

Ubuntu 7.10 Small Business Server (version 2.0)

This is version 2.0 of my original guide. I am including the original guide with additional notes and modifications. Version 2.0 of the guide also incorporates the addition of Windows shares, Windows login scripts, and NFS mounts. I will go into detail for configuring a Windows XP Professional SP2 client computer and an Ubuntu client computer. RAID1 will be used to ensure data integrity for our user home directories and for our LDAP database. Please note that this is an optional modification to the guide.

Table of Contents

Section #	Section Name
1	Introduction
2	Warranty and Legal
3	Goals
4	Before You Begin
5	My Setup
6	Step 1: Install Ubuntu Server
7	Step 2: Post-installation Configuration
8	Step 3: Configure LDAP Data Directory and LDAP User Home Directories
9	Step 4: Configure RAID1 (Mirroring)
10	Step 5: Install Postfix Mail Agent
11	Step 6: Install OpenLDAP
12	Step 7: Install SAMBA
13	Step 8: Configure OpenLDAP for use with SAMBA
14	Step 9: Configure SAMBA
15	Step 10: Configure the SMLDAP-TOOLS package
16	Step 11: Populate LDAP using smbldap-tools
17	Step 12: Add an LDAP User to the System
18	Step 13: Configure LDAP Authentication on the Server
19	Step 14: Install the BIND DNS Server
20	Step 15: Install and Configure NFS Server Support
21	Step 16: Install Webmin
22	Step 17: Configure BIND9 and the Primary DNS Zone
23	Step 18: Configure the Server to use Itself for DNS
24	Some notes and conclusions
25	Install and Configure Apache2 + PHPLDAPAdmin
26	Configure Ubuntu Server 7.10 (client) to Mount NFS Shares
27	Configure Ubuntu Server 7.10 (client) for LDAP Authentication
28	Configure SAMBA to Share /ldaphome
29	Configure SAMBA - Enable the 'Netlogon' Share

30	Create a Simple Windows Logon Script
31	Appendix A: Final /etc/samba/smb.conf File
32	Appendix B: Final /etc/ldap/slapd.conf File
33	Appendix C: Windows XP Professional SP2 Client Configuration Notes
34	Appendix D: Ubuntu Server 7.10 (Client) Configuration Notes
35	Appendix E: Final Notes and Observations From The Author

Introduction

Top

This is version 2.0 of my original guide. I am including the original guide with additional notes and modifications. Version 2.0 of the guide also incorporates the addition of Windows shares, Windows login scripts, and NFS mounts. I will go into detail for configuring a Windows XP Professional SP2 client computer and an Ubuntu client computer. RAID1 will be used to ensure data integrity for our user home directories and for our LDAP database. Please note that this is an optional modification to the guide.

Much of the work on this guide has been done for my own amusement and proof of concept, as I am a computer consultant that continually looks for the best way to serve my customers. As such the guide will need to be customized for your exact scenario. Also note that because I put this guide on the internet it means I believe in it and that I know it works. If you go through my guide and copy/paste every command then this WILL work without issue. If you make a change you must ensure that you follow the change throughout the guide.

Please note, and this is very important, this guide only applies to the SAMBA3 branch. SAMBA4 is in development and will supposedly make most of this guide obsolete. When that happens count on a new guide based on the new technology found in SAMBA4.

Top

Warranty and Legal

Top

I provide this guide with absolutely zero liability on my side. I do not warranty this guide. By following this guide you agree that I cannot be held responsible for the end results (unless those results are good, in which case you may send me a check or cash for my troubles. :-). Now back to the serious part. I do not have the time to provide free support in response to this guide.

HOWEVER, you may feel free to send me an email or post a response in some manner to this guide and I will be happy to help you through your issue as I have the time. This guide has been written by Richard Maloley II. This is my intellectual property, in addition to my words. Please respect my work. Do not claim it as your own or anyone elses. You may copy, print, use, reuse, study, adore, and distribute this guide to your content. Although my name must remain as the author and you must keep my copyright in place.

Top

Goals

Top

The overall goal is to have a server computer with the role of "domain controller." My definition of domain controller is a server computer with a central user database that client computers can authenticate against. This guide will accomplish the following goals:

1. Central user authentication using an LDAP database
2. Central storage of users home directories using a combination of NFS and SAMBA
3. The creation of a SAMBA domain that Windows XP Professional SP2 computers can "join" and participate in
4. A DNS server that can be used on your network
5. Data integrity from the use of RAID1 arrays for user and LDAP data

Top

Before You Begin

Top

You must understand that this guide was written to be a proof-of-concept for a fully opensource domain controller for small businesses. I am a computer consultant who wants to deliver the very best solution at the lowest cost to my customers. Whatever solution I deliver must lower their total cost of ownership and must be able to be managed by someone other than myself. There are some pieces of this guide that will not apply to you. I will do my best to point these out and to give you good notes to follow. This guide is also written in such a way that you can take a piece from here and there and use it in your own system. Hopefully this will help you. I wrote this guide for my own use and I am giving it to the opensource community for their use as well. I believe that good documentation is something that is missing from opensource software, therefore I do my best to give back to the community. Once again I must stress that I can only give limited support in regards to this guide. I do not warranty it. Follow this guide under your own supervision! With all that said we can begin :-).

Top

My Setup

Top

Let me explain how I have my environment configured because this will give you a better idea on how to do things. I highly recommend configuring a virtual test environment before moving on to a physical testing environment and definitely before moving on to a production environment. I do not care how small your network is, doing this on production machines without testing is DUMB. I do not support DUMB configurations. I do things by the book and if you follow this guide then I ask that you also do it "by the book."

- I use VMWare Server (the free edition) for all my testing.
- In VMWare I configured a single server for this guide. This server was configured with 384MB of RAM (I recommend increasing this to 512MB if you encounter issues on boot up where slapd hangs) and five (5) SCSI virtual hard drives. Each virtual hard drive was defined as having four (4) GB of space.
- I installed Ubuntu Server 7.10 32bit to the first hard drive that was defined. I left everything at default. I let the Ubuntu installer partition the hard drive however it wanted to.

- The username that I defined during the install was "sysadmin"
- The password that I defined during the install was "12345"
- Yes, that is a very insecure password. I use that password because this is a tutorial and frankly typing my normal password 500 times when writing this password was cumbersome. Next time I will only use "1" for the password because it is even easier!!!!
- During the install I did not install any additional software.
- After the installation I configured a static IP address.
- After the installation I configured /etc/apt/sources.list so that it did not use the CD and so that all extra repositories were enabled. I am not an expert in APT so I have no clue what enabling the repositories really does. All I know is that this way I can get all the software I need without issue. I recommend you do the same.
- After the installation I updated apt (apt-get update) and then I updated my system (apt-get upgrade).
- After the system was updated I installed the OpenSSH Server package for remote access (apt-get install openssh-server).
- At this point I did everything via remote terminal (PUTTY in Windows).
- In my setup I configured two RAID1 arrays from the four (4) additional SCSI hard drives that I defined in VMWare. One array was to be used for the LDAP data (which I mounted at /ldap_data). The second array was to be used for the LDAP Users home directories (which I mounted at /ldaphome). Please note that this entire step is optional (well, the RAID is, I did in fact move the LDAP data and LDAP home directories so you will need to configure the folders at some point.)
- In VMWare I also configured a second Ubuntu server for the client. I also configured a Windows XP Professional SP2 client. Because I'm writing this you can be sure that both clients worked on the domain. The result was that the Ubuntu client mounted the NFS share for the LDAP users home directories and my LDAP users were able to access their files and save their files on the server. The Windows client was able to log in and a login script mounted their home folder as their H: drive. They were also able to read and write to their home folder. However the Windows client was unable to modify the Access Control List and I'm still trying to figure that one out. Version 3.0 of the guide will hopefully have that one figured out correctly, but that depends greatly on feedback given to me and if the SAMBA team will help out. At this time I am unsure of who to contact regarding the SAMBA team on fixing this little issue.

I tried to keep my setup simple and to the point. As I've said, my setup is probably different from yours. Therefore when following this guide you will need to adapt it to your own setup. This will probably lead to problems. I ask that you read through the entire guide at least once before you attempt to try it. Then write down the changes that you know you'll need to make. Then make sure to follow through with those changes throughout the guide. For example, if you change the LDAP password in the beginning then you must use the same password EVERYWHERE.

OK - Now we can actually get into the meat of this article - how do you do it all?

Top

Step 1: Install Ubuntu Server

Top

Usually I would assume that this has already been accomplished, however experience has shown that I need to include this. The steps here are pretty simple and straightforward.

1. Begin the installation by using your wonderful Ubuntu Server 7.10 CD. Please note that this is the 32bit version. I AM NOT USING 64BIT FOR THIS TUTORIAL! If you'd like to test using 64bit then I would happily accept your results for version 3.0.
2. Naturally we wish to install Ubuntu, so tell the installer menu that you'll be installing Ubuntu to the hard disk today.
3. These are the exact options that I ended up using:
 1. English
 2. United States
 3. No
 4. U.S. English
 5. U.S. English
 6. dc01-ubuntu
 7. Guided - use entire disk
 8. SCSI3 - (0,0,0) (sda)
 9. Yes
 10. Eastern
 11. Yes
 12. sysadmin
 13. sysadmin
 14. 12345
 15. 12345
 16. Do not choose any extra software to install. (Continue)
 17. Remove CD when told to and Reboot.
4. Ubuntu Server is now installed and should be rebooting.
5. Ensure that you can log in to the system using "sysadmin" (or whichever username you assigned) and the password that you assigned. If you cannot log in then you might need to redo the installation...

Top

Step 2: Post-installation Configuration

Top

Congratulations! You made it to step 2. Now we need to configure some of the basics. Here is an overview: Configure APT; configure a static IP address; update the server; install OpenSSH Server;

configure external time sync... Please note that I am not using SUDO in front of my commands. I find it easier to do everything from a root bash prompt. To get there type "sudo bash" and enter your password.

Configure APT

First create a backup of the /etc/apt/sources.list file:

```
cp /etc/apt/sources.list /etc/apt/sources.list.original
```

Now we need to edit the file. I use VIM to do all my editing. If you don't know how to use VIM then I recommend using NANO to edit the file as it is a lot easier to use.

```
vim /etc/apt/sources.list
```

We will be commenting out the CD-ROM lines and will uncomment all the extra repositories. I'm posting a copy of my file for your reference.

```
/etc/apt/sources.list
```

```
#
# deb cdrom:[Ubuntu-Server 7.10 _Gutsy Gibbon_ - Release i386 (20071016)]/ gutsy
main restricted

#deb cdrom:[Ubuntu-Server 7.10 _Gutsy Gibbon_ - Release i386 (20071016)]/ gutsy
main restricted
# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to
# newer versions of the distribution.

deb http://us.archive.ubuntu.com/ubuntu/ gutsy main restricted
deb-src http://us.archive.ubuntu.com/ubuntu/ gutsy main restricted

## Major bug fix updates produced after the final release of the
## distribution.
deb http://us.archive.ubuntu.com/ubuntu/ gutsy-updates main restricted
deb-src http://us.archive.ubuntu.com/ubuntu/ gutsy-updates main restricted

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team, and may not be under a free licence. Please satisfy yourself as to
## your rights to use the software. Also, please note that software in
## universe WILL NOT receive any review or updates from the Ubuntu security
## team.
deb http://us.archive.ubuntu.com/ubuntu/ gutsy universe
deb-src http://us.archive.ubuntu.com/ubuntu/ gutsy universe
deb http://us.archive.ubuntu.com/ubuntu/ gutsy-updates universe
deb-src http://us.archive.ubuntu.com/ubuntu/ gutsy-updates universe

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team, and may not be under a free licence. Please satisfy yourself as to
## your rights to use the software. Also, please note that software in
## multiverse WILL NOT receive any review or updates from the Ubuntu
## security team.
deb http://us.archive.ubuntu.com/ubuntu/ gutsy multiverse
deb-src http://us.archive.ubuntu.com/ubuntu/ gutsy multiverse
deb http://us.archive.ubuntu.com/ubuntu/ gutsy-updates multiverse
deb-src http://us.archive.ubuntu.com/ubuntu/ gutsy-updates multiverse

## Uncomment the following two lines to add software from the 'backports'
## repository.
## N.B. software from this repository may not have been tested as
## extensively as that contained in the main release, although it includes
## newer versions of some applications which may provide useful features.
## Also, please note that software in backports WILL NOT receive any review
```

```
## or updates from the Ubuntu security team.
deb http://us.archive.ubuntu.com/ubuntu/ gutsy-backports main restricted
universe multiverse
deb-src http://us.archive.ubuntu.com/ubuntu/ gutsy-backports main restricted
universe multiverse

## Uncomment the following two lines to add software from Canonical's
## 'partner' repository. This software is not part of Ubuntu, but is
## offered by Canonical and the respective vendors as a service to Ubuntu
## users.
deb http://archive.canonical.com/ubuntu gutsy partner
deb-src http://archive.canonical.com/ubuntu gutsy partner

deb http://security.ubuntu.com/ubuntu gutsy-security main restricted
deb-src http://security.ubuntu.com/ubuntu gutsy-security main restricted
deb http://security.ubuntu.com/ubuntu gutsy-security universe
deb-src http://security.ubuntu.com/ubuntu gutsy-security universe
deb http://security.ubuntu.com/ubuntu gutsy-security multiverse
```

Update APT

```
apt-get update
```

Update the System

```
apt-get upgrade
```

Install OpenSSH Server

```
apt-get install openssh-server
```

Configure a Static IP Address

I will be using the IP address 192.168.0.60. Now, you will obviously need to use an IP address that works on your network. Take special note here because you must remember YOUR assigned IP address because you will be making use of it later on.

The file you need to edit is `/etc/network/interfaces`. I will post a copy of my edited file for you to use as a reference.

```
/etc/network/interfaces
```

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
```

```
# The loopback network interface
auto lo
iface lo inet loopback
```

```
# The primary network interface
auto eth0
#iface eth0 inet dhcp
iface eth0 inet static
    address 192.168.0.60
    netmask 255.255.255.0
    gateway 192.168.0.1
```

Configure A Fully Qualified Domain Name

We need to change our hostname to be a fully qualified domain name (FQDN). The safe way to do this is to add it to the `/etc/hosts` file and then edit the `/etc/hostname` file to reflect the change. Your

FQDN if you follow this guide exactly will be dc01-ubuntu.example.local.

Once again I will post the command and my resulting file for your reference.

