SENECA COLLEGE OF APPLIED ARTS AND TECHNOLOGY

FACULTY OF TECHNOLOGY

SCHOOL OF ELECTRONICS AND COMPUTER ENGINEERING TECHNOLOGY

SUBJECT: **COMPUTER PERIPHERAL SYSTEMS PER 452 A B C**

Student Name: ,
(Last name) (First name)

Student Number: **Workstation Number:**

LAB 6: Accessing Linux Shared resources from Windows Computers

I. Sharing a folder and CD-ROM on a Linux computer

1. In this lab, again set up two computers per group. Refer to the information given in Part III, share a folder and the CD-ROM drive on a Linux computer so that a Windows computer connected to the LAN can access the shared resource on the Linux computer.

2. In Part II, **each student** of the group will write a lab procedure in detail that shows:

   a) All the steps to create a shared file and a CD-ROM folder on a Linux computer.

   b) The steps for any Windows computer to access the shared file created in a).

3. When your group have configured, set up, and tested the sharing. Go to the Windows computer to browse the shared resources on the Linux computer and demonstrated the following to the lab supervisor:

   - Read(open) the text file that you created on a Linux computer from a Windows computer.
   - Access the files of the CD on a Linux computer from a Windows computer.
   - Install application software or a device driver on a Windows computer from the shared folder or the CD-ROM drive of a Linux computer. (For example, you can download an application/game program for Windows into the shared folder on the Linux computer. Then install the program by accessing it over the network on the Windows computer.)

4. Write the name of the software program or device driver that you have successfully installed for this lab below.

____

VERIFICATION PART I: ______________
Lab Instructor initials

II. Handwrite the procedure (legibility counts). Each student should submit your own hand-written procedure with the completed lab.

The procedure should show all the steps to create a shared file and a CD-ROM folder on a Linux computer and to allow a Windows computer in the username as **Win2sam** to access the shared resource on the Linux computer with **Read-Only** permission.

If you prefer, you could use a word processor application program to type the procedure and print it out on paper. However, you must hand in the printed procedure together with your hand-written draft.

III. Guided Experiment as Reference information

NOTE: It is recommended that you follow the information in this part to try out the setting of Samba so that you can carry out the tasks and ready to complete the hand-written procedure later.

A. On a Linux computer

1. Create a regular user account called **student** if your Linux doesn't have one.

2. From the top menu bar, select **System** – **Administration** – (**Server Settings** –) **Samba**. This will open the Samba Server Configuration window. (You can also start the Samba Server Configuration window by typing **system-config-samba** at the command line prompt.) Find out the following settings:

   Workgroup: ____________________________

   Server String: __________________________

   Now, close the Samba Server Configuration window.

3. Start a terminal window and change the current working directory to /etc/samba. We are going to manually modify the samba configuration file.

4. Before we start making any changes, let's back it up the file so that we can restore it in the future if required. To do this, at the command prompt, type

```
cp   smb.conf    smb.conf.org
```

5. Use an editor to modify the workgroup and server string setting under the [global] section in smb.conf as follows:

```
[global]

        workgroup = PER452
        server string = Sambaxxy
```

   where **xx** is your work station number, and **y** is your lab section).

6. Save smb.conf and exit the editor. **There is no need to restarting the samba services after having modified the smb.conf**. This is because that Samba rereads its configuration every time when a client connects to it.

7. Now re-open the Samba Server Configuration window to verify the workgroup and server string setting. Have they been changed?

   Y/N

8. Click the **Security** tab under **Preferences** – **Server Settings**. Check the default settings but leave them unchanged (Authentication Mode: User; Encrypt Passwords: Yes; Guest Account: No guest account)

9. In the earlier version of smb.conf file, one example section for a shared folder with **valid user(s)** who can place files in the shared folder over the LAN network is as similar to the one shown below:

[folder name]	**[gearth]**
(comment)	**Google Earth Application**
(path)	**/usr/local/bin**
	browseable = yes
	writeable = yes
	valid users = dummy

10. We could just follow this example to create a shared folder by manually modifying the smb.conf file as we did in Steps 4 and 5 above. **Instead**, we are going to do it in the GUI of Samba Server Configuration window in next step. You can also do it in the GUI of SWAT, if you have installed it in your RedHat Linux.

