



*Linux Integration
with Windows (Samba)*
(Course Code QLX26)

Student Exercises

ERC 4.0

IBM Certified Course Material

Trademarks

IBM® is a registered trademark of International Business Machines Corporation.

The following are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

AIX®

Hummingbird®

Perform™

PowerPC®

PS/2®

SPT™

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

Pentium is a trademark of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States and other countries.

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

October 2003 Edition

The information contained in this document has not been submitted to any formal IBM test and is distributed on an “as is” basis without any warranty either express or implied. 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 result elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk. The original repository material for this course has been certified as being Year 2000 compliant.

© Copyright International Business Machines Corporation 1999, 2003. All rights reserved.

This document may not be reproduced in whole or in part without the prior written permission of IBM.

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Trademarks.	v
Exercises Description	vii
Exercise 1. Installing Samba	1-1
Exercise 2. Network Browsing	2-1
Exercise 3. Authentication	3-1
Exercise 4. File Sharing	4-1
Exercise 5. Printer Sharing	5-1
Exercise 6. Windows Domain Support	6-1
Exercise 7. Configuring Samba with SWAT	7-1

Trademarks

The reader should recognize that the following terms, which appear in the content of this training document, are official trademarks of IBM or other companies:

IBM® is a registered trademark of International Business Machines Corporation.

The following are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

AIX®

Hummingbird®

Perform™

PowerPC®

PS/2®

SP™

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

Pentium is a trademark of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States and other countries.

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

Exercises Description

Each exercise in this course is divided into sections as described below. Select the section that best fits your method of working in labs. You may elect to use a combination of these sections as appropriate.

Exercise Instructions — This section contains what it is you are to accomplish. There are no definitive details on how to perform the tasks. You are given the opportunity to work through the exercise given, using what you learned in the unit presentation, utilizing the Student Notebook for the unit, your past experience and maybe a little intuition.

Exercise Instructions With Hints — This section is an exact duplicate of the Exercise Instructions section except that in addition, specific details and/or hints are provided to help you step through the exercise. A combination of using the Instructions section along with Instructions With Hints section can make for a rewarding combination providing you with hints only when you need them.

Exercise 1. Installing Samba

What This Exercise is About

In this exercise you are going to install Samba from source and from the RPM files that came with your distribution.

What You Should Be Able to Do

At the end of the lab, you should be able to:

- Compile and install a Samba source tarball
- Install a Samba binary RPM

Required Materials

- <http://www.samba.org>: Samba source tarball
- <http://www.redhat.com>: Samba RPMs

Exercise Instructions

Installing Samba from source

In this first part of the exercise you are going to follow all of the steps that are required to install Samba using the original source tarball which is taken from the Samba website. After this, you are briefly going to test this installation and uninstall it. The reason for this is that we want everybody in the classroom to use the same configuration and that can only be assured if everybody installs from the binary RPM that is provided with your operating system.

- ___ 1. Log in to your Linux machine as root. The password is **ibmlnx**, unless your instructor indicates otherwise.
- ___ 2. Start a graphical user environment and start a browser (Konqueror, Netscape, Mozilla or Galeon).

Your instructor might indicate that you need to configure your browser for a proxy or socks server, since the classroom might be behind a firewall.

Once you can browse the web, go to the Samba website (<http://www.samba.org>). Browse the site. Go to the download section and download the latest stable Samba release. Store it in /root

Note: In order to save download time, your instructor might already have downloaded this release and stored it locally. Check with the instructor if this is the case.

- ___ 3. Go to the /usr/src directory and unpack the source tarball.
- ___ 4. Go into the Samba source directory and look through the file "Manifest" until you see the installation directions. Read these carefully.
- ___ 5. Go into the "source" directory and run the **./configure --help** script. View the different configuration options carefully: these options decide where all the files go and what features of Samba (including experimental features) will be enabled.
- ___ 6. Run the **./configure** script without any options. Not supplying any options will ensure that we can deinstall Samba easily later on. The disadvantage is that not all binaries will end up in a place that is included in our \$PATH.

This script will take a minute or so.

- ___ 7. Run the **make** command so that all your binaries are built. (This will take 5-20 minutes, depending on the speed of your system. You might want to get a cup of coffee while the system is compiling.)
- ___ 8. Run the **make install** command so that everything is installed. Note that all files are installed relative to a directory /usr/local/samba.
- ___ 9. Create a basic smb.conf file in the directory where the config file should go (with a regular install, /usr/local/samba/lib). If you don't know where the config file should

go, run the **testparm** command which was installed as part of the **make install**. You can read the directory from the output of the testparm command.

The smb.conf file should contain a **netbios name** and a **workgroup** statement.

- ___ 10. Run the **testparm** command again.
- ___ 11. Start the **smbd** and **nmbd** daemons, both with the **-D** option so that they are started as daemons.
- ___ 12. Use the **smbclient** command to verify that your daemons are indeed functioning.
- ___ 13. Kill all Samba daemons.
- ___ 14. Use the **make uninstall** command to uninstall all Samba programs. Verify that the /usr/local/samba directory is indeed empty, except for the smb.conf file that we created ourselves, some documentation and a few lock- and logfiles. Then remove the /usr/local/samba directory completely.

Installing Samba from RPM

- ___ 15. Depending on the lab setup, the Samba RPMs that came with your distribution are located in /root or are stored on a network server. Alternatively, your instructor may have CDs to distribute to you. In any case, install all Samba RPMs and list the information on these RPMs and the files they contain.
- ___ 16. Browse the contents of the default smb.conf file that was configured on your system.
- ___ 17. Use your distributions start script to start Samba.
- ___ 18. Use the **smbclient** command again to test your local configuration.
- ___ 19. Stop the Samba daemons using your distributions scripts.

END OF LAB

Exercise Instructions With Hints

Installing Samba from source

In this first part of the exercise you are going to follow all of the steps that are required to install Samba using the original source tarball which is taken from the Samba website. After this, you are briefly going to test this installation and uninstall it. The reason for this is that we want everybody in the classroom to use the same configuration and that can only be assured if everybody installs from the binary RPM that is provided with your operating system.

- ___ 1. Log in to your Linux machine as root. The password is **ibmlnx**, unless your instructor indicates otherwise.

 » Login as **root** and provide **ibmlnx** as root password.

- ___ 2. Start a graphical user environment and start a browser (Konqueror, Netscape, Mozilla or Galeon).

Your instructor might indicate that you need to configure your browser for a proxy or socks server, since the classroom might be behind a firewall.

Once you can browse the web, go to the Samba website (<http://www.samba.org>). Browse the site. Go to the download section and download the latest stable Samba release. Store it in /root

Note: In order to save download time, your instructor might already have downloaded this release and stored it locally. Check with the instructor if this is the case.

- ___ 3. Go to the /usr/src directory and unpack the source tarball.

