



Hands On Labs+ Samba Version 4 with Windows And Linux Client and Server

Table of Contents

Introduction	2
Goals	2
Packages, Resources and Prerequisites	2
Document Conventions	3
General Process	4
Windows File Share Server to Ubuntu Samba Client.....	7
Windows 2008 R2 Server – Disable the Firewall	7
Windows 2008 R2 Server – Create a Fileshare	9
Ubuntu 13.10 Client – Samba Client Installation	10
Ubuntu 13.10 Client – Samba Client Configuration	10
Ubuntu 13.10 Client – Mount the Windows Share.....	11
Ubuntu File Share Server to Windows Client.....	13
Ubuntu 13.10 Server – Samba Server Installation	13
Ubuntu Server 13.10 – Samba Server Configuration	13
Windows 2008 R2 Client – Access the Share	14
APPENDIX A – Configuration Files.....	16
Sample /etc/samba/smb.conf File	16

Introduction

File sharing between Windows and Linux systems has been filled with difficult configuration, poor performance, poor security or some combination of all of these challenges. With Samba Version 4, you will find that great strides have been made in all of those categories.

We will show how to set up both Windows and Ubuntu Linux as a file share server as well as how to configure each as a client for the other's shares.

Goals

This Hands On Lab will show you how to set up Windows Server 2008 shares so that they are available to a Linux client. Additionally, we will demonstrate how to set up Ubuntu Linux as a file share server for Windows clients.

Additionally, we will talk about how to secure those shares in both directions by setting up account access and using credentials files to secure the password exchange between our systems. Finally, we will discuss all the services that have to be installed and running on both Windows and Linux for both operating systems to function as either file servers or clients.

PLEASE NOTE: The lab servers you will use are Ubuntu (Debian based) distributions and Windows 2008 R2 Standard Server. Although the process and packages for RPM distributions or other Windows operating systems will be very similar, there may be differences, see your distribution documentation for specifics.

Packages, Resources and Prerequisites

- cifs-utils
- winbind
- smbclient
- samba
- samba-common

The resources you will be accessing during the course of this lab are:

- Ubuntu 13.10 Server: Test Server and Client
 - You will use this to test your Windows Server to Ubuntu Client and Ubuntu Server to Windows Client Samba shares
- Windows 2008 R2 Standard Server: Test Server and Client
 - You will use this to test your Windows Server to Ubuntu Client and Ubuntu Server to Windows Client Samba shares

Prerequisites to this lab:

- A LinuxAcademy.com Lab+ Subscription
- Internet Access, SSH Client and Windows Remote Desktop Client (built in Windows Remote Desktop Client or other RDP equivalent)
 - You will need to connect to the public IP of the servers in order to configure Samba and File Sharing
 - SSH client can be Windows (i.e. Putty) or from another Linux system shell
- For Windows, the Remote Desktop Client
- Login Information (Provided When The Server Starts Up)

Document Conventions

Just a couple of housekeeping items to go over so you can get the most out of this Hands On Lab without worrying about how to interpret the contents of the document.

When we are demonstrating a command that you are going to type in your shell while connected to a server, you will see a box with a dark blue background and some text, like this:

```
linuxacademy@ip-10-0-0-0:~$ sudo apt-get install package  
[sudo] password for linuxacademy: <PASSWORD PROVIDED>
```

That breaks down as follows:

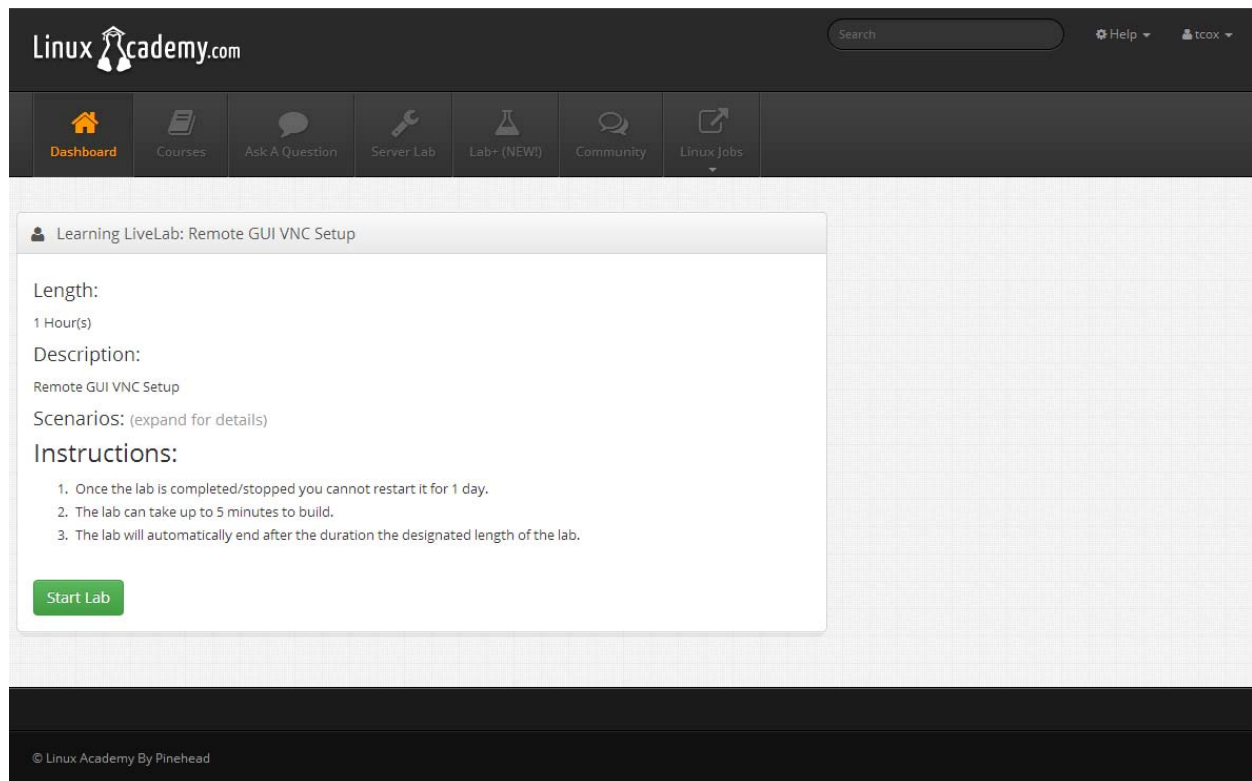
- The white text that looks something like “linuxacademy@ip-10-0-0-0:~\$: “should be interpreted as the console prompt your cursor will be sitting at when you are logged into the server. You are not typing that into the console, it is just there. Note that the portion that says “ip-10-0-0-0” will be different for you based on the IP address of your system.
- The bold yellow text in any command example is the actual command you will type in your shell.
- Any lines subsequent to the bold yellow command that you type in is to be considered as the response you can expect from your command entry.

If you do not see the command prompt as the first line in the example, then all the white text is an example of a text, script or configuration file and is intended to be typed in its entirety (in the event you are instructed to create a previously non-existent file) or to compare against the contents of the file on your system.

One final convention, if you see a “~” at the end of a line in a command or text example, that will indicate that the line overflowed the end of line. Meaning you should just keep typing without hitting enter to start a new line until the natural end of the command.

General Process

When you are ready to begin the Hands On Lab, log into your Linux Academy Lab+ subscription and navigate to the “Live Labs” section on the Dashboard. Once you choose the “Samba Version 4 Window and Linux Client and Server” Lab from the list, you will see the screen below:



[Linux Academy Live Labs+ - Start Screen]

A few things to note before you start this process:

- When you launch the lab from this screen, it may take up to FIVE MINUTES for your servers to be deployed and be available for your use.
- Do not leave your desktop and come back, once the servers are launched, you will only have THREE HOURS to complete this lab from start to finish. After that point, the servers time out and are deleted permanently. Any and all work that you have done will then be lost.
- You can only use this lab ONCE PER DAY. If you try to use it more than that after completing the lab or the servers timing out, the screen will tell you when it will be available to you again.

- Other than those descriptions, you may retry any of the Labs+ labs as many times as you wish as long as you are a subscriber.

Once you have clicked on the ‘Start Lab’ button that you see above, a process will launch on our servers that will deploy the two servers we will use in our lab for testing. After a few minutes of processing (and you will see a status message that says “Creating Lab... Please Wait”), you should see a screen that looks like this one:

The screenshot shows the Linux Academy website interface. At the top is a navigation bar with the Linux Academy logo, a search bar, and links for Help and tcox. Below the navigation bar is a menu with icons for Dashboard, Courses, Ask A Question, Server Lab, Lab+ (NEW!), Community, and Linux Jobs. The main content area displays the details for a lab titled 'Learning LiveLab: Remote GUI VNC Setup'. The lab details include: Length: 1 Hour(s); Description: Remote GUI VNC Setup; Scenarios: (expand for details); Instructions: 1. Once the lab is completed/stopped you cannot restart it for 1 day. 2. The lab can take up to 5 minutes to build. 3. The lab will automatically end after the duration the designated length of the lab. Lab Connection Information: Server 1 [Public IP: 54.84.29.129] [Private IP: 10.0.0.133], Server 2 [Public IP:] [Private IP:]. Access credentials: Server 1: [user: linuxacademy] [password: 123456], Server 2: [user: linuxacademy] [password: 123456]. There are two buttons: 'Complete Lab' and 'Download Lab Guide'. On the right side, there is a 'Lab Expiration' section showing 'Time left:' with a progress bar indicating 0 hours, 52 minutes, and 51 seconds. The footer of the page shows '© Linux Academy By Pinehead'.