```
vim /etc/hosts
/etc/hosts

127.0.0.1      localhost
127.0.1.1      dc01-ubuntu dc01-ubuntu.example.local

# The following lines are desirable for IPv6 capable hosts
::1           ip6-localhost ip6-loopback
fe00::0       ip6-localnet
ff00::0       ip6-mcastprefix
ff02::1       ip6-allnodes
ff02::2       ip6-allrouters
ff02::3       ip6-allhosts
vim /etc/hostname
/etc/hostname

dc01-ubuntu.example.local
```

Configure External Time Sync

This step can be optional if you prefer. I feel as though this should be required, however. In a network with a client/server model you want every device to have the exact same time. Otherwise concurrent file access and other items could run into unexpected problems. From a security stand point you want to make sure that all devices have the same time to track file changes in the case of an intruder. As I said, this is optional but I highly recommend it.

First install the NTP service. This is a small install and is very easy to configure.

```
apt-get install ntp
```

Now we need to edit the file /etc/ntp.conf and add an additional line to the file. Add "server pool.ntp.org" below "server ntp.ubuntu.com". Here is the command:

```
vim /etc/ntp.conf
```

Here is a copy of my file after making the change.

```
/etc/ntp.conf

# /etc/ntp.conf, configuration for ntpd

driftfile /var/lib/ntp/ntp.drift

# Enable this if you want statistics to be logged.
#statsdir /var/log/ntpstats/

statistics loopstats peerstats clockstats
filegen loopstats file loopstats type day enable
filegen peerstats file peerstats type day enable
filegen clockstats file clockstats type day enable

# You do need to talk to an NTP server or two (or three).
server ntp.ubuntu.com
server pool.ntp.org

# By default, exchange time with everybody, but don't allow configuration.
# See /usr/share/doc/ntp-doc/html/acopt.html for details.
restrict -4 default kod notrap nomodify nopeer noquery
restrict -6 default kod notrap nomodify nopeer noquery
```

```
# Local users may interrogate the ntp server more closely.
restrict 127.0.0.1
restrict ::1

# Clients from this (example!) subnet have unlimited access,
# but only if cryptographically authenticated
#restrict 192.168.123.0 mask 255.255.255.0 notrust

# If you want to provide time to your local subnet, change the next line.
# (Again, the address is an example only.)
#broadcast 192.168.123.255

# If you want to listen to time broadcasts on your local subnet,
# de-comment the next lines. Please do this only if you trust everybody
# on the network!
#disable auth
#broadcastclient
Now we will reboot the server to ensure that everything is working properly.

shutdown -r now
OR

reboot
Top
```

Step 3: Configure LDAP Data Directory and LDAP User Home Directories

Top

We will be making two directories. However, pay attention here, because this is important. The `/ldaphome` directory **MUST** be created, do not skip that. The `/ldap_data` directory is optional depending on how you wish to install and configure OpenLDAP. In that section I show you two different ways for configuring OpenLDAP. If you will be leaving OpenLDAP in the default directory then you do not need to create the `/ldap_data` directory.

Run the following commands to create the directories:

```
mkdir /ldaphome
mkdir /ldap_data
Top
```

Step 4: Configure RAID1 (Mirroring)

Top

This is an optional step. I'm including these notes for those of you who have the hard drives and would like the data integrity and security. Basically we are going to use a program called CFDISK to partition and configure our hard drives. We will then use the program MDADM to setup each of our RAID arrays. We will then configure the MDADM configuration file so that our arrays are recognized automatically in the future. Then we will format each array and mount each array in their designated directories. The final step will be to configure our `/etc/fstab` configuration file to automatically mount our arrays at bootup. Once again, this is optional. If you are not using RAID then you can safely ignore this step.

Install the MDADM software package.

```
apt-get install mdadm
```

Next we need to use CFDISK to partition and configure our hard drives. Basically each hard drive needs a partition. Make it a primary partition. You will be using type "fd" for Linux raid. Please be sure to put the correct /dev/xxx in the command. I recommend writing out what you'll be doing and going off that sheet so it is less confusing.

```
cfdisk /dev/sdb
cfdisk /dev/sdc
cfdisk /dev/sdd
cfdisk /dev/sde
```

Now we can create the first array.

```
mdadm --create --verbose /dev/md0 --level=1 --raid-devices=2 /dev/sdb1 /dev/sdc1
```

OK - that command is definitely confusing. Here is what it all means. We are invoking the program and telling it to create a new RAID device. The program is going to give us as much information as possible. The device it is going to create is /dev/md0. RAID1 will be used for the device. Only two devices are going to be participating in the array. Those two devices are /dev/sdb1 and /dev/sdc1.

Next format the array with the ext3 filesystem. Naturally you can use whatever filesystem you want, but this is what I am familiar with.

```
mkfs.ext3 /dev/md0
```

Now we can create the second array.

```
mdadm --create --verbose /dev/md1 --level=1 --raid-devices=2 /dev/sdd1 /dev/sde1
```

Next format the array with the ext3 filesystem. Naturally you can use whatever filesystem you want, but this is what I am familiar with.

```
mkfs.ext3 /dev/md1
```

Great! Now we have our two arrays. The next thing we need to do is define these two arrays in our /etc/mdadm.conf file.

```
vim /etc/mdadm.conf
/etc/mdadm.conf
```

```
DEVICE          /dev/sdb1 /dev/sdc1 /dev/sdd1 /dev/sde1
ARRAY           /dev/md0 devices=/dev/sdb1,/dev/sdc1
ARRAY           /dev/md1 devices=/dev/sdd1,/dev/sde1
```

Alright, go ahead and try mounting the RAID arrays to their respective folders. In my case /dev/md0 will be mounted at /ldap_data and /dev/md1 will be mounted at /ldaphome.

```
mount /dev/md0 /ldap_data
mount /dev/md1 /ldaphome
```

Does it work? If not then you have your work cut out for you. If yes then continue.

Let's add the mounting information to the /etc/fstab file. We will be adding the following lines:

```
# Custom RAID entries
/dev/md0 /ldap_data ext3 defaults,errors=remount-ro 0 1
/dev/md1 /ldaphome ext3 defaults,errors=remount-ro 0 1
```

```
vim /etc/fstab
/etc/fstab
```

```
# /etc/fstab: static file system information.
```

```
#
```

```
#
```

```
proc          /proc          proc          defaults      0            0
```

```
# /dev/sda1
```

```
UUID=09afe0b0-d7df-4322-bd07-fa0854041a6f /              ext3
```

```
defaults,errors=remount-ro 0            1
```

```
# /dev/sda5
```

```
UUID=d557816b-8149-46ea-b6fb-dd674231e597 none          swap          sw
```

```
0          0
/dev/scd0   /media/cdrom0   udf,iso9660 user,noauto,exec 0      0
/dev/fd0    /media/floppy0   auto      rw,user,noauto,exec 0      0
```

```
# Custom RAID entries
/dev/md0 /ldap_data ext3 defaults,errors=remount-ro 0 1
/dev/md1 /ldaphome ext3 defaults,errors=remount-ro 0 1
Now reboot the server and ensure that everything mounts correctly!
```

reboot
Top

Step 5: Install Postfix Mail Agent

Top

We will be installing Postfix for several reasons. One, the system needs a mailserver in order to email reports about the RAID arrays and other items of interest. Two, you might wish to use a mail server for other tasks. Three, it just makes things easier. Four, the reason I chose to install Postfix is because it is the only mail server that I am familiar with. Like something else? Good for you, use it.

I guess that the first thing to do would be to actually install it:

```
apt-get install postfix mailx
```

During the installation it will ask you some questions. Answer as follows:

```
Internet site
dc01-ubuntu.example.local
```

Naturally you will want to customize those answers to tailor to your environment, but if you are following this guide exactly then the answers I provide should be sufficient.

Top

Step 6: Install OpenLDAP

Top

You might notice that this step is very similar to Step 2 in the original guide. What I've done in version 2.0 is change the order slightly and move some steps into their own sections to simplify the entire guide. My hope is that this will be easier to follow and use.

OK, well we need to install OpenLDAP at this point. We're using OpenLDAP as opposed to other LDAP servers for one reason and one only: This is the only program that I found good documentation for in regards to SAMBA and other services. I'm fairly certain that you can use Novell and other LDAP servers in place of OpenLDAP. Please be advised that those are beyond my comprehension at this time and I'd rather stick to the standard - OpenLDAP in this case.

There are two ways to configure OpenLDAP. In one configuration we will have OpenLDAP store its data in a different directory than default. I do this so that the directory can be on its own hard drive for backup purposes. Others may wish to "leave it as it is." That is fine. This guide will work either way. Therefore I have two sub-sections here. The first section describes how to install and configure OpenLDAP with the default directory. The second section shows you how to customize it.

OpenLDAP with the Default Directory

Install OpenLDAP:

```
apt-get install slapd ldap-utils migrationtools
```

This installs more than just OpenLDAP - it installs other utilities that can be of assistance to you.

During the installation you will be prompted to supply an Admin password and then to confirm it:

```
Admin password: 12345
```

```
Confirm password: 12345
```

Now we need to reconfigure OpenLDAP and customize it to our needs.

```
dpkg-reconfigure slapd
```

Naturally this will also prompt you for some information. Here are the answers that I am using.

Please note that when you deviate here you must also follow suit everywhere else! If you change the domain name then change it everywhere else!

```
No
```

```
DNS domain name: example.local
```

```
Name of your organization: example.local
```

```
Admin password: 12345
```

```
Confirm password: 12345
```

```
OK
```

```
BDB
```

```
No
```

```
Yes
```

```
No
```

And now you have OpenLDAP installed!

OpenLDAP with a Customized Directory

Install OpenLDAP:

```
apt-get install slapd ldap-utils migrationtools
```

This installs more than just OpenLDAP - it installs other utilities that can be of assistance to you.

During the installation you will be prompted to answer some questions. Here are the answers that I am using:

```
Admin password: 12345
```

```
Confirm password: 12345
```

```
Reconfigure OpenLDAP:
```

```
dpkg-reconfigure slapd
```

```
Answers:
```

```
No
```

```
DNS domain name: example.local
```

```
Name of your organization: example.local
```

```
Admin password: 12345
```

```
Confirm password: 12345
```

```
OK
```

```
BDB
```

```
Yes
```

```
Yes
```

```
No
```

Stop OpenLDAP:

```
/etc/init.d/slapd stop
```

Edit the file `/etc/ldap/slapd.conf` and change the directory. In the file find the first "directory `/var/lib/ldap`" and change it to "directory `/ldap_data`"

```
vim /etc/ldap/slapd.conf
```

Copy all the current DB files into our new directory:

```
cp -R /var/lib/ldap/* /ldap_data/
```

Set the correct permissions on the new directory and files:

```
chown -R openldap:openldap /ldap_data/  
Yes, we need to reconfigure OpenLDAP yet again.
```

```
dpkg-reconfigure slapd
```

Answers:

```
No  
DNS domain name: example.local  
Name of your organization: example.local  
Admin password: 12345  
Confirm password: 12345  
OK  
BDB  
Yes  
Yes  
No
```

Now start OpenLDAP:

```
/etc/init.d/slapd start
```

Here is a copy of my /etc/ldap/slapd.conf file after this initial change:

```
/etc/ldap/slapd.conf
```

```
# This is the main slapd configuration file. See slapd.conf(5) for more  
# info on the configuration options.
```

```
#####  
# Global Directives:
```

```
# Features to permit  
#allow bind_v2
```

```
# Schema and objectClass definitions  
include /etc/ldap/schema/core.schema  
include /etc/ldap/schema/cosine.schema  
include /etc/ldap/schema/nis.schema  
include /etc/ldap/schema/inetorgperson.schema
```

```
# Where the pid file is put. The init.d script  
# will not stop the server if you change this.  
pidfile /var/run/slapd/slapd.pid
```

```
# List of arguments that were passed to the server  
argsfile /var/run/slapd/slapd.args
```

```
# Read slapd.conf(5) for possible values  
loglevel 0
```

```
# Where the dynamically loaded modules are stored  
modulepath /usr/lib/ldap  
moduleload back_bdb
```

```
# The maximum number of entries that is returned for a search operation  
sizelimit 500
```

```
# The tool-threads parameter sets the actual amount of cpu's that is used  
# for indexing.  
tool-threads 1
```

```
#####  
# Specific Backend Directives for bdb:
```

```

# Backend specific directives apply to this backend until another
# 'backend' directive occurs
backend          bdb
checkpoint 512 30

#####
# Specific Backend Directives for 'other':
# Backend specific directives apply to this backend until another
# 'backend' directive occurs
#backend

#####
# Specific Directives for database #1, of type bdb:
# Database specific directives apply to this database until another
# 'database' directive occurs
database        bdb

# The base of your directory in database #1
suffix          "dc=nodomain"

# rootdn directive for specifying a superuser on the database. This is needed
# for syncrepl.
# rootdn        "cn=admin,dc=nodomain"

# Where the database file are physically stored for database #1
#directory      "/var/lib/ldap"
directory       "/ldap_data"

# For the Debian package we use 2MB as default but be sure to update this
# value if you have plenty of RAM
dbconfig set_cachesize 0 2097152 0

# Sven Hartge reported that he had to set this value incredibly high
# to get slapd running at all. See http://bugs.debian.org/303057
# for more information.

