

Re: Sync AD with System.DirectoryServices

Source:

http://www.tech-archive.net/Archive/Windows/microsoft.public.windows.server.active_directory/2006-08/msg02364

- *From:* "Joe Kaplan \ (MVP – ADSI)" <joseph.e.kaplan@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Sat, 19 Aug 2006 16:44:04 –0500
-

You can't really do this. There is no way to access the user's password programmatically like that. There are some third party products out there that can capture them as they are changed and store them in other data stores (I think psynch does this, along with its normal function of self service password reset), but that is really the only option here.

Note also that it is pretty hard to do as good of a job at securing the passwords in a different store as AD or ADAM does. You have to put some serious effort into this in order to not compromise your security.

However, I think you might be better off considering creating a disaster recovery plan around AD. Perhaps you could replicate AD to another location off site and switch over that AD instance in the event of a disaster? The great advantage here is that you are still using AD, so you don't necessarily have to change all of your applications (and all of the other things that might depend on AD, which can be a LOT, depending on how you use it).

You might consider starting a different thread here regarding some approaches for disaster recovery, as a lot of the other experts in this group are very familiar with that type of thing.

Best of luck,

Joe K.

—
Joe Kaplan—MS MVP Directory Services Programming
Co—author of "The .NET Developer's Guide to Directory Services Programming"
<http://www.directoryprogramming.net>
—

<jason.michael.perry@xxxxxxxxxx> wrote in message
news:1156010016.631201.84980@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

I have a client running AD internally for user authentication. They want to create a single sign on system that syncs/duplicates AD in to SQL server on an external server.

Re: Sync AD with System.DirectoryServices

The idea here is to still allow access to the Intranet (hosted externally) in the case of a natural disaster (hurricane).

I can easily get all the user data I need from this using System.DataServices. However, this approach does not give me access to the user's password. I need a way to authenticate a user if AD dies...any ideas?