

Active Directory Project

Imran Hakim Roslan
imrh@protonmail.com

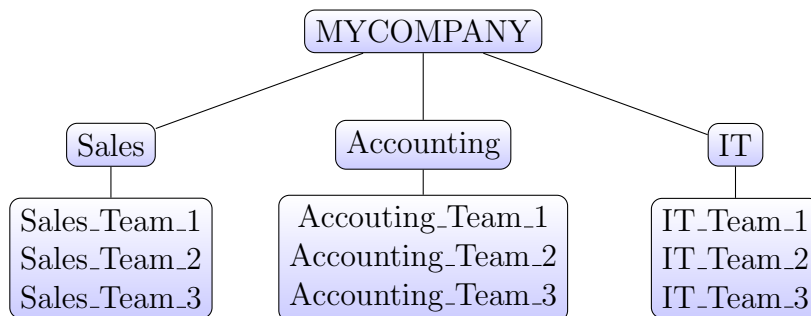
January 30, 2021

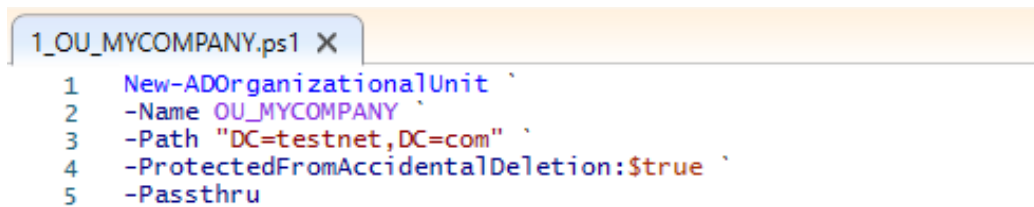
1 Introduction

An Active Directory project to demonstrate my skills in managing Windows Server Active Directory. Windows Server 2012 R2 is used as the environment with single domain named testnet.com.

2 Organization Structure

An organization named MYCOMPANY consists of three distinct departments which are Sales, Accounting, and IT. Each department has three teams to manage the company work flow.



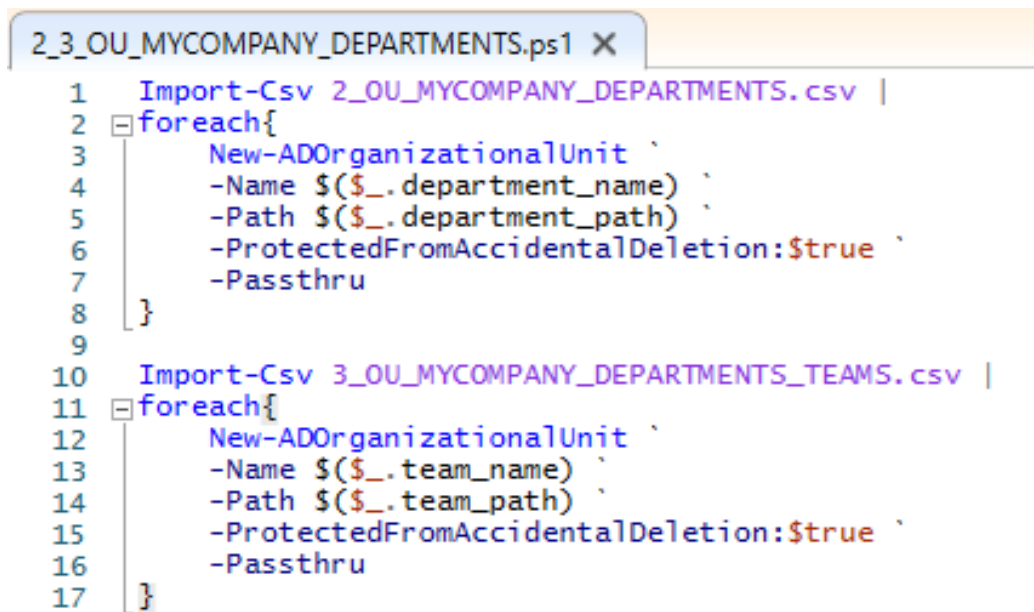


```

1  New-ADOrganizationalUnit `
2  -Name OU_MYCOMPANY `
3  -Path "DC=testnet,DC=com" `
4  -ProtectedFromAccidentalDeletion:$true `
5  -Passthru

```

Figure 1: MYCOMPANY OU code snippet



```

1  Import-Csv 2_OU_MYCOMPANY_DEPARTMENTS.csv |
2  foreach{
3      New-ADOrganizationalUnit `
4      -Name $($_.department_name) `
5      -Path $($_.department_path) `
6      -ProtectedFromAccidentalDeletion:$true `
7      -Passthru
8  }
9
10 Import-Csv 3_OU_MYCOMPANY_DEPARTMENTS_TEAMS.csv |
11 foreach{
12     New-ADOrganizationalUnit `
13     -Name $($_.team_name) `
14     -Path $($_.team_path) `
15     -ProtectedFromAccidentalDeletion:$true `
16     -Passthru
17 }

```

Figure 2: Departments and Teams OU code snippet

3 Constructing Active Directory Organizational Unit

3.1 Create MYCOMPANY Organizational Unit (OU)

A Powershell code snippet in Figure 1 to create MYCOMPANY OU.