# Number of objects that can be locked at the same time.
dbconfig set_lk_max_objects 1500
# Number of locks (both requested and granted)
dbconfig set_lk_max_locks 1500
# Number of lockers
dbconfig set_lk_max_lockers 1500

# Indexing options for database #1
index           objectClass eq

# Save the time that the entry gets modified, for database #1
lastmod        on

# Where to store the replica logs for database #1
# relogfile     /var/lib/ldap/replog

# The userPassword by default can be changed
# by the entry owning it if they are authenticated.
# Others should not be able to see it, except the
# admin entry below
# These access lines apply to database #1 only
access to attrs=userPassword,shadowLastChange
        by dn="cn=admin,dc=nodomain" write
        by anonymous auth
        by self write
        by * none

```

```

# Ensure read access to the base for things like
# supportedSASLMechanisms. Without this you may
# have problems with SASL not knowing what
# mechanisms are available and the like.
# Note that this is covered by the 'access to *'
# ACL below too but if you change that as people
# are wont to do you'll still need this if you
# want SASL (and possible other things) to work
# happily.
access to dn.base="" by * read

# The admin dn has full write access, everyone else
# can read everything.
access to *
    by dn="cn=admin,dc=nodomain" write
    by * read

# For Netscape Roaming support, each user gets a roaming
# profile for which they have write access to
#access to dn=".*,ou=Roaming,o=morsnet"
#    by dn="cn=admin,dc=nodomain" write
#    by dnattr=owner write

#####
# Specific Directives for database #2, of type 'other' (can be bdb too):
# Database specific directives apply to this database until another
# 'database' directive occurs
#database

# The base of your directory for database #2
#suffix "dc=debian,dc=org"
I'm a firm believer in fully testing everything. Therefore I recommend rebooting. If you don't wish
to perform a full reboot then go ahead and just restart OpenLDAP.

reboot
OR:
/etc/init.d/slapd restart
Now OpenLDAP is installed and it should be functional. You can verify that it is running by
scanning your server with a portscanner, like NMAP.

Top

```

Step 7: Install SAMBA

Top

We want to install SAMBA because we wish to have a domain the Windows clients can participate in. We also want to share files, etc... SAMBA is a good program for this. One thing to look forward to is the fact that SAMBA now has access to Microsoft documents that detail the SMB protocol. What does this mean? Well it hopefully means that in the future SAMBA and Windows will be able to interoperate without issues.

It has been pointed out that this step could be optional in some situations. For example, if you are running a Linux only network then yes, this part could be optional. And so will several other parts. Also, if you wish to separate your services and run SAMBA on a different server. Therefore look at these directions as a guide in those situations and for the second server example you should be able to follow most of the same steps without issue and have it work, providing DNS works that is.

For the majority of people following this guide then this is a required step. Please don't deviate unless you know what you are doing.

Install the required software:

```
apt-get install samba smbldap-tools smbclient samba-doc
```

There should be no prompts for answers or any additional configuration.

Top

Step 8: Configure OpenLDAP for use with SAMBA

Top

By default OpenLDAP is not configured to work with SAMBA. We need to tell OpenLDAP that SAMBA is there and how to talk to it. We do this by installing a schema file for OpenLDAP that describes SAMBA.

Run the following commands to install the file in the correct location:

```
cp /usr/share/doc/samba-doc/examples/LDAP/samba.schema.gz /etc/ldap/schema/  
gzip -d /etc/ldap/schema/samba.schema.gz
```

Now we need to edit the OpenLDAP configuration file, again. I wish this step could have been earlier but if we did that then OpenLDAP complains about missing items.

```
vim /etc/ldap/slapd.conf
```

Find the lines that begin with "include" - you'll notice that this is how OpenLDAP knows about other configuration files. Now add the following two lines below the other "include" lines:

```
include          /etc/ldap/schema/samba.schema  
include          /etc/ldap/schema/misc.schema
```

While in the file we need to change another line. Find the line that says "access to attribute=userPassword,shadowLastChange" and change it to:

```
access to attrs=userPassword,shadowLastChange,sambaNTPassword,sambaLMPassword
```

Now we can either reboot the server or just restart the service:

```
reboot
```

OR:

```
/etc/init.d/slapd restart
```

Top

Step 9: Configure SAMBA

Top

This step can become complicated so be sure to read through it and figure out what you want to do. The only file that we will be editing is the file /etc/samba/smb.conf. We will make a backup of this file before we begin, so in case of a screw up you can just restore the backup. In this file we will configure the domain name, how LDAP works, etc... Please be sure to verify every aspect of the file otherwise you will run into problems.

First enter the SAMBA directory:

```
cd /etc/samba/
```

Now backup the smb.conf file:

```
cp smb.conf smb.conf.original
```

Open the smb.conf file for editing:

vim smb.conf

OK - this next part is not exactly copy and paste. First and foremost, find the following items and change them to what I have:

```
workgroup = EXAMPLE
security = user
passdb backend = ldapsam:ldap://localhost/
obey pam restrictions = no
```

Now copy and paste the following lines just below the line "obey pam restrictions = no":

```
#####
#COPY AND PASTE THE FOLLOWING UNDERNEATH "OBEY PAM RESTRICTIONS = NO"
#####
#
#       Begin: Custom LDAP Entries
#
ldap admin dn = cn=admin,dc=example,dc=local
ldap suffix = dc=example, dc=local
ldap group suffix = ou=Groups
ldap user suffix = ou=Users
ldap machine suffix = ou=Computers
ldap idmap suffix = ou=Users
; Do ldap passwd sync
ldap passwd sync = Yes
passwd program = /usr/sbin/smbldap-passwd %u
passwd chat = *New*password* %n\n *Retye*new*password* %n\n
*all*authentication*tokens*updated*
add user script = /usr/sbin/smbldap-useradd -m "%u"
ldap delete dn = Yes
delete user script = /usr/sbin/smbldap-userdel "%u"
add machine script = /usr/sbin/smbldap-useradd -w "%u"
add group script = /usr/sbin/smbldap-groupadd -p "%g"
delete group script = /usr/sbin/smbldap-groupdel "%g"
add user to group script = /usr/sbin/smbldap-groupmod -m "%u" "%g"
delete user from group script = /usr/sbin/smbldap-groupmod -x "%u" "%g"
set primary group script = /usr/sbin/smbldap-usermod -g "%g" "%u"
domain logons = yes
#
#       End: Custom LDAP Entries
#
#####
#STOP COPYING HERE!
#####
```

Obviously in the previous two smb.conf configuration steps you'll want to change the information to suit your needs. Please remember this!

Now comment out the following line. This is a very important step! Fail to do this and you WILL NOT BE ABLE TO JOIN A WINDOWS CLIENT TO THE DOMAIN!!!

Change:

```
invalid users = root
```

To:

```
;invalid users = root
```

Add the following line to the file (examples of the line should be there somewhere, I recommend sticking it there). This line disables roaming profiles for Windows.

```
logon path =
```

For reference here is a copy of my edited /etc/samba/smb.conf file for your viewing pleasure:

/etc/samba/smb.conf

```

#
# Sample configuration file for the Samba suite for Debian GNU/Linux.
#
#
# This is the main Samba configuration file. You should read the
# smb.conf(5) manual page in order to understand the options listed
# here. Samba has a huge number of configurable options most of which
# are not shown in this example
#
# Any line which starts with a ; (semi-colon) or a # (hash)
# is a comment and is ignored. In this example we will use a #
# for commentary and a ; for parts of the config file that you
# may wish to enable
#
# NOTE: Whenever you modify this file you should run the command
# "testparm" to check that you have not made any basic syntactic
# errors.
#

#===== Global Settings =====

[global]

## Browsing/Identification ###

# Change this to the workgroup/NT-domain name your Samba server will part of
#   workgroup = MSHOME
workgroup = EXAMPLE

# server string is the equivalent of the NT Description field
  server string = %h server (Samba, Ubuntu)

# Windows Internet Name Serving Support Section:
# WINS Support - Tells the NMBD component of Samba to enable its WINS Server
;   wins support = no

# WINS Server - Tells the NMBD components of Samba to be a WINS Client
# Note: Samba can be either a WINS Server, or a WINS Client, but NOT both
;   wins server = w.x.y.z

# This will prevent nmbd to search for NetBIOS names through DNS.
  dns proxy = no

# What naming service and in what order should we use to resolve host names
# to IP addresses
;   name resolve order = lmhosts host wins bcst

#### Networking ####

# The specific set of interfaces / networks to bind to
# This can be either the interface name or an IP address/netmask;
# interface names are normally preferred
;   interfaces = 127.0.0.0/8 eth0

# Only bind to the named interfaces and/or networks; you must use the
# 'interfaces' option above to use this.
# It is recommended that you enable this feature if your Samba machine is
# not protected by a firewall or is a firewall itself. However, this
# option cannot handle dynamic or non-broadcast interfaces correctly.
;   bind interfaces only = true

```

```

#### Debugging/Accounting ####

# This tells Samba to use a separate log file for each machine
# that connects
    log file = /var/log/samba/log.%m

# Put a capping on the size of the log files (in Kb).
    max log size = 1000

# If you want Samba to only log through syslog then set the following
# parameter to 'yes'.
;    syslog only = no

# We want Samba to log a minimum amount of information to syslog. Everything
# should go to /var/log/samba/log.{smbd,nmbd} instead. If you want to log
# through syslog you should set the following parameter to something higher.
    syslog = 0

# Do something sensible when Samba crashes: mail the admin a backtrace
    panic action = /usr/share/samba/panic-action %d

##### Authentication #####

# "security = user" is always a good idea. This will require a Unix account
# in this server for every user accessing the server. See
# /usr/share/doc/samba-doc/htmldocs/Samba3-HOWTO/ServerType.html
# in the samba-doc package for details.
;    security = user
security = user

# You may wish to use password encryption. See the section on
# 'encrypt passwords' in the smb.conf(5) manpage before enabling.
    encrypt passwords = true

# If you are using encrypted passwords, Samba will need to know what
# password database type you are using.
#    passdb backend = tdbsam
passdb backend = ldapsam:ldap://localhost/

#    obey pam restrictions = yes
obey pam restrictions = no

#####
#COPY AND PASTE THE FOLLOWING UNDERNEATH "OBEY PAM RESTRICTIONS = NO"
#####
#
#    Begin: Custom LDAP Entries
#
ldap admin dn = cn=admin,dc=example,dc=local
ldap suffix = dc=example, dc=local
ldap group suffix = ou=Groups
ldap user suffix = ou=Users
ldap machine suffix = ou=Computers
ldap idmap suffix = ou=Users
; Do ldap passwd sync
ldap passwd sync = Yes
passwd program = /usr/sbin/smbldap-passwd %u
passwd chat = *New*password* %n\n *Retype*new*password* %n\n
*all*authentication*tokens*updated*

```

```

add user script = /usr/sbin/smbldap-useradd -m "%u"
ldap delete dn = Yes
delete user script = /usr/sbin/smbldap-userdel "%u"
add machine script = /usr/sbin/smbldap-useradd -w "%u"
add group script = /usr/sbin/smbldap-groupadd -p "%g"
delete group script = /usr/sbin/smbldap-groupdel "%g"
add user to group script = /usr/sbin/smbldap-groupmod -m "%u" "%g"
delete user from group script = /usr/sbin/smbldap-groupmod -x "%u" "%g"
set primary group script = /usr/sbin/smbldap-usermod -g "%g" "%u"
domain logons = yes
#
#           End: Custom LDAP Entries
#
#####
#STOP COPYING HERE!
#####

;   guest account = nobody
;   invalid users = root

# This boolean parameter controls whether Samba attempts to sync the Unix
# password with the SMB password when the encrypted SMB password in the
# passdb is changed.
;   unix password sync = no

# For Unix password sync to work on a Debian GNU/Linux system, the following
# parameters must be set (thanks to Ian Kahan < for
# sending the correct chat script for the passwd program in Debian Sarge).
    passwd program = /usr/bin/passwd %u
    passwd chat = *Enter\snew\sUNIX\spassword:* %n\n
*Retype\snew\sUNIX\spassword:* %n\n *passwd:*password\supdated\ssuccessfully* .

# This boolean controls whether PAM will be used for password changes
# when requested by an SMB client instead of the program listed in
# 'passwd program'. The default is 'no'.
;   pam password change = no

##### Domains #####

# Is this machine able to authenticate users. Both PDC and BDC
# must have this setting enabled. If you are the BDC you must
# change the 'domain master' setting to no
#
;   domain logons = yes
#
# The following setting only takes effect if 'domain logons' is set
# It specifies the location of the user's profile directory
# from the client point of view)
# The following required a [profiles] share to be setup on the
# samba server (see below)
;   logon path = \\%N\profiles\%U
# Another common choice is storing the profile in the user's home directory
;   logon path = \\%N%\%U\profile
logon path =

# The following setting only takes effect if 'domain logons' is set
# It specifies the location of a user's home directory (from the client
# point of view)
;   logon drive = H:

```

```

; logon home = \\%N%\%U

# The following setting only takes effect if 'domain logons' is set
# It specifies the script to run during logon. The script must be stored
# in the [netlogon] share
# NOTE: Must be store in 'DOS' file format convention
; logon script = logon.cmd

# This allows Unix users to be created on the domain controller via the SAMR
# RPC pipe. The example command creates a user account with a disabled Unix
# password; please adapt to your needs
; add user script = /usr/sbin/adduser --quiet --disabled-password --gecos "" %u

##### Printing #####

# If you want to automatically load your printer list rather
# than setting them up individually then you'll need this
; load printers = yes

# lpr(ng) printing. You may wish to override the location of the
# printcap file
; printing = bsd
; printcap name = /etc/printcap

# CUPS printing. See also the cupsaddsmb(8) manpage in the
# cupsys-client package.
; printing = cups
; printcap name = cups

# When using [print$], root is implicitly a 'printer admin', but you can
# also give this right to other users to add drivers and set printer
# properties
; printer admin = @lpadmin

##### Misc #####

# Using the following line enables you to customise your configuration
# on a per machine basis. The %m gets replaced with the netbios name
# of the machine that is connecting
; include = /home/samba/etc/smb.conf.%m

# Most people will find that this option gives better performance.
# See smb.conf(5) and /usr/share/doc/samba-doc/htmldocs/Samba3-HOWTO/speed.html
# for details
# You may want to add the following on a Linux system:
#     SO_RCVBUF=8192 SO_SNDBUF=8192
#     socket options = TCP_NODELAY

# The following parameter is useful only if you have the linpopup package
# installed. The samba maintainer and the linpopup maintainer are
# working to ease installation and configuration of linpopup and samba.
; message command = /bin/sh -c '/usr/bin/linpopup "%f" "%m" %s; rm %s' &