[Linux Academy Live Labs+ - Start Screen]

You will see all the information you need to access your servers from another system. Specifically, you need:

- The servers public IP address
- Access credentials

One thing to note is that, in addition to the IP that you see above, the server will have another IP assigned to it in the 10.0.0.x subnet. This is a private IP address and will not route outside of your

private server pool. Your Ubuntu server will have a static private address of 10.0.0.100. We will be using the external IP address to connect over SSH when configuring Samba Server and Client. The Windows 2008 Server will need you to connect over RDP (Remote Desktop) as previously discussed.

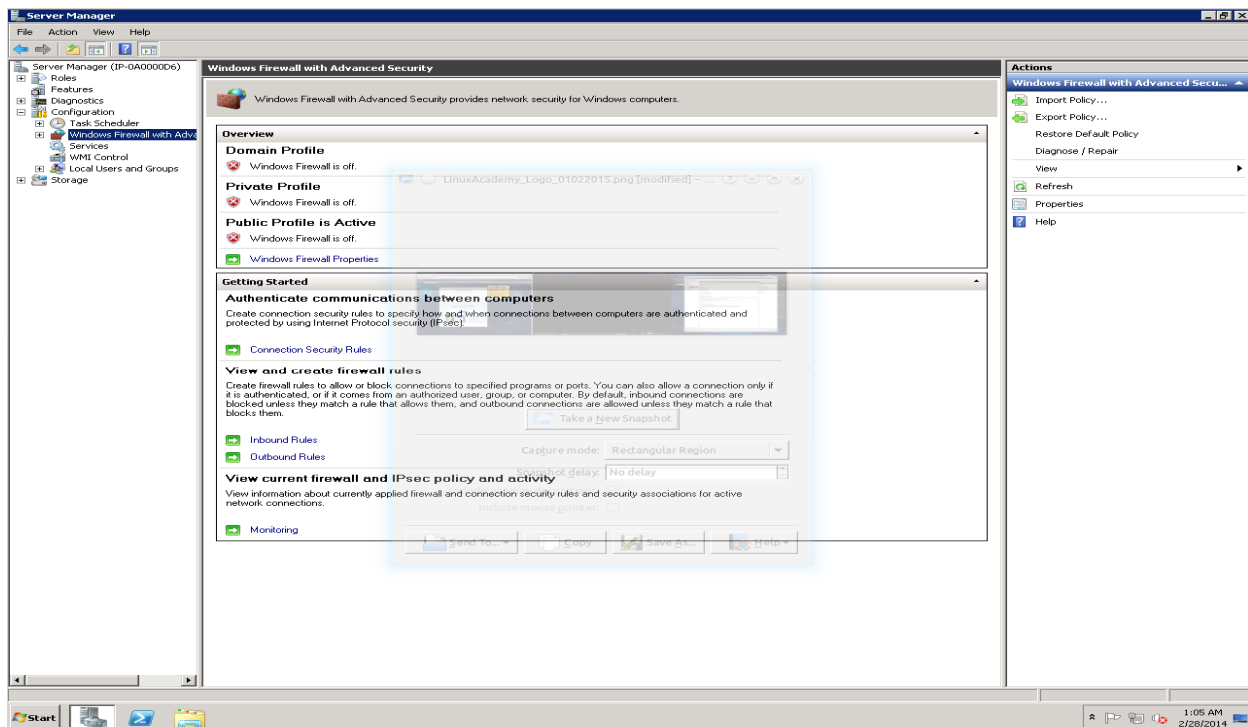
Windows File Share Server to Ubuntu Samba Client

We will start by configuring our Windows 2008 R2 Standard Server as a Server for file shares. This is done in the standard method for creating Windows shares, which we will illustrate in the following sections. Keep in mind, although we are not configuring AD Accounts (Active Directory) and using them, the method of accessing a share for an AD user will be noted in the appropriate section if you choose to go beyond the scope of this lab and experiment with AD account share access from Ubuntu.