 » # **cd /usr/src**

 » # **tar -zxvf /root/samba-version.tar.gz**

- ___ 4. Go into the Samba source directory and browse the file “Manifest” until you see the installation directions. Read these carefully.

 » # **cd samba-version**

 » # **less Manifest**

- ___ 5. Go into the “source” directory and run the **./configure --help**. script. View the different configuration options carefully: these options decide where all the files go and what features of Samba (including experimental features) will be enabled.

 » # **cd source**

 » # **./configure --help | less**

- ___ 6. Run the **./configure** script without any options. Not supplying any options will ensure that we can deinstall Samba easily later on. The disadvantage is that not all binaries will end up in a place that is included in our \$PATH.

This script will take a minute or so.

» # **./configure**

- ___ 7. Run the **make** command so that all your binaries are built. (This will take 5-20 minutes, depending on the speed of your system. You might want to get a cup of coffee while the system is compiling.)

» # **make**

- ___ 8. Run the **make install** command so that everything is installed. Note that all files are installed relative to a directory `/usr/local/samba`.

» # **make install**

- ___ 9. Create a basic `smb.conf` file in the directory where the config file should go (with a regular install, `/usr/local/samba/lib`). If you don't know where the config file should go, run the **testparm** command which was installed as part of the **make install**. You can read the directory from the output of the `testparm` command.

The `smb.conf` file should contain a **netbios name** and a **workgroup** statement.

» # **cd /usr/local/samba/bin**

» # **./testparm**

» # **cd /usr/local/samba/lib**

» # **vi smb.conf**

Edit the file so that it looks like this

```
[global]
netbios name = system1
workgroup = lx26
```

- ___ 10. Run the **testparm** command again.

» # **/usr/local/samba/bin/testparm**

- ___ 11. Start the **smbd** and **nmbd** daemons, both with the **-D** option so that they are started as daemons.

» # **/usr/local/samba/bin/smbd -D**

» # **/usr/local/samba/bin/nmbd -D**

- ___ 12. Use the **smbclient** command to verify that your daemons are indeed functioning.

» # **/usr/local/samba/bin/smbclient -L system1 -N**

- ___ 13. Kill all Samba daemons.

» # **killall smbd**

» # **killall nmbd**

- ___ 14. Use the **make uninstall** command to uninstall all Samba programs. Verify that the `/usr/local/samba` directory is indeed empty, except for the `smb.conf` file that we created ourselves, some documentation and a few lock- and logfiles. Then remove the `/usr/local/samba` directory completely.

» # **cd /usr/src/samba-version/source**

```
» # make uninstall
» # cd /usr/local/samba
» # ls -lR | less
» # cd /root
» # rm -fr /usr/local/samba
```

Installing Samba from RPM

___ 15. Depending on the lab setup, the Samba RPMs that came with your distribution are located in /root or are stored on a network server. Alternatively, your instructor may have CDs to distribute to you. In any case, install all Samba RPMs and list the information on these RPMs and the files they contain.

```
» # ls samba*.rpm
» # rpm -qip samba*.rpm | less
» # rpm -qlp samba*.rpm | less
» # rpm -ivh samba*.rpm
```

___ 16. Browse the contents of the default smb.conf file that was configured on your system.

```
» # less /etc/samba/smb.conf
```

___ 17. Use your distributions start script to start Samba.

```
» redhat# service smb start
suse# rcnmb start; rcsmb start
```

___ 18. Use the **smbclient** command again to test your local configuration.

```
» # smbclient -L localhost -N
```

___ 19. Stop the Samba daemons using your distributions scripts.

```
» redhat# service smb stop
suse# rcsmb stop; rcnmb stop
```

END OF LAB

Exercise 2. Network Browsing

What This Exercise is About

In this exercise you are going to configure Samba for network browsing.

What You Should Be Able to Do

At the end of the lab, you should be able to:

- Configure Samba for network browsing

Exercise Instructions

Configuring Master Browsers

In this part of the exercise you need to team up with one or two partner teams, and configure your two or three systems in the same workgroup. You are going to play with various browsing settings so that different machines will (predictably) become local and domain master browser.

- ___ 1. Find one or two partner teams to work with and, if necessary, introduce yourself to each other.
- ___ 2. Move the default `smb.conf` file that came with your distribution to a safe location so that you can start off with a blank `smb.conf` file. In this new, blank file, configure a **netbios name** (your own hostname) and a **workgroup** (a name you decide on together with your partner teams). Also include a **log level = 2** statement, and make sure the log output goes to `/var/log/samba/log.%m`.
- ___ 3. Start your graphical user environment, if this has not yet been started. Open at least two terminal windows. In one window, start a **tail -f** of the file `/var/log/samba/log.nmbd`, and in another window, run the **smbclient -L localhost -N** continuously with a one second interval.
- ___ 4. Run the **uptime** command to determine which system has been running the longest. Start the Samba daemons. Wait a minute or so to see the browser elections in action. How long does it take before a master browser is elected for your workgroup? Why so long? Which system was elected and why?
- ___ 5. Stop all Samba daemons on all systems. On one system, configure Samba to be a Local Master Browser, with an **os level** of 32. Start all Samba daemons again. Which system became LMB? Why? How long did it take this time?
- ___ 6. Leave the system that became LMB running. On a non-LMB system, stop the Samba daemons and configure this system as a **preferred master** with an **os level** of 33. Start this daemon. Does this system become LMB?
- ___ 7. While you were doing these exercises, did you also see other master browsers appear in the output of the **smbclient** command? How long did this take?

WINS Service

- ___ 8. Your instructor will appoint one of the systems (or his/her own server) in the classroom as the WINS server. Configure all your systems to use this WINS server, and then run the same simulation again. Does this make a difference in the time it takes?

Clean-up

- ___ 9. Change your smb.conf file so that you are a simple system with **os level** zero, and part of the lx26 workgroup. Do not attempt to become master browser.

END OF LAB

Exercise Instructions With Hints

Configuring Master Browsers

In this part of the exercise you need to team up with one or two partner teams, and configure your two or three systems in the same workgroup. You are going to play with various browsing settings so that different machines will (predictably) become local and domain master browser.

