The latest SAMBA-LDAP-PDC How-to

Assembled by
David Trask
Technology Coordinator
Vassalboro Community School
Vassalboro, Maine USA

Context of this How-to

This How-To aims at helping to configure an OpenLDAP/Samba Primary Domain Controller for Microsoft Windows Workstations (and, using nss ldap and pam ldap, a unique source of authentication for all workstations, including Linux and other UNIX systems).

For our example, we'll use the following:

- The K12LTSP (www.k12ltsp.org) distribution version 3.1 based on RedHat 9.0 (we used this as many of the necessary packages are already included and saved us from having to download them)
- All workstations and servers are in the same LAN 192.168.1.0/24
- We want to configure the Microsoft Windows NT Domain named SAMPLE-NT
- We will have a central Primary Domain Controller named SAMPLE-PDC (NetBIOS name) on the host 192.168.1.1/32
- We want this Primary Domain Controller to be the WINS server and the Master Browser
- Server of the SAMPLE-NT domain
- All authentication objects (users and groups) will be stored on an OpenLDAPserver, using the base DN : dc=SAMPLE,dc=ORG
- Samba Users accounts will be stored inou=Users,dc=SAMPLE,dc=ORG
- Samba Computers accounts will be stored in ou=Computers,dc=SAMPLE,dc=ORG
- Samba Groups accounts will be stored in ou=Groups,dc=SAMPLE,dc=ORG

Separating Samba accounts (Users and Computers) and Groups is a optional way to do the job. We could store all of this data under the same DN, but we made this distinction to make the LDAP tree more human-readable. Additionally there is also a potential issue with computer management via LDAP. Feel free to change those statements (Microsoft Windows NT Domain Name, LDAP tree) for something that fits your situation better.

DNS resolution must be ok to use Samba without spending hours trying to understand why that think is supposed to work and doesn't!

What you'll need and want:

OpenLDAP packages…when you install RH 9 via the K12LTSP distribution try doing a custom install and selecting individual packages. It is assumed that you have some experience in performing installations. Check "Select individual packages" and then make sure to select the following:

Openldap
Openldap-clients
Openldap-servers
Nss-ldap
Pam-ldap
Samba (I used version 2.2.7a)
Samba-client
Samba-common
*Samba-server (this may not be required, but I used it anyway)

(optional, but highly desirable is…. Directory_administrator)

In the event that you are using a different distribution or that the packages above do not come on the disks you have they may be downloaded from the Internet.

http://www.openldap.org

http://www.samba.org

or try http://www.rpmfind.net for specific RedHat packages

Feel free to look around for others that may correspond with your distribution.

smbldap-tools

smbldap-tools is a package containing some useful scripts to manage users/groups when you're using LDAP as source of users/groups data (for Unix and for Samba). We used those scripts in this How-to to add/del/modify users and groups.
These tools can be found at http://idealx.org or more specifically (as of this writing)

http://www.idealx.org/prj/samba/index.en.html

Compiling Samba:

Unfortunately Samba needs to be compiled with LDAP enabled. This is kinda' hard to do once it's installed. Here's what I did. I downloaded Pre-compiled versions of Samba with LDAP support (I installed them right on top of the current Samba...worked fine)

http://ftp.freshrpms.net/pub/freshrpms/redhat/testing/samba-ldap/

Download the samba packages corresponding to the names mention above and install those as well (right on top of the Samba you installed when setting up your server). This will give you an LDAP enabled version of Samba.

Sample compile instructions for Samba source- (if that's the route you choose..adjust for your situation)

1) cd /usr/local/src

2) tar -xvzf samba-latest.tar.gz

3) cd samba-latest/source/

4) ./configure -with-acl-support -with-profile -disable-static -with-msdfs -with-ldapsam

5) make

6) make install

Configuring OpenLDAP

You'll need to configure your OpenLDAPserver to serve as SAM database for Samba.

Following our example, we must to configure it to :

•accept the Samba LDAP v3 schema,
•run on the base DN dc=SAMPLE,dc=ORG,
•contain the minimal entries needed to start using it.

For the needs of this HOWTO example, we have used the following LDAP tree :
(using Relative DN notation)

```
dc=SAMPLE,dc=ORG
        |
        ‘--- ou=Users : to store user accounts (both posixAccount and
        |        sambaAccount) for Unix and Windows systems
        |
        ‘--- ou=Computers : to store computer accounts (sambaAccount) for Windows
        |        systems
        |
        ‘--- ou=Groups : to store system groups (posixGroup) for Unix and Windows
                 systems (or for any other LDAP-aware systems)
```

You may choose to use another LDAP tree to store objects : for example, all accounts (shadowAccounts and sambaAccounts) ”under” the same DN. We thought it was simpler to understand like this (and was not a problem for an Unix-nss ldap do deal with). Additionally, using shadowAccount is not mandatory: if you don’t use shadow passwords on your Unix systems, you should use posixAccounts instead.