Windows 2008 R2 Server – Disable the Firewall

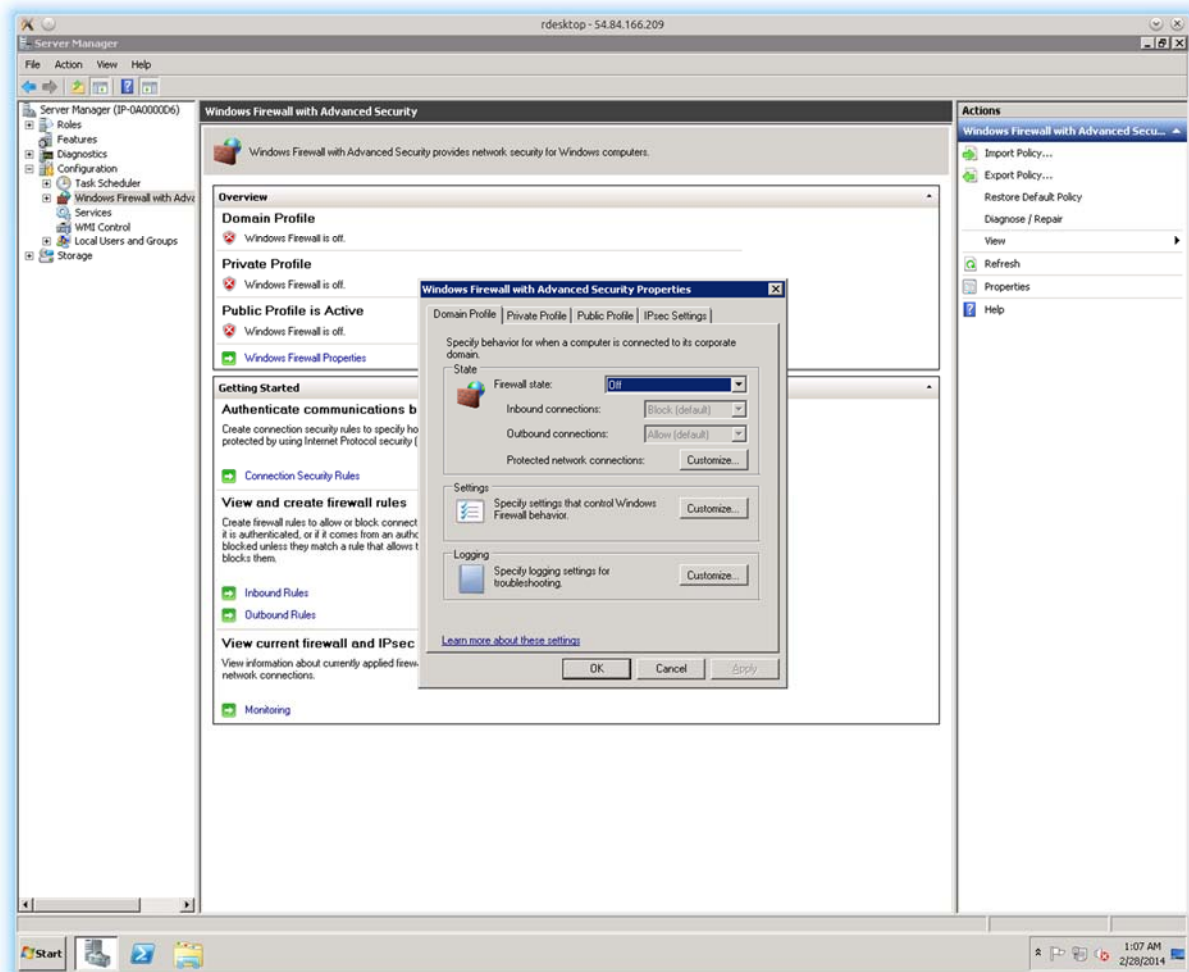
The first thing we need to do, is to adjust our Windows 2008 Server security policy. By default, the Windows Firewall will be active and attached to all Windows processes running in the Public, Private and Domain space. Although there are methods of allowing file sharing traffic through the appropriate interfaces in the firewall (namely traffic over ports 139 and 445), those activities are beyond the scope of this lab. Additionally, disabling the firewall completely applied equally to Windows Desktops as it does to Windows Servers, so for consistency in exploring sharing our Windows filesystems to Ubuntu, we will simply turn it off.

Once you sign into the Windows Desktop using the “linuxacademy” username and password, you will open the System Roles Management Console. In that console, you will see an entry called “Windows Firewall with Advanced Security” which will look like the screen shot below:



[Linux Academy Live Labs+ - Windows Firewall with Advanced Security]

You will notice that in this screen shot, the firewall for all three interface types mentioned above (Public, Private and Domain) are already disabled. At the very least, your firewall will have the “Public” and “Domain” firewall rules enabled. Click the “Windows Firewall Properties” link in the center window. You will see a dialog box that looks like the one in the following screen shot:

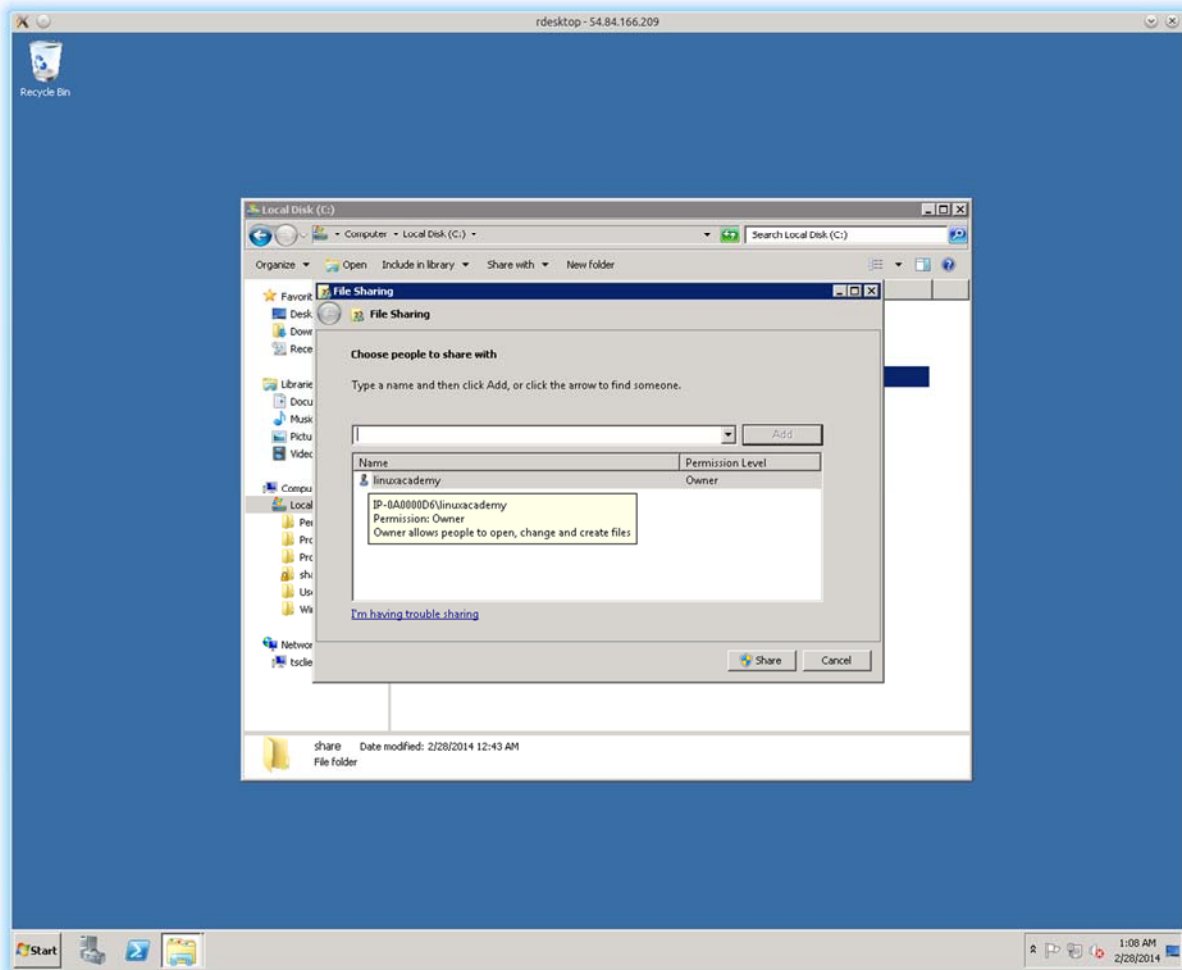


[Linux Academy Live Labs+ - Windows Firewall Properties]

Choose the tab on the properties window for each of the profiles and in the dropdown box called “Firewall Status”, choose the “Off” setting for each of them. Finally, click “Apply” and then “OK” and our firewall is now disabled.

Windows 2008 R2 Server – Create a Fileshare

Creating a Windows fileshare is something you have likely done before. Open Windows Explorer (or My Computer) and on the C:\ drive, create a folder called “share”. Once you have created the folder, right click on the folder with your mouse and choose “Share With...” and then from the resulting pop up, click on “Specific People”. You will see a screen that looks like the following:



[Linux Academy Live Labs+ - Share With...]

At this point, you will notice that we can just click “Share” since the “linuxacademy” account is already populated in the share with list. However, we could add any account, local or domain, that is available on the server in order to allow access to the folder we just created. We can also have as many folders shared (or entire drives or printers) as we want on this system. In this case, we are sharing the folder called “share” we just created and then “locking” access to just the “linuxacademy” account.

At this point, our Windows Server configuration is complete. This system will now serve this fileshare to anyone asking for it with the right user credentials and protocol. This is where our Ubuntu client comes in the picture.

