

# Linux Network Services: Samba

*This material is based on work supported by the  
National Science Foundation under Grant No. 0802551*



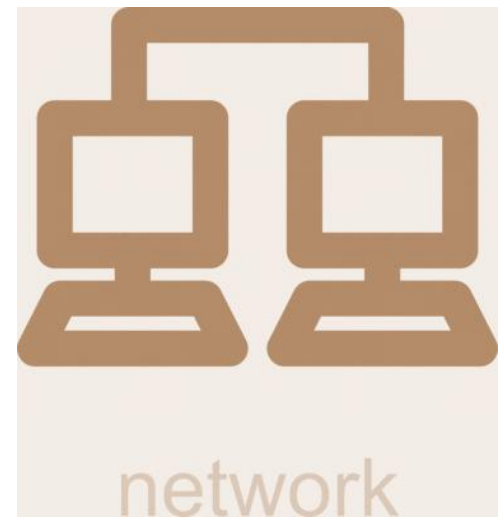
*Any opinions, findings, and conclusions or recommendations expressed in this material are those of  
the author (s) and do not necessarily reflect the views of the National Science Foundation*

# Lesson Overview

The Windows operating system has a strong market share in home and business environments. Windows file sharing and accessibility options are easy to implement. The simplicity of the Windows system allows even a novice computer user to share files from a local host or browse and access files from a remote server.

Like the Windows system, Linux has a file sharing and access tool called SAMBA that provides various capabilities including printing through the TCP/IP protocol.

In this lesson, you will explore SAMBA and the capabilities it offers to Unix and Windows-based systems. SAMBA services are frequently installed on home and business computers. So understanding its use and implementation is necessary for all Linux administrators.



# Objective

You should know what will be expected of you when you complete this lesson. These expectations are presented as objectives. Objectives are short statements of expectations that tell you what you must be able to do, perform, learn, or adjust after reviewing the lesson.

## **Lesson Objective:**

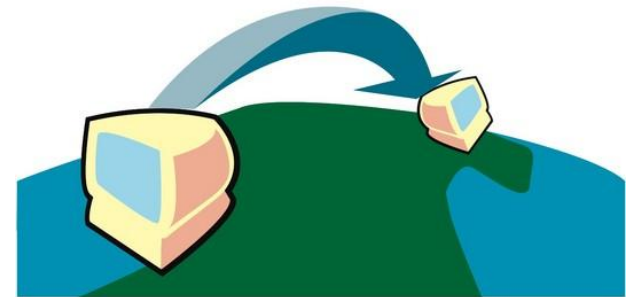
Given a need for file and print sharing, a student will be able to install and configure a SAMBA file and printer server as per industry standards.



# Lesson Outline

In this lesson, you will explore:

- ❖ Overview of Windows share
- ❖ Introduction to Samba
- ❖ Installation of Samba
- ❖ Configuration of Samba (using SWAT)
- ❖ Comparison of Windows share process to Samba
- ❖ Installation and utilization of SWAT to configure Samba



# Resources and Notes

You will need to do a fresh installation of the latest versions of Ubuntu and Fedora before attempting the lab activities in this lesson. Use your favorite virtual machine and create the required operating systems.

The labs and information in this lesson should be tested on your personal home computers, and not on computers used in a production environment.

You will be using Wireshark in the bonus/challenge lab. Wireshark is a powerful program and could lead to potential security risks. Wireshark should never be used in a production or work environment. Most companies will consider it a security risk that could result in instant termination.

Considerable time should be set aside for the labs and discussions in this lesson as compared to previous lessons. While the reading requirements may be less than previous assignments, labs and discussions require time and thought to supplement the concepts and enhance the learning experience. The labs and discussions were designed to be practical, interesting, educational and extremely close to what students may encounter on a regular basis in the real world.