Using Samba and OpenLDAP, we will store:

- Windows user accounts using sambaAccount object class (samba.schema)
- Windows computer accounts (ie. workstations) using sambaAccount object class
- Unix-only user accounts using shadowAccount object class (nis.schema)
- Users groups (Windows and Unix, as it seems there is no difference in Samba release 2.2.4 using posixGroup object class.)

Schemas

First, copy the samba.schema to /etc/openldap/schema/samba.schema.
You’ll find thisSamba schema shipped with the Samba release (on my K12LTSP box it was located in /usr/share/doc/samba-2.2.7a/LDAP/samba.schema)

Configuration

Create your /etc/openldap/slapd.conf to configure your server :

```
# /etc/openldap/slapd.conf file for SAMBA-LDAP

 include /etc/openldap/schema/core.schema
include /etc/openldap/schema/cosine.schema
include /etc/openldap/schema/inetorgperson.schema
include /etc/openldap/schema/nis.schema
include /etc/openldap/schema/samba.schema
```

```
database ldbm
suffix "dc=SAMPLE,dc=ORG"
rootdn "cn=Manager,dc=SAMPLE,dc=ORG"
rootpw secret
directory /var/lib/ldap

index objectClass,rid,uid,uidNumber,gidNumber,memberUid eq
index cn,mail,surname,givenname eq,subinitial

# - The End
```

This is a copy of my slapd.conf file to give you some more ideas.

```
include           /etc/openldap/schema/core.schema
include           /etc/openldap/schema/cosine.schema
include           /etc/openldap/schema/inetorgperson.schema
include           /etc/openldap/schema/nis.schema
include           /etc/openldap/schema/redhat/rfc822-MailMember.schema
include           /etc/openldap/schema/redhat/autofs.schema
include           /etc/openldap/schema/redhat/kerberosobject.schema
include           /etc/openldap/schema/samba.schema

# Define global ACLs to disable default read access.

# Do not enable referrals until AFTER you have a working directory
# service AND an understanding of referrals.
#referral   ldap://root.openldap.org

#pidfile    //var/run/slapd.pid
#argsfile   //var/run/slapd.args

# Create a replication log in /var/lib/ldap for use by slurpd.
#replogfile /var/lib/ldap/master-slapd.replog

# Load dynamic backend modules:
# modulepath      /usr/sbin/openldap
# moduleload      back_ldap.la
# moduleload      back_ldbm.la
# moduleload      back_passwd.la
# moduleload      back_shell.la

#
# The next three lines allow use of TLS for connections using a dummy test
# certificate, but you should generate a proper certificate by changing to
# /usr/share/ssl/certs, running "make slapd.pem", and fixing permissions on
# slapd.pem so that the ldap user or group can read it.
# TLSCertificateFile /usr/share/ssl/certs/slapd.pem
# TLSCertificateKeyFile /usr/share/ssl/certs/slapd.pem
# TLSCACertificateFile /usr/share/ssl/certs/ca-bundle.crt
#
# Sample Access Control
#     Allow read access of root DSE
#     Allow self write access
#     Allow authenticated users read access
#     Allow anonymous users to authenticate
#
#access to dn="" by * read
```

```
#access to *
#     by self write
#     by users read
#     by anonymous auth
#
# if no access controls are present, the default is:
#     Allow read by all
#
# rootdn can always write!

#######################################################################
# ldbm database definitions
#######################################################################

database    ldbm
suffix      "dc=vassalboro,dc=org"
#suffix           "o=My Organization Name,c=US"
rootdn      "cn=Manager,dc=vassalboro,dc=org"
#rootdn           "cn=Manager,o=My Organization Name,c=US"
# Cleartext passwords, especially for the rootdn, should
# be avoided.  See slappasswd(8) and slapd.conf(5) for details.
# Use of strong authentication encouraged.
rootpw      secret
# rootpw          {crypt}ijFYNcSNctBYg
# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd/tools. Mode 700 recommended.
directory   /var/lib/ldap
# Indices to maintain
index objectClass,rid,uid,uidNumber,gidNumber,memberUid      eq
index cn,mail,surname,givenname                  eq,subinitial
# Replicas to which we should propagate changes
#replica host=ldap-1.example.com:389 tls=yes
#     bindmethod=sasl saslmech=GSSAPI
#     authcId=host/ldap-master.example.com@EXAMPLE.COM
```

Then, edit your/etc/openldap/ldap.conf to indicate your base DN and default server:

```
# /etc/openldap/ldap.conf for samba-ldap
#
# LDAP Defaults

HOST 127.0.0.1
BASE dc=SAMPLE,dc=ORG

# - The End
```