Powershell code snippet to create departments and teams OUs is shown in Figure 2.

3.2 Create OUs to Store Users and Computers Group

Figure 3 shows OUs are created to store users and computers groups.

```

1  Import-Csv 4_OU_MYCOMPANY_DEPARTMENTS_TEAMS_User_Computer.csv |
2  foreach{
3      New-ADOrganizationalUnit `
4      -Name $($_.ou_name) `
5      -Path $($_.ou_path) `
6      -ProtectedFromAccidentalDeletion:$true `
7      -Passthru
8  }
9
10 Import-Csv 5_OU_MYCOMPANY_DEPARTMENTS_TEAMS_User_Computer_GROUPS.csv |
11 foreach{
12     New-ADGroup `
13     -Name $($_.group_name) `
14     -Path $($_.group_path) `
15     -GroupCategory Security `
16     -GroupScope Global
17 }

```

Figure 3: Separate OUs for department user and computer

3.3 Structuring a Parent Organizational Unit for Users

A separate parent organizational unit (OU) is formed to store the organization users. This provides a centralized OU to store users container with respect to their assigned department.

3.4 Structuring a Parent Organizational Unit for Computers

A centralized parent organizational unit (OU) for the organization's computers also been formed to store the computers container.

3.5 Organizational Units Result Structure

The resulting OUs structure is demonstrated in Figure 4.

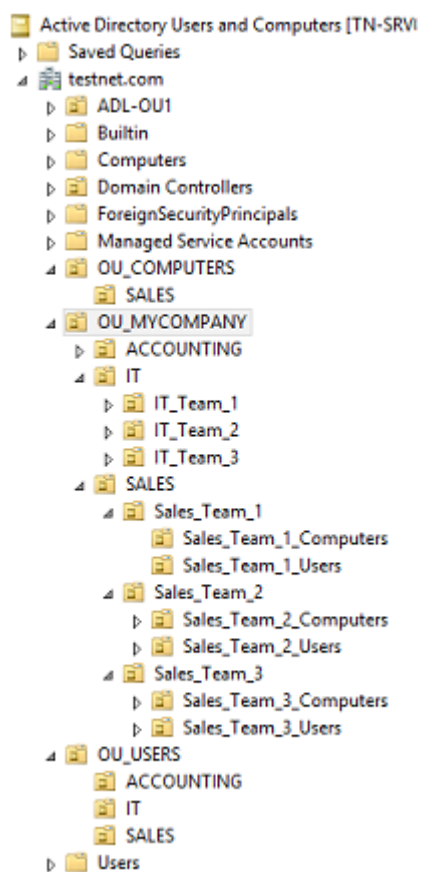
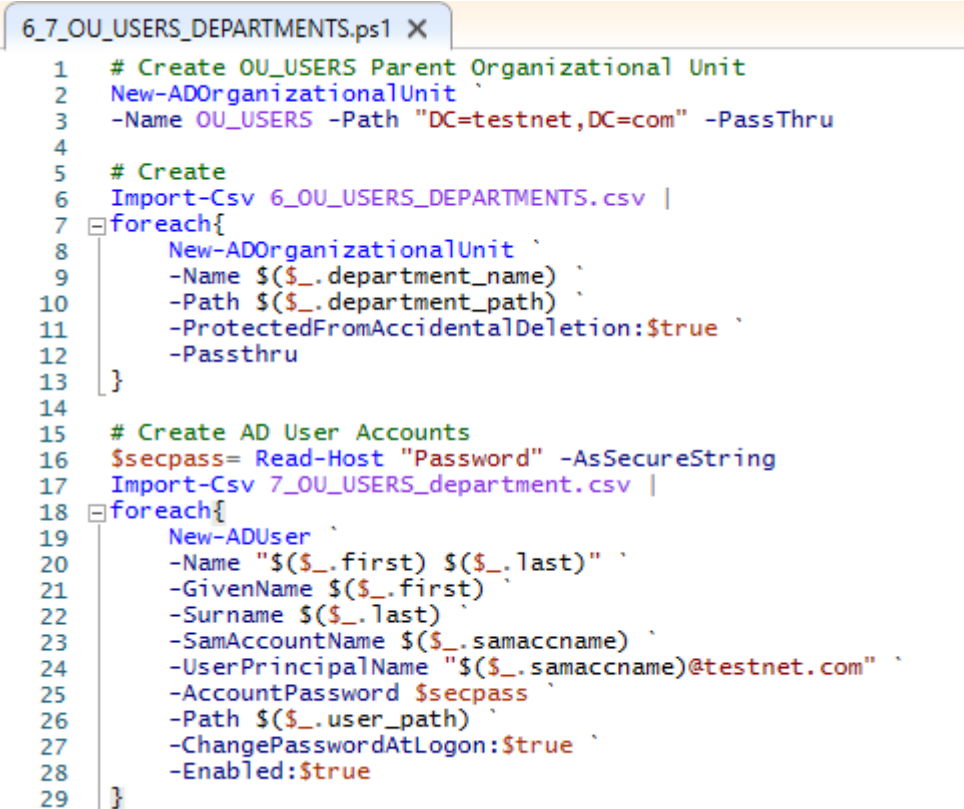


