Lab 4: User Management

Server System Management - Windows Server Labs

<STUDENT NAME>

Academic Year: 2021-2022

Table of Contents

[Introduction 3](#_Toc96899265)

[Learning Goals 3](#_Toc96899266)

[Knowledge (what you need to know) 3](#_Toc96899267)

[Abilities (what you need to be able to do) 3](#_Toc96899268)

[Requirements 3](#_Toc96899269)

[Accessing Active Directory 4](#_Toc96899270)

[Creating & Managing Users & Groups 4](#_Toc96899271)

[Extensions - Optional Assignments 7](#_Toc96899272)

## Introduction

In this lab we will populate our AD environment with users and groups. In the first part we try and enumerate all information in our Windows environment. Afterwards we will add more users (both manually and with some sort of automation).

## Learning Goals

# Knowledge (what you need to know)

* What (user & computer) information can you enumerate in an AD domain.
* What attributes are required when creating users & groups

# Abilities (what you need to be able to do)

* CRUD operations on users & groups through the GUI and with PowerShell
* Enumerating information when having access to an AD domain

## Requirements

The three virtual machines, fully configured as they should be at the end of Lab 3 – installing an AD domain.

## Accessing Active Directory

In this part of the lab you will learn to access and consult objects from Active Directory using the ADU&C console and PS-cmdlets. The best practice is to do everything from your Windows 11 client and logged in as the domain administrator, but you are free to do this first part of the lab on your GUI server as well to save resources (the other virtual machines are not required for now).

1. We will start by **enumerating** information from Active Directory through a GUI tool. Open the ADU&C tool and search for the following objects:   
   1. All user(groups) whose name starts with the letter “a”

Graphical user interface, application

Description automatically generated

* 1. All computers

Graphical user interface, text, application, email

Description automatically generated

* 1. All disabled accounts

Graphical user interface, text, application, email

Description automatically generated

* 1. All members of the Administrators Group

Graphical user interface, text, application

Description automatically generated

1. Activate the “advanced features” view in the AD U&C window to enumerate **all attributes** of an object and open the properties window for the **domain administrator** account. In this window the most frequently used attributes of your account are shown. In the “**attribute editor**” tab you can also perform a filter. Try to create a filter that only shows attributes for fields that have a value. Write down the distinguished name of the domain administrator account.

CN=Administrator,CN=Users,DC=corp-ben,DC=serverlabs,DC=be

1. Use PowerShell cmdlets to perform the following exercises. Write down the cmdlets below the questions.
   1. Show all computers in your domain in a non-interactive way (i.e. without asking for additional information). In the cmdlet, use only the parameters that are strictly necessary to do this.

Get-ADComputer -Filter \*

* 1. Like most cmdlets, the cmdlet used in the previous question only shows a limited number of properties/attributes of the desired object. Adjust your previous cmdlet in such a way that **all properties** of **all computers** are displayed.

Get-ADComputer -Filter \* -Properties \*

* 1. Show all attributes of the Windows 11 computer.

Get-ADComputer -Filter 'Name -like "WIN11\*"' -Properties \*

* 1. Show only the name, SID, the IPv4 address and OS of all computers in your domain. Make sure that these properties are shown in a column form (the output). **Paste a screenshot of the output below (make sure everything is clearly visible: prompt & required properties)**

Get-ADComputer -Filter \* -Properties \* | FT Name,SID,IPv4Address,OperatingSystem

Text

Description automatically generated

## Creating & Managing Users & Groups

From now on boot at least your Windows 11 client machine and perform the following tasks from there. The core server is once again not required.

1. (Recap question as it is important 😉) Open the ADU&C console (fastest method is to use **run** > **dsa.msc**) and choose the “advanced features view”. Select the built-in domain administrator account. The custom view gives you a few extra tabs. Which tab should you click to see all attributes of the account?

Attribute Editor

1. Click on this tab. You will find that many attributes have no value. Make sure that only see the attributes that have a filled-in value. Put a screenshot here of the GUID of the built-in domain administrator account.

Graphical user interface, text, application, email

Description automatically generated

1. Use a PS-cmdlet to request a list of all domain users. Make sure that your cmdlet does not require any extra input.

Get-ADUser -Filter \*

1. If you have used the correct cmdlet, you will notice that (unless specified otherwise) a specific subset of so-called **default properties** are displayed from each domain user. Is “the number of times a domain user logged in” a default property?

No

1. Use a PS-cmdlet to show the **number** of domain users.

(Get-ADUser -Filter \*).count

1. Use a PS-cmdlet to retrieve **all** properties (default and extended properties) of the built-in domain administrator account.

Get-ADUser -Filter \* -Properties \*

1. Edit your previous command and find out how many times the built-in domain administrator account has already logged in (this may take a moment).

