Set up Samba to serve as a domain controller

By Jack Wallen
April 27, 2012, 5:09 PM PDT

Takeaway: Samba can serve as an NT4-style domain controller. Jack Wallen shows how to do this on a Ubuntu Server 11.04 machine.

Samba is one of the most versatile tools in all of Linux/Unix/Mac land. It enables those machines to connect to Windows shares and other heterogeneous services/machines.

Samba can also serve other, more advanced needs such as that of a domain controller. A domain controller is a machine that serves as a responder to security authentication requests. This is usually handled by a Microsoft Windows-based machine, but for those who want the security and reliability of a Linux machine handling this task, it's completely (and rather easily) possible.

Samba cannot act as an Active Directory Primary domain controller; it can serve as an NT4-style domain controller. In this tutorial, I describe how this can be done on a Ubuntu Server 11.04 machine.

Step 1: Install Samba

Follow these steps:

1. Open a terminal (or log in to the machine if it's GUI-less).
2. Issue the command *sudo apt-get install samba libpam-smbpass*.
3. Type the sudo password and hit Enter.
4. Accept any dependencies.

Step 2: Configure Samba

There is quite a lot of configuration to tackle, so we'll take this section by section. You should open the /etc/samba/smb.conf file in your favorite editor (mine is nano) and edit the following sections accordingly.

Globals

You only need to modify these directives:

- security =
- Workgroup =

Security must be set to users, and Workgroup can be set to any descriptive word for your organization.

Domains

In this part of the configuration file, look for this section:

```
domain logons = yes
logon path = \\%N\%U\profile
logon drive = H:
logon home = \\%N\%U
logon script = logon.cmd
add machine script = sudo /usr/sbin/useradd -N -g machines -c Machine -d
/var/lib/samba -s /bin/false %u
```

Here are the descriptions for each directive:

- **domain logons:** Provides the netlogon service that makes Samba act as a domain controller.
- **logon path:** Places the user's Windows profile into their home directory.
- **logon drive:** Specifies the home directory local path.
- **logon home:** Specifies the home directory location.
- **logon script:** Determines the script to be run locally once a user has logged in. **Note:** This script must be placed in the [netlogon] share (more on this in a bit).
- **add machine script:** Automatically creates the Machine Trust Account needed for a workstation to join the domain. **Note:** The machines group must be added to the new domain controller for this to work. Use the addgroup command to create this group.

Share definitions

There are two sections to focus on here: [homes] and [netlogon]. The [homes] section should be uncommented and look like this:

```
[homes]
comment = Home Directories
browseable = no
read only = no
create mask = 0700
directory mask = 0700
valid users = %S
```

The netlogon section should look like this:

```
[netlogon]
comment = Network Logon Service
path = /srv/samba/netlogon
guest ok = yes
read only = yes
share modes = no
```

The path for the [netlogon] section probably doesn't exist on your machine, so you'll need to create it with this command: *sudo mkdir -p /srv/samba/netlogon*.

The logon.cmd script must be referred from the [netlogon] section. That script does not exist yet, so create an empty file to be used with the command: *sudo touch /srv/samba/netlogon/logon.cmd*.

As for the content of the logon.cmd script, any Windows logon script command will work. For more information on logon scripts, read Brian Posey's TechRepublic article "Simplifying network options through logon scripts."

Step 3: Restart Samba

Issue these two commands to restart the necessary services:

```
sudo restart smbd
sudo restart nmbd
```

Step 4: Configure rights

For security purposes, root is disabled by default. Because of this, a system group needs to be mapped to the Windows Domain Admin group; this is done with the Linux net command like so: *sudo net groupmap add ntgroup="Domain Admins" unixgroup=sambaadmin rid=512 type=d*.

The group sambaadmin probably doesn't exist, so you'll need to create it. You will also need to add whatever user will be used to join the domains to this new group (as well as to the admin and sudo group). That user will also need Samba credentials, which is done with this command *sudo smbpasswd -a username* (where username is the name of the user who will be used to join the domains).

Finally, rights need to be explicitly provided to the Domain Admins group to allow the add machine script (and other admin functions) to work. This is accomplished by issuing this one command (where EXAMPLE is the actual domain): *net rpc rights grant -U sysadmin "EXAMPLE\Domain Admins" SeMachineAccountPrivilege SePrintOperatorPrivilege SeAddUsersPrivilege SeDiskOperatorPrivilege SeRemoteShutdownPrivilege*.

You should now be able to join your Windows clients to the domain