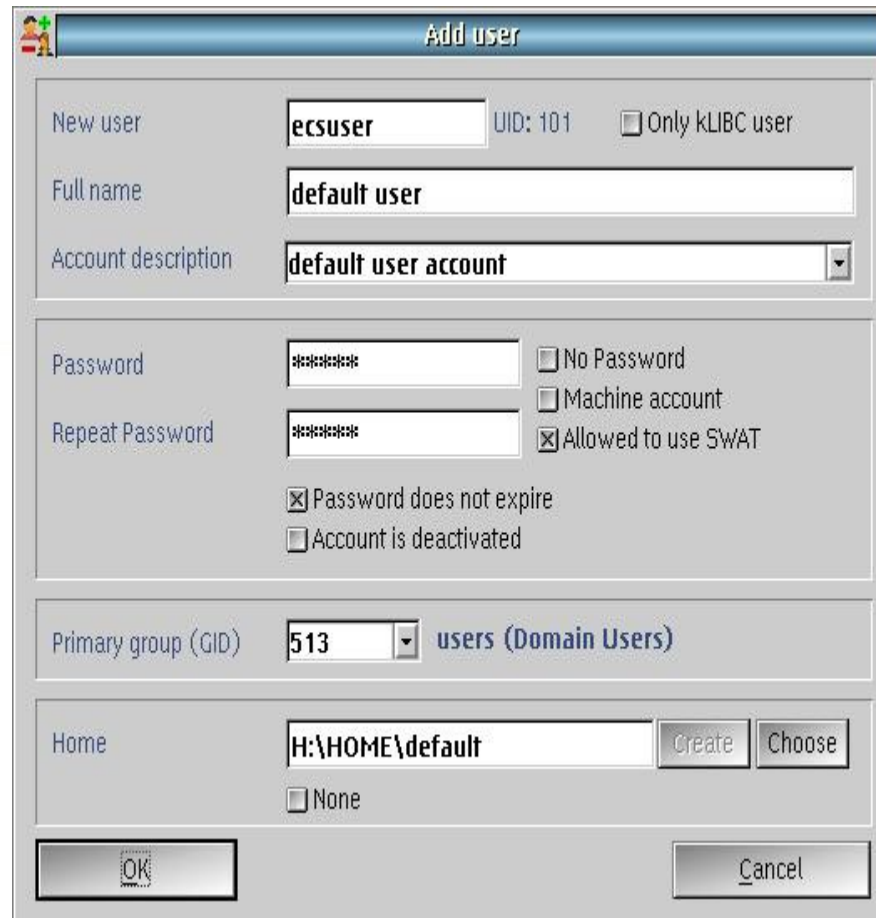
☐ Account is deactivated

Primary group (GID):

Home:

☐ None

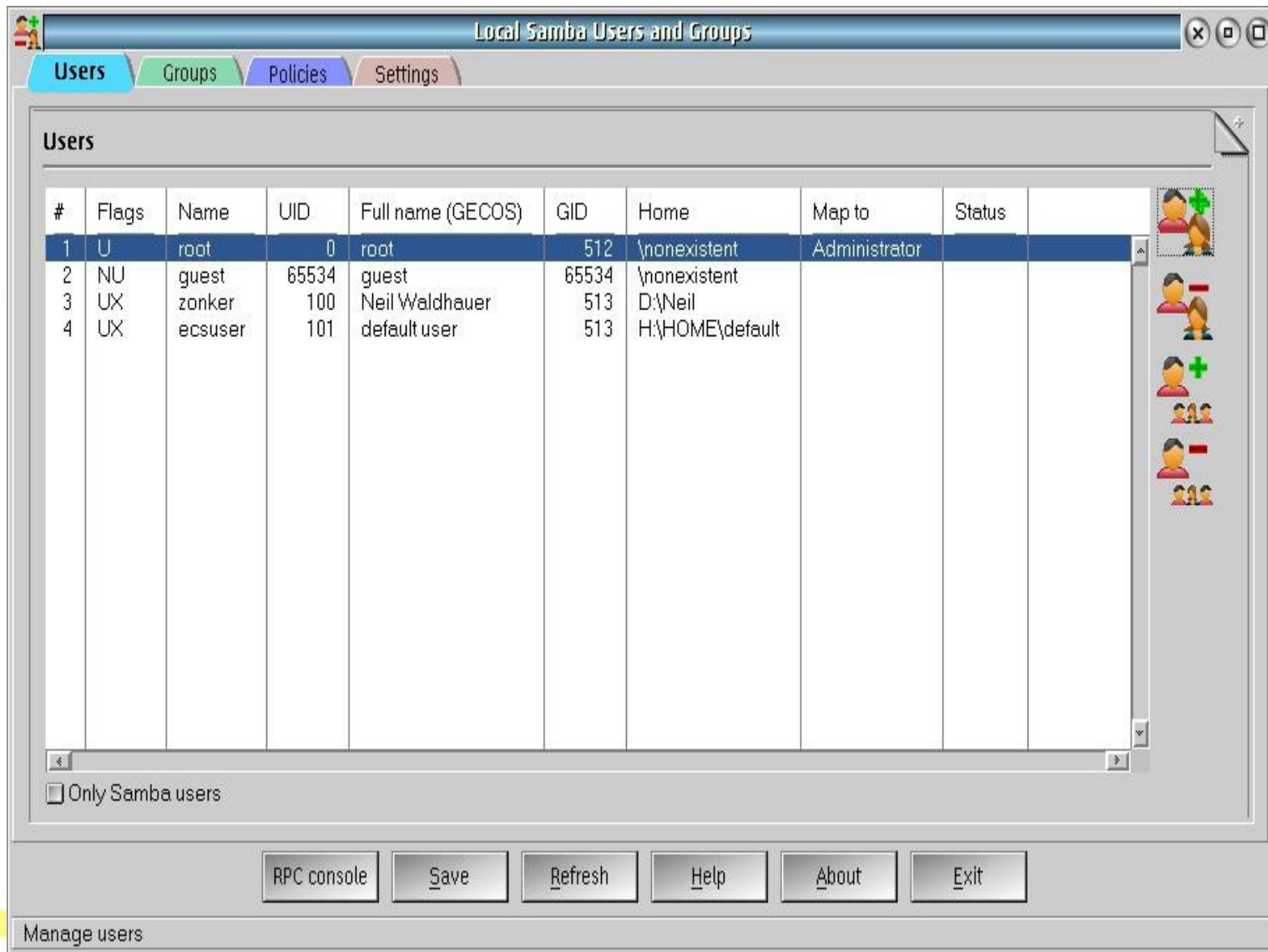
# Add a user



The screenshot shows a 'Add user' dialog box with the following fields and options:

- New user:**  UID: 101 ☐ Only kLIBC user
- Full name:**
- Account description:**
- Password:**  ☐ No Password
- Repeat Password:**  ☐ Machine account
- ☒ Allowed to use SWAT
- ☒ Password does not expire
- ☐ Account is deactivated
- Primary group (GID):**  **users (Domain Users)**
- Home:**     
☐ None
-

# user list



# Conclusion

It's October 2015, and I've been using Samba successfully in my Local Network. I can interoperate between iOS, Linux, Mac OS, OS/2, eComStation and Windows.

I have not yet tested with Android, but it is likely to work.