Ubuntu 13.10 Client – Samba Client Installation

We have a couple of utilities that need to be installed in order for our Ubuntu client to be able to access the Windows fileshare over Samba. In this case, we are going to install the necessary Samba client utilities and supporting packages. To do this, simply log in to the Ubuntu client and run the following command:

Next, we need to edit a configuration file in order to configure the VNC desktop we are going to connect to. Edit the file “/etc/sysconfig/vncservers” and change it to look like the following example (substitute your user as appropriate):

```
sudo apt-get install cifs-utils winbind smbclient
```

At this point, Ubuntu will go out and install these packages and all the supporting applications and libraries we will need to access our Windows shares. Once all of these are complete (and there are a couple dozen so it will take a moment or two), we can move on to the configuration.

Ubuntu 13.10 Client – Samba Client Configuration

The first thing that needs to be changed is to make an addition to how files are “looked up” and accessed on our system. Since we added the “winbind” package (a package that allows us to add an Ubuntu system to an Active Directory and authenticate users with AD – something that we will not specifically do in our case but the process should be something you do when setting up Windows and Linux mixed environments), we will need to make a quick change to the “/etc/nsswitch.conf” file. Using ‘vi’, edit the file and change the following line:

```
hosts: files dns      ← What it will look like  
  
CHANGE TO:  
  
hosts: files wins dns ← What you need to change it to
```

The next thing we need to do is restart the network. Easy enough, simply execute the following command:

```
sudo service networking restart
```

NOTE: Make sure you execute a restart, this will recycle the networking service without disconnecting you. If you execute a 'stop', you will never get the chance to start it as it will stop networking and disconnect you. If that happens, you will have to end the lab and try again the next day or contact support and ask nicely for us to reset it for you explaining your error.

Believe it or not, that is all there is to the client set up. Before we start talking about mounting our fileshare, let's do a quick test to be sure we can see what is available. At the command line, execute the following command and you will see something like this:

```
smbclient -L //10.0.0.251 -U linuxacademy ← Your IP there
Enter linuxacademy's password:
Domain=[WORKGROUP] OS=[Windows 2008 Server] Server=[Windows 2008]

  Sharename      Type            Comment
  -----
  C$              Disk            Default Share
  share           Disk
NetBIOS over TCP disabled - no workgroup available
```

As long as you see the 'share' that we created on the Windows server, you are ready to mount it locally. If you get any error (NT_STATUS_LOGON_FAILURE, NT_STATUS_CONNECT_FAILURE, etc), then check the previous configuration session and be sure you have installed all the client packages, edited the nsswitch.conf file and restarted your network.

Ubuntu 13.10 Client – Mount the Windows Share

So now we can actually mount the share our user has access to. Let's create a local directory that we can mount it to like so:

```
sudo mkdir -p /mnt/test
```

Now we can manually mount the remote share to this local directory by executing this command:

```
sudo mount -t cifs //10.0.0.251/share /mnt/test -o ~
username=linuxacademy,password=123456,rw,nounix,file_mode=0777, ~
dir_mode=0777
```

You can now change to the "/mnt/test" directory and create a file, any file, in 'vi'. If you then switch back over to your Windows file server and check the "C:\share" directory, you will see your text file there.

The command line that we used to mount this share can be translated, in plain English, to read as "mount a filesystem of type CIFS from the IP 10.0.0.251 called share to my local directory called

/mnt/test using the username 'linuxacademy' and the password '123456', mount it read/write as a non-unix filesystem and give everyone full read/write/execute to all files and directories existing and created on this mount". That may seem like a mouthful and you can change multiple parts of it. The most likely change will be that you will want to change the 'file_mode' and 'dir_mode' parameters to restrict local users on the Ubuntu system from being able to make changes or even access this share since it is connected with a local account's credentials. That is for you to decide.

Finally, we can mount this share permanently so that when we reboot, it will automatically be available by adding a simply entry to our '/etc/fstab' file. However, we would need to pass user credentials to the mount line in the fstab file, which is bad security practice. Let's create a credentials file that we can use. In the '/root' directory, using 'vi', let's create a file called .smbcredentials which contains the following two lines:

```
username=linuxacademy
password=123456
```

Now we can add the following line to our '/etc/fstab' file. As the last line in the file, add the following:

```
# remote windows file share
//10.0.0.251/share /mnt/test cifs ~
credentials=/root/.smbcredentials,rw,nounix,file_mode=0777, ~
dir_mode=0777 0 0
```

We can test the mount and credentials by executing a 'sudo mount -a' and checking the directory for our previously created file. Again, you can change the mount directory and permissions as you desire for the use. This creates a mount that is available to everyone to access fully, but your mileage and needs will vary. That's all there is to our Windows server to Ubuntu client for Samba. Let's flip the tables now!

Ubuntu File Share Server to Windows Client

Now we are going to look at the reverse of our previous configuration. This time, we will set up Ubuntu as our Samba server and make the 'linuxacademy' user home directory available as a share that the appropriate account credentials can then mount on a Windows server or desktop.