- ___ 1. Find one or two partner teams to work with and, if necessary, introduce yourself to each other.
- ___ 2. Move the default `smb.conf` file that came with your distribution to a safe location so that you can start off with a blank `smb.conf` file. In this new, blank file, configure a **netbios name** (your own hostname) and a **workgroup** (a name you decide on together with your partner teams). Also include a **log level = 2** statement, and make sure the log output goes to `/var/log/samba/log.%m`.

```
» # mv /etc/samba/smb.conf /etc/samba/smb.conf.old
» # vi /etc/samba/smb.conf
```

The contents of the file should look like:

```
[global]
netbios name = system1
workgroup = ourgroup
log level = 2
log file = /var/log/samba/log.%m
» # testparm
```

- ___ 3. Start your graphical user environment, if this has not yet been started. Open at least two terminal windows. In one window, start a **tail -f** of the file `/var/log/samba/log.nmbd`, and in another window, run the **smbclient -L localhost -N** continuously with a one second interval.

```
» # startx
» In one xterm/nxterm/rxvt/Konsole/gnome-terminal window:
# tail -f /var/log/samba/log.nmbd
» In another xterm/nxterm/rxvt/Konsole/gnome-terminal window:
# while true
> do
> clear
> smbclient -L localhost -N
> sleep 1
> done
```

- ___ 4. Run the **uptime** command to determine which system has been running the longest. Start the Samba daemons. Wait a minute or so to see the browser elections in action. How long does it take before a master browser is elected for your workgroup? Why so long? Which system was elected and why?
- » # **uptime**
 - » redhat# **service smb restart**
 - suse: **rcnmb start; rcsmb start**
- ___ 5. Stop all Samba daemons on all systems. On one system, configure Samba to be a Local Master Browser, with an **os level** of 32. Start all Samba daemons again. Which system became LMB? Why? How long did it take this time?
- » redhat# **service smb stop**
 - suse# **rcsmb stop; rcnmb stop**
 - » # **vi /etc/samba/smb.conf**
- Add the following lines:
- ```
local master = yes
os level = 32
```
- » # **testparm**
  - » redhat# **service smb start**
  - suse: **rcnmb start; rcsmb start**
- \_\_\_ 6. Leave the system that became LMB running. On a non-LMB system, stop the Samba daemons and configure this system as a **preferred master** with an **os level** of 33. Start this daemon. Does this system become LMB?
- » redhat# **service smb stop**
  - suse: **rcsmb stop; rcnmb stop**
  - » # **vi /etc/samba/smb.conf**
- Add the following lines:
- ```
local master = yes
os level = 33
preferred master = yes
```
- » # **testparm**
 - » redhat# **service smb start**
 - suse# **rcnmb start; rcsmb start**
- ___ 7. While you were doing these exercises, did you also see other master browsers appear in the output of the **smbclient** command? How long did this take?

WINS Service

- ___ 8. Your instructor will appoint one of the systems (or his/her own server) in the classroom as the WINS server. Configure all your systems to use this WINS server,

and then run the same simulation again. Does this make a difference in the time it takes?

```
» redhat# service smb stop
suse: rcsmb stop; rcnmb stop
» # vi /etc/samba/smb.conf
```

If you are the WINS server, add the following line:

```
wins support = yes
```

If you are a WINS client, add the following line:

```
wins server = <IP address of WINS server>
```

```
» # testparm
» redhat# service smb start
suse:# rcnmb start; rcsmb start
```

Clean-up

___ 9. Change your smb.conf file so that you are a simple system with **os level** zero, and part of the lx26 workgroup. Do not attempt to become master browser.

```
» # vi /etc/samba/smb.conf
```

Change the file so that it looks like this:

```
[global]
netbios name = system1
workgroup = lx26
```

```
» # testparm
» redhat# service smb restart
suse# rcnmb restart; rcsmb restart
```

END OF LAB

Exercise 3. Authentication

What This Exercise is About

This unit will be used to familiarize the student with Windows authentication and its implementation in Samba.

What You Should Be Able to Do

At the end of the lab, you should be able to:

- Configure basic authentication
- Configure encrypted passwords
- Configure a guest account

Exercise Instructions

Basic Authentication

- ___ 1. Verify that on your system two user accounts, samba1 and samba2, are added, and that the passwords of these accounts are samba1 and samba2, respectively.
- ___ 2. Configure your smb.conf file so that Samba is in **security = user** mode. Use **smbclient** to access Samba as user samba1. Use the password "samba1". Does this work?
- ___ 3. Now try to log on with the password "SAMBA1" (all capitals). Does this work? Why?
- ___ 4. Change the UNIX password of samba1 to "Samba1" (with a capital S). Then try to log on again with passwords "samba1", "Samba1" and "SAMBA1". Does this work? Change the **password level** to 1 and try these passwords again. Do they work now?
- ___ 5. Try different UNIX passwords (with multiple capitals) for samba1 and various **password level** settings until you understand how this works.
- ___ 6. Set the password of samba1 back to samba1.
- ___ 7. Edit the file /etc/samba/smbusers and add an alias for samba1 so that the Windows username "Samba User 1" is used for it. Change the smb.conf file so that the /etc/samba/smbusers file is used. Then try to log on to Samba with the Windows username "Samba User 1".

Encrypted Passwords

- ___ 8. Configure Samba to use encrypted passwords. Use the **smbadduser** to add samba1 and samba2 to the Samba password files. View the resulting /etc/samba/smbpasswd file.
- ___ 9. Try to log on to Samba as samba1. Does this work? Now change the UNIX password of samba1 to something completely different. Try to log on again. Does this work?
- ___ 10. Set the password of samba1 back to samba1.

Guest Accounts

- ___ 11. Try to login to Samba with an invalid username. What happens?
- ___ 12. Add a UNIX account "pcguest" to your system. DO NOT give this account a password. Then change the smb.conf file so that this account is used when a user supplies an invalid username. Try to log in to Samba with an invalid username again. What happens now?

Logging onto Samba from a Windows 95/98/ME machine

Depending on local circumstances, one or two machines in the classroom are set up as Windows 95/98/ME machines, or you will have a preconfigured VMware image on your system with Windows 95/98/ME. The instructor can tell you what is used in your classroom.

- ___ 13. Boot the Windows 95/98/ME machine. When you need to log on, do so as “samba1” with password “samba1”. Start the Windows Explorer and browse your network neighborhood. (Note: It might take several minutes after booting before you can actually access the network neighborhood: Windows machines do browser elections too...) Browse your own Samba server. Do you get access?
- ___ 14. Log out (Start; Shut Down; Close all programs and log on as a different user) and log in with a username that does not exist on the Samba server. Browse the network neighborhood again and try to access your system. Does this work? Why?
- ___ 15. If there is still time, then disable the guest account on the Samba server and try to access your Samba server from the Windows machine again. Does this work now?

END OF LAB

Exercise Instructions With Hints

Basic Authentication

- ___ 1. Verify that on your system two user accounts, samba1 and samba2, are added, and that the passwords of these accounts are samba1 and samba2, respectively.
 - » # **id samba1**
 - » # **id samba2**
 - » Switch to a free virtual terminal with <Alt-F6> or <Ctrl-Alt-F6> and try to log in as samba1 with password samba1. Do the same for samba2.

- ___ 2. Configure your smb.conf file so that Samba is in **security = user** mode. Use **smbclient** to access Samba as user samba1. Use the password "samba1". Does this work?
 - » redhat# **service smb stop**
suse: **rcsmb stop; rcnmb stop**
 - » # **vi /etc/samba/smb.conf**

Add the following line:

security = user
 - » # **testparm**
 - » redhat# **service smb start**
suse# **rcnmb start; rcsmb start**
 - » # **smbclient -L localhost -U samba1**

Password: **samba1**

- ___ 3. Now try to log on with the password "SAMBA1" (all capitals). Does this work? Why?