This is a copy of my /etc/openldap/ldap.conf file to give you some more ideas.

```
#
# LDAP Defaults
#

# See ldap.conf(5) for details
# This file should be world readable but not world writable.
```

```
#BASE dc=example, dc=com
#URI  ldap://ldap.example.com ldap://ldap-master.example.com:666

#SIZELIMIT  12
#TIMELIMIT  15
#DEREF      never
HOST 127.0.0.1
BASE dc=vassalboro,dc=org
```

Finally, start your OpenLDAPserver :

/etc/init.d/ldap start

or

service ldap start

Everything should work fine. If not :

- verify your schemas
- verify that /var/lib/ldap exist and is owned by the user who run sladp (ldap user for RedHat OpenLDAPpackages)
- consult the OpenLDAPdocumentation

Initial entries

Next, we'll inject some initial entries on the brand new OpenLDAP server configured and started above.

A sample LDIF file is presented below…copy/paste it on a file named base.ldif and add it using:

ldapadd -x -h localhost -D "cn=manager,dc=SAMPLE,dc=ORG" –f base.ldif -W

(type your admin DN password, 'secret' to complete the command)

you can put the base.ldif anywhere for the most part…I created a folder in my /root directory called LDIF and put it in there for safe-keeping

This is a copy of my base.ldif….use it for ideas or copy and modify…another one is located on the idealx.org site in the pdf version of the how-to.

```
dn: dc=vassalboro,dc=org
objectClass: domain
dc: vassalboro

dn: ou=Groups,dc=vassalboro,dc=org
objectClass: top
```

```
objectClass: organizationalunit
ou: Groups
description: System groups

dn: ou=Users,dc=vassalboro,dc=org
objectclass: top
objectclass: organizationalunit
ou: Users
description: Users of the Organization

dn: ou=Computers,dc=vassalboro,dc=org
objectclass: top
objectclass: organizationalunit
ou: Computers
description: Windows Domain Computers


dn: dc=vassalboro,dc=org
objectclass: dcobject
objectclass: organization
o: vassalboro org.
dc: vassalboro

dn: ou=manager,dc=vassalboro,dc=org
objectclass: organizationalrole
cn: manager

dn: cn=Domain Admins,ou=Groups,dc=vassalboro,dc=org
objectClass: posixGroup
gidNumber: 200
cn: Domain Admins
memberUid: administrator
description: Windows Domain Users

dn: cn=Domain Users,ou=Groups,dc=vassalboro,dc=org
objectClass: posixGroup
gidNumber: 201
cn: Domain Users
description: Windows Domain Users
```

smbldap-tools configuration

Finally, you must configure your smbldap-tools to match your system and LDAP configuration:

(If you haven’t done so yet….download them and install them using
rpm Uvh smbldap-tools.0.7-2.i386.rpm
By default, this will install the tools to /usr/local/sbin/)

smbldap_tools.pm

In the version of the smbldap-tools that I got, there is a problem on line 520 of /usr/local/sbin/ smbldap_tools.pm. The line looks like this:

```
die ``Cannot open <<$filename>> for writing: $!\n'';
```

and should look like this:

```
die ``Cannot open $filename for writing: $!\n'';
```

```
I used emacs to edit as you can get the line number on the bottom of the screen.
```

smbldap-useradd.pl

Edit /usr/local/sbin/smbldap-useradd.pl

Uncomment lines regarding RedHat style groups if desired.

smbldap_conf.pm

Edit the /usr/local/sbin/smbldap_conf.pm and configure it according to your LDAP configuration (RootDN password and LDAP server @IP address). See my sample below…

You'll find two confusing entries: slaveLDAP and masterLDAP. For our example, those two LDAP servers will be the same one, but in a real life configuration, you may want to have a slave server to serve all your read request, and one dedicated to write request. Anyway, in the current example, as we build the PDC using Samba and OpenLDAPon the same host, you should specify 127.0.0.01 for the two LDAP servers.

You'll find some other configuration options in this configuration file: those are the default values used by smbldap-tools when creating an account (user or computer). Feel free to change those values if desired.