## Resources for Virtual Box:

- [Download Virtualbox](#)
- [Virtualbox for Linux Hosts](#)
- [Install Fedora to Virtualbox](#)
- [Virtualbox manual](#)
- [Using Virtualbox with Ubuntu](#) (process similar to Fedora)

## Resources for Samba:

- [Ubuntu Samba Guide](#)
- [File-sharing in Ubuntu](#)
- [Samba in Ubuntu](#)
- [Windows, Linux, Samba](#)

## YouTube Samba Resources:

- [Share files anonymously](#)
- [File-sharing with Samba](#)
- [Share Linux folders](#)
- [Install & Setup Samba](#)

# What is Samba?

Samba's website describe Samba in this way:

“Samba is an Open Source/Free Software suite that provides seamless file and print services to SMB/CIFS clients.” Samba is freely available, unlike other SMB/CIFS implementations, and allows for interoperability between Linux/Unix servers and Windows-based clients.”

Samba is one of the most important aspects of the Linux operating system because it allows users to comfortably interact with Windows file systems and share/access networked files and folders in an easy manner. Samba allows Windows and Linux users on the same network to access all the network resources without problems. Samba minimizes or eliminates the frustration involved with basic configuration of network shared access to resources.

Samba is an extension of a Server Message Block (SMB) protocol server that runs on most popular Linux flavors. Samba was designed to allow relatively seamless integration with Windows machines and Network Neighborhood (newer versions of Windows may use the term *My Network Places*).

## Recommended Reading

- [Samba Overview](#)
- [SMB Protocol](#)
- [Network Neighborhood](#)
- [CIFS](#)

# Understanding Windows Share

Most computer users should be familiar with Windows and the share process that allows files, folders, and common resources to be shared on a local network. Microsoft has mastered this capability by streamlining the installation and initialization of shared services.

To understand the SAMBA service, you should know a few terms:

- ❖ **Network share** – the term used for any device or resource that can be easily accessed remotely on a local area network.
- ❖ **File sharing** – the ability to provide easy access to file over a local area network by using file share services.
- ❖ **SMB** (Server Message Block) – A protocol originally developed by IBM and then modified by Microsoft for conventional network shares. Originally designed to run on top of the NetBios/NetBeui API. In Windows 2000 and above, SMB is designed to run by default over TCP.(May be referred to as Common Internet File System (CIFS). Commonly uses port 139.

*Interesting side note...* Microsoft modified the original IBM SMB multiple times throughout the early 90's including a major modification to allow it to work with their LAN manager in Windows for Workgroups. Linux enthusiasts (SAMBA Project) counter-acted to those Microsoft changes over the years by modifying the Linux SMB to communicate with the Windows “protocol.”

## Recommended Reading

- [Network Share](#)

# Windows Share (Contd)

Continued from previous . . .

- ❖ **NetBios** (Network Basic Input Output System) – Implemented at the session layer of the OSI model. NetBios allows applications on separate computers to communicate over the local area network.
- ❖ **NetBios name** – the ASCII, 15 character name given to a computer on a local area network. It commonly is used via port 137 (UDP) for local LAN identification.
- ❖ **Host Name** – The hostname is a unique name given to a computer connected to the Internet. The name is divided in two parts. In the case of *www.someipname.com*, the first part (*www*) is the local name, and the second part (*someipname.com*) is the local domain.  
*Note: Case is irrelevant in hostnames.*
- ❖ **Name Resolution** -TCP/IP networks identify systems by IP addresses and always associate these addresses with more human-readable text names.
- ❖ **Workgroup** - workgroup is a group name tag that identifies an arbitrary collection of computers and their resources on an SMB network.
- ❖ **Server** – A computer that has a resource to share

## Recommended Reading

- [NetBios](#)
- [Host Name](#)
- [Name Resolution](#)
- [Workgroup](#)



# Windows Share (Contd)

Continued from previous . . .

- ❖ **Client** – A computer requesting information or a resource from a server. (Note: A computer can be a server, client or both.)
- ❖ **Resource** – a resource is an often overlooked and mistaken term in the IT field. A resource can be anything used with a computer or on a networked system. Examples include memory, files, folders, hard drives, printers, services and more. The people who run or administer computers are the most overlooked resource in most businesses!
- ❖ **Workgroups** - A workgroup is a collection of computers that each maintains its own security information.
- ❖ **Domain** - A domain is a collection of computers where security is handled centrally. The domain controllers maintain account information related to users/clients such as account names, encrypted passwords, authorized hours of use, and groups to which the user belongs. Domains provide a centralized location for administrators to manage account and access information.