# Domain Master specifies Samba to be the Domain Master Browser. If this
# machine will be configured as a BDC (a secondary logon server), you
# must set this to 'no'; otherwise, the default behavior is recommended.
; domain master = auto

# Some defaults for winbind (make sure you're not using the ranges
# for something else.)
; idmap uid = 10000-20000

```

```

; idmap gid = 10000-20000
; template shell = /bin/bash
;
; The following was the default behaviour in sarge
; but samba upstream reverted the default because it might induce
; performance issues in large organizations
; See #368251 for some of the consequences of *not* having
; this setting and smb.conf(5) for all details
;
; winbind enum groups = yes
; winbind enum users = yes

#===== Share Definitions =====

# Un-comment the following (and tweak the other settings below to suit)
# to enable the default home directory shares. This will share each
# user's home directory as \\server\username
[homes]
; comment = Home Directories
; browseable = no

# By default, \\server\username shares can be connected to by anyone
# with access to the samba server. Un-comment the following parameter
# to make sure that only "username" can connect to \\server\username
# This might need tweaking when using external authentication schemes
; valid users = %S

# By default, the home directories are exported read-only. Change next
# parameter to 'yes' if you want to be able to write to them.
; writable = no

# File creation mask is set to 0700 for security reasons. If you want to
# create files with group=rw permissions, set next parameter to 0775.
; create mask = 0700

# Directory creation mask is set to 0700 for security reasons. If you want to
# create dirs. with group=rw permissions, set next parameter to 0775.
; directory mask = 0700

# Un-comment the following and create the netlogon directory for Domain Logons
# (you need to configure Samba to act as a domain controller too.)
[netlogon]
; comment = Network Logon Service
; path = /home/samba/netlogon
; guest ok = yes
; writable = no
; share modes = no

# Un-comment the following and create the profiles directory to store
# users profiles (see the "logon path" option above)
# (you need to configure Samba to act as a domain controller too.)
# The path below should be writable by all users so that their
# profile directory may be created the first time they log on
[profiles]
; comment = Users profiles
; path = /home/samba/profiles
; guest ok = no
; browseable = no
; create mask = 0600
; directory mask = 0700

[printers]

```

```

comment = All Printers
browseable = no
path = /var/spool/samba
printable = yes
public = no
writable = no
create mode = 0700

# Windows clients look for this share name as a source of downloadable
# printer drivers
[print$]
comment = Printer Drivers
path = /var/lib/samba/printers
browseable = yes
read only = yes
guest ok = no
# Uncomment to allow remote administration of Windows print drivers.
# Replace 'ntadmin' with the name of the group your admin users are
# members of.
; write list = root, @ntadmin

# A sample share for sharing your CD-ROM with others.
;[cdrom]
; comment = Samba server's CD-ROM
; writable = no
; locking = no
; path = /cdrom
; public = yes

# The next two parameters show how to auto-mount a CD-ROM when the
# cdrom share is accessed. For this to work /etc/fstab must contain
# an entry like this:
# /dev/scd0 /cdrom iso9660 defaults,noauto,ro,user 0 0
# The CD-ROM gets unmounted automatically after the connection to the
# If you don't want to use auto-mounting/unmounting make sure the CD
# is mounted on /cdrom
# preexec = /bin/mount /cdrom
# postexec = /bin/umount /cdrom
Now we can restart the SAMBA service.

```

```
/etc/init.d/samba restart
```

Very important! We need to tell SAMBA what the "admin" password for the OpenLDAP server is. Hint: If you changed your "admin" password to be different from mine then you MUST replicate that change here! I guarantee you that someone will do this step and will have issues with SAMBA... this might be why!

```
smbpasswd -w 12345
```

Go ahead and reboot the server and make sure that everything still works correctly.

```
reboot
```

Top

Step 10: Configure the SMLDAP-TOOLS package

Top

The smbldap-tools package is one of the most important packages that we will be configuring today.

This is a collection of scripts that we will use to add users, groups, and computers to the LDAP directory. Of course this will require careful configuration. Many mistakes can be made here. I recommend doing everything that I do and then going back through another time to make your own customizations. If you are not careful here then you will run into issues. Good luck!

Open up the "examples" directory:

```
cd /usr/share/doc/smbldap-tools/examples/
```

Copy the configuration files to the correct directory and unzip them.:

```
cp smbldap_bind.conf /etc/smbldap-tools/  
cp smbldap.conf.gz /etc/smbldap-tools/  
gzip -d /etc/smbldap-tools/smbldap.conf.gz
```

Open up the smbldap-tools directory:

```
cd /etc/smbldap-tools/
```

Now you need to get the Security ID (SID) for your SAMBA domain. Write this string down (copy and paste it somewhere) because you will need it for the next step.

```
net getlocalsid
```

This results in (example): SID for domain DC01-UBUNTU is: S-1-5-21-949328747-3404738746-3052206637

Open up the file /etc/smbldap-tools/smbldap.conf for editing:

```
vim smbldap.conf
```

Alright, now we need to edit the file. You can't just copy and paste here, you need to edit the specific lines according to your individual setup. I will include my file for reference as well:

```
SID="S-1-5-21-949328747-3404738746-3052206637" ## This line must have the same  
SID as when you ran "net getlocalsid"
```

```
sambaDomain="EXAMPLE"
```

```
ldapTLS="0"
```

```
suffix="dc=example,dc=local"
```

```
sambaUnixIdPooldn="sambaDomainName=EXAMPLE,${suffix}" ## Be careful with this  
section!!
```

```
userHome="/ldaphome/%U" ## This is found in the UNIX section.
```

```
userSmbHome=
```

```
userProfile=
```

```
userHomeDrive=
```

```
userScript=
```

```
mailDomain="example.local"
```

```
/etc/smbldap-tools/smbldap.conf
```

```
# $Source: /opt/cvs/samba/smbldap-tools/smbldap.conf,v $
```

```
# $Id: smbldap.conf,v 1.18 2005/05/27 14:28:47 jtournier Exp $
```

```
#
```

```
# smbldap-tools.conf : Q & D configuration file for smbldap-tools
```

```
# This code was developed by IDEALX (http://IDEALX.org/) and  
# contributors (their names can be found in the CONTRIBUTORS file).
```

```
#
```

```
# Copyright (C) 2001-2002 IDEALX
```

```
#
```

```
# This program is free software; you can redistribute it and/or  
# modify it under the terms of the GNU General Public License  
# as published by the Free Software Foundation; either version 2  
# of the License, or (at your option) any later version.
```

```
#
```

```
# This program is distributed in the hope that it will be useful,  
# but WITHOUT ANY WARRANTY; without even the implied warranty of  
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the  
# GNU General Public License for more details.
```

```

#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307,
# USA.

# Purpose :
#         . be the configuration file for all smbldap-tools scripts

#####
#
# General Configuration
#
#####

# Put your own SID. To obtain this number do: "net getlocalsid".
# If not defined, parameter is taking from "net getlocalsid" return
#SID="S-1-5-21-4205727931-4131263253-1851132061"
SID="S-1-5-21-4052000378-234799737-4288018487"

# Domain name the Samba server is in charged.
# If not defined, parameter is taking from smb.conf configuration file
# Ex: sambaDomain="IDEALX-NT"
#sambaDomain="IDEALX-NT"
sambaDomain="EXAMPLE"

#####
#
# LDAP Configuration
#
#####

# Notes: to use to dual ldap servers backend for Samba, you must patch
# Samba with the dual-head patch from IDEALX. If not using this patch
# just use the same server for slaveLDAP and masterLDAP.
# Those two servers declarations can also be used when you have
# . one master LDAP server where all writing operations must be done
# . one slave LDAP server where all reading operations must be done
#   (typically a replication directory)

# Slave LDAP server
# Ex: slaveLDAP=127.0.0.1
# If not defined, parameter is set to "127.0.0.1"
slaveLDAP="127.0.0.1"

# Slave LDAP port
# If not defined, parameter is set to "389"
slavePort="389"

# Master LDAP server: needed for write operations
# Ex: masterLDAP=127.0.0.1
# If not defined, parameter is set to "127.0.0.1"
masterLDAP="127.0.0.1"

# Master LDAP port
# If not defined, parameter is set to "389"
masterPort="389"

# Use TLS for LDAP
# If set to 1, this option will use start_tls for connection
# (you should also used the port 389)
# If not defined, parameter is set to "1"

```

```

#ldapTLS="1"
ldapTLS="0"

# How to verify the server's certificate (none, optional or require)
# see "man Net::LDAP" in start_tls section for more details
verify="require"

# CA certificate
# see "man Net::LDAP" in start_tls section for more details
cafile="/etc/opt/IDEALX/smbldap-tools/ca.pem"

# certificate to use to connect to the ldap server
# see "man Net::LDAP" in start_tls section for more details
clientcert="/etc/opt/IDEALX/smbldap-tools/smbldap-tools.pem"

# key certificate to use to connect to the ldap server
# see "man Net::LDAP" in start_tls section for more details
clientkey="/etc/opt/IDEALX/smbldap-tools/smbldap-tools.key"

# LDAP Suffix
# Ex: suffix=dc=IDEALX,dc=ORG
#suffix="dc=idealx,dc=org"
suffix="dc=example,dc=local"

# Where are stored Users
# Ex: usersdn="ou=Users,dc=IDEALX,dc=ORG"
# Warning: if 'suffix' is not set here, you must set the full dn for usersdn
usersdn="ou=Users,${suffix}"

# Where are stored Computers
# Ex: computersdn="ou=Computers,dc=IDEALX,dc=ORG"
# Warning: if 'suffix' is not set here, you must set the full dn for computersdn
computersdn="ou=Computers,${suffix}"

# Where are stored Groups
# Ex: groupsdn="ou=Groups,dc=IDEALX,dc=ORG"
# Warning: if 'suffix' is not set here, you must set the full dn for groupsdn
groupsdn="ou=Groups,${suffix}"

# Where are stored Idmap entries (used if samba is a domain member server)
# Ex: groupsdn="ou=Idmap,dc=IDEALX,dc=ORG"
# Warning: if 'suffix' is not set here, you must set the full dn for idmapdn
idmapdn="ou=Idmap,${suffix}"

# Where to store next uidNumber and gidNumber available for new users and groups
# If not defined, entries are stored in sambaDomainName object.
# Ex: sambaUnixIdPooldn="sambaDomainName=${sambaDomain},${suffix}"
# Ex: sambaUnixIdPooldn="cn=NextFreeUnixId,${suffix}"
#sambaUnixIdPooldn="sambaDomainName=IDEALX-NT,${suffix}"
sambaUnixIdPooldn="sambaDomainName=EXAMPLE,${suffix}" ## Be careful with this
section!!

# Default scope Used
scope="sub"

# Unix password encryption (CRYPT, MD5, SMD5, SSHA, SHA, CLEARTXT)
hash_encrypt="SSHA"

# if hash_encrypt is set to CRYPT, you may set a salt format.
# default is "%s", but many systems will generate MD5 hashed
# passwords if you use "$1$.8s". This parameter is optional!
crypt_salt_format="%s"

```

```

#####
#
# Unix Accounts Configuration
#
#####

# Login defs
# Default Login Shell
# Ex: userLoginShell="/bin/bash"
userLoginShell="/bin/bash"

# Home directory
# Ex: userHome="/home/%U"
#userHome="/home/%U"
userHome="/ldaphome/%U" ## This is found in the UNIX section.

# Default mode used for user homeDirectory
userHomeDirectoryMode="700"

# Gecos
userGecos="System User"

# Default User (POSIX and Samba) GID
defaultUserGid="513"

# Default Computer (Samba) GID
defaultComputerGid="515"

# Skel dir
skeletonDir="/etc/skel"

# Default password validation time (time in days) Comment the next line if
# you don't want password to be enable for defaultMaxPasswordAge days (be
# careful to the sambaPwdMustChange attribute's value)
defaultMaxPasswordAge="45"

#####
#
# SAMBA Configuration
#
#####

# The UNC path to home drives location (%U username substitution)
# Just set it to a null string if you want to use the smb.conf 'logon home'
# directive and/or disable roaming profiles
# Ex: userSmbHome="//PDC-SMB3/%U"
#userSmbHome="//PDC-SRV/%U"
userSmbHome=

# The UNC path to profiles locations (%U username substitution)
# Just set it to a null string if you want to use the smb.conf 'logon path'
# directive and/or disable roaming profiles
# Ex: userProfile="//PDC-SMB3/profiles/%U"
#userProfile="//PDC-SRV/profiles/%U"
userProfile=

# The default Home Drive Letter mapping
# (will be automatically mapped at logon time if home directory exist)
# Ex: userHomeDrive="H:"
#userHomeDrive="H:"
userHomeDrive=

```

```
# The default user netlogon script name (%U username substitution)
# if not used, will be automatically username.cmd
# make sure script file is edited under dos
# Ex: userScript="startup.cmd" # make sure script file is edited under dos
#userScript="logon.bat"
userScript=
```

```
# Domain appended to the users "mail"-attribute
# when smbldap-useradd -M is used
# Ex: mailDomain="idealx.com"
#mailDomain="idealx.com"
mailDomain="example.local"
```

```
#####
#
# SMBLDAP-TOOLS Configuration (default are ok for a RedHat)
#
#####
```

```
# Allows not to use smbpasswd (if with_smbpasswd == 0 in smbldap_conf.pm) but
# prefer Crypt::SmbHash library
with_smbpasswd="0"
smbpasswd="/usr/bin/smbpasswd"
```

```
# Allows not to use slappasswd (if with_slappasswd == 0 in smbldap_conf.pm)
# but prefer Crypt:: libraries
with_slappasswd="0"
slappasswd="/usr/sbin/slappasswd"
```

```
# comment out the following line to get rid of the default banner
# no_banner="1"
```

Open the file /etc/smbldap-tools/smbldap_bind.conf file for editing:

```
vim smbldap_bind.conf
```

Edit the file so the following is correct according to your setup. I will also include a copy of my file for reference.

