

Re: Help me replace some Windows installations

Source: <http://linux.derkeiler.com/Newsgroups/comp.os.linux.setup/2004-09/0410.html>

From: Abdullah Ramazanoglu (abdullah_at_ramazanoglu.tr)

Date: 09/13/04

Date: Mon, 13 Sep 2004 14:12:08 +0300

begin ZnU <znu@acedsl.com> dedi ki:
> In article <hs9f12-025.ln1@triangulo.it.uc3m.es>,
> ptb@oboe.it.uc3m.es (P.T. Breuer) wrote:
>> ZnU <znu@acedsl.com> wrote:
--8<--

Again set follow up to c.o.l.misc

> Look, I don't understand what the miscommunication is here. Basically,
> what I want to do, is let a user log in and have appropriate access to his
> home directory, and appropriate lack of access to the home directories of
> other users. The way Windows and OS X handle this is by mounting home
> directories as share points using the authentication information for the
> user logging in on each machine. In other words, if I log in as 'znu' on
> Client-1, Client-1 logs into the file server using the 'znu' account, and
> the 'Users' share is mounted with the privileges defined for 'znu' on the
> server.

Possible with untrusted clients in SMB, and trusted clients in NFS.

> With NFS, my understanding is that I'd create a system-wide entry to mount
> the 'Users' share at, say, /home. If I did that, I'd have to grant the
> client full read-write access on the entire share, because it has to be
> able to grant the appropriate access to any user who might sit down and
> log in, and that potentially requires being able to read from and write to
> any home directory. I then have to trust the client to make sure that user
> A doesn't delete files from user B's home directory or whatever. I can't
> necessarily trust every client that might be connected to this network.

Subtly wrong. You have to trust the *administrator* of the client machines. There's a misconception of "trusting a user on the client" and "trusting the administrator of the client". If you administer both clients and servers, then there's no risk of user X accessing home of user Y. It's because with NFS (plain NFS) granting access to a remote machine just enables NFS sharing for that machine in general, but it doesn't allow for *any* user on the client to access any exported directory. Still, user ABC on client, accesses to server with rights of the user ABC on server. This scheme is only risky if someone can install his own Linux client and,

being root of his own machine, create arbitrary user accounts on this client, which enables him to access to NFS server as any user he chooses. Plain NFS is to be used in trusted-administrator environments, historically. If you can't trust the admin of a remote machine, then you don't grant NFS access to that machine. If you can't trust your LAN environment, then don't use plain NFS. Either use some sort of secured NFS, like kerberized NFS which Peter already told you about, or use SMB for Linux to Linux shares. NFS is not Linux, like SMB is not Windows. They're just RFC'ed file sharing protocols.

>> > *OpenDirectory is basically OpenLDAP with an Apple schema that provides everything OS X needs. I *think* this schema is a superset of RFC 2307 (which is what I'd want for Linux clients, right?), but it's hard to*
>>
>> *I have no idea. If you have an ldap server, then linux can use it for authentication. It's a question of putting a few entries in the pam.d files, and arranging that nsswitch.conf is rigged to refer libc getpwent through ldap for the passwords.*
>>
> *See, this is the kind of thing that makes people say Linux is not ready for prime time.*

You've started this thread with "I've got a chance to replace 30 Windows machines, please help me..." and now you seem to be more interested in making comparisons rather than get help and get going. I don't know what are you after. Just argue for the sake of argument? A cat in the bag? If you are really interested in a solution, then why are you keeping on posting to c.o.l.advocacy, which has nothing to do with your technical questions, and why didn't you replied to my post in c.o.l.misc?

> *With OS X clients, this same procedure requires *no* client-side configuration, aside from clicking a single checkbox to tell the machine to obtain LDAP server information automatically from DHCP.*
>
> *With Windows clients, the equivalent procedure involves running a wizard and filling in a couple of obvious things in its fields.*

As for the comparisons, you don't have to do anything on a decent Linux distro to enable LDAP based authentication on client side. As I've told in my previous post, Mandrake installer asks you if you'd like to have it enabled, and does the rest automatically without even you having to know what LDAP is.

>> > *find documentation for this sort of thing. I was hoping someone here had*
>>
>> *All the docs are on the ldap site for linux (and presumably in the howto).*
>>
>> > *done this and could tell me.*
>> >

comp.os.linux.setup: Re: Help me replace some Windows installations

>> > *I've managed to Google up lots of information about using a Linux
>> > server with OS X clients, but I've found practically nothing about
>> > doing things the other way around.*
>>
>> ???

The question remains open though. Why you would want to employ a client grade machine for server tasks, while it is natural and excellent practice to employ Linux there. As long as you use OS X as the server, the only remaining question is how to introduce Linux clients into this environment (your original question). And the answer is:

- Use smbclient facilities (smbmount) on Linux clients to access the homes as SMB shares on OS X.
- Use LDAP authentication on Linux clients to authenticate against LDAP on OS X.

The rest was meant to help you in implementing server side on Linux. A wise move afterall. But if don't want to do it, then why argue about it?

--
Abdullah | aramazan@ |
Ramazanoglu | myrealbox |
_____ | D.O.T cöm |__