Ubuntu 13.10 Server – Samba Server Installation

We need to install the appropriate Samba server applications for our shares to be available to Windows systems. Open a command prompt and execute the following installation command::

```
sudo apt-get install samba samba-common
```

This will take a few minutes and install a number of supporting libraries and other applications. Once installed, it will automatically create the appropriate configuration file defaults as well as start the Samba process (called 'smbd').

Ubuntu Server 13.10 – Samba Server Configuration

At this point, we are going to edit the Samba server configuration file that will make our local user's '/home' directory available to a Windows system providing the appropriate credentials. So, using 'vi', edit the file called '/etc/samba/smb.conf' and look for the commented line reading "# security = user". Place the following items right under that section:

```
# security = user
hosts allow =

[homes]
comment = Local User Home Directories
browseable = no
writable = yes

[share]
comment = Default share
path = /share/
force user = samba
force group = samba
read only = No
hosts allow =
```

Once you save the file, restart the 'smbd' service like so:

```
sudo service smbd restart
```

The last step is to add our local 'linuxacademy' account to the Samba password configuration. Easy enough, execute the following:

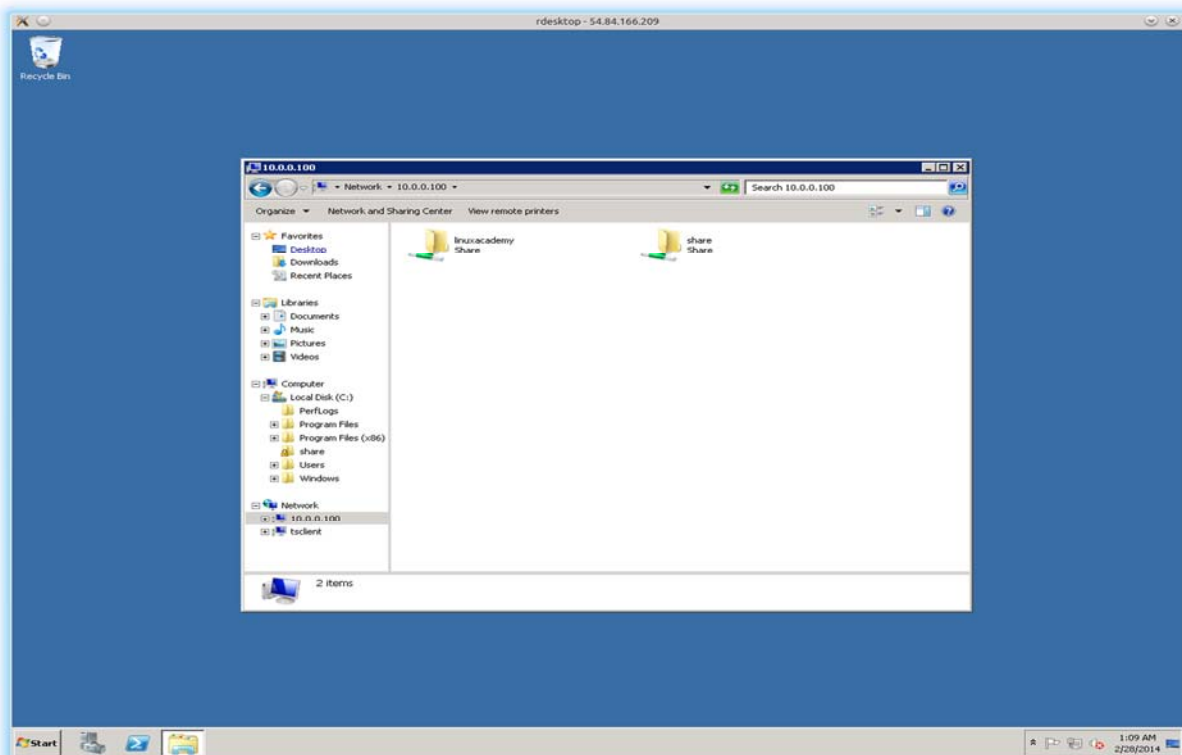
```
sudo smbpasswd -a linuxacademy
New SMB password:      ← give it the default '123456' password
Retype new SMB password:
SMB password for user linuxacademy added.
```

Believe it or not, our sample share is now complete. We are now able to go to our Windows client system and, provided we use the correct credentials, see the share and map it as a network drive.

Windows 2008 R2 Client – Access the Share

Once the server has been configured, the client configuration is straight forward. Please note, this section assumes that you have already disabled the Windows Advanced Firewall in our previous section. If you have not, please return to that section and complete the steps to make sure the firewall will not block the incoming share ports for your client.