```
slaveDN="cn=admin,dc=example,dc=local"
slavePw="12345"
masterDN="cn=admin,dc=example,dc=local"
masterPw="12345"
```

```
/etc/smbldap-tools/smbldap_bind.conf
```

```
#####
# Credential Configuration #
#####
```

```
# Notes: you can specify two different configuration if you use a
# master ldap for writing access and a slave ldap server for reading access
# By default, we will use the same DN (so it will work for standard Samba
# release)
```

```
#slaveDN="cn=Manager,dc=idealx,dc=org"
#slavePw="secret"
#masterDN="cn=Manager,dc=idealx,dc=org"
#masterPw="secret"
```

```
slaveDN="cn=admin,dc=example,dc=local"
slavePw="12345"
masterDN="cn=admin,dc=example,dc=local"
masterPw="12345"
```

Set the correct permissions on the above files:

```
chmod 0644 /etc/smbldap-tools/smbldap.conf
chmod 0600 /etc/smbldap-tools/smbldap_bind.conf
Top
```

Step 11: Populate LDAP using smbldap-tools

Top

This is another simple step but it is very important. When doing this step if you encounter errors then it is most likely because you failed the previous step. Just a hint.

Run the command to populate the directory:

```
smbldap-populate -u 30000 -g 30000
```

When doing so it will prompt you to assign a password to the user "root" - remember to use the password that you've been using to keep things simple.

```
12345
```

Verify that you have several new entries in your LDAP directory by running the command:

```
ldapsearch -x -b dc=example,dc=local | less
```

Awesome, now we have some default entries in our LDAP directory. This is a good thing!

Top

Step 12: Add an LDAP User to the System

Top

Run the following command to add a new user to the LDAP. Please note that you should edit this user information to suit your needs. This will add a standard user, not an administrative user.

```
smbldap-useradd -a -m -M ricky -c "Richard M" ricky
```

Here is an explanation of the above command switches:

- a allows Windows as well as Linux login
- m makes a home directory, leave this off if you do not need local access. PAM will be configured to automatically create a home directory.
- M sets up the username part of their email address
- c specifies their full name

Now we need to set the password for this new account:

```
smbldap-passwd ricky
# I will be using "12345" for the password.
```

Now that we have a user in our LDAP directory we will need to configure the system to authenticate via LDAP.

Top

Step 13: Configure LDAP Authentication on the Server

Top

The basic steps for this section came from the Ubuntu Forums (<http://ubuntuforums.org/showthread.php?t=597056>). Thanks to all who contributed to that thread! Basically we need to tell our server to use LDAP authentication as one of its options. Be careful with this! It can cause your server to break! This is why we always have a backup around.

Install the necessary software used to accomplish this feat:

apt-get install auth-client-config libpam-ldap libnss-ldap
You will be prompted to answer some questions. Use the following answers (or your own if you changed things before!):

```
Should debconf manage LDAP configuration?: Yes
LDAP server Uniform Resource Identifier: ldapi://127.0.0.1
Distinguished name of the search base: dc=example,dc=local
LDAP version to use: 3
Make local root Database admin: Yes
Does the LDAP database require login? No
LDAP account for root: cn=admin,dc=example,dc=local
LDAP root account password: 12345
Create a backup of the file /etc/ldap.conf:
```

```
cp /etc/ldap.conf /etc/ldap.conf.original
Open the file /etc/ldap.conf for editing in your favorite editor:
```

```
vim /etc/ldap.conf
```

Please note that you cannot just copy and paste the following into your file. Find the referenced lines and modify them so that they are correct. I will include a copy of my file for reference.

```
host 127.0.0.1
base dc=example,dc=local
uri ldap://127.0.0.1/
rootbinddn cn=admin,dc=example,dc=local
bind_policy soft
/etc/ldap.conf
```

```
###DEBCONF###
```

```
##
```

```
## Configuration of this file will be managed by debconf as long as the
## first line of the file says '###DEBCONF###'
```

```
##
```

```
## You should use dpkg-reconfigure to configure this file via debconf
```

```
##
```

```
#
```

```
# @(#) $Id: ldap.conf,v 1.38 2006/05/15 08:13:31 lukeh Exp $
```

```
#
```

```
# This is the configuration file for the LDAP nameservice
# switch library and the LDAP PAM module.
```

```
#
```

```
# PADL Software
```

```
# http://www.padl.com
```

```
#
```

```
# Your LDAP server. Must be resolvable without using LDAP.
# Multiple hosts may be specified, each separated by a
# space. How long nss_ldap takes to failover depends on
# whether your LDAP client library supports configurable
# network or connect timeouts (see bind_timelimit).
host 127.0.0.1
```

```
# The distinguished name of the search base.
```

```
#base dc=padl,dc=com
```

```
base dc=example,dc=local
```

```
# Another way to specify your LDAP server is to provide an
# uri with the server name. This allows to use
# Unix Domain Sockets to connect to a local LDAP Server.
```

```
uri ldap://127.0.0.1/
```

```
#uri ldaps://127.0.0.1/
```

```
#uri ldapi://%2fvar%2frun%2fldapi_sock/
```

```
# Note: %2f encodes the '/' used as directory separator

# The LDAP version to use (defaults to 3
# if supported by client library)
ldap_version 3

# The distinguished name to bind to the server with.
# Optional: default is to bind anonymously.
#binddn cn=proxyuser,dc=padl,dc=com

# The credentials to bind with.
# Optional: default is no credential.
#bindpw secret

# The distinguished name to bind to the server with
# if the effective user ID is root. Password is
# stored in /etc/ldap.secret (mode 600)
rootbinddn cn=admin,dc=example,dc=local

# The port.
# Optional: default is 389.
#port 389

# The search scope.
#scope sub
#scope one
#scope base

# Search timelimit
#timelimit 30

# Bind/connect timelimit
#bind_timelimit 30

# Reconnect policy: hard (default) will retry connecting to
# the software with exponential backoff, soft will fail
# immediately.
#bind_policy hard
bind_policy soft

# Idle timelimit; client will close connections
# (nss_ldap only) if the server has not been contacted
# for the number of seconds specified below.
#idle_timelimit 3600

# Filter to AND with uid=%s
#pam_filter objectclass=account

# The user ID attribute (defaults to uid)
#pam_login_attribute uid

# Search the root DSE for the password policy (works
# with Netscape Directory Server)
#pam_lookup_policy yes

# Check the 'host' attribute for access control
# Default is no; if set to yes, and user has no
# value for the host attribute, and pam_ldap is
# configured for account management (authorization)
# then the user will not be allowed to login.
#pam_check_host_attr yes
```

```
# Check the 'authorizedService' attribute for access
# control
# Default is no; if set to yes, and the user has no
# value for the authorizedService attribute, and
# pam_ldap is configured for account management
# (authorization) then the user will not be allowed
# to login.
#pam_check_service_attr yes

# Group to enforce membership of
#pam_groupdn cn=PAM,ou=Groups,dc=padl,dc=com

# Group member attribute
#pam_member_attribute uniquemember

# Specify a minimum or maximum UID number allowed
#pam_min_uid 0
#pam_max_uid 0

# Template login attribute, default template user
# (can be overridden by value of former attribute
# in user's entry)
#pam_login_attribute userPrincipalName
#pam_template_login_attribute uid
#pam_template_login nobody

# HEADS UP: the pam_crypt, pam_nds_passwd,
# and pam_ad_passwd options are no
# longer supported.
#
# Do not hash the password at all; presume
# the directory server will do it, if
# necessary. This is the default.
pam_password md5

# Hash password locally; required for University of
# Michigan LDAP server, and works with Netscape
# Directory Server if you're using the UNIX-Crypt
# hash mechanism and not using the NT Synchronization
# service.
#pam_password crypt

# Remove old password first, then update in
# cleartext. Necessary for use with Novell
# Directory Services (NDS)
#pam_password clear_remove_old
#pam_password nds

# RACF is an alias for the above. For use with
# IBM RACF
#pam_password racf

# Update Active Directory password, by
# creating Unicode password and updating
# unicodePwd attribute.
#pam_password ad

# Use the OpenLDAP password change
# extended operation to update the password.
#pam_password exop

# Redirect users to a URL or somesuch on password
```

```

# changes.
#pam_password_prohibit_message Please visit http://internal to change your
password.

# RFC2307bis naming contexts
# Syntax:
# nss_base_XXX          base?scope?filter
# where scope is {base,one,sub}
# and filter is a filter to be &'d with the
# default filter.
# You can omit the suffix eg:
# nss_base_passwd      ou=People,
# to append the default base DN but this
# may incur a small performance impact.
#nss_base_passwd      ou=People,dc=padl,dc=com?one
#nss_base_shadow      ou=People,dc=padl,dc=com?one
#nss_base_group       ou=Group,dc=padl,dc=com?one
#nss_base_hosts       ou=Hosts,dc=padl,dc=com?one
#nss_base_services   ou=Services,dc=padl,dc=com?one
#nss_base_networks   ou=Networks,dc=padl,dc=com?one
#nss_base_protocols  ou=Protocols,dc=padl,dc=com?one
#nss_base_rpc        ou=Rpc,dc=padl,dc=com?one
#nss_base_ethers     ou=Ethers,dc=padl,dc=com?one
#nss_base_netmasks   ou=Networks,dc=padl,dc=com?ne
#nss_base_bootparams ou=Ethers,dc=padl,dc=com?one
#nss_base_aliases    ou=Aliases,dc=padl,dc=com?one
#nss_base_netgroup   ou=Netgroup,dc=padl,dc=com?one

# attribute/objectclass mapping
# Syntax:
#nss_map_attribute    rfc2307attribute      mapped_attribute
#nss_map_objectclass  rfc2307objectclass    mapped_objectclass

# configure --enable-nds is no longer supported.
# NDS mappings
#nss_map_attribute uniqueMember member

# Services for UNIX 3.5 mappings
#nss_map_objectclass posixAccount User
#nss_map_objectclass shadowAccount User
#nss_map_attribute uid msSFU30Name
#nss_map_attribute uniqueMember msSFU30PosixMember
#nss_map_attribute userPassword msSFU30Password
#nss_map_attribute homeDirectory msSFU30HomeDirectory
#nss_map_attribute homeDirectory msSFUHomeDirectory
#nss_map_objectclass posixGroup Group
#pam_login_attribute msSFU30Name
#pam_filter objectclass=User
#pam_password ad

# configure --enable-mssfuf-schema is no longer supported.
# Services for UNIX 2.0 mappings
#nss_map_objectclass posixAccount User
#nss_map_objectclass shadowAccount user
#nss_map_attribute uid msSFUName
#nss_map_attribute uniqueMember posixMember
#nss_map_attribute userPassword msSFUPassword
#nss_map_attribute homeDirectory msSFUHomeDirectory
#nss_map_attribute shadowLastChange pwdLastSet
#nss_map_objectclass posixGroup Group
#nss_map_attribute cn msSFUName
#pam_login_attribute msSFUName

```

```
#pam_filter objectclass=User
#pam_password ad

# RFC 2307 (AD) mappings
#nss_map_objectclass posixAccount user
#nss_map_objectclass shadowAccount user
#nss_map_attribute uid sAMAccountName
#nss_map_attribute homeDirectory unixHomeDirectory
#nss_map_attribute shadowLastChange pwdLastSet
#nss_map_objectclass posixGroup group
#nss_map_attribute uniqueMember member
#pam_login_attribute sAMAccountName
#pam_filter objectclass=User
#pam_password ad

# configure --enable-authpassword is no longer supported
# AuthPassword mappings
#nss_map_attribute userPassword authPassword

# AIX SecureWay mappings
#nss_map_objectclass posixAccount aixAccount
#nss_base_passwd ou=aixaccount,?one
#nss_map_attribute uid userName
#nss_map_attribute gidNumber gid
#nss_map_attribute uidNumber uid
#nss_map_attribute userPassword passwordChar
#nss_map_objectclass posixGroup aixAccessGroup
#nss_base_group ou=aixgroup,?one
#nss_map_attribute cn groupName
#nss_map_attribute uniqueMember member
#pam_login_attribute userName
#pam_filter objectclass=aixAccount
#pam_password clear

# Netscape SDK LDAPS
#ssl on

# Netscape SDK SSL options
#sslpath /etc/ssl/certs

# OpenLDAP SSL mechanism
# start_tls mechanism uses the normal LDAP port, LDAPS typically 636
#ssl start_tls
#ssl on

# OpenLDAP SSL options
# Require and verify server certificate (yes/no)
# Default is to use libldap's default behavior, which can be configured in
# /etc/openldap/ldap.conf using the TLS_REQCERT setting. The default for
# OpenLDAP 2.0 and earlier is "no", for 2.1 and later is "yes".
#tls_checkpeer yes

# CA certificates for server certificate verification
# At least one of these are required if tls_checkpeer is "yes"
#tls_cacertfile /etc/ssl/ca.cert
#tls_cacertdir /etc/ssl/certs

# Seed the PRNG if /dev/urandom is not provided
#tls_randfile /var/run/egd-pool

# SSL cipher suite
# See man ciphers for syntax
```

```

#tls_ciphers TLSv1

# Client certificate and key
# Use these, if your server requires client authentication.
#tls_cert
#tls_key

# Disable SASL security layers. This is needed for AD.
#sasl_secprops maxssf=0

# Override the default Kerberos ticket cache location.
#krb5_ccname FILE:/etc/.ldapcache

# SASL mechanism for PAM authentication - use is experimental
# at present and does not support password policy control
#pam_sasl_mech DIGEST-MD5
Now we need to copy the file /etc/ldap.conf to the file /etc/ldap/ldap.conf. First we will backup the
file (/etc/ldap/ldap.conf) and then we will copy the new file.

cp /etc/ldap/ldap.conf /etc/ldap/ldap.conf.original
cp /etc/ldap.conf /etc/ldap/ldap.conf
OK, create a new file by running the following command. You will need to edit the first part of the
command to use your favorite editor.

vim /etc/auth-client-config/profile.d/open_ldap
This file is the new OpenLDAP authentication profile. Copy and paste EXACTLY the following
lines:

[open_ldap]
nss_passwd=passwd: compat ldap
nss_group=group: compat ldap
nss_shadow=shadow: compat ldap
pam_auth=auth          required          pam_env.so
auth          sufficient  pam_unix.so likeauth nullok
auth          sufficient  pam_ldap.so use_first_pass
auth          required    pam_deny.so
pam_account=account   sufficient    pam_unix.so
account       sufficient  pam_ldap.so
account       required    pam_deny.so
pam_password=password sufficient    pam_unix.so nullok md5 shadow use_authtok
password      sufficient  pam_ldap.so use_first_pass
password      required    pam_deny.so
pam_session=session  required     pam_limits.so
session       required    pam_mkhome.so skel=/etc/skel/ umask=0077
session       required    pam_unix.so
session       optional    pam_ldap.so

```

Backup the /etc/nsswitch.conf file:

```

cp /etc/nsswitch.conf /etc/nsswitch.conf.original

```

Backup the files in /etc/pam.d:

```

cd /etc/pam.d/
mkdir bkup
cp * bkup/

```

Enable the new OpenLDAP profile by running the following command. If you did all the previous steps correctly then this will run without issue.

