AUTOMATION

Creating two users in a HR dirctory using dsadd

dsadd user "CN=Fred Clarke,OU=HR,DC=etechd,DC=com" -samid Fred -upn Fred@etechd.com -fn Fred -ln Clarke -pwd Password14 -mustchpwd yes -canchpwd yes -disabled no

dsadd user "CN=Ke Clark,OU=HR,DC=etechd,DC=com" -samid Ken -upn ke@etechd.com -fn Ke -ln Clarke -pwd Password13 -mustchpwd yes -canchpwd yes -disabled no

Using ldifde to automate users in the HR OU

Enter the data below into notepad and save as ldifde file

Use your own domain name in place of etechd

==========

dn: CN=Karl Foster, OU=Department,DC=etechd,DC=COM
changetype: add
cn: Karl Foster
objectClass: user
samAccountName: KFoster

dn: CN=Fabian Alleyne, OU=Department,DC=etechd,DC=COM
changetype: add
cn: Fabian Alleyne
objectClass: user
samAccountName: Fabian

dn: CN=Paul Lashley, OU=Department,DC=etechd,DC=COM
changetype: add
cn: Paul Lashley
objectClass: user
samAccountName: Plashley

Save your file as source.ldf at the root of your Server directory. Example

Example

M:\>source.ldf

To import the usernames

At your command prompt type

Ldifde –i –f source.ldf

(i stands for import, f stands for file)

ADDING NAMES USING CSVDE

USE CSVDE TO AUTOMATE YOUR OWN ENTRY

Enter the information below in an excel file and save as a CSV file.

objectClass	sAMAccountName	dn
user	Petergr	CN=Peter Graham,OU=Newport,DC=cp,dc=com
user	Janiebo	CN=Janie Bourne,OU=Newport,DC=cp,dc=com
user	Edgardu	CN=Edgar Dunn,OU=Newport,DC=cp,dc=com
user	Belindaha	CN=Belinda Hart,OU=Newport,DC=cp,dc=com
user	Mayja	CN=May Jamieson,OU=Newport,DC=cp,dc=com
user	Leroyot	CN=Leroy Ota,OU=Newport,DC=cp,dc=com

To import the names in AD use the following command

Csvde –i –f source.csv

Csvde limitations

Csvde is unable to to add passwords

I use csvde to import my user objects then I use dsmod to put in everything else.

For example

dsmod user " CN=Peter Graham,OU=Newport,DC=cp,dc=com " -pwd Password01 -mustchpwd -disabled no

*Tips

- To convert columns to text use the following formula

  =A1 & "," & A2 & "," & A3 And so on

- To convert text to columns in excel use Data, text to columns

Use Windows PowerShell to create a new user account in Active Directory and enable the new account.

When you use the **New-ADUser** cmdlet to create a user account, the new account is disabled and cannot be enabled unless either of the following has occurred:

- A valid password has been set for the account.
- The **-PasswordNotRequired** parameter has been set to true.

To create a user account for Phil Gibbins, specify a password, and enable the new account, you can use the following command:

```
PS C:\> New-ADUser -Name "Phil Gibbins" -GivenName Phil -Surname Gibbins `
-SamAccountName pgibbins -UserPrincipalName pgibbins@corp.contoso.com `
-AccountPassword (Read-Host -AsSecureString "AccountPassword") `
-PassThru | Enable-ADAccount
```

The **-PassThru** parameter, which has been added to the **New-ADUser** command just shown, returns the newly created user account object so that it can be piped into the **Enable-ADAccount** cmdlet to enable the new account

Creating and managing user accounts is a common Active Directory administration task. Windows PowerShell provides considerable flexibility in how this can be done on the Windows Server 2012 platform. Typing **Get-Command *ADUser** at a Windows PowerShell prompt shows there are four cmdlets for managing users accounts:

- **New-ADUser** Creates a new Active Directory user
- **Get-ADUser** Gets one or more Active Directory users so that you can perform some action with them
- **Set-ADUser** Modifies the properties of an existing Active Directory user
- **Remove-ADUser** Removes the specified user from Active Directory

Any administration of user accounts using Windows PowerShell involves using one or more of these cmdlets

Create a user

To be able to generate and create hundreds of users first always try to create only one, to see you have the necessary permissions, network connectivity, etc.

There are lots of AD properties available from this command so let’s check a couple.

AccountPassword: Have to provide a SecureString here, simple plaintext won’t be enough. Use the ConvertTo-SecureString command to store submit your password or convert it on the fly:

ConvertTo-SecureString “MyPassword1!” -AsPlainText -Force

ChangePasswordAtLogon: For test users I always use $false so I can log on with them without any hassle.

Enabled: I always use $true, so I can use them immediately.

Path: This defines the OU where the user will be created. I you omit this, Windows uses the default user container. Submit a distinguished name here. Check the OU with the attribute editor in AD Administrative Center:

Company, Title and MobilePhone, etc are pretty straightforward, but I always struggle with names so here is a rough overview.

New-ADUser property name	AD property on the GUI (ADAC)	LDAP attribute
DisplayName	Display name	displayName
GivenName	First name	givenName
Initials	Middle initials	initials
Name	Full name	name
OtherName	-	middleName
SamAccountName	User SamAccountName logon	sAMAccountName
Surname	Last name	sn

With this in hand an example user creation goes like this:

New-ADUser
-AccountPassword (ConvertTo-SecureString “MyPassword1!” -AsPlainText -Force)
-ChangePasswordAtLogon $false
-City London
-company “Letitknow Ltd.”
-DisplayName “Smith, John”
-Enabled $true
-MobilePhone “+1 11 555 5555″
-Name “Smith, John”
-SamAccountName smithj
-Title CFO
-Path “OU=Users,OU=Company,DC=home,DC=local”
-givenname John

-surname Smith
-userprincipalname (“smithj” + “@home.local”)
-department “IT”
-description “My created user”
-office “HQ”

Bulk User Creation with Powershell

Import-Csv .\mycsv.csv | foreach-object {$userprinicpalname = $_.SamAccountName + "@DeanLashley.com"; New-ADUser -SamAccountName $_.SamAccountName -UserPrincipalName $userprinicpalname -Name $_.name -DisplayName $_.name -GivenName $_.cn -SurName $_.sn -Department $_.Department -Path "CN=users,DC=DeanLashley,DC=com" -AccountPassword (ConvertTo-SecureString "Microsoft~1;" -AsPlainText -force) -Enabled $True -PasswordNeverExpires $True -PassThru }

Name	samAccountName	cn	sn	Description	Department	Employee	Pat
Peter Parris	PParris	Peter	Parris				cn=
Veronica Belgrave	VBelgrave	Veronica	Belgrave				cn=

Save the above excel File as a csv file

Import-Csv .\usercreationfile.csv | foreach-object {
$userprinicpalname = $_.SamAccountName + "@{domainname}.com" ;
New-ADUser –SamAccountName $_.SamAccountName -UserPrincipalName $userprinicpalname -Name $_.name -DisplayName $_.name -GivenName $_.cn -SurName $_.sn -Department $_.Department -Path "CN=Users,DC=biogen,DC=com" -AccountPassword (ConvertTo-SecureString "Microsoft~1;" -AsPlainText -force) -Enabled $True -PasswordNeverExpires $True -PassThru }

Run the script above in powershell for Active Directory.