- ___ 4. Change the UNIX password of samba1 to "Samba1" (with a capital S). Then try to log on again with passwords "samba1", "Samba1" and "SAMBA1". Does this work? Change the **password level** to 1 and try these passwords again. Do they work now?
 - » # **passwd samba1**

New password: **Samba1**

Retype new password: **Samba1**
 - » # **smbclient -L localhost -U samba1**

Log on with password samba1, Samba1 and SAMBA1.
 - » # **vi /etc/samba/smb.conf**

Add the following line:

password level = 1
 - » # **testparm**
 - » redhat# **service smb restart**
suse# **rcnmb restart; rcsmb restart**

» # **smbclient -L localhost -U samba1**

Log on with password samba1 and SAMBA1.

___ 5. Try different UNIX passwords (with multiple capitals) for samba1 and various **password level** settings until you understand how this works.

___ 6. Set the password of samba1 back to samba1.

» # **passwd samba1**

New password: **samba1**

Retype new password: **samba1**

___ 7. Edit the file /etc/samba/smbusers and add an alias for samba1 so that the Windows username "Samba User 1" is used for it. Change the smb.conf file so that the /etc/samba/smbusers file is used. Then try to logon to Samba with the Windows username "Samba User 1".

» # **vi /etc/samba/smbusers**

Add the following line:

```
samba1 = "Samba User 1"
```

» # **vi /etc/samba/smb.conf**

Add the following line:

```
username map = /etc/samba/smbusers
```

» # **testparm**

» redhat# **service smb restart**

suse# **rcnmb restart; rcsmb restart**

» # **smbclient -L localhost -U "Samba User 1"**

Encrypted Passwords

___ 8. Configure Samba to use encrypted passwords. Use the **smbadduser** to add samba1 and samba2 to the Samba password files. View the resulting /etc/samba/smbpasswd file.

» # **vi /etc/samba/smb.conf**

Add the following lines:

```
encrypt passwords = yes
```

```
smb passwd file = /etc/samba/smbpasswd
```

» # **testparm**

» redhat# **service smb restart**

suse: **rcnmb restart; rcsmb restart**

» # **smbadduser samba1:samba1**

New SMB password: **samba1**

Retype new SMB password: **samba1**

» # **smbadduser samba2:samba2**

New SMB password: **samba2**

Retype new SMB password: **samba2**

» # **less /etc/samba/smbpasswd**

___ 9. Try to log on to Samba as samba1. Does this work? Now change the UNIX password of samba1 to something completely different. Try to log on again. Does this work?

» # **smbclient -L localhost -U samba1**

» # **passwd samba1**

New password: **something completely different**

Retype new password: **something completely different**

» # **smbclient -L localhost -U samba1**

___ 10. Set the password of samba1 back to samba1.

» # **passwd samba1**

New password: **samba1**

Retype new password: **samba1**

Guest Accounts

___ 11. Try to log in to Samba with an invalid username. What happens?

» # **smbclient -L localhost -U "An invalid username"**

Password: **bla**

___ 12. Add a UNIX account "pcguest" to your system. DO NOT give this account a password. Then change the smb.conf file so that this account is used when a user supplies an invalid username. Try to log in to Samba with an invalid username again. What happens now?

» # **useradd pcguest**

» # **vi /etc/samba/smb.conf**

Add the following lines:

```
guest account = pcguest
```

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

» # **testparm**

» redhat# **service smb restart**

suse# **rcnmb restart; rcsmb restart**

» # **smbclient -L localhost -U "An invalid username"**

Password: **bla**

Logging onto Samba from a Windows 95/98/ME machine

Depending on local circumstances, one or two machines in the classroom are set up as Windows 95/98/ME machines, or you will have a preconfigured VMware image on your system with Windows 95/98/ME. The instructor can tell you what is used in your classroom.

- ___ 13. Boot the Windows 95/98/ME machine. When you need to log on, do so as “samba1” with password “samba1”. Start the Windows Explorer and browse your network neighborhood. (Note: It might take several minutes after booting before you can actually access the network neighborhood: Windows machines do browser elections too...) Browse your own Samba server. Do you get access?
- ___ 14. Log out (Start; Shut Down; Close all programs and log on as a different user) and log in with a username that does not exist on the Samba server. Browse the network neighborhood again and try to access your system. Does this work? Why?
- ___ 15. If there is still time, then disable the guest account on the Samba server and try to access your Samba server from the Windows machine again. Does this work now?

END OF LAB

Exercise 4. File Sharing

What This Exercise is About

In this exercise you will practice the techniques given in the lecture for configuring Samba to share directories on the Linux host. You will experiment with dynamic home directories, permission issues, and other characteristics of disk shares.

What You Should Be Able to Do

At the end of the lab, you should be able to:

- Properly set up the defaults for dynamic home directory shares
- Configure disk shares for read-only, public access, and so on

Exercise Instructions

The [homes] template share

As shown in the lecture, the [homes] share is actually a template which is used for sharing the home directory of the user to that particular user.

- ___ 1. Modify the smb.conf file so that the [homes] share is added.
- ___ 2. Use the **smbclient** command to retrieve a list of shares on your own server, while logging in anonymously, as samba1, as samba2 and with an incorrect username.
- ___ 3. Use the **smbclient** command to log in to the home directory of samba1. Upload the file /etc/passwd. Log out and view the permissions on the file /home/samba1/passwd. Are these permissions what you expected?
- ___ 4. Use the **smbmount** command to access the home directory of samba2. Upload the file /etc/passwd. Disconnect and view the permissions on the file /home/samba2/passwd.

Note: This only works under Linux (not any other UNIX) and only if your Linux kernel supports the smbfs filesystem.

- ___ 5. On a Windows 95/98/ME system, log in as samba1 with password samba1. Browse your network neighborhood and go to your own server. What shares do you see? Access the samba1 share and see what files are in there. Can you read and write files?
- ___ 6. On a Windows 95/98/ME system, start an MS-DOS prompt. Run the command **net use h: \\servername\homes**. What does this command do? Does it work? Why would you use "homes" instead of your own username?

Other file shares

- ___ 7. Edit the smb.conf file and add a [tmp] share as shown in the lecture. Access this share, either using **smbclient** or **smbmount**, or from a Windows 95/98/ME system. Can you add and delete your own files? Can you add and delete files from others?
- ___ 8. Create a new UNIX group "samba". Add only samba1 to this group as its secondary group. Create a group directory /groups/samba for this group with the correct group ownership and permissions. Then add a share to the smb.conf file for this directory. Browse your server as samba1 and samba2. Which users can see and access the share? Now add samba2 to the group as well. Is there anything else you need to do to allow samba2 access to the group?
- ___ 9. Configure a share which exports your local CD-ROM drive. Samba should automatically close the CD tray and mount the CD when somebody connects to it, and unmount and eject the CD when the share is disconnected. Then try to see if this works from a Windows 95/98/ME system.