```

auth-client-config -a -p open_ldap

```

The final step is to simply reboot the server. When the server is running again then test to see if you can log in with your new LDAP user. No matter what you should be able to log in with a local user (unless the system is hung). If the system hangs then reboot HARD and try again.

reboot
Top

Step 14: Install the BIND DNS Server

Top

We will be using the BIND DNS server because it is the only DNS server that I know how to configure. We will be using WebMIN to configure it (Webmin will be installed later and we will configure BIND in a later step). Why do we need a DNS server? Well, DNS makes it easier to manage the hosts on the network. LDAP works great when you can use DNS. DNS must be there in order for a Windows client to join the domain.

Install the software:

```
apt-get install bind9
```

Top

Step 15: Install and Configure NFS Server Support

Top

By this point LDAP authentication is working without issue and LDAP user home folders are located in /ldaphome. If this is not correct then you will want to go back through and fix things.

Now we will be installing and configuring our NFS server. Thanks to everyone in the thread <http://ubuntuforums.org/showthread.php?t=249889> for the help with this section.

First install the software:

```
apt-get install nfs-kernel-server nfs-common portmap
```

Now we need to reconfigure portmap.

```
dpkg-reconfigure portmap
```

Answer as follows to the prompt:

```
no
```

Restart portmap:

```
/etc/init.d/portmap restart
```

Open up the /etc/exports file for editing. This is where we define our NFS shares (or exports).

```
vim /etc/exports
```

Add the following line to the file. What this line does is allow unrestricted access to the /ldaphome share from any computer. I will also include a copy of my file for reference.

```
/ldaphome *(rw,async)
/etc/exports
```

```
# /etc/exports: the access control list for filesystems which may be exported
#                to NFS clients.  See exports(5).
#
```

```
# Example for NFSv2 and NFSv3:
```

```
# /srv/homes      hostname1(rw,sync) hostname2(ro,sync)
```

```
#
```

```
# Example for NFSv4:
```

```
# /srv/nfs4      gss/krb5i(rw,sync,fsid=0,crossmnt)
```

```
# /srv/nfs4/homes gss/krb5i(rw,sync)
```

```
#
```

```
/ldaphome *(rw,async)
```

Restart the NFS service.

```
/etc/init.d/nfs-kernel-server restart
```

Now we have NFS enabled and configured. If you have a client up and running at the moment you can give it a test. Otherwise just continue with this guide.

Top

Step 16: Install Webmin

Top

Webmin is a very useful program. We can use it to control installed services, monitor the system, and help ease administration.

Download the package from the Webmin website:

```
wget http://superb-east.dl.sourceforge.net/sourceforge/webadmin/webmin_1.400_all.deb
```

We need to install some required packages first.

```
apt-get install openssl libauthen-pam-perl libio-pty-perl libmd5-perl libnet-ssleay-perl
```

Now we can install Webmin:

```
dpkg -i webmin_1.400_all.deb
```

You should see a message similar to the following when it successfully installs:

```
"Webmin install complete. You can now login to https://dc01-ubuntu.example.local:10000/ as root with your root password, or as any user who can use sudo to run commands as root."
```

The Webmin installation is now complete.

Top

Step 17: Configure BIND9 and the Primary DNS Zone

Top

We now want to create our DNS zone so that we are in charge of it and can make use of it. I prefer using a GUI to do this as opposed to editing the zone files.

In a web browser navigate to: <https://192.168.0.60:10000> (Please use the IP address that YOU assigned to your server.)

Login as "sysadmin" and "12345"

Servers > BIND DNS Server

Under "Existing DNS Zones" click "Create master zone"

Enter in the following information (customize to your needs!):

```
Zone type: Forward (Names to Addresses)
Domain name / Network: example.local
Records file: Automatic
Master server: dc01-ubuntu.example.local
Email address: sysadmin@example.local
Click "Create" button
```

Click "Apply Changes" button

Click "Address (0)" at the top

Fill in with this information (customize to your needs!):

Name: dc01-ubuntu

Address: 192.168.0.60

Click "Create" button

Click "Return to record types"

Click "Apply Changes" button.

Top

Step 18: Configure the Server to use Itself for DNS

Top

DNS doesn't do a whole lot of good if we don't use it. In this section we point our `/etc/resolv.conf` file to ourselves. I also recommend leaving in a known working DNS server as the secondary source just in case something screws up. In some of my trials I did notice that the server would hang trying to start BIND9.

Backup the `/etc/resolv.conf` file before editing it!

```
cp /etc/resolv.conf /etc/resolv.conf.original
```

Open the `/etc/resolv.conf` file for editing:

```
vim /etc/resolv.conf
```

Edit the file so that the only lines in the file are the following. I will also include a copy of my file for reference.

```
search example.local
```

```
nameserver 192.168.0.60
```

Reboot the server and then test DNS to ensure everything is working the way it should be.

```
reboot
```

Top

Some notes and conclusions

Top

You should now have a fully functional SAMBA domain controller. All you need to do now is add a workstation account, join machines to the network, and voila, DOMAIN! The next few sections go through some other items of interest (Windows logon script, configuring a Linux client, configuring a Windows client, etc...)

Top

Install and Configure Apache2 + PHPLDAPAdmin

Top

Apache is a nice server to have installed. By having it installed you'll be able to host your own websites, etc... PHPLDAPAdmin is a very nice LDAP management tool. So far the best use that I have gotten from it is the ability to view my LDAP directory. This way I can confirm that items that should be there really are there.

Install the software:

```
apt-get install apache2 phpldapadmin
```

Open the file `/etc/apache2/httpd.conf` for editing:

```
vim /etc/apache2/httpd.conf
```

Add the following line to the very top of the file. It will stop an annoying message when Apache starts up. Please customize this according to your configuration.

```
ServerName dc01-ubuntu.example.local
```

Restart Apache:

```
/etc/init.d/apache2 restart
```

Copy the PHPLDAPAdmin folder into the /var/www/ directory. This way we can access PHPLDAPAdmin more easily.

```
cp -R /usr/share/phpldapadmin/ /var/www/phpldapadmin
```

Access PHPLDAPAdmin my going to: <http://192.168.0.60/phpldapadmin/>. The username is "cn=admin,dc=example,dc=local" - customize that if you changed the LDAP domain properties.

Top

Configure Ubuntu Server 7.10 (client) to Mount NFS Shares

Top

In order for our whole system to work the correct way we need to have access to the user files stored on the server. For Linux clients we will be using NFS to accomplish this. One thing to note is that this section assumes that your client has Linux installed, that it can resolve DNS entries against your server, and that the client works on it's own.

Install NFS support:

```
apt-get install portmap nfs-common
```

Restart the associated services:

```
/etc/init.d/portmap restart
```

```
/etc/init.d/nfs-common restart
```

Create the /ldaphome directory:

```
cd /
```

```
mkdir ldaphome
```

Try to manually mount the ldaphome NFS share:

```
mount dc01-ubuntu.example.local:/ldaphome /ldaphome
```

Now go ahead and add the necessary entries into /etc/fstab so that the directory is mounted at boot.

I'm also including a copy of my file for reference.

```
vim /etc/fstab
```

Add the following lines to the bottom of the file:

```
# Custom NFS mount for home directories.
```

```
dc01-ubuntu.example.local:/ldaphome /ldaphome nfs
```

```
rsize=8192,wsiz=8192,timeo=14,intr
```

```
/etc/fstab
```

```
# /etc/fstab: static file system information.
```

```
#
```

```
#
```

```
proc /proc proc defaults 0 0
```

```
# /dev/sda1
```

```
UUID=fd12bae1-adda-4b61-9ce9-ed4e9a1f52aa / ext3
```

```
defaults,errors=remount-ro 0 1
```

```
# /dev/sda5
```

```
UUID=86661b5c-c34f-9fad-c85d-ccbc61e5fb0d none swap sw
```

```
0 0
```

```
/dev/scd0 /media/cdrom0 udf,iso9660 user,noauto,exec 0 0
```

```
/dev/fd0          /media/floppy0  auto    rw,user,noauto,exec 0      0
```

```
# Custom NFS mount for home directories.  
dc01-ubuntu.example.local:/ldaphome /ldaphome nfs  
rsize=8192,wsize=8192,timeo=14,intr  
Reboot the client to ensure that everything is working.
```

reboot

Top

Configure Ubuntu Server 7.10 (client) for LDAP Authentication

Top

Now that you have this server it only makes sense to also have an LDAP client, right? Well, here we go. I'm going to shorten this section and only give you the relevant parts. I'm assuming that since you made it through the initial guide you are pretty confident in your ability to install Ubuntu and configure the basics.

Assumptions/Requirements:

- Your hostname and host file need to be configured correctly. Your hostname should be "client-linux.example.local" - I'm going to assume that you are in the domain "example.local" and that your hostname is "client-linux" - Please customize this to your own scenario. Your hosts file needs to have your FQDN in it otherwise you may run into issue.
- You have your /etc/resolv.conf file configured so that it is looking at your server for DNS and that it is searching your domain. For my setup I used the same /etc/resolv.conf as I did for the server.
- You can PING the server by name and by IP.
- You installed and configured NTP for time synchronization. This is important in a domain environment!
- Because of the nature of our home directories you MUST have NFS set up and configured on the client FIRST. The previous section describes how to do this.

OK, now we can begin.

Install the software:

```
apt-get install auth-client-config libpam-ldap libnss-ldap
```

Answer the questions with the following (customize if you need to):

```
Should debconf manage LDAP configuration?: Yes  
LDAP server Uniform Resource Identifier: ldapi://dc01-ubuntu.example.local  
Distinguished name of the search base: dc=example,dc=local  
LDAP version to use: 3  
Make local root Database admin: Yes  
Does the LDAP database require login? No  
LDAP account for root: cn=admin,dc=example,dc=local  
LDAP root account password: 12345  
Create a backup of the file /etc/ldap.conf:
```

```
cp /etc/ldap.conf /etc/ldap.conf.original
```

Open the file /etc/ldap.conf for editing in your favorite editor:

```
vim /etc/ldap.conf
```

Please note that you cannot just copy and paste the following into your file. Find the referenced lines and modify them so that they are correct. I will include a copy of my file for reference.

```
host dc01-ubuntu.example.local
base dc=example,dc=local
uri ldap://dc01-ubuntu.example.local/
rootbinddn cn=admin,dc=example,dc=local
bind_policy soft
/etc/ldap.conf

###DEBCONF###
##
## Configuration of this file will be managed by debconf as long as the
## first line of the file says '###DEBCONF###'
##
## You should use dpkg-reconfigure to configure this file via debconf
##

#
# @(#) $Id: ldap.conf,v 1.38 2006/05/15 08:13:31 lukeh Exp $
#
# This is the configuration file for the LDAP nameservice
# switch library and the LDAP PAM module.
#
# PADL Software
# http://www.padl.com
#

# Your LDAP server. Must be resolvable without using LDAP.
# Multiple hosts may be specified, each separated by a
# space. How long nss_ldap takes to failover depends on
# whether your LDAP client library supports configurable
# network or connect timeouts (see bind_timelimit).
#host 127.0.0.1
host dc01-ubuntu.example.local

# The distinguished name of the search base.
#base dc=padl,dc=com
base dc=example,dc=local

# Another way to specify your LDAP server is to provide an
# uri with the server name. This allows to use
# Unix Domain Sockets to connect to a local LDAP Server.
uri ldap://dc01-ubuntu.example.local/
#uri ldaps://127.0.0.1/
#uri ldapi://%2fvar%2frun%2fldapi_sock/
# Note: %2f encodes the '/' used as directory separator

# The LDAP version to use (defaults to 3
# if supported by client library)
ldap_version 3

# The distinguished name to bind to the server with.
# Optional: default is to bind anonymously.
#binddn cn=proxyuser,dc=padl,dc=com

# The credentials to bind with.
# Optional: default is no credential.
#bindpw secret

# The distinguished name to bind to the server with
# if the effective user ID is root. Password is
```

```
# stored in /etc/ldap.secret (mode 600)
rootbinddn cn=admin,dc=example,dc=local

# The port.
# Optional: default is 389.
#port 389

# The search scope.
#scope sub
#scope one
#scope base

# Search timelimit
#timelimit 30

# Bind/connect timelimit
#bind_timelimit 30

# Reconnect policy: hard (default) will retry connecting to
# the software with exponential backoff, soft will fail
# immediately.
#bind_policy hard
bind_policy soft

# Idle timelimit; client will close connections
# (nss_ldap only) if the server has not been contacted
# for the number of seconds specified below.
#idle_timelimit 3600

# Filter to AND with uid=%s
#pam_filter objectclass=account

# The user ID attribute (defaults to uid)
#pam_login_attribute uid

# Search the root DSE for the password policy (works
# with Netscape Directory Server)
#pam_lookup_policy yes

# Check the 'host' attribute for access control
# Default is no; if set to yes, and user has no
# value for the host attribute, and pam_ldap is
# configured for account management (authorization)
# then the user will not be allowed to login.
#pam_check_host_attr yes

# Check the 'authorizedService' attribute for access
# control
# Default is no; if set to yes, and the user has no
# value for the authorizedService attribute, and
# pam_ldap is configured for account management
# (authorization) then the user will not be allowed
# to login.
#pam_check_service_attr yes

# Group to enforce membership of
#pam_groupdn cn=PAM,ou=Groups,dc=padl,dc=com

# Group member attribute
#pam_member_attribute uniquemember

# Specify a minimum or maximum UID number allowed
```

```
#pam_min_uid 0
#pam_max_uid 0

# Template login attribute, default template user
# (can be overridden by value of former attribute
# in user's entry)
#pam_login_attribute userPrincipalName
#pam_template_login_attribute uid
#pam_template_login nobody

# HEADS UP: the pam_crypt, pam_nds_passwd,
# and pam_ad_passwd options are no
# longer supported.
#
# Do not hash the password at all; presume
# the directory server will do it, if
# necessary. This is the default.
pam_password md5

# Hash password locally; required for University of
# Michigan LDAP server, and works with Netscape
# Directory Server if you're using the UNIX-Crypt
# hash mechanism and not using the NT Synchronization
# service.
#pam_password crypt

# Remove old password first, then update in
# cleartext. Necessary for use with Novell
# Directory Services (NDS)
#pam_password clear_remove_old
#pam_password nds

# RACF is an alias for the above. For use with
# IBM RACF
#pam_password racf

# Update Active Directory password, by
# creating Unicode password and updating
# unicodePwd attribute.
#pam_password ad

# Use the OpenLDAP password change
# extended operation to update the password.
#pam_password exop

# Redirect users to a URL or somesuch on password
# changes.
#pam_password_prohibit_message Please visit http://internal to change your
password.