11. Go back to the Samba Server Configuration window. Click the **Add** button to add a Samba share. In the **Basic** tab's Directory field, enter **/usr/local** (this is the name of a local directory that we or you want to share). In the Share name field, enter **local** (this is the name that we/you want to the Windows PCs on the LAN to see). And in the Description field, enter **local share**. Click on the **Writable** and **Visible** option boxes to allow this share to be accessed as read-write and be visible when browsing on the network. (Note: In some GUI version of Samba Server Configuration, you may need to uncheck the **Read Only** button to make it Writable!) Now click the **Access** tab and select **Allow access to everyone** just for now. Click **OK** to add the share.

12. To verify that this has been done in the smb.conf, at the command prompt, type

```
less  /etc/samba/smb.conf
```

Look for the last few lines in the file, which should contain the configuration statements for this share. What are these statements? List them here:

____

____

____

____

____

13. Although we did not specifically issue a command to start the **smbd** and **nmbd** services that Samba uses, creating the share in the step above should have caused the **smb** services to start automatically. To verify this, at the command prompt, you can type **ps auxw | grep smbd** and **ps auxw | grep nmbd**. You can also go to the **Server Setting** then **Services** or **Service Configuration** to check the status of the **smb** and **nmb** services (NOT just the smb!!).

14. Back to the Samba Server Configuration utility window. From the **Preferences** menu select **Samba Users**. At the Samba Users window click **Add User**. At the Create New Samba User window, click on the drop-down menu for Unix Username, select and add a few users (student, user1, user2, etc.) other than **root** to be a Samba user. From the Samba Users list, select the user called **student** that you created ealier. (For earlier versions of Linux before Fedora 7, you can add **root** as a Samba user). Click **Edit User** on the selected Samba user (**student**). In the Windows Username field enter **student** as username that you would like to map to this Linux Samba user you selected, enter a Samba password and confirm it (**this does not have to be the same as the Linux user's password**). (Write this password on a piece of paper so that you will not forget it.) Similarly, select **user1**, edit it with a Windows Username **user1** and a different password for Windows access.

    Note: For earlier versions of Linux, you can specify a different Windows Username such as **Windows2sam** as username to map to the selected Samba user **student**. For Fedora 14, the Windows Username has to be the same as the selected Samba user name, regardless what you put in to the box for Windows Username. However, the password used by the Windows computer to access the share can be different from the one used on the Linux computer.)

15. Recall in Step 11, we selected **Allow access to everyone** for the **local** share. We will just leave it as it is to allow all Samba Users to have access to the share. (Later when we want to limit one or two specific users for the **local** share, we can come back to the Samba Server Configuration window to modify the setting of the **local** share.) After you have done, close the Samba Server Configuration utility window.

16. Re-open the **smb.conf** file. List the section that you have found inside sam.conf of sharing a folder with **valid user(s)** who can place files in the shared folder over the LAN network: (An example of correct format to have more than one user is shown in Step 9.)

    (folder name) ______________________________

    (comment) ______________________________

    (path) ______________________________

    ____

    ____

    ____

    ____

17. To understand **SELinux**, we want to setup **SELinux** to allow Samba to share any file/directory. Locate the setting called **SELinux Management** or **SELinux**

Administration or **Security Level and Firewall**. Enable it if you disabled it in the previous labs. Open **Boolean** section. Select and expand the Samba sub-section and check the box for "Allow Samba to share any file/directory". You should select **Permissive** rather than **Enforcing** for SELinux if you want to allow Sambe to share. (If you are working with **Fedora 6**, locate the **Security Level and Firewall** setting. Open **Modify SELinux Policy** under **SELinux**. Select and expand **Samba** section. Check the box of **Disable SELinux protection for smbd daemon**. We do not have to disable the protection for **nmbd** daemon here since that smbd is the one that is responsible for implementing the SMB file and print sharing.)

18. Let's just check that the **smb** services has indeed started correctly in case there is anything wrong in the **smb.conf** file causing **smb** failed to start. We can use the **service** command or do it differently as explained in Step 13 above.

```
        service   smb   status        and        service   nmb   status
Or
        /sbin/service   smb   status    and    /sbin/service   nmb   status
```

19. It's a good idea to test your Linux Samba share locally on your Linux computer by using the **smbclient** utility to access your Samba share similarly as you did in Lab 5 to access a Windows share:

```
        smbclient    //xxx.yyy.zzz.nnn/local   -U username
```

where xxx.yyy.zzz.nnn is the IP of your Linux computer and local is the Share name that you specified in Step 11. But you should know which user name that you should use and in what password. Option **–U** may be needed if the current login user name is different from the valid user.