Note

For unknown reasons, the root postexec command does not seem to work correctly. This is under investigation.

END OF LAB

Exercise Instructions With Hints

The [homes] template share

As shown in the lecture, the [homes] share is actually a template which is used for sharing the home directory of the user to that particular user.

__ 1. Modify the smb.conf file so that the [homes] share is added.

» # **vi /etc/samba/smb.conf**

Add the following:

```
[homes]
path = /home/%u
comment = Home Directory of %U
browsable = no
writable = yes
```

» # **testparm**

» redhat# **service smb restart**

suse# **rcnmb restart; rc smb restart**

__ 2. Use the **smbclient** command to retrieve a list of shares on your own server, while logging in anonymously, as samba1, as samba2 and with an incorrect username.

» # **smbclient -L localhost -N**

» # **smbclient -L localhost -U samba1**

» # **smbclient -L localhost -U samba2**

» # **smbclient -L localhost -U "invalid username"**

__ 3. Use the **smbclient** command to log in to the home directory of samba1. Upload the file /etc/passwd. Log out and view the permissions on the file /home/samba1/passwd. Are these permissions what you expected?

» # **smbclient //localhost/samba1 -U samba1%samba1**

smb: \> **put /etc/passwd passwd**

smb: \> **quit**

» # **ls -l /home/samba1/passwd**

__ 4. Use the **smbmount** command to access the home directory of samba2. Upload the file /etc/passwd. Disconnect and view the permissions on the file /home/samba2/passwd.

Note: This only works under Linux (not any other UNIX) and only if your Linux kernel supports the smbfs filesystem.

» # **mkdir /mnt/smb**

» # **smbmount //localhost/samba2 /mnt/smb -o username=samba2**

Password: **samba2**

» # **cp /etc/passwd /mnt/smb**

- » # **umount /mnt/smb**
- » # **ls -l /home/samba2/passwd**

- ___ 5. On a Windows 95/98/ME system, log in as samba1 with password samba1. Browse your network neighborhood and go to your own server. What shares do you see? Access the samba1 share and see what files are in there. Can you read and write files?
- ___ 6. On a Windows 95/98/ME system, start an MS-DOS prompt. Run the command **net use h: \\servername\homes**. What does this command do? Does it work? Why would you use “homes” instead of your own username?

Other file shares

- ___ 7. Edit the smb.conf file and add a [tmp] share as shown in the lecture. Access this share, either using **smbclient** or **smbmount**, or from a Windows 95/98/ME system. Can you add and delete your own files? Can you add and delete files from others?

- » # **vi /etc/samba/smb.conf**

Add the following share:

```
[tmp]
comment = Free disk space here
writable = yes
browsable = yes
public = yes
path = /tmp
```

- » # **testparm**
- » redhat# **service smb restart**
- » suse# **rcnmb restart; rc smb restart**

- ___ 8. Create a new UNIX group “samba”. Add only samba1 to this group as its secondary group. Create a group directory /groups/samba for this group with the correct group ownership and permissions. Then add a share to the smb.conf file for this directory. Browse your server as samba1 and samba2. Which users can see and access the share? Now add samba2 to the group as well. Is there anything else you need to do to allow samba2 access to the group?

- » # **groupadd samba**
- » # **usermod -G samba samba1**
- » # **mkdir -p /groups/samba**
- » # **chmod 2770 /groups/samba**
- » # **chgrp samba /groups/samba**
- » # **vi /etc/samba/smb.conf**

Add the following share:

```
[samba]
comment = samba group share
```

```
writable = yes
browsable = yes
public = no
path = /groups/samba
valid users = @samba
» # testparm
» redhat# service smb restart
suse# rcnmb restart; rcsmb restart
» # smbclient -L localhost -N
» # smbclient //localhost/samba -N
» # smbclient -L localhost -U samba1%samba1
» # smbclient //localhost/samba -U samba1%samba1
» # smbclient -L localhost -U samba2%samba2
» # smbclient //localhost/samba -U samba2%samba2
» # usermod -G samba samba2
» # smbclient -L localhost -U samba2%samba2
» # smbclient //localhost/samba -U samba2%samba2
```

- ___ 9. Configure a share which exports your local CD-ROM drive. Samba should automatically close the CD tray and mount the CD when somebody connects to it, and unmount and eject the CD when the share is disconnected. Then try to see if this works from a Windows 95/98/ME system.

Note

For unknown reasons, the root postexec command does not seem to work correctly. This is under investigation.

```
» # vi /etc/samba/smb.conf

Add the following share:

[CD]
comment = CD-ROM
writable = no
browsable = yes
public = no
path = /mnt/cdrom
root preexec = eject -t ; mount /mnt/cdrom
root postexec = umount /mnt/cdrom ; eject
» # testparm
» redhat# service smb restart
suse# rcnmb restart; rcsmb restart
```

END OF LAB

Exercise 5. Printer Sharing

What This Exercise is About

In this exercise you will configure a Linux Samba server to support all of the printers currently supported by the host Linux operating system. In addition, you will learn how to restrict which printers are actually shared.

What You Should Be Able to Do

At the end of the lab, you should be able to:

- Configure a basic Samba setup which exposes all of the Linux-configured printers to the SMB network clients
- Restrict the above configuration to selected printers only

Exercise Instructions

Configuring a UNIX printer

Considering the wide variety of classrooms that this course will be taught in, it is not possible to include instructions here on how to configure your system so that a UNIX printer (a printer that can be accessed with the regular UNIX commands) is available. Your instructor will therefore tell you how to configure your UNIX printer.

Red Hat Linux uses the **redhat-config-printer** utility, while SuSE configures printers through **YaST2 Control Center**.

As an alternative, if no printers are available at all in your classroom, you can print to a virtual terminal (for instance `/dev/tty9`) or `xterm (/dev/pts/n;` figure this out with the **tty** command) window. If this is the case, make sure that the printing subsystem user (“lp”) has access to this device (**chown lp /dev/tty9**).

- ___ 1. Configure your UNIX printer using the instructions that will be provided by your instructor.
- ___ 2. Create a test file, `/root/testfile` and put something in here, such as a poem, so that you are able to identify this file as your own.
- ___ 3. Send the test file to the printer using the regular UNIX commands to verify that your printing subsystem works.

Configuring the [printers] template share

- ___ 4. Modify your `smb.conf` file so that all printers in your `printcap` file are automatically shared. Then restart Samba.
- ___ 5. Use the **smbclient** command to browse your own Samba server. Do you see the printer share? What is the name of the printer share, and why?
- ___ 6. Use the **smbclient** command to print the test file. Log in as `samba1`. Verify that the file was indeed printed.
- ___ 7. Switch over to a Windows 95/98/ME system and browse your network neighborhood again. Install the printer which was shared on your Samba server and try to print a file.