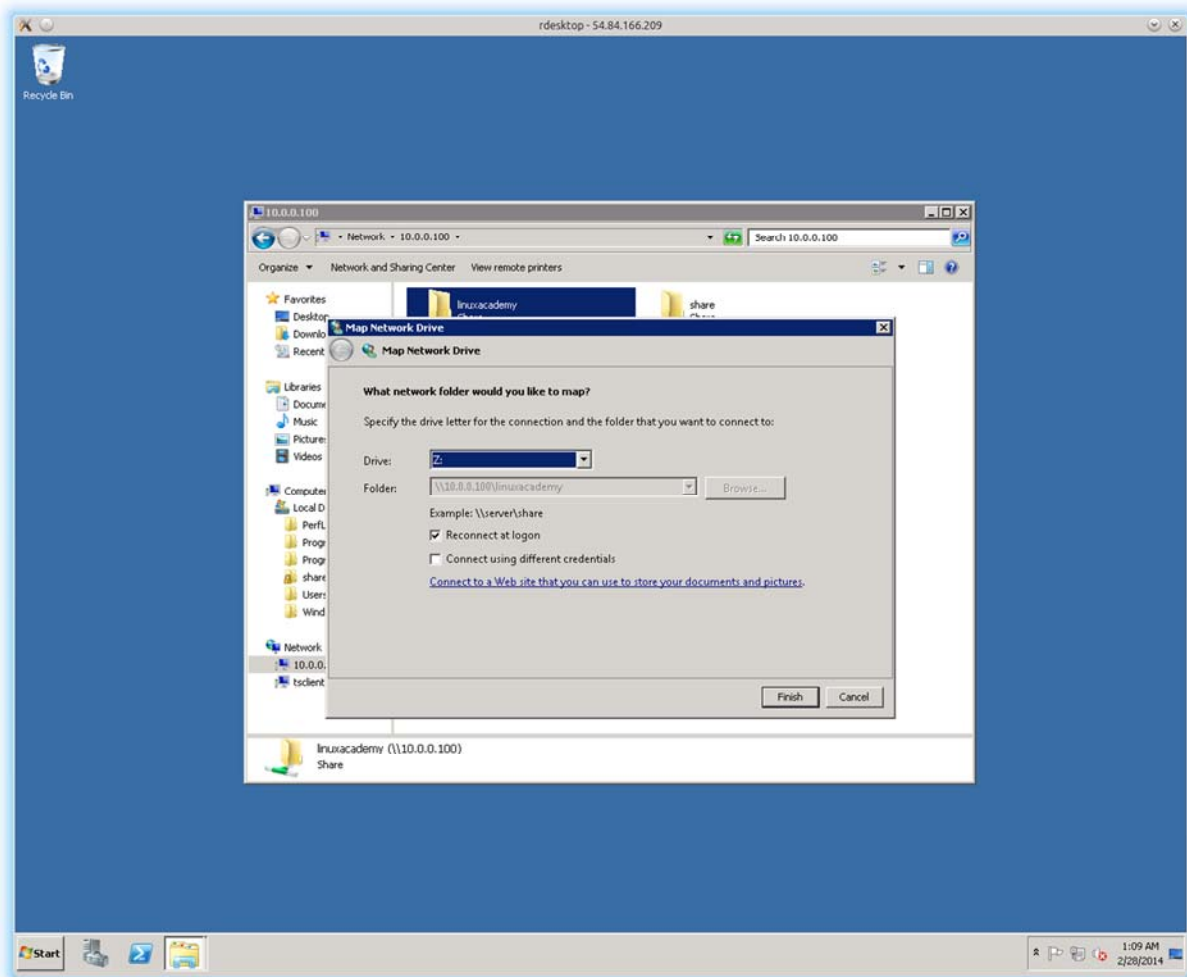
Since we are already logged in as the 'linuxacademy' user AND our local account password matches the Samba password on the Ubuntu server, we can easily access the share (if we had given it a different password on the Ubuntu side, we would be prompted for a user and password to connect to the share with). You will see a screen that looks like this:



Double clicking on that share called 'linuxacademy' will give you access to the share. You can create a file in that folder and then visit your '/home/linuxacademy' directory on the Ubuntu server and see the file you created!

Strangely enough, if you try to double click the 'share' share, you will get an access error message. That is because although we defined the share in the Samba configuration file, we did not create the expected directory on the filesystem. If you want, you can go back and create that directory with appropriate permissions and restart the Samba daemon and observe what happens.

Finally, we can take that Samba share and map a network drive by right clicking and choosing "Map Network Drive", that would look like:



Assigning it the desired drive letter and checking the "Reconnect at Login" is the Windows equivalent of adding an entry to the '/etc/fstab' file in Linux, it will make the share persistently available after reboot.

That's all there is to it, have fun experimenting further!

APPENDIX A – Configuration Files

Sample /etc/samba/smb.conf File

```
# security = user

hosts allow =

[homes]

comment = Local User Home Directories

browseable = no

writable = yes


[share]

comment = Default share

path = /share/

force user = samba

force group = samba

read only = No

hosts allow =
```