This is a copy of my smbldap_conf.pm file….use it for ideas or modify to your system…*(I've highlighted items you need to pay particular attention to…most of them can be left to the default…except the obvious ones like names and so forth)*

```
#!/usr/bin/perl
use strict;
package smbldap_conf;

# $Id: smbldap_conf.pm,v 1.14 2002/06/01 04:30:48 olem Exp $
```

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

#  This code was developped 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

use vars qw($VERSION @ISA @EXPORT @EXPORT_OK %EXPORT_TAGS
$UID_START $GID_START $smbpasswd $slaveLDAP $masterLDAP
$with_smbpasswd $mk_ntpasswd
$ldap_path $ldap_opts $ldapsearch $ldapsearchnobind
$ldapmodify $ldappasswd $ldapadd $ldapdelete $ldapmodrdn
$suffix $usersdn $computersdn
$groupsdn $scope $binddn $bindpasswd
$slaveDN $slavePw $masterDN $masterPw
$_userLoginShell $_userHomePrefix $_userGecos
$_defaultUserGid $_defaultComputerGid
$_skeletonDir $_userSmbHome
$_userProfile $_userHomeDrive
$_userScript $usersou $computersou $groupsou
);

use Exporter;
$VERSION = 1.00;
@ISA = qw(Exporter);

@EXPORT = qw(
$UID_START $GID_START $smbpasswd $slaveLDAP $masterLDAP
$with_smbpasswd $mk_ntpasswd
$ldap_path $ldap_opts $ldapsearch $ldapsearchnobind $ldapmodify $ldappasswd
$ldapadd $ldapdelete $ldapmodrdn $suffix $usersdn
$computersdn $groupsdn $scope $binddn $bindpasswd
$slaveDN $slavePw $masterDN $masterPw
$_userLoginShell $_userHomePrefix $_userGecos
$_defaultUserGid $_defaultComputerGid $_skeletonDir
$_userSmbHome $_userProfile $_userHomeDrive $_userScript
$usersou $computersou $groupsou
);


##############################################################################
#
# General Configuration
```

```
#
##############################################################################

#
# UID and GID starting at...
#

$UID_START = 1000;
$GID_START = 1000;

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

# Notes: to use to dual ldap servers backend for Samba, you must patch
# Samba with the dual-head patch from vassalboro. If not using this patch
# just use the same server for slaveLDAP and masterLDAP.
#
# Slave LDAP : needed for read operations
#
# Ex: $slaveLDAP = "127.0.0.1";
$slaveLDAP = "localhost";

#
# Master LDAP : needed for write operations
#
# Ex: $masterLDAP = "127.0.0.1";
$masterLDAP = "localhost";

#
# LDAP Suffix
#
# Ex: $suffix = "dc=vassalboro,dc=ORG";
$suffix = "dc=vassalboro,dc=org";

#
# Where are stored Users
#
# Ex: $usersdn = "ou=Users,$suffix"; for ou=Users,dc=vassalboro,dc=ORG
$usersou = q(Users);

$usersdn = "ou=$usersou,$suffix";

#
# Where are stored Computers
#
# Ex: $computersdn = "ou=Computers,$suffix"; for ou=Computers,dc=vassalboro,dc=ORG
$computersou = q(Computers);

$computersdn = "ou=$computersou,$suffix";

#
# Where are stored Groups
#
# Ex $groupsdn = "ou=Groups,$suffix"; for ou=Groups,dc=vassalboro,dc=ORG
$groupsou = q(Groups);

$groupsdn = "ou=$groupsou,$suffix";

#
```

```
# Default scope Used
#
$scope = "sub";

#
# Credential Configuration
#
# Bind DN used
# Ex: $binddn = "cn=Manager,$suffix"; for cn=Manager,dc=vassalboro,dc=org
$binddn = "cn=Manager,$suffix";
#
# Bind DN passwd used
# Ex: $bindpasswd = 'secret'; for 'secret'
$bindpasswd = "secret";

#
# Notes: if using dual ldap patch, you can specify to different configuration
# By default, we will use the same DN (so it will work for standard Samba
# release)
#
$slaveDN = $binddn;
$slavePw = $bindpasswd;
$masterDN = $binddn;
$masterPw = $bindpasswd;

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

# Login defs
#
# Default Login Shell
#
# Ex: $_userLoginShell = q(/bin/bash);
$_userLoginShell = q(/bin/bash);

#
# Home directory prefix (without username)
#
#Ex: $_userHomePrefix = q(/home/);
$_userHomePrefix = q(/home/);

#
# Gecos
#
$_userGecos = q(System User);

#
# Default User (POSIX and Samba) GID
#
$_defaultUserGid = 100;

#
# Default Computer (Samba) GID
#
$_defaultComputerGid = 553;

#
# Skel dir
#
```

```
$_skeletonDir = q(/etc/skel);

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

#
# The UNC path to home drives location without the username last extension
# (will be dynamically prepended)
# Ex: q(\\\\My-PDC-netbios-name\\homes) for \\My-PDC-netbios-name\homes
$_userSmbHome = q(\\\\MARVIN-PDC\\homes);

#
# The UNC path to profiles locations without the username last extension
# (will be dynamically prepended)
# Ex: q(\\\\My-PDC-netbios-name\\profiles) for \\My-PDC-netbios-name\profiles
$_userProfile = q(\\\\MARVIN-PDC\\profiles\\);