## Recommended Reading

- [Server & Client](#)

# Samba Users

Anyone using a Linux-based machine in a large enterprise environment is going to be implementing some aspect of Samba. Corporations, banks and other financial institutions, government, military organizations, schools, public libraries, and research facilities all use Samba as one of the services to access network resources.

Any network configuration that may (or may not) contain a Windows machine may also use Samba services. It is important to note that Samba does not require Windows or Windows machines to work. Samba can be used on a Linux only network to share information.

## Recommended Reading

- [Intro to Samba](#)

# Capabilities of Samba

Samba provides the following services:

- ❖ Share one or more directory trees
- ❖ Share printers installed on the server among Windows clients on the network
- ❖ Participate in NetBIOS browsing and master browser elections.
- ❖ Authenticate clients logging onto a Windows domain
- ❖ Assist with Windows Internet Name Service (WINS) name-server resolution
- ❖ Act as a file/print server for Windows networks. Samba can replace expensive Windows NT file/print servers in this role creating a less expensive solution (Samba is free!).
- ❖ Provide a gateway for the synchronization of UNIX and Windows NT(based) passwords.

At a more specific level, Samba does the following:

- ❖ Open and close files on remote systems
- ❖ Create and delete files and directories on remote systems
- ❖ Read and write files on remote systems
- ❖ Search for files on a network
- ❖ Queue and cancel files in a print spool

## Recommended Reading

- [Samba Capabilities](#)

# What Can Samba Do For Me?

- ❖ Provide equivalent Windows server functions and services at no additional software cost
- ❖ Eliminate the additional licensing costs Client Access Licenses - (CALs) that Microsoft requires for each Windows client to access a Windows server
- ❖ Provide a common area and infrastructure for data or user directories to transition from a Windows environment or Windows server to a Unix one, or possibly vice versa
- ❖ Provide service to share printers and other resources among Windows and Unix workstations and clients
- ❖ Provide unilateral support for a group of computer users who have any combination of Windows NT, 2000, XP, and Unix based computers
- ❖ Allow you to integrate Unix and Windows authentication, which results in a single, centralized database of user accounts that works with both systems
- ❖ Allow you to network various Unix, Windows, Macintosh (OS X) combinations and other systems using one protocol

## Recommended Reading

- [Client Access Licenses](#)

# Components of Samba

- ❖ **smbd** – a daemon that handles file and printer sharing and provides authentication and authorization for SMB clients. (Make sure SmbD is configured to start as a daemon on startup (manually via [x]inetd)
- ❖ **nmbd** – a daemon that supports NetBIOS Name Service and WINS, which is Microsoft's implementation of a NetBIOS Name Server (NBNS). It also assists with network browsing. (*Note: The above listed daemons are the minimum required components for Samba services.*)
- ❖ **winbindd** – a daemon that is used along with the name service switch to get information on users and groups from a Windows NT server and allows Samba to authorize users through a Windows NT/2000 server
- ❖ **findsmb** – a program that searches the local network for computers that respond to SMB protocol and prints information on them
- ❖ **net** – A new program distributed with Samba 3.0 that can be used to perform remote administration of servers
- ❖ **nmblookup** – A program that provides NBT name lookups to find a computer's IP address when given its machine name

## Recommended Reading

- [Samba Suite](#)

# Components of Samba

*Continued from previous . . .*

- ❖ **pdbedit** – A new program distributed with Samba 3.0 that is helpful for managing user accounts held in SAM databases
- ❖ **smbclient** – An *FTP*-like Unix client that can be used to connect to SMB shares and operate on them
- ❖ **smbcontrol** – A simple administrative utility that sends messages to *nmbd* or *smbd*
- ❖ **smbpasswd** – A program that allows an administrator to change the passwords used by Samba
- ❖ **smbsh** – A tool that functions like a command shell to allow access to a remote SMB filesystem and allow Unix utilities to operate on it
- ❖ **smbspool** – A print-spooling program used to send files to remote printers that are shared on the SMB network
- ❖ **smbstatus** – A program that reports the current network connections to the shares on a Samba server

## Recommended Reading

- [Samba Suite](#)

# Simple SMB Connection

For two computers to connect in a SMB process, three actions must occur. They are:

1. Establish a NetBIOS session.
2. Negotiate the protocol variant.
3. Set session parameters and make a tree connection to a resource.