Get-ADUser -Filter 'Name -like "Administrator\*"' -Properties \* | FT logonCount

1. As you may know, the guest account is disabled by default. Activate this account using a PS-cmdlet.

Enable-ADAccount -Identity "Guest"

1. Using the GUI, verify that your command from the previous question was successful. Afterwards disable the guest account in the GUI

Disabled accounts have a down arrow.

1. Using the GUI, create the following personal domain user:
   1. First name: <firstname> (ex: Thomas)
   2. Last name: <lastname> (ex: Clauwaert)
   3. Don’t use initials
   4. User logon name: your lastname + the first letter of your first name, no spaces or special characters (ex clauwaertt)
   5. Password: Friday13th!
   6. No need to reset password
   7. Make sure the account expires on the first day of 2023
   8. Company: Howest
   9. (Tip: You might get errors from your win11. Verify/do the magic on your GUI if needed)
2. Use the GUI to verify your new user. Which groups does (s)he belongs to?

Graphical user interface, text, application, email

Description automatically generated

1. Show, using a PS-cmdlet, all groups of which the domain administrator account is a member.

Get-ADPrincipalGroupMembership Administrator | select name

1. Use the GUI to make the new domain user part of all the groups you found in the previous question.

Graphical user interface, text, application, email

Description automatically generated

1. Use a PS-cmdlet to print all groups of the new domain user. Make sure the format is the full DN name of every Group.

Text

Description automatically generated

1. Sign out on your windows 11 client and sign in using your new personal domain account. Configure your custom prompt (refer to lab 3, question 18).
2. One of the advantages of using a domain is that you can also manage (group) policies, such as password policies for all domain users. On your windows 11 client there are 2 tools to manage group policies:
   1. The **local group policy editor** – this one is typically not used in a domain setting

Text, application

Description automatically generated

* 1. **Group Policy Management** – use this one to manage group policies in the domain

Open (a) the local group policy editor and search for a policy that indicates the minimum password length of local users on the machine. Paste a screenshot of the window where you found this policy.

1. Try to adjust the minimum length in this tool. Does this work?

no

1. Now open the correct tool and try to change the minimum password length from 7 to 8. Paste a screenshot below to show where you changed this and that the length has changed to 8.

Graphical user interface, text, application

Description automatically generated

1. Since this policy is a computer configuration setting, it will not be applied until you restart your computer. However you can do it immediately by running the following legacy command

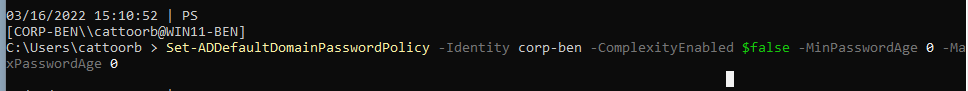
**gpupdate /force**

Now open the local group policy editor and check if the minimum length is set to 8 there as well. If not, restart your windows 11 virtual machine and test again. Finally, use a PS-cmdlet to request the default domain password policy. The result of this cmdlet should also show the desired 8. This might take some time and/or reboots.

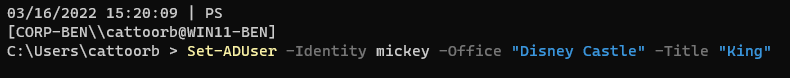
Text

Description automatically generated

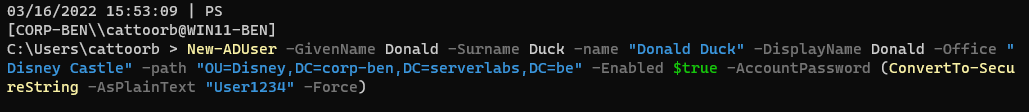
1. Because this is a test environment we will make things easier for us by reducing the complexity of passwords. This is obviously not a best practice, and should not be done in a production environment/real life scenario! In other words use a PS-cmdlet to change the default password policy in such a way that complex passwords are not needed and the minimum and maximum age is not applicable/defined.



1. Use the GUI (ADU&C and right click on your corp-name.serverlabs.be) to create a new Organizational Unit (OU) with the name **Disney** and create a new domain user account with the following properties in this OU:
   1. First name: Mickey
   2. Last name: Mouse
   3. No initials
   4. User logon name: mickey
   5. Password: Friday13th!
   6. No need to change the password
   7. Password never expires
   8. Description: “Mickey was created by <firstname>” (replace <firstname> by your own name)
   9. Office: Howest
   10. Make sure Mickey is only allowed to log in on the windows 11 client and on workdays from 8h till 18h.
   11. Make sure he is a member of the domain users group.
2. Change the following settings for the mickey user using a single PS-cmdlet.
   1. Change the Office to “Disney Castle”
   2. Add a title: “King”