#
# The default Home Drive Letter mapping
# (will be automatically mapped at logon time if home directory exist)
# Ex: q(U:) for U:
$_userHomeDrive = q(F:);

#
# The default user netlogon script name
# if not used, will be automatically username.cmd
#
#$_userScript = q(startup.bat); # make sure script file is edited under dos


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

# Allows not to use smbpasswd (if $with_smbpasswd == 0 in smbldap_conf.pm) but
# prefer mkntpwd... most of the time, it's a wise choice :-)
$with_smbpasswd = 0;
$smbpasswd = "/usr/bin/smbpasswd";
$mk_ntpasswd = "/usr/local/sbin/mkntpwd";

$ldap_path = "/usr/bin";
$ldap_opts = "-x";
$ldapsearch = "$ldap_path/ldapsearch $ldap_opts -h $slaveLDAP -D '$slaveDN' -w
'$slavePw'";
$ldapsearchnobind = "$ldap_path/ldapsearch $ldap_opts -h $slaveLDAP";
$ldapmodify = "$ldap_path/ldapmodify $ldap_opts -h $masterLDAP -D '$masterDN' -w
'$masterPw'";
$ldappasswd = "$ldap_path/ldappasswd $ldap_opts -h $masterLDAP -D '$masterDN' -w
'$masterPw'";
$ldapadd = "$ldap_path/ldapadd $ldap_opts -h $masterLDAP -D '$masterDN' -w
'$masterPw'";
$ldapdelete = "$ldap_path/ldapdelete $ldap_opts -h $masterLDAP -D '$masterDN' -w
'$masterPw'";
$ldapmodrdn = "$ldap_path/ldapmodrdn $ldap_opts -h $masterLDAP -D '$masterDN' -w
'$masterPw'";
```

```
1;

# - The End
```

Configuring Linux

You need to tell your Linux box to use LDAP (pam ldap and nss ldap). Then, you should run nscd and finish your system LDAP configuration.

pam ldap, nssldap and nscd

Use ’authconfig’ (in the terminal) to activate pam ldap :

- Cache Information
- Use LDAP
- don’t select ’Use TSL’
- Server: 127.0.0.1
- Base DN: dc=SAMPLE,dc=ORG
- Use Shadow Passwords
- Use MD5 Passwords
- Use LDAP Authentication
- Server : 127.0.0.1
- Base DN: dc=SAMPLE,dc=ORG

Cache Information means you’re using nscd (man nscd for more info) : if you’re going to use pam ldap and nss ldap, you should really use it for optimization.

Warning: If you screw this up you may be forced to boot into single-user mode in order to undo it.

/etc/ldap.conf

Edit your /etc/ldap.conf file to configure your LDAP parameters :

```
# /etc/ldap.conf for using local LDAP server for authentification

# Your LDAP server. Must be resolvable without using LDAP.
host 127.0.0.1

# The distinguished name of the search base.
base dc=SAMPLE,dc=ORG

# RFC2307bis naming contexts
# we use ?sub (and not the default ?one) because we
# separated sambaAccounts on ou=Computers,dc=SAMPLE,dc=org
# and ou=Users,dc=SAMPLE,dc=org
nss_base_passwd dc=SAMPLE,dc=ORG?sub
nss_base_shadow dc=SAMPLE,dc=ORG?sub
nss_base_group ou=Groups,dc=SAMPLE,dc=ORG?one

ssl no
pam_password md5

# - The End
```

Test your system

To test your system, we'll create a system account in LDAP (say 'testuser'), and will try login as this new user.

To create a system account in LDAP, use the smbldap-tool named smbldap-useradd.pl (should be in your path, but is located at /usr/local/sbin…if not)

```
[root@pdc-srv tmp]# smbldap-useradd.pl -m testuser1
```

adding new entry "uid=testuser1,ou=Users,dc=SAMPLE,dc=ORG" (this line does not show up in my terminal, but may in yours…in any case it does work ok)

```
[root@pdc-srv tmp]# smbldap-passwd.pl testuser1

Changing password for testuser1
New password for user testuser1:
Retype new password for user testuser1:
all authentication tokens updated successfully
```

Now, try to login on your system (Unix login) as testuser1 (using another console, or using ssh). Everything should work fine :

```
[user@host-one:~]$ ssh testuser1@pdc-srv

testuser1@pdc-srv's password:

Last login: Sun Dec 23 15:49:40 2001 from host-one

[testuser1@pdc-srv testuser1]$ id
uid=1000(testuser1) gid=100(users) groups=100(users)
```

Dont forget to delete this testuser1 after having completed your tests:

```
[root@pdc-srv]# smbldap-userdel.pl testuser1
```

Configuring Samba

Here, we'll configure Samba as a Primary Domain Controller for the Microsoft Windows NT Domain named SAMPLE-NT with the SAM database stored in our OpenLDAPserver.

Configuration

We need to configure /etc/samba/smb.conf like in the example below, assuming that :

- Our Microsoft Windows NT Domain Name will be : SAMPLE-NT
- Our server Netbios Name will be : SAMPLE-PDC
- Our server will allow roving/roaming profiles
- All samba share will rely on /opt/samba/* excepted for home directories (always on /home/USERNAME).
- We really want our Samba-LDAP PDC server to be the domain browser on the LAN.

Before beginning this section you must create some directories, according to your/etc/smb.conf use the following commands :

```
mkdir /opt/samba
mkdir /opt/samba/netlogon
mkdir /opt/samba/profiles
chmod 1757 /opt/samba/profiles
```

Edit your/etc/samba/smb.conf like in the example below to configure your Samba server.

Some remarks about this file:

The global section

This section allows you to configure the global parameters of the server. Here are all the parameters we defined in the previous paragraph. We also have defined the program used for a user to change his password (passwd program) and the dialog used between the server and the user during the change. The option ”add user script” allow smbd to add, as root, a new machine. When a machine contacts the domain, this script is called and the new machine is added to the domain. This makes the administration of a machine's account easy. For security, not all the machines could logged to the domain, but only a administrator who has a privilege account. For French users, we added a line that allows smbd to map incoming filenames from a DOS code page. This option is very useful if you want the files and directories in your profiles saved with all the accents they have. Don't forget to read the man page for more details: this option is a Western European UNIX character set. The parameter client code page MUST be set to code page 850 in order for the conversion to the UNIX character set to be done correctly.

```
[global]
workgroup = SAMPLE-NT
netbios name = SAMPLE-PDC
server string = SAMPLE-LDAP PDC Server

passwd program = /usr/local/sbin/smbldap-passwd.pl -o %u
passwd chat = *new*password* %n\n *new*password* %n\n *successfully*
unix password sync = Yes

; SAMBA-LDAP declarations
ldap suffix = dc=SAMPLE,dc=ORG
ldap admin dn = cn=Manager,dc=SAMPLE,dc=ORG
ldap port = 389
ldap server = 127.0.0.1
ldap ssl = No
add user script = /usr/local/sbin/smbldap-useradd.pl -m -d /dev/null -g 1000 -s /bin/false %u

character set = iso8859-1
```

The shares section

Here are all the share sections. In particular, we can define all the user's home directories which are defined by the [homes] section:

```
[homes]
comment = Home Directories
valid users = %S
read only = No
create mask = 0664
directory mask = 0775
browseable = No
```

The profiles section

Next is the path to the profile's directory. Profiles of all users will be stored here. This is the root directory for profiles and the ldap variable profilePath specifies exactly the path for each user. For example if the profilePath is set to \\SAMPLE-PDC\profiles\testuser, then the profile directory for user testuser is /opt/samba/profiles/testuser/. Make sure to have the right permissions for this directory. The sticky bit must be set. Make a simple

```
chmod 1757  /opt/samba/profiles
```

(detailed above when you created the directories) and it will be ok. Don't forget that the system doesn't reflect this change immediately. You should wait several minutes before any profile takes place.

```
[profiles]
path = /opt/samba/profiles
writeable = yes
browseable = no
create mode = 0644
directory mode = 0755
guest ok = yes
```

The netlogon section

If you want command's file to be downloaded and ran when a user successfully logs in, you have to define a netlogon section and a netlogon script. The netlogon script must take place in the global section and the script must be a relative path to the [netlogon] service. For example, if the [netlogon] service specifies a path of/opt/samba/netlogon (like in our example), than if the script is defined as logon script = STARTUP.BAT, then the file that will be downloaded is /opt/samba/netlogon/ STARTUP.BAT.

For an example of more you can do we defined a doc section that authorized everybody to browse the / usr/share/doc documentation directory.

```
[global]
...
logon script = STARTUP.BAT (add this to the global section)
...