Configuring individual printers

- ___ 8. Edit your `smb.conf` file so that printers are not loaded automatically anymore. Then configure two printers, one named “`nolinenum`” and one named “`linenumbers`”. For both, add a **print command** statement so that they both print to the same printer, but the second one should add line numbers to the print job. Then use the **smbprint** command to print your test file to the printer.

Working with pseudo-printers (optional)

By modifying the print command for an individual printer, we can do a lot of strange things. A typical example is sending the print job to a fax instead of a printer. We don't have a fax in our classroom, so we will do something else, just to get the idea.

- ___ 9. Configure a printer share named "mailtoroot". If jobs are submitted to this printer, they need to be mailed to root.
- ___ 10. On the Windows system, install this additional printer as a Generic text-only printer and print a small text file (create one with WordPad or Notepad, if necessary) to this printer. Then, as root, check your mailbox with the **mail** command. Did it work?
- ___ 11. Make a list of files in /var/spool/samba. Do you see any files here? Why? How would you solve this?

END OF LAB

Exercise Instructions With Hints

Configuring a UNIX printer

Considering the wide variety of classrooms that this course will be taught in, it is not possible to include instructions here on how to configure your system so that a UNIX printer (a printer that can be accessed with the regular UNIX commands) is available. Your instructor will therefore tell you how to configure your UNIX printer.

Red Hat Linux uses the **redhat-config-printer** utility, while SuSE configures printers through **YaST2 Control Center**.

As an alternative, if no printers are available at all in your classroom, you can print to a virtual terminal (for instance `/dev/tty9`) or xterm (`/dev/pts/n`; figure this out with the **tty** command) window. If this is the case, make sure that the printing subsystem user (“lp”) has access to this device (**chown lp /dev/tty9**).

- ___ 1. Configure your UNIX printer using the instructions that will be provided by your instructor.
 - » All hints assume that the printer configured has the name “lp”. If this is not the case, change all printer queue names to the corresponding names and don't forget to add the **-P queue** option to all relevant commands (**lpr**, **lpq**, **lpc** and so forth).
- ___ 2. Create a test file, `/root/testfile` and put something in here, such as a poem, so that you are able to identify this file as your own.
 - » # **vi /root/testfile**
- ___ 3. Send the test file to the printer using the regular UNIX commands to verify that your printing subsystem works.
 - » # **lpr /root/testfile**

Configuring the [printers] template share

- ___ 4. Modify your `smb.conf` file so that all printers in your `printcap` file are automatically shared. Then restart Samba.
 - » # **vi /etc/samba/smb.conf**

In the `[global]` section, add the following lines:

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

At the end of the file, add the following share:

```
[printers]
path = /var/spool/samba
read only = true
```

```

    printable = true
    guest ok = true
  » # testparm
  » redhat# service smb restart
    suse# rcnmb restart; rcsmc restart

```

- ___ 5. Use the **smbclient** command to browse your own Samba server. Do you see the printer share? What is the name of the printer share, and why?
- ```

 » # smbclient -L localhost -N

```
- \_\_\_ 6. Use the **smbclient** command to print the test file. Log in as samba1. Verify that the file was indeed printed.
- ```

  » # smbclient //localhost/lp -U samba1%samba1
    smb: \> print /root/testfile
    smb: \> exit

```
- ___ 7. Switch over to a Windows 95/98/ME system and browse your network neighborhood again. Install the printer which was shared on your Samba server and try to print a file.

Configuring individual printers

- ___ 8. Edit your smb.conf file so that printers are not loaded automatically anymore. Then configure two printers, one named “nolinenumber” and one named “linenumbers”. For both, add a **print command** statement so that they both print to the same printer, but the second one should add line numbers to the print job. Then use the **smbprint** command to print your test file to the printer.

```

  » # vi /etc/samba/smb.conf

```

Change the load printers statement to:

```
load printers = no
```

Add the following two shares:

```

[nolinenumbers]
path = /var/spool/samba
writable = no
guest ok = yes
printable = yes
print command = lpr -r -P lp %s

```

```

[linenumbers]
path = /var/spool/samba
writable = no
guest ok = yes

```

```
printable = yes
print command = nl %s | lpr -r -P lp
» # testparm
» redhat# service smb restart
suse# rcnmb restart; rcsmb restart
» # smbclient //localhost/nolinenumbers -U samba1%samba1

smb: \> print /root/testfile

smb: \> exit

» # smbclient //localhost/linenumbers -U samba1%samba1

smb: \> print /root/testfile

smb: \> exit
```

Working with pseudo-printers (optional)

By modifying the print command for an individual printer, we can do a lot of strange things. A typical example is sending the print job to a fax instead of a printer. We don't have a fax in our classroom, so we will do something else, just to get the idea.

- ___ 9. Configure a printer share named "mailtoroot". If jobs are submitted to this printer, they need to be mailed to root.

```
» # vi /etc/samba/smb.conf

Add the following share:

[mailtoroot]
path = /var/spool/samba
writable = no
guest ok = yes
printable = yes
print command = cat %s | mail -s "Print file from %U" root
» # testparm
» redhat# service smb restart
suse# rcnmb restart; rcsmb restart
```

- ___ 10. On the Windows system, install this additional printer as a Generic text-only printer and print a small text file (create one with WordPad or Notepad, if necessary) to this printer. Then, as root, check your mailbox with the **mail** command. Did it work?

- ___ 11. Make a list of files in /var/spool/samba. Do you see any files here? Why? How would you solve this?

```
» # ls -l /var/spool/samba
```

END OF LAB

Exercise 6. Windows Domain Support

What This Exercise is About

This workshop will provide you with experience in configuring the Samba software to act as a domain controller in a moderately complex network.

What You Should Be Able to Do

At the end of the lab, you should be able to:

- Configure Samba to act as a Primary Domain Controller
- Configure logon scripts
- Configure roaming profiles

Exercise Instructions

Configuring Samba as PDC

In this section you are going to configure your Samba server as a Primary Domain Controller. Since there can only be one PDC in a domain, you are going to change your workgroup from LX26 to something unique.

- ___ 1. Configure your smb.conf file so that you are in a unique workgroup, and that your system is able to accept domain logons.
- ___ 2. Switch over to a Windows 95/98/ME system and configure this for domain logons. Delete all *.PWL files from C:\Windows. Reboot the machine and log on to your domain as samba1 with an invalid password. Does this work? Then log on to the domain as samba1 with the valid password.

Logon scripts

- ___ 3. Modify your smb.conf file so that a client that logs on to the domain uses the logon script /home/netlogon/logon.bat. Create the logon script so that it contains the MS-DOS command **net use h: \\server\homes**.
- ___ 4. On your Windows 95/98/ME system, log off and log on again. After you have logged on, start the Windows Explorer. Do you see the H:-disk? Can you modify and add files?