1. Create a new domain user account with the following attributes, using a single (non-interactive) PS-cmdlet:
   1. First name: Donald
   2. Last name: Duck
   3. Logon name: Donald
   4. Office: Disney Castle
   5. OU: Disney
   6. Account: enabled
   7. Part of the domain users group
   8. You can use the following parameter to set the password correctly:   
      **“-AccountPassword (ConvertTo-SecureString -AsPlainText “Friday13th!” -Force)”**

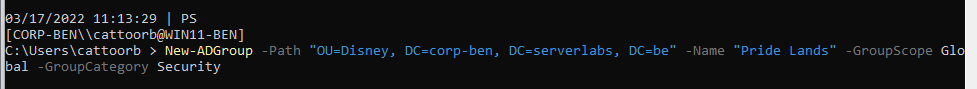


1. Create a new global security group called “Agrabah” in the Disney OU using the GUI.

Graphical user interface, application

Description automatically generated

1. Create a new global security group called “Pride Lands” in the Disney OU using a PS-cmdlet (so “Pride lands” and “Agrabah” should be in the same container/at the same level).



1. Write a PowerShell script that automatically creates the users, specified in the “disneyusers.csv” file, and adds them to the correct containers (OU “Aladdin” and/or OU “LionKing”, which are part of the “Disney” OU). You are free to pimp your script as you want (for example also the creation of the OU’s) but at the very least it should:
   1. Import the csv
   2. Pipe it to a foreach
   3. Fill in the correct attributes of the new-aduser cmdlet

Put a screenshot of the “Aladdin” OU with its users clearly visible down below

param([Parameter(Mandatory)][string] $inputfile)

$users = Import-Csv $inputfile -Delimiter ";"

foreach($user in $users) {

Remove-ADUser -Identity $user.Givenname

$Enabled = [System.Convert]::ToBoolean($user.Enabled -replace '\$')

$ChangePasswordAtLogon = [System.Convert]::ToBoolean($user.mustChangePassword -replace '\$')

$path = ($user.ParentOU -replace 'thomas', 'ben')

$mail = ($user."E-mail" -replace 'thomas', 'ben')

New-ADUser -Name $user.Name -Surname $user.Name -GivenName $user.Givenname -EmailAddress $mail -SamAccountName $user.samAccountName -Path $path -AccountPassword (ConvertTo-SecureString -AsPlainText $($user.Password) -Force) -Enabled $Enabled -ChangePasswordAtLogon $ChangePasswordAtLogon -Description $user.Description

}

Graphical user interface, text, application

Description automatically generated

1. Use a PS-cmdlet (and piping) to add all users in the Aladdin OU to the Agrabah Security group. Tip: A possible solution is to enumerate and filter with get-aduser and pipe it to a foreach loop followed by adding the user to the group.

Get-ADUser -SearchBase ‘OU=Aladdin,OU=Disney,DC=corp-ben,DC=serverlabs,DC=be’ -Filter \* | ForEach-Object {Add-ADGroupMember -Identity ‘Agrabah’ -Members $\_ }

1. Now add all the users from the “LionKing” OU to the “Pride Lands” security group, this time **using the GUI**. Do this for all LionKing users at once, not one by one! Tip: use the advanced option and search on the description (the value in the description field is the same for all LionKing users).

Graphical user interface, text, application

Description automatically generated

1. Use a PS-cmdlet to display the following items:
   1. All groups in your domain

Get-ADGroup -Filter \*

* 1. The sam account names and the scopes of all groups in your domain

Get-ADGroup -Filter \* | FT SamAccountName, GroupScope

* 1. List below the 3 types of “group scopes” used by Windows and search online the differences

Table

Description automatically generated

* 1. The sam account names and categories of all groups in your domain, what is the difference between the categories? Search online if needed.

Everything is security

* 1. All properties of the domain admins group

Get-ADGroup -Identity "Domain Admins" | FL \*

* 1. All members of the domain admins group

Get-ADGroupMember -Identity "Domain Admins" | fl \*

* 1. What kind of names are typically shown of these members?

The distinguishedName, name and SamAccountName

1. Run the Pester Script and put a screenshot (or multiple ones if needed) below. To get a list of variables you can always open the script and read the first few lines. Update your $PROFILE file with a domainusername variable and the other ones as well if you didn’t already.

$PesterPreference = New-PesterConfiguration;

$PesterPreference.output.verbosity = 'Detailed';

$username="<firstname>" ; $domain="corp-<firstname>"; $domainusername="<lastname+firstletteroffirstname>"

Text

Description automatically generated

## Extensions - Optional Assignments

* Redo the first part of the lab after adding all other uses & groups.
* Is it possible to create a local user on the Core server remotely?
* Explore ADSI Edit under tools in server manager.
* Try to perform an operation that enumerates AD information and sniff it with Wireshark. Can you find sensitive information?
* Create a script that shows the domain administrators which users are working the hardest on the network. You are free to determine how.