A program such as Wireshark can be used to monitor the session connections and aid in troubleshooting.

Select **PLAY** below for a video on Wireshark packet capture.



View Video  
VideoLesson5PacketCapture  
(C4L5S15).mp4

# Viewing the NetBios Name

When using Wireshark to capture packets from two or more computers, a user may have difficulties identifying which packet belongs to which computer. Using the NetBios names to match data packets will help with the identification. Follow the steps below to view the NetBIOS table:

1. Open Command Prompt.
2. At the command prompt, type:

**nbtstat -n**

The NetBIOS local name table for this computer is displayed as command-line output. The name type is listed to indicate whether each name is a unique name or a group name. Also, the status for each name is given, indicating whether the name is registered or unregistered with the network.

## Recommended Reading

- [NetBios Name](#)



# GUI for Samba: SWAT

SWAT, Samba Web Administration Tool, is the popular GUI interface used to manage and configure Linux shared resources.

SWAT allows system administrators to easily configure Samba services by changing the following Samba attributes:

- ❖ **Globals:** Provides access to the global parameters in smb.conf
- ❖ **View:** Lets you view the current configuration file
- ❖ **Shares:** Lets you create, edit, or drop shares (share management)
- ❖ **Wizard:** Assists users for quick share configurations
- ❖ **Status:** Shows you which services are running: usually smbd (the Samba daemon itself) and nmbd (the NetBIOS nameserver support daemon), and possibly Winbind (to allow a Linux box to become a Windows domain member)
- ❖ **Printers:** Used with printers to share print resources
- ❖ **Password:** Allows you to create, delete, enable, or disable local Samba users and change passwords for a local or remote server.

## Recommended Reading

- [SWAT for Samba](#)
- [Samba & SWAT](#)

# GUI for Samba: Swat (Contd)

## SWAT Tips:

1. Make sure to use the commit changes button whenever you use the SWAT configuration page. You will also need to go to the Status page and stop all services then restart them for changes to take effect.
2. Fedora SWAT – You must enable SWAT to run in the configuration file.
3. SWAT runs in two different modes. The user mode gives the user permission for his/her level of system access and only provides basic capabilities similar to viewing system state and status. The root mode gives the user the capability to change and reconfigure the Samba setup.
4. SWAT requires a web server to provide an interface to the OS. SWAT works very well with Apache (just remember to allow port 901 on your firewall)

Select **PLAY** below for a video on SWAT.



View Video  
VideoLesson5SWAT  
(C4L5S18).mp4

# **SAMBA Installation**

With SWAT

# Installation Notes

There is both good news and bad news to installing Samba and SWAT. The good news is most popular flavors of Linux allow users to install both applications via the standard package management utility. The bad news is that some of the new versions, specifically Ubuntu based versions, have some configuration issues. Administrators should carefully review postings on their particular version of Linux and make sure there is consistency with information available on the Internet. The author of this lesson found numerous errors and contradictions in postings on the Internet. It is suggested that you view all of the postings in blogs and pay particular attention to feedback and responses to suggestions. Ubuntu 10.10 was one version that appeared to have some challenges with configurations and actually induced Samba configuration problems when upgraded from previous working versions of Ubuntu.

Samba and Ubuntu are popular combinations and chances are that somewhere, someone has experienced the same issues you may encounter. Set some time aside for research and testing before attempting the lab activities in this lesson. As always, be extremely careful in a production environment and make sure to perform due diligence before upgrading to newer versions of operating systems or services.

**Tech Tip:** Before installing Samba or SWAT, students should disable all firewalls, SELinux and IPTables if they are in service. Students should then attempt to ping to and from all computers that will be networked together for file sharing. These actions will simplify the initial installation, configuration and troubleshooting. Once the services are confirmed to be running properly, firewalls, SELinux and IPTables should be added, configured and tested one at a time. Beginners will find this process to be more reliable and easier. Even seasoned Linux professionals encounter problems when configuring shares, especially on newer OS distributions.