20. When prompted for password, you know which password should be entered, right? If not, recall how did you connect to Windows computer to access and list the contents of a shared folder of that particular Windows computer using the **smbclient** utility. Can you list the contents of the Linux Samba shares folder right within your Linux computer?

Y/N

21. What if you can successfully access your Linux Samba share locally on your Linux computer or other Linux computer, **but fail to access the Linux Samba share from Windows computers**? It was found in one case that the samba configuration file **smb.conf** was corrupted by the Samba Server Configuration utility. To verify if this is the case, open the **smb.conf** in a text editor, check the workgroup setting, server string setting, and the setting of the Linux Samba share folder(s). If you have found a problem or problems from the above settings, you **have to** modify **smb.conf** in the text editor but **NOT** by running the Samba Server Configuration utility to fix the problem. (You could, however, try to re-install or upgrade Samba Server Configuration utility to see if the problem can be fixed without modifying in the text editor.) For some reasons, the Samba server graphic tool or smb deamon in FC7 seems automatically commenting out the workgroup setting in **smb.conf**. Therefore, if you can not access your Linux share under PER452 workgroup within Windows' Network Place, try to change the workgroup to the default Windows workgroup (**WORKGROUP**). Another "problem" was found with Samba server graphic tool is the visible/browseable setting. Once you select the property for the Linux share as writeable, the tool will automatically unckeck the Visible radio point so that the browseable setting for the share will be commented out or changed to no as browseable = no

while writeable = yes. This seems does not have any real problem for the share to be seen or access from Windows computers.

B. On a Windows computer

Step 1: Boot to Windows Professional.

Step 2: Browse **"MY NETWORK PLACES"**. Look for the **PER452** workgroup and your partner's Linux Samba server **Sambaxxy**

Step 3: Click on the Samba server's icon, log in with the Windows Username and the Samba password specified by your partner. You should be able to log in with the Windows Username for either Linux Samba user **student** or **user1** since that you and/or your partner selected "Allow access to everyone" when setting up the Linux Samba share.

Step 4: You should now be able to see/open the files in the shared folder of the Linux computer. If your partner has shared CD ROM folder of his/her Linux computer, you can run an application or access the files that are contained on the CD ROM disc on the Linux computer.

C. To limit only one of the Linux Samba users to have access to a Linux Samba share

1. On the Linux computer, open the Samba Server Configuration window. Click to select a Samba share, say /**usr**/**local** the one was created in Section A.

2. Click on the **Properties** icon to open the **Edit Samba Share** window for the share. Click the **Access** tab and select **Only allow access to specific users** and check the box for **user2** only. Click **OK** to close the Edit window.

3. Now, edit **user2** with a Windows Username and a Windows access password.

4. Go back to your partner's Windows*. Click on the Samba server's icon, log in with the Windows Username and the Samba password specified for Linux Samba user **user2**. You can access the share as you did in Section B. However, you can not access the share with the Windows Username for Linux Samba user **student** or **user1** since that now only Samba user **user2** has the permission to access the share.

   ***Note: You may need to reboot your Windows computer so that the Windows OS can refresh its good/bad "memory" of the last connection to a Linux Samba server. Sometime, you need to wait for a few minutes before the Windows OS can discover all the Samba shares.**

5. Again, try to test your Linux Samba share locally on your Linux computer by using the **smbclient** utility as you did before:

```
smbclient    // xxx.yyy.zzz.nnn/local   -U  username
```

6. What user name should be used here if required? And, when prompted for password, what should you enter?

7. As you did in Lab 5, start the **Places** pull-down menu. Select **Connect to Server…**. Fill in the **Service type** and the **Server name…**. You should be able to

connect to the Samba share on your own Linux, or a Samba share on any other Linux computer, or any Windows share on a Windows computer. Try to connect to the Samba share on your Linux computer from this GUI.

8. Can you list the contents of the Linux Samba shares folder right within your Linux computer now?

Y/N

Reminder: Bring and hand-in your hand-written procedure with your completed lab for the next PER452 lab.