[netlogon]
comment = Network Logon Service
path = /opt/samba/netlogon
guest ok = Yes
[doc]
path=/usr/share/doc
public=yes
writable=no
read only=no
create mask = 0750
guest ok = Yes
```

For example, we could have the STARTUP.BAT script that set the documentation directory mounted on the J volume on windows clients. Another useful command set windows time synchronized to the server's one:

```
NET USE F:\\SAMPLE-PDC\home
NET USE J: \\SAMPLE-PDC\doc
NET TIME \\SAMPLE-PDC /SET /YES
```

The Startup.bat file MUST be created in DOS or Windows notepad or it will not work!

Initial entries

Samba must know the passwd of the ldap admin dn(cn=Manager,dc=SAMPLE,dc=ORG) you’ve specified in smb.conf to be able to add/modify accounts stored in the LDAP SAM.

To do so, use the following command (assuming ’secret’ is the ldap admin dn password, see your /etc/openldap/slapd.conf configuration file to be sure):

```
[root@pdc-srv samba]# smbpasswd -w secret
```

```
Setting stored password for "cn=Manager,dc=SAMPLE,dc=ORG" in secrets.tdb
```

Samba will store this data in /etc/samba/secrets.tbd.

Now, you should create your’Administrator’ user:

First let’s create the Domain Admins group

```
[root@pdc-srv samba]# smbldap-groupadd.pl -g 200 Domain Admins
```

Now, let’s actually create your’Administrator’ user:

```
[root@pdc-srv samba]# smbldap-useradd.pl -a -m -g 200 administrator
adding new entry "uid=administrator,ou=Users,dc=SAMPLE,dc=ORG"
modifying entry "uid=administrator,ou=Users,dc=SAMPLE,dc=ORG"  these may not show up
modifying entry "uid=administrator,ou=Users,dc=SAMPLE,dc=ORG"
```

```
[root@pdc-srv samba]# smbldap-passwd.pl administrator
```

```
Changing password for administrator
New password :
Retype new password :
all authentication tokens updated successfully
```

In fact, any user placed in the ”Domain Admins” group will be granted Windows admin rights.

Testing

To validate your Samba configuration, use testparm who should return ’Loaded services file OK.’ without any warnings nor unknown parameters. See man testparmfor more info.

Start-Stop servers

Assuming you're following this HOW-TO, we use:

start/stop the OpenLDAPserver : /etc/init.d/ldap start/stop
start/stop the Sambaserver : /etc/init.d/smb start/stop
start/stop the nscdserver : /etc/init.d/nscd start/stop

other methods such as ***service ldap start*** will work as well.

Create a Computer account

To create a computer account, you can use smbldap-tools to manually add accounts:

```
[root@pdc-srv root]# smbldap-useradd.pl -w testcomputer1

modifying entry "uid=testcomputer1$,ou=Computers,dc=IDEALX,dc=ORG"
```

You can also use the automatic procedure within you Microsoft Windows client (I have been unable to get this to work…let me know if you figure it out) To make things easier for me in a lab situation O created a simple shell script to add multiple machines:

```
#!/bin/sh
smbldap-useradd.pl -w labcomputer1
smbldap-useradd.pl -w labcomputer2
smbldap-useradd.pl -w labcomputer3
smbldap-useradd.pl -w labcomputer4
and so forth
```

I called it machineadd.sh and run it from the terminal ./machineadd.sh

Make your own and try it out…it's much faster.

Delete a Computer account

To delete a computer account, just use smbldap-tools:

```
[root@pdc-srv root]# smbldap-userdel.pl testcomputer1
```

Instead of removing the computer account, you may want to de-activate the Samba Account.
To do that, use an LDAP browser and modify the acctFlags from [W] to [WD] ('D' indicating 'Disabled'). To re-activate the computer account, just modify [WD] to [W]. Sometimes, de/re-activation is a better means to temporary disable the workstation.

Roaming/Roving profiles

When a Microsoft Windows NT/2K/XP user joins the SAMPLE-NT domain, his profile is stored in the directory defined in the profile section of the samba configuration file. He has to log out for this to be saved. This is a roaming profile: he can use this profile from any computer he wants, hence the name “roaming profiles”. Roaming profiles are very useful as they consist of user data such as “Favorites”, “History”, desktop wallpaper, “My Documents”, and more. If a users personal configuration changes, it will be integrated in his roaming profile.

In this Howto, we used roaming profiles: the LDAP ProfilePath indicate to Samba where to look for those roaming profiles…example: (SAMPLE-PDC\profiles\testsmbuser2)

and the [profiles] section of the /etc/samba/smb.conf indicates to samba how to

Mandatory profiles

The mandatory profile is created the same way as the roaming profile. The difference is that this profile is made ‘read only’ by the administrator so that the user can have only one fixed profile on the domain.

To do so, rename the file NTuser.dat to NTuser.man (for MANdatory profile), and remove the right access bit. For our testsmbuser1 user, you’ll have to do:

```
mv /opt/samba/profiles/testsmbuser1/NTUSER.DAT

/opt/samba/profiles/testsmbuser1/NTUSER.MAN

chmod -w /opt/samba/profiles/testsmbuser1/NTUSER.MAN
```

A Mandatory profile allows you to set up a common user profile for every user on the Domain…useful to give everyone a common starting point or for specifying specific settings without having to do so for each user.

Logon Scripts

To use Logon Scripts (.BAT or .CMD), just specify the relative path from the netlogon share to the command script desired in the scriptPathattribute for the user. This is done easily in the program Directory Administrator.

NFS Exports

We will set up NFS to export the home directories so Linux users will access the same home directories as Windows/Samba users.

1. Add the following line to /etc/exports

   /home 192.168.0.0/24(rw)   use your own IP addresses on this line

2. Restart the NFS services

service nfs restart

Another method:

On your file/ldap server, add the following lines to /etc/exports

```
/home  10.1.2.3/255.255.255.255(rw)
```

Replace "10.1.2.3" with the IP address of the "client" server. You can spec a whole range of course, such as "10.0.0.0/255.0.0.0" for all of the 10.x.x.x addresses.

Now run "exportfs -a".

Also double-check that you are not firewalling off access to NFS & portmap

(UDP ports 111 & 2049).

Client Set-Up

Linux Client

To set up a linux client to use the primary domain controller for authentication and home directory, do the following:

Automounter

Configure the automounter to mount the home directories as needed.

1. Edit /etc/auto.master add:
2. `/home      /etc/auto.homes      --timeout=60`
3. Edit /etc/auto.homes
4. `*          192.168.0.1:/home/&      you should use your own IP addresses`
5. Restart the automounter:

   service autofs restart

Authconfig

The authconfig program is a semi-graphical setup for the Pluggable Authentication Modules (PAM) subsystem. Set the following options in authconfig:

1. Cache Information

2. Use LDAP (Server: 192.168.0.1, Base DN: dc=sample,dc=org)

3. Use shadow passwords

4. Use MD5 passwords

5. Use LDAP authentication

Windows 98

Adding a windows 98 box to the domain is exactly the same as adding a windows 98 box to a NT/ 2000/XP domain.

1. Open the network control panel

2. Select ``client for Microsoft networks" and click ``properties."

3. Select ``log on to windows NT domain"

4. Enter domain name (SAMPLE-NT)

5. Restart the computer

LDAP or not LDAP?

Perhaps, you'll want to use an alternative system policy concerning profiles : granting some user the roaming profile privilege across the domain, while some other may have only roaming profiles on one PDC server, and some other won't use roaming profiles at all. This alternative way is possible thanks to Samba which will search in the LDAP sambaAccount for the profile location if no information is given by the 'logon drive', 'logon script' and 'logon path' directives of smb.conf.

RequireSignOrSeal

This registry key (gathered from the Samba-tng lists) is needed for Windows 2000 and XP clients to join and logon to a Samba domain :

It is suggested that you check the following registry entries which should be set to (0). This is the default under W2K (but check to confirm) however under XP the default is (1) and definitely needs changing:

```
[HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services\Netlogon\Parameters]
"requirestrongkey"=dword:00000000
"requiresignorseal"=dword:00000000
[HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Services\Netlogon\Parameters]
"requirestrongkey"=dword:00000000
"requiresignorseal"=dword:00000000
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Netlogon\Parameters]
"requirestrongkey"=dword:00000000
"requiresignorseal"=dword:00000000
```

You can change this in the Local or Domain policy editor in Windows 2000/XP. (regedit)

Fake user root

To allow Microsoft Windows 2000 and XP workstations to join the domain, a root user must exit (uid=0) and be used when joining a client to the domain

To create this false user (false because the user root should be present on you're system files, not in LDAP), just issue the following commands:

```
smbldap-useradd.pl -a -m -g 200 root
smbldap-usermod.pl -u 0 -g 0 root
smbldap-passwd.pl root
```

More Information

1. Excellent and more complete documentation on building a production Samba/LDAP server is available at:
   http://www.idealx.org/prj/samba/index.en.html

2. There are some graphical tools for administering SambaLDAP accounts.

   - There is a webmin module available from:
     http://webmin.idealx.org

   - Webmin itself is available from:
     http://www.webmin.com/

   - gq - a graphical ldap browser
     http://biot.com/gq

3. Another how-to document is posted at:
   http://network.gouldacademy.org/

4. More useful sites:
   http://www.samba.org/
   http://www.openLDAP.org/

5. Information about the K12LTSP distribution is available at http://www.k12ltsp.org

Special thanks to:

Derek Dresser of Gould Academy in Bethel, Maine USA for his fantastic quickstart how-to located at http://network.gouldacademy.org

Olivier Lemaire for his Samba/LDAP document at idealx.org (much of this document comes from that one)

Eric Harrison and Paul Nelson for their work on the K12LTSP distribution and their suggestions on getting LDAP/Samba to work with K12LTSP

And everyone else in the awesome Open Source Community!