# Command Line Installation (Ubuntu)

At a command prompt, type: **sudo apt-get install samba**

Then follow the default prompts to complete a default Samba server. Samba uses a separate set of passwords than the standard Linux system accounts (stored in `/etc/samba/smbpasswd`), so you'll need to create a Samba password for your username using this command:

**sudo smbpasswd -a USERNAME**

Once you have created your password, you need to edit your `/etc/samba/smb.conf` file, the configuration file for Samba. Begin by creating a folder named **ubuntutest** on your home folder. Use the command below to create the folder:

**mkdir /home/USERNAME/test**

(Important: You do not want to use **sudo** to create the folder, because then the owning user and group will be set as 'root', which means you won't be able to access the folder using your Samba username and password.)

**sudo gedit /etc/samba/smb.conf**

Select **PLAY** below for a video on Ubuntu setup.



View Video  
VideoLesson5UbuntuSetup(C4L5S21).  
mp4

# Command Line Installation (Ubuntu)

*Continued from previous . . .*

Once `smb.conf` has loaded, add the text below to the very end of the file:

```
[test1]
path = /home/USERNAME/test1
available = yes
read only = no
browseable = yes
public = yes
writable = yes
```

There should be no spaces between the lines there should be a single space both before and after each of the equal signs. These settings will share the `test1` folder we created earlier and give your username permission to read and write to the folder. Save the `smb.conf` and restart Samba with this command:

```
sudo restart smbd
```

Once Samba has restarted, use this command to check your `smb.conf` for any syntax errors:

```
sudo testparm
```

If this “passes” you should be able to share files via this folder. (This command tests the `smb.conf` configuration file for correct syntax but does not check the file to make sure it really does work within the samba daemon.)

# Installing Samba on Fedora 14

From the terminal:

```
sudo yum install samba samba-client cifs-utils
```

You must edit `/etc/samba/smb.conf` as root:

```
$ sudo gedit /etc/samba/smb.conf
```

Set your Windows Workgroup name in [global] section.

Example:

```
[c_drive]  
path = /media/c_drive  
public = yes  
writable = no[netshare]  
path = /data/  
public = yes  
writable = yes[test1]  
path = /data/  
public = yes  
writable = yes
```

## Required Reading

- [SWAT & Fedora](#)

# Installing Samba on Fedora 14 (Contd)

To access shares, you must be a valid user. Add valid *users* and *passwords* using the **smbpasswd** command.

The login name and password you create will be the login name and password you use from Windows to access your Linux computer. The password does NOT need to match your Linux password.

```
$ sudo smbpasswd -a username  
New SMB password:  
Retype new SMB password:  
account_policy_get: (warnings ignored)  
Added user username.
```

(Note: 'username' MUST be a valid account on the Fedora machine)

Enter the command below to run samba and check for errors:

```
$ sudo /etc/init.d/smb start
```



# Installing Samba on Fedora 14 (Contd)

Restart Samba for every change to users/passwords or 'smb.conf'  
(Restarting is extremely important because configuration changes are not applied until you restart the services.)

```
$ sudo /etc/init.d/smb restart
```

## Firewall Configuration

The system firewall will block Samba by default. To allow Samba to run, enter the following command: **\$ system-config-firewall**

Note: To allow Samba access to work through the firewall, you must set *Samba* as a *Trusted Service*.

## SELinux Configuration

SELinux has significant control over restricting different parts of Samba and can really be challenging. It is suggested you disable both SELinux and your firewall application while doing the initial setup. Once you confirm that you are able to share, SELinux and your firewall can be enabled and tweaked for proper service and access to resources.

Select **PLAY** below for a video on Ubuntu setup.



View Video  
VideoLesson5FedoraSwat  
Setup(C4L5S25).mp4

# Sample File (SMB.CONF)

```
# Samba config file created using SWAT
# from UNKNOWN (127.0.0.1)
# Date: 2011/04/26 21:34:41

[global]
  workgroup = WORKGROUP
  netbios name = FEDORA1
  server string = Fedora shares
  interfaces = 192.168.15.103, 127.0.0.1
  security = SHARE
  encrypt passwords = No
  guest account = student
  log file = /var/log/samba/log.%m
  max log size = 50
  name resolve order = bcast hosts lmhosts wins
  guest ok = Yes
  hosts allow = 127.0.0.1, 192.168.15. /24
  cups options = raw
```

*File sample continued on next ...*

```
; /etc/smb.conf
;
; Make sure and restart the server after making
; /etc/rc.d/init.d/smb stop
; /etc/rc.d/init.d/smb start

[global]
; Uncomment this if you want a guest account
; guest account = nobody
  log file = /var/log/samba-log.%m
  lock directory = /var/lock/samba
  share modes = yes

[homes]
  comment = Home Directories
  browseable = no
  read only = no
  create mode = 0750

[tmp]
  comment = Temporary file space
  path = /tmp
  read only = no
  public = yes
```

Image taken from [tldp.org](http://tldp.org).