Roaming profiles

- ___ 5. Modify your smb.conf file so that roaming profiles are supported, both for Windows 95/98/ME clients and Windows NT/2000 clients.
- ___ 6. Go to your Windows 95/98/ME system and configure it for roaming profiles. Reboot the system, then log on as samba1. Answer “yes” to the question about whether you will want to use roaming profiles. Change something in your environment (such as the background pattern) and log out.

Joining a Samba server in a domain

For this part of the exercise you will have to work together with a partner team.

- ___ 7. Discuss with your partner team who is going to be the PDC for the domain, and whose server is going to join the domain. Exchange information regarding netbios names, domain names, root passwords and so forth.
- ___ 8. Configure the PDC so that it allows systems to automatically join the domain.
- ___ 9. Create a machine trust account on the PDC for the other server.

- ___ 10. Configure the other server so that it supports domain logons, and join it to the domain.
- ___ 11. From any Samba server, use the **smbclient** command to view the information on the samba server (not the PDC). Log in as samba1. What password do you need? On the PDC, change the password of the user samba1 to something else. Then try to log on to the other server again. Are passwords really checked on the PDC?

END OF LAB

Exercise Instructions With Hints

Configuring Samba as PDC

In this section you are going to configure your Samba server as a Primary Domain Controller. Since there can only be one PDC in a domain, you are going to change your workgroup from LX26 to something unique.

- ___ 1. Configure your smb.conf file so that you are in a unique workgroup, and that your system is able to accept domain logons.

» # **vi /etc/samba/smb.conf**

Change the file so that it looks like this:

```
[global]
netbios name = system1
workgroup = workgroup1
security = user
os level = 64
domain master = yes
local master = yes
preferred master = yes
domain logons = yes
```

```
[netlogon]
writable = no
public = no
path = /home/netlogon
```

Any print options and shares that you configured earlier may be left in place.

» # **mkdir /home/netlogon**

» # **testparm**

» redhat# **service smb restart**

suse# **rcnmb restart; rcsmb restart**

- ___ 2. Switch over to a Windows 95/98/ME system and configure this for domain logons. Delete all *.PWL files from C:\Windows. Reboot the machine and log on to your domain as samba1 with an invalid password. Does this work? Then log on to the domain as samba1 with the valid password.

- » Go to your Network Properties. Select "Client for Microsoft Networks" and click on Properties. Enable "Logon to a Windows NT domain".
- » Delete all *.PWL files from C:\Windows using the Windows Explorer.
- » Reboot the system.
- » Log on as samba1 with an invalid password.
- » Log on as samba1 with a valid password.

Logon scripts

- ___ 3. Modify your smb.conf file so that a client that logs on to the domain uses the logon script /home/netlogon/logon.bat. Create the logon script so that it contains the MS-DOS command **net use h: \\server\homes**.

» # vi /etc/samba/smb.conf

Add the following line to the [global] section:

```
logon script = logon.bat
```

» # testparm

» redhat# **service smb restart**

```
suse# rcnmb restart; rcsmb restart
```

» # vi /home/netlogon/logon.bat

The file should contain:

```
@echo off
```

```
net use h: \\servername\homes
```

Make sure the file is in MS-DOS format. This can be forced in vi with the command **:set tx** after which you can save the file.

- ___ 4. On your Windows 95/98/ME system, log off and log on again. After you have logged on, start the Windows Explorer. Do you see the H:-disk? Can you modify and add files?

Roaming profiles

- ___ 5. Modify your smb.conf file so that roaming profiles are supported, both for Windows 95/98/ME clients and Windows NT/2000 clients.

» # **mkdir /home/profile**

» # **mkdir /home/profile/samba1**

» # **chown samba1:samba1 /home/profile/samba1**

» # **mkdir /home/profile/samba2**

» # **chown samba2:samba2 /home/profile/samba2**

» # vi /etc/samba/smb.conf

Remove the `logon script` line, since our current logon script will interfere with the **logon drive** parameter.

Add the following lines in the [global] section:

```
logon path = \\server1\profile\%U
```

```
logon drive = H:
```

```
logon home = \\server1\%U
```

```
preserve case = yes
```

```
short preserve case = yes
```

```
case sensitive = no
```

Add the following share:

```
[profile]
path = /home/profile
create mode = 0600
directory mode = 0770
browsable = yes
writable = yes
```

» # **testparm**

» redhat# **service smb restart**

suse# **rcnmb restart; rcsmb restart**

___ 6. Go to your Windows 95/98/ME system and configure it for roaming profiles. Reboot the system, then log on as samba1. Answer “yes” to the question about whether you will want to use roaming profiles. Change something in your environment (such as the background pattern) and log out.

» Click on Start; Settings; Control Panel. Click on Passwords. Go to the tab User Profiles and select the level of roaming profiles you want. Click OK, but do not reboot the computer just yet.

» In the Control Panel, click on Network. Verify that the Primary Logon is set to “Client for Microsoft-Networks”.

» Now reboot your system.

» Log in as samba1 with password samba1. When a message appears asking you whether you want to enable roaming profiles, click Yes.

» Change something in your environment, such as the background pattern and log out.

» On the PDC:

```
# ls -l /home/samba1
```

» Log in again as samba1 on your Windows machine. Did everything work?

» Log in as samba1 to another Windows machine in the classroom which has also been set up for roaming profiles (don't forget to specify your own domain name). Does this work too?

Joining a Samba server in a domain

For this part of the exercise you will have to work together with a partner team.

___ 7. Discuss with your partner team who is going to be the PDC for the domain, and whose server is going to join the domain. Exchange information regarding netbios names, domain names, root passwords and so forth.

___ 8. Configure the PDC so that it allows systems to automatically join the domain.

» # **vi /etc/samba/smb.conf**

Add the following line in the [global] section (on one line):

```
add user script = /usr/sbin/useradd -g 100 -d /dev/null -c "%U"
-s /bin/false -M %u
```

- » # **testparm**
- » redhat# **service smb restart**
- suse# **rcnmb restart; rcsmb restart**
- » # **smbadduser root:root**

New SMB password: **ibmlnx**

Retype new SMB password: **ibmlnx**

__ 9. Create a machine trust account on the PDC for the other server.

- » # **useradd -g 100 -d /dev/null -c "System 2" -s /bin/false -M system2\$**
- » # **passwd -l system2\$**
- » # **smbpasswd -a -m system2**

__ 10. Configure the other server so that it supports domain logons, and join it to the domain.

- » redhat# **service smb stop**
- suse# **rcsmb stop; rcnmb stop**
- » # **mv /etc/samba/smb.conf /etc/samba/smb.conf.pdc**
- » # **vi /etc/samba/smb.conf**

Create this file again so that it looks like this:

```
[global]
netbios name = system2
workgroup = workgroup1
security = domain
password server = *
encrypt passwords = yes
```

- » # **testparm**
- » # **smbpasswd -j workgroup1 -r system1 -U root**
- » redhat# **service smb start**
- suse# **rcnmb start; rcsmb start**

__ 11. From any Samba server, use the **smbclient** command to view the information on the samba server (not the PDC). Log in as samba1. What password do you need? On the PDC, change the password of the user samba1 to something else. Then try to log on to the other server again. Are passwords really checked on the PDC?