Figure 4: Result OU Structure

4 MYCOMPANY Active Directory User Accounts

4.1 Creating Users

The following PowerShell code snippet in Figure 5 is used to create Active Directory user accounts.

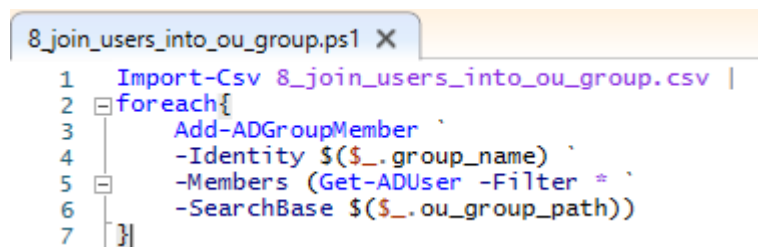


```
1 # Create OU_USERS Parent Organizational Unit
2 New-ADOrganizationalUnit `
3 -Name OU_USERS -Path "DC=testnet,DC=com" -PassThru
4
5 # Create
6 Import-Csv 6_OU_USERS_DEPARTMENTS.csv |
7 foreach{
8     New-ADOrganizationalUnit `
9     -Name $($_.department_name) `
10    -Path $($_.department_path) `
11    -ProtectedFromAccidentalDeletion:$true `
12    -Passthru
13 }
14
15 # Create AD User Accounts
16 $secpass= Read-Host "Password" -AsSecureString
17 Import-Csv 7_OU_USERS_department.csv |
18 foreach{
19     New-ADUser `
20     -Name "$($_.first) $($_.last)" `
21     -GivenName $($_.first) `
22     -Surname $($_.last) `
23     -SamAccountName $($_.samaccname) `
24     -UserPrincipalName "$($_.samaccname)@testnet.com" `
25     -AccountPassword $secpass `
26     -Path $($_.user_path) `
27     -ChangePasswordAtLogon:$true `
28     -Enabled:$true
29 }
```

Figure 5: Active Directory user account creation with PowerShell

4.2 Joining User Accounts to Department User Groups

Joining users to their assigned team accordingly is useful for managing user accounts in Active Directory. This provides ease of delegating permissions by grouping users together and managing them in single location. Figure 6 shows the code listing for the joining user accounts method.



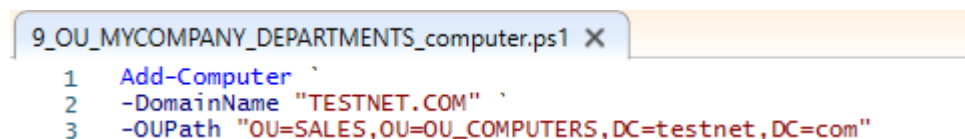
```
8_join_users_into_ou_group.ps1 X
1  Import-Csv 8_join_users_into_ou_group.csv |
2  foreach{
3      Add-ADGroupMember `
4      -Identity $($_.group_name) `
5      -Members (Get-ADUser -Filter * `
6      -SearchBase $($_.ou_group_path))
7  }
```

Figure 6: Joining users into their assigned groups

5 MYCOMPANY Active Directory Computer Accounts

5.1 Create and Join Computer Account into Active Directory Domain Controller

The example PowerShell code snippet in Figure 7 is used to create computer account and join into the Active Directory domain with path set to "OU=SALES,OU=OU_COMPUTERS,DC=