# RFC2307bis naming contexts
# Syntax:
# nss_base_XXX          base?scope?filter
# where scope is {base,one,sub}
# and filter is a filter to be &'d with the
# default filter.
# You can omit the suffix eg:
# nss_base_passwd      ou=People,
# to append the default base DN but this
# may incur a small performance impact.
#nss_base_passwd      ou=People,dc=padl,dc=com?one
#nss_base_shadow      ou=People,dc=padl,dc=com?one
```

```

#nss_base_group          ou=Group,dc=padl,dc=com?one
#nss_base_hosts         ou=Hosts,dc=padl,dc=com?one
#nss_base_services     ou=Services,dc=padl,dc=com?one
#nss_base_networks     ou=Networks,dc=padl,dc=com?one
#nss_base_protocols    ou=Protocols,dc=padl,dc=com?one
#nss_base_rpc          ou=Rpc,dc=padl,dc=com?one
#nss_base_ethers       ou=Ethers,dc=padl,dc=com?one
#nss_base_netmasks     ou=Networks,dc=padl,dc=com?ne
#nss_base_bootparams   ou=Ethers,dc=padl,dc=com?one
#nss_base_aliases      ou=Aliases,dc=padl,dc=com?one
#nss_base_netgroup     ou=Netgroup,dc=padl,dc=com?one

# attribute/objectclass mapping
# Syntax:
#nss_map_attribute      rfc2307attribute      mapped_attribute
#nss_map_objectclass   rfc2307objectclass   mapped_objectclass

# configure --enable-nds is no longer supported.
# NDS mappings
#nss_map_attribute uniqueMember member

# Services for UNIX 3.5 mappings
#nss_map_objectclass posixAccount User
#nss_map_objectclass shadowAccount User
#nss_map_attribute uid msSFU30Name
#nss_map_attribute uniqueMember msSFU30PosixMember
#nss_map_attribute userPassword msSFU30Password
#nss_map_attribute homeDirectory msSFU30HomeDirectory
#nss_map_attribute homeDirectory msSFUHomeDirectory
#nss_map_objectclass posixGroup Group
#pam_login_attribute msSFU30Name
#pam_filter objectclass=User
#pam_password ad

# configure --enable-mssfu-schema is no longer supported.
# Services for UNIX 2.0 mappings
#nss_map_objectclass posixAccount User
#nss_map_objectclass shadowAccount user
#nss_map_attribute uid msSFUName
#nss_map_attribute uniqueMember posixMember
#nss_map_attribute userPassword msSFUPassword
#nss_map_attribute homeDirectory msSFUHomeDirectory
#nss_map_attribute shadowLastChange pwdLastSet
#nss_map_objectclass posixGroup Group
#nss_map_attribute cn msSFUName
#pam_login_attribute msSFUName
#pam_filter objectclass=User
#pam_password ad

# RFC 2307 (AD) mappings
#nss_map_objectclass posixAccount user
#nss_map_objectclass shadowAccount user
#nss_map_attribute uid sAMAccountName
#nss_map_attribute homeDirectory unixHomeDirectory
#nss_map_attribute shadowLastChange pwdLastSet
#nss_map_objectclass posixGroup group
#nss_map_attribute uniqueMember member
#pam_login_attribute sAMAccountName
#pam_filter objectclass=User
#pam_password ad

# configure --enable-authpassword is no longer supported

```

```
# AuthPassword mappings
#nss_map_attribute userPassword authPassword

# AIX SecureWay mappings
#nss_map_objectclass posixAccount aixAccount
#nss_base_passwd ou=aixaccount,?one
#nss_map_attribute uid userName
#nss_map_attribute gidNumber gid
#nss_map_attribute uidNumber uid
#nss_map_attribute userPassword passwordChar
#nss_map_objectclass posixGroup aixAccessGroup
#nss_base_group ou=aixgroup,?one
#nss_map_attribute cn groupName
#nss_map_attribute uniqueMember member
#pam_login_attribute userName
#pam_filter objectclass=aixAccount
#pam_password clear

# Netscape SDK LDAPS
#ssl on

# Netscape SDK SSL options
#sslpath /etc/ssl/certs

# OpenLDAP SSL mechanism
# start_tls mechanism uses the normal LDAP port, LDAPS typically 636
#ssl start_tls
#ssl on

# OpenLDAP SSL options
# Require and verify server certificate (yes/no)
# Default is to use libldap's default behavior, which can be configured in
# /etc/openldap/ldap.conf using the TLS_REQCERT setting. The default for
# OpenLDAP 2.0 and earlier is "no", for 2.1 and later is "yes".
#tls_checkpeer yes

# CA certificates for server certificate verification
# At least one of these are required if tls_checkpeer is "yes"
#tls_cacertfile /etc/ssl/ca.cert
#tls_cacertdir /etc/ssl/certs

# Seed the PRNG if /dev/urandom is not provided
#tls_randfile /var/run/egd-pool

# SSL cipher suite
# See man ciphers for syntax
#tls_ciphers TLSv1

# Client certificate and key
# Use these, if your server requires client authentication.
#tls_cert
#tls_key

# Disable SASL security layers. This is needed for AD.
#sasl_secprops maxssf=0

# Override the default Kerberos ticket cache location.
#krb5_ccname FILE:/etc/.ldapcache

# SASL mechanism for PAM authentication - use is experimental
# at present and does not support password policy control
#pam_sasl_mech DIGEST-MD5
```

Now we need to copy the file /etc/ldap.conf to the file /etc/ldap/ldap.conf. First we will backup the file and then we will copy the new file.

```
cp /etc/ldap/ldap.conf /etc/ldap/ldap.conf.original
cp /etc/ldap.conf /etc/ldap/ldap.conf
```

OK, create a new file by running the following command. You will need to edit the first part of the command to use your favorite editor.

```
vim /etc/auth-client-config/profile.d/open_ldap
```

This file is the new OpenLDAP authentication profile. Copy and paste EXACTLY the following lines:

```
[open_ldap]
nss_passwd=passwd: compat ldap
nss_group=group: compat ldap
nss_shadow=shadow: compat ldap
pam_auth=auth          required          pam_env.so
auth                  sufficient        pam_unix.so likeauth nullok
auth                  sufficient        pam_ldap.so use_first_pass
auth                  required          pam_deny.so
pam_account=account   sufficient        pam_unix.so
account               sufficient        pam_ldap.so
account               required          pam_deny.so
pam_password=password sufficient        pam_unix.so nullok md5 shadow use_authtok
password              sufficient        pam_ldap.so use_first_pass
password              required          pam_deny.so
pam_session=session  required          pam_limits.so
session               required          pam_mkhome.so skel=/etc/skel/ umask=0077
session               required          pam_unix.so
session               optional          pam_ldap.so
```

Backup the /etc/nsswitch.conf file:

```
cp /etc/nsswitch.conf /etc/nsswitch.conf.original
```

Backup the files in /etc/pam.d:

```
cd /etc/pam.d/
mkdir bkup
cp * bkup/
```

Enable the new OpenLDAP profile by running the following command. If you did all the previous steps correctly then this will run without issue.

```
auth-client-config -a -p open_ldap
```

The final step is to simply reboot the client. When the client is running again then test to see if you can log in with your new LDAP user. No matter what you should be able to log in with a local user (unless the system is hung). If the system hangs then reboot HARD and try again.

```
reboot
```

Top /etc/pam.d/common-password

```
password sufficient pam_ldap.so
password required pam_unix.so nullok obscure min=4 max=8 md5
```

For solving login issue

Configure SAMBA to Share /ldaphome

Top

Since this entire project is to create a domain for Windows PCs it only makes sense to configure the server so that the user home directories are available to Windows clients. This section will configure

SAMBA so that the /ldaphome directory is shared.

Add the following lines to the bottom of the /etc/samba/smb.conf file:

```
# LDAPHOME share definition
```

```
[ldaphome]
```

```
path = /ldaphome
```

```
writeable = yes
```

```
browseable = yes
```

```
security mask = 0777
```

```
force security mode = 0
```

```
directory security mask = 0777
```

```
force directory security mode = 0
```

SAMBA should automatically update its configuration after about 2 minutes. From a Windows computer you should be able to access the server as an LDAP user. You will then have access to your home folder.

Top

Configure SAMBA - Enable the 'Netlogon' Share

Top

Create a directory for the netlogon share to use:

```
mkdir /home/samba
```

```
mkdir /home/samba/netlogon
```

Open the file /etc/samba/smb.conf for editing:

```
vim /etc/samba/smb.conf
```

Uncomment the netlogon lines by changing:

```
:[netlogon]
; comment = Network Logon Service
; path = /home/samba/netlogon
; guest ok = yes
; writable = no
; share modes = no
```

To:

```
[netlogon]
comment = Network Logon Service
path = /home/samba/netlogon
guest ok = yes
writable = no
share modes = no
```

Top

Create a Simple Windows Logon Script

Top

We will create the logon script in the new Netlogon shared folder.

```
vim /home/samba/netlogon/allusers.bat
```

Copy and paste the following lines into that new file. Customize as necessary!

```
@echo off
REM # SYNC THE TIME WITH THE SERVER
net time \\dc01-ubuntu.example.local /set /y
REM # DELETE ALL MAPPED DRIVES
net use h: /delete
```

```
REM # MAP ALL NECESSARY DRIVES
net use h: "\\dc01-ubuntu.example.local\ldaphome\%username%"
We need to install an extra program to convert this file to a file that Windows can use.
```

```
apt-get install flip
Use this program to convert the file:
```

```
flip -m /home/samba/netlogon/allusers.bat
Now we need to tell Samba about this logon script.
```

```
vim /etc/samba/smb.conf
Change the line: ; logon script = logon.cmd
```

```
To: logon script = allusers.bat
```

Please note that I removed the semicolon (;) and changed the name of the file.

Now when Windows clients log in to the domain the script will run.

Top

Appendix A: Final /etc/samba/smb.conf File

Top

Here is a copy of my final /etc/samba/smb.conf file for your reference. This has all my customization in it already.

```
#
# Sample configuration file for the Samba suite for Debian GNU/Linux.
#
#
# This is the main Samba configuration file. You should read the
# smb.conf(5) manual page in order to understand the options listed
# here. Samba has a huge number of configurable options most of which
# are not shown in this example
#
# Any line which starts with a ; (semi-colon) or a # (hash)
# is a comment and is ignored. In this example we will use a #
# for commentary and a ; for parts of the config file that you
# may wish to enable
#
# NOTE: Whenever you modify this file you should run the command
# "testparm" to check that you have not made any basic syntactic
# errors.
#

#===== Global Settings =====

[global]

## Browsing/Identification ###

# Change this to the workgroup/NT-domain name your Samba server will part of
#   workgroup = MSHOME
workgroup = EXAMPLE

# server string is the equivalent of the NT Description field
#   server string = %h server (Samba, Ubuntu)

# Windows Internet Name Serving Support Section:
# WINS Support - Tells the NMBD component of Samba to enable its WINS Server
#   wins support = no
```

```

# WINS Server - Tells the NMBD components of Samba to be a WINS Client
# Note: Samba can be either a WINS Server, or a WINS Client, but NOT both
; wins server = w.x.y.z

# This will prevent nmbd to search for NetBIOS names through DNS.
dns proxy = no

# What naming service and in what order should we use to resolve host names
# to IP addresses
; name resolve order = lmhosts host wins bcst

#### Networking ####

# The specific set of interfaces / networks to bind to
# This can be either the interface name or an IP address/netmask;
# interface names are normally preferred
; interfaces = 127.0.0.0/8 eth0

# Only bind to the named interfaces and/or networks; you must use the
# 'interfaces' option above to use this.
# It is recommended that you enable this feature if your Samba machine is
# not protected by a firewall or is a firewall itself. However, this
# option cannot handle dynamic or non-broadcast interfaces correctly.
; bind interfaces only = true

#### Debugging/Accounting ####

# This tells Samba to use a separate log file for each machine
# that connects
log file = /var/log/samba/log.%m

# Put a capping on the size of the log files (in Kb).
max log size = 1000

# If you want Samba to only log through syslog then set the following
# parameter to 'yes'.
; syslog only = no

# We want Samba to log a minimum amount of information to syslog. Everything
# should go to /var/log/samba/log.{smbd,nmbd} instead. If you want to log
# through syslog you should set the following parameter to something higher.
syslog = 0

# Do something sensible when Samba crashes: mail the admin a backtrace
panic action = /usr/share/samba/panic-action %d

##### Authentication #####

# "security = user" is always a good idea. This will require a Unix account
# in this server for every user accessing the server. See
# /usr/share/doc/samba-doc/htmldocs/Samba3-HOWTO/ServerType.html
# in the samba-doc package for details.
; security = user
security = user

# You may wish to use password encryption. See the section on
# 'encrypt passwords' in the smb.conf(5) manpage before enabling.
encrypt passwords = true

```