- » # **smbclient -L server2 -U samba1%samba1**

(On the PDC)

- » # **smbpasswd samba1**

New SMB password: **somethingelse**

Retype new SMB password: **somethingelse**

(On any system)

- » # **smbclient -L server2 -U samba1%samba1**
- » # **smbclient -L server2 -U samba1%somethingelse**

END OF LAB

Exercise 7. Configuring Samba with SWAT

What This Exercise is About

This exercise will use the graphical **SWAT** environment to configure the Samba server.

What You Should Be Able to Do

At the end of the lab, you should be able to:

- Install **SWAT** and configure **xinetd.conf**
- Use **SWAT** to configure file shares
- Use **SWAT** to configure printer shares

Exercise Instructions

Installing and configuring SWAT

Red Hat Linux bundles SWAT in a separate RPM than the Samba server and client software, while SuSE includes SWAT in the main Samba server RPM.

- ___ 1. **Red Hat Linux Only:** Verify that the **SWAT** RPM was installed. If it was not installed, install it now.
- ___ 2. Enable **SWAT**.

Running SWAT

Your goal in this lab is to recreate the `smb.conf` file that you've generated so far by using **SWAT**. You'll start with an empty configuration file and step through the **SWAT** screens, filling in the parameters to recreate disk shares and printer shares.

In the answers for these lab steps, we've listed the parameters you're likely to have changed, but they may not be complete or you may have used a synonym at some point (for example, `writable = yes` and `read only = no` are synonyms). So the answer may not contain all of the settings you need to create a configuration file exactly like the one you have built over the last couple of days.

- ___ 3. Rename the **smb.conf** file to **smb.conf.orig** so that your original settings are saved. Then create a new `/etc/smb.conf` for **SWAT** to use.
- ___ 4. Start a web browser (Netscape, Konqueror or Lynx) and access **http://localhost:901**. Log in as root with the root password. Browse around the different screens until you feel comfortable with the site setup.

Note: If you configured your browser earlier to use a proxy server, then you might need to disable this proxy server now.
- ___ 5. Visit the section for global parameters and fill in the fields with the values you've been using so far. It would be best to do this from memory, or by reviewing previous labs, but if you get stuck, you always have your `/etc/samba/smb.conf.orig` file to rely on. You can always visit the **View** link to see what the configuration file looks like so far.
- ___ 6. Now go to the section for disk shares and create the shares that you've been using in the class.
- ___ 7. At this point, you should be able to restart the server and use a client to verify the work you've done so far. Try it and see how it works!
- ___ 8. Now add all printer shares as well. Depending on the contents of your **printcap** file, the last step could have been quite easy, because Samba will just read that file and create the printer shares on the fly. If you wish, you may go back to the global section and turn off **load printers**, as you did in a previous lab, and create those shares by yourself.

Comparing the files

- ___ 9. Try comparing the **/etc/smb.orig** and **/etc/smb.conf** files to see which parameters are different. What are you going to do about the comments?

- ___ 10. Is there a program which prints out all of the configuration parameters, but ignores any comments? If so, you could run that program against the **smb.orig** file and save the results. Then run it again on the **smb.conf** file and save those results, too. When you compare those two sets of results, it will be accurate because the comments won't be there. Do that now.

END OF LAB

Exercise Instructions With Hints

Installing and configuring SWAT

Red Hat Linux bundles SWAT in a separate RPM than the Samba server and client software, while SuSE includes SWAT in the main Samba server RPM.

- ___ 1. **Red Hat Linux Only:** Verify that the **SWAT** RPM was installed. If it was not installed, install it now.
 - » redhat# **rpm -qi samba-swat**
 - If you don't see the name and version number of the RPM, then install it with the command **rpm -ivh samba-swat-version.rpm**.
- ___ 2. Enable **SWAT**.
 - » redhat# **chkconfig swat on**
 - » redhat# **service xinetd restart**

 - » suse# **chkconfig samba on**
 - » suse# **rcxinetd restart**

Running SWAT

Your goal in this lab is to recreate the `smb.conf` file that you've generated so far by using **SWAT**. You'll start with an empty configuration file and step through the **SWAT** screens, filling in the parameters to recreate disk shares and printer shares.

In the answers for these lab steps, we've listed the parameters you're likely to have changed, but they may not be complete or you may have used a synonym at some point (for example, `writable = yes` and `read only = no` are synonyms). So the answer may not contain all of the settings you need to create a configuration file exactly like the one you have built over the last couple of days.

- ___ 3. Rename the **smb.conf** file to **smb.conf.orig** so that your original settings are saved. Then create a new `/etc/smb.conf` for **SWAT** to use.
 - » # **mv /etc/samba/smb.conf /etc/samba/smb.conf.orig**
 - » # **touch /etc/samba/smb.conf**
- ___ 4. Start a web browser (Netscape, Konqueror or Lynx) and access **http://localhost:901**. Log in as root with the root password. Browse around the different screens until you feel comfortable with the site setup.

Note: If you configured your browser earlier to use a proxy server, then you might need to disable this proxy server now.
- ___ 5. Visit the section for global parameters and fill in the fields with the values you've been using so far. It would be best to do this from memory, or by reviewing previous labs, but if you get stuck, you always have your **/etc/samba/smb.conf.orig** file to

rely on. You can always visit the **View** link to see what the configuration file looks like so far.

- ___ 6. Now go to the section for disk shares and create the shares that you've been using in the class.
- ___ 7. At this point, you should be able to restart the server and use a client to verify the work you've done so far. Try it and see how it works!
- ___ 8. Now add all printer shares as well. Depending on the contents of your **printcap** file, the last step could have been quite easy, because Samba will just read that file and create the printer shares on the fly. If you wish, you may go back to the global section and turn off **load printers**, as you did in a previous lab, and create those shares by yourself.

Comparing the files

- ___ 9. Try comparing the **/etc/smb.orig** and **/etc/smb.conf** files to see which parameters are different. What are you going to do about the comments?
 - » # **diff /etc/samba/smb.conf /etc/samba/smb.conf.orig**
- ___ 10. Is there a program which prints out all of the configuration parameters, but ignores any comments? If so, you could run that program against the **smb.orig** file and save the results. Then run it again on the **smb.conf** file and save those results, too. When you compare those two sets of results, it will be accurate because the comments won't be there. Do that now.
 - » **testparm /etc/samba/smb.conf.orig > /tmp/smb.conf.orig**
 - » **testparm /etc/samba/smb.conf > /tmp/smb.conf**
 - » **diff /tmp/smb.conf /tmp/smb.conf.orig**

The output of the previous command is the configuration parameters that still need to be changed.

END OF LAB