# Sample File (SMB.CONF)

```
[homes]
  comment = Home Directories
  read only = No
  browseable = No

[printers]
  comment = All Printers
  path = /var/spool/samba
  printable = Yes
  browseable = No

[samba]
  path = /home/student/samba
  read only = No
  hosts allow = 192.168.15.

[fedoratest]
  path = /home/student/Desktop/fedoratest
  read list = ALL
  write list = ALL
  read only = No
```

Select **PLAY** below for a video on SMB.conf file.



View Video  
VideoLesson5SambaConf  
(C4L5S27).mp4

# Samba Troubleshooting

Installing and configuring Samba normally falls in to one of two categories. It either goes really well or there are big challenges. Your success will depend on how well you follow instructions, how well you pay attention, and what distribution of Linux you are using. At a minimum, the following tips should be followed on every installation of Samba.

1. Temporarily disable your network firewall, SELinux and IPTables. See Video on right.
2. Use fresh virtual machines.
3. Install both NMAP and Wireshark for troubleshooting purposes later. Doing this now will prevent the need for reconfiguring your Guest adapter settings to access the Internet once the labs are in progress.
4. Use internal network in Virtual machine for the share setup.
5. Ping back and forth between all Virtual Machines prior to configuring Samba and shares. (Shares will not work unless the computers see each other.)
6. Research several different sources for information on installing Samba on your particular distribution of Linux. Do not work from one tutorial. *Some of the sources researched for this lesson were actually incorrect on the Internet.*
7. Install Apache web server as you install SWAT. Search and check the SWAT configuration file to make sure SWAT is not disabled in the conf file (It is in Fedora 14)
8. Use the same IP schema and subnet (ie... 192.168.15.X 255.255.255.0)
9. Capture some packets from a working shared folder and the interface to use for comparison and troubleshooting at a later time.

View Video  
VideoLesson5Samba  
Firewall(C4L5S28).m  
p4



# Samba Troubleshooting

1. Remember to restart your samba services after every change to the configuration.
2. Patience and research are key.
3. Make sure to select a working interface to use for sharing (ie... eth0 or eth1).
4. Once you get a working shared folder up and running, use Wireshark to capture packets on the adapter. Then use your firewall to block ports used as part of the share process. Block ports on both the *server* and *client* machines. Analyze the packets and save for a future date. Knowing the hand shaking process is a big advantage in troubleshooting shared resources and services.

Select **PLAY** below for a video on Samba troubleshooting.



View Video  
VideoLesson5SambaTroubleshooting(  
C4L5S29).mp4

## Troubleshooting Resources

- ❖ [Troubleshooting Samba](#)
- ❖ [Basic Tips](#)
- ❖ [Samba Checklist](#)
- ❖ [SELinux/Samba](#)
- ❖ [Samba Security](#)
- ❖ [Solaris Troubleshooting](#)

# Lesson Summary

Course 4 Lesson 5 introduced one of the most popular Linux services called Samba. A thorough understanding of Samba and the Windows SMB/share processes is a necessity for technicians entering the IT field.

Samba is recognized as one of the most important services provided by the Windows environment, consequently, technicians must be able to implement and manage shared network resources.

Specifically, lesson 5 allowed students to:

- ❖ Review basic Windows networking terminology
- ❖ Review Windows and Linux shared services
- ❖ Configure Windows, Ubuntu, and Fedora operating systems to share a common folder
- ❖ (Bonus lab) use Wireshark to capture packets used in a shared network environment. Additionally, students were required to review the shared process at a packet level.

## Recommended Reading

- ❖ [Samba Docs](#)