```

# If you are using encrypted passwords, Samba will need to know what
# password database type you are using.
#   passwd backend = tdbsam
passwd backend = ldapsam:ldap://localhost/

#   obey pam restrictions = yes
obey pam restrictions = no

#####
#COPY AND PASTE THE FOLLOWING UNDERNEATH "OBEY PAM RESTRICTIONS = NO"
#####
#
#       Begin: Custom LDAP Entries
#
ldap admin dn = cn=admin,dc=example,dc=local
ldap suffix = dc=example, dc=local
ldap group suffix = ou=Groups
ldap user suffix = ou=Users
ldap machine suffix = ou=Computers
ldap idmap suffix = ou=Users
; Do ldap passwd sync
ldap passwd sync = Yes
passwd program = /usr/sbin/smbldap-passwd %u
passwd chat = *New*password* %n\n *Retype*new*password* %n\n
*all*authentication*tokens*updated*
add user script = /usr/sbin/smbldap-useradd -m "%u"
ldap delete dn = Yes
delete user script = /usr/sbin/smbldap-userdel "%u"
add machine script = /usr/sbin/smbldap-useradd -w "%u"
add group script = /usr/sbin/smbldap-groupadd -p "%g"
delete group script = /usr/sbin/smbldap-groupdel "%g"
add user to group script = /usr/sbin/smbldap-groupmod -m "%u" "%g"
delete user from group script = /usr/sbin/smbldap-groupmod -x "%u" "%g"
set primary group script = /usr/sbin/smbldap-usermod -g "%g" "%u"
domain logons = yes
#
#       End: Custom LDAP Entries
#
#####
#STOP COPYING HERE!
#####

;   guest account = nobody
;   invalid users = root

# This boolean parameter controls whether Samba attempts to sync the Unix
# password with the SMB password when the encrypted SMB password in the
# passwd is changed.
;   unix password sync = no

# For Unix password sync to work on a Debian GNU/Linux system, the following
# parameters must be set (thanks to Ian Kahan < for
# sending the correct chat script for the passwd program in Debian Sarge).
    passwd program = /usr/bin/passwd %u
    passwd chat = *Enter\snew\sUNIX\spassword:* %n\n
*Retype\snew\sUNIX\spassword:* %n\n *passwd:*password\supdated\ssuccessfully* .

```

```

# This boolean controls whether PAM will be used for password changes
# when requested by an SMB client instead of the program listed in
# 'passwd program'. The default is 'no'.
; pam password change = no

##### Domains #####

# Is this machine able to authenticate users. Both PDC and BDC
# must have this setting enabled. If you are the BDC you must
# change the 'domain master' setting to no
#
; domain logons = yes
#
# The following setting only takes effect if 'domain logons' is set
# It specifies the location of the user's profile directory
# from the client point of view)
# The following required a [profiles] share to be setup on the
# samba server (see below)
; logon path = \\%N\profiles\%U
# Another common choice is storing the profile in the user's home directory
; logon path = \\%N\%U\profile
logon path =

# The following setting only takes effect if 'domain logons' is set
# It specifies the location of a user's home directory (from the client
# point of view)
; logon drive = H:
; logon home = \\%N\%U

# The following setting only takes effect if 'domain logons' is set
# It specifies the script to run during logon. The script must be stored
# in the [netlogon] share
# NOTE: Must be store in 'DOS' file format convention
; logon script = logon.cmd
logon script = allusers.bat

# This allows Unix users to be created on the domain controller via the SAMR
# RPC pipe. The example command creates a user account with a disabled Unix
# password; please adapt to your needs
; add user script = /usr/sbin/adduser --quiet --disabled-password --gecos "" %u

##### Printing #####

# If you want to automatically load your printer list rather
# than setting them up individually then you'll need this
; load printers = yes

# lpr(ng) printing. You may wish to override the location of the
# printcap file
; printing = bsd
; printcap name = /etc/printcap

# CUPS printing. See also the cupsaddsmb(8) manpage in the
# cupsys-client package.
; printing = cups
; printcap name = cups

# When using [print$], root is implicitly a 'printer admin', but you can
# also give this right to other users to add drivers and set printer
# properties
; printer admin = @lpadmin

```

Misc

```
# Using the following line enables you to customise your configuration
# on a per machine basis. The %m gets replaced with the netbios name
# of the machine that is connecting
; include = /home/samba/etc/smb.conf.%m

# Most people will find that this option gives better performance.
# See smb.conf(5) and /usr/share/doc/samba-doc/htmldocs/Samba3-HOWTO/speed.html
# for details
# You may want to add the following on a Linux system:
#     SO_RCVBUF=8192 SO_SNDBUF=8192
#     socket options = TCP_NODELAY
```

```
# The following parameter is useful only if you have the linpopup package
# installed. The samba maintainer and the linpopup maintainer are
# working to ease installation and configuration of linpopup and samba.
; message command = /bin/sh -c '/usr/bin/linpopup "%f" "%m" %s; rm %s' &
```

```
# Domain Master specifies Samba to be the Domain Master Browser. If this
# machine will be configured as a BDC (a secondary logon server), you
# must set this to 'no'; otherwise, the default behavior is recommended.
; domain master = auto
```

```
# Some defaults for winbind (make sure you're not using the ranges
# for something else.)
```

```
; idmap uid = 10000-20000
; idmap gid = 10000-20000
; template shell = /bin/bash
;
```

```
; The following was the default behaviour in sarge
; but samba upstream reverted the default because it might induce
; performance issues in large organizations
; See #368251 for some of the consequences of *not* having
; this setting and smb.conf(5) for all details
;
```

```
; winbind enum groups = yes
; winbind enum users = yes
```

===== Share Definitions =====

```
# Un-comment the following (and tweak the other settings below to suit)
# to enable the default home directory shares. This will share each
# user's home directory as \\server\username
```

```
:[homes]
; comment = Home Directories
; browseable = no
```

```
# By default, \\server\username shares can be connected to by anyone
# with access to the samba server. Un-comment the following parameter
# to make sure that only "username" can connect to \\server\username
# This might need tweaking when using external authentication schemes
; valid users = %S
```

```
# By default, the home directories are exported read-only. Change next
# parameter to 'yes' if you want to be able to write to them.
; writable = no
```

```
# File creation mask is set to 0700 for security reasons. If you want to
# create files with group=rw permissions, set next parameter to 0775.
; create mask = 0700
```

```

# Directory creation mask is set to 0700 for security reasons. If you want to
# create dirs. with group=rw permissions, set next parameter to 0775.
;   directory mask = 0700

# Un-comment the following and create the netlogon directory for Domain Logons
# (you need to configure Samba to act as a domain controller too.)
[netlogon]
    comment = Network Logon Service
    path = /home/samba/netlogon
    guest ok = yes
    writable = no
    share modes = no

# Un-comment the following and create the profiles directory to store
# users profiles (see the "logon path" option above)
# (you need to configure Samba to act as a domain controller too.)
# The path below should be writable by all users so that their
# profile directory may be created the first time they log on
;[profiles]
;   comment = Users profiles
;   path = /home/samba/profiles
;   guest ok = no
;   browseable = no
;   create mask = 0600
;   directory mask = 0700

[printers]
    comment = All Printers
    browseable = no
    path = /var/spool/samba
    printable = yes
    public = no
    writable = no
    create mode = 0700

# Windows clients look for this share name as a source of downloadable
# printer drivers
[print$]
    comment = Printer Drivers
    path = /var/lib/samba/printers
    browseable = yes
    read only = yes
    guest ok = no
# Uncomment to allow remote administration of Windows print drivers.
# Replace 'ntadmin' with the name of the group your admin users are
# members of.
;   write list = root, @ntadmin

# A sample share for sharing your CD-ROM with others.
;[cdrom]
;   comment = Samba server's CD-ROM
;   writable = no
;   locking = no
;   path = /cdrom
;   public = yes

# The next two parameters show how to auto-mount a CD-ROM when the
#   cdrom share is accessed. For this to work /etc/fstab must contain
#   an entry like this:
#
#   /dev/scd0 /cdrom iso9660 defaults,noauto,ro,user 0 0

```

```

#
# The CD-ROM gets unmounted automatically after the connection to the
#
# If you don't want to use auto-mounting/unmounting make sure the CD
#     is mounted on /cdrom
#
;   preexec = /bin/mount /cdrom
;   postexec = /bin/umount /cdrom

# LDAPHOME share definition
[ldaphome]
path = /ldaphome
writeable = yes
browseable = yes
security mask = 0777
force security mode = 0
directory security mask = 0777
force directory security mode = 0
Top

```

Appendix B: Final /etc/ldap/slapd.conf File

Top

Here is a copy of my final /etc/ldap/slapd.conf file for your reference.

```

# This is the main slapd configuration file. See slapd.conf(5) for more
# info on the configuration options.

#####
# Global Directives:

# Features to permit
#allow bind_v2

# Schema and objectClass definitions
include      /etc/ldap/schema/core.schema
include      /etc/ldap/schema/cosine.schema
include      /etc/ldap/schema/nis.schema
include      /etc/ldap/schema/inetorgperson.schema
include      /etc/ldap/schema/samba.schema
include      /etc/ldap/schema/misc.schema

# Where the pid file is put. The init.d script
# will not stop the server if you change this.
pidfile      /var/run/slapd/slapd.pid

# List of arguments that were passed to the server
argsfile     /var/run/slapd/slapd.args

# Read slapd.conf(5) for possible values
loglevel     0

# Where the dynamically loaded modules are stored
modulepath   /usr/lib/ldap
moduleload   back_bdb

# The maximum number of entries that is returned for a search operation
sizelimit    500

# The tool-threads parameter sets the actual amount of cpu's that is used

```

```

# for indexing.
tool-threads 1

#####
# Specific Backend Directives for bdb:
# Backend specific directives apply to this backend until another
# 'backend' directive occurs
backend      bdb
checkpoint 512 30

#####
# Specific Backend Directives for 'other':
# Backend specific directives apply to this backend until another
# 'backend' directive occurs
#backend

#####
# Specific Directives for database #1, of type bdb:
# Database specific directives apply to this database until another
# 'database' directive occurs
database     bdb

# The base of your directory in database #1
suffix      "dc=example,dc=local"

# rootdn directive for specifying a superuser on the database. This is needed
# for syncrepl.
# rootdn    "cn=admin,dc=example,dc=local"

# Where the database file are physically stored for database #1
#directory  "/var/lib/ldap"
directory   "/ldap_data"

# For the Debian package we use 2MB as default but be sure to update this
# value if you have plenty of RAM
dbconfig set_cachesize 0 2097152 0

# Sven Hartge reported that he had to set this value incredibly high
# to get slapd running at all. See http://bugs.debian.org/303057
# for more information.

# Number of objects that can be locked at the same time.
dbconfig set_lk_max_objects 1500
# Number of locks (both requested and granted)
dbconfig set_lk_max_locks 1500
# Number of lockers
dbconfig set_lk_max_lockers 1500

# Indexing options for database #1
index      objectClass eq

# Save the time that the entry gets modified, for database #1
lastmod    on

# Where to store the replica logs for database #1
# relogfile /var/lib/ldap/repllog

# The userPassword by default can be changed
# by the entry owning it if they are authenticated.
# Others should not be able to see it, except the
# admin entry below
# These access lines apply to database #1 only

```

```

access to attrs=userPassword,shadowLastChange,sambaNTPassword,sambaLMPassw
    by dn="cn=admin,dc=example,dc=local" write
    by anonymous auth
    by self write
    by * none

# Ensure read access to the base for things like
# supportedSASLMechanisms. Without this you may
# have problems with SASL not knowing what
# mechanisms are available and the like.
# Note that this is covered by the 'access to *'
# ACL below too but if you change that as people
# are wont to do you'll still need this if you
# want SASL (and possible other things) to work
# happily.
access to dn.base="" by * read

# The admin dn has full write access, everyone else
# can read everything.
access to *
    by dn="cn=admin,dc=example,dc=local" write
    by * read

# For Netscape Roaming support, each user gets a roaming
# profile for which they have write access to
#access to dn=".*,ou=Roaming,o=morsnet"
#     by dn="cn=admin,dc=example,dc=local" write
#     by dnattr=owner write

#####
# Specific Directives for database #2, of type 'other' (can be bdb too):
# Database specific directives apply to this databasse until another
# 'database' directive occurs
#database

# The base of your directory for database #2
#suffix          "dc=debian,dc=org"
Top

```

Appendix C: Windows XP Professional SP2 Client Configuration Notes

Top

Anyone that has configured a Windows XP computer for use on a Windows domain will have no problems here. The main thing you have to remember is a) Make sure the network is working. b) Make sure DNS is working. c) Join the computer to the correct domain.

Go ahead and join the computer to the domain like you normally would.

1. Log into the computer as an Administrative user (most likely Administrator)
2. Right click "My Computer" and select "Properties"
3. Select the "Computer Name" tab at the top
4. Click the "Change" button near the bottom
5. In this new window select the "Domain:" radio button in the "Member of" section
6. Type in your domain name - in our example the domain name to enter is simply "example"

7. Click the "OK" button
8. A window should pop up asking you for a username and password. Use "root" and your root password which should still be "12345" unless you changed it
9. After a few seconds you should see a pop-up that says "Welcome to the example domain" or something similar to that effect
10. Click "OK"
11. Click "OK" again
12. Reboot the computer
13. When it boots you will be at a login prompt. The first time you try to log in you'll want to ensure that you are logging on to the DOMAIN, not the LOCAL COMPUTER.

Follow those simple steps and you should have a Windows client on your domain in no time.

Top

Appendix D: Ubuntu Server 7.10 (Client) Configuration Notes

Top

Setting up a client is very similar to setting up the server. Basically you will want to install the operating system, install updates, configure networking options, make sure DNS resolves correctly, etc... Once you are at that stage you can follow the directions from earlier sections to configure NFS mounting and LDAP authentication. At that point all you need to do is deploy your client to where it needs to be.

In fact, once you have a client setup and configured you could potentially create a master image that you can then deploy to other systems at a later date. Just an idea.

The most important thing about the client is that DNS resolution works! This is an absolute. You also want to ensure that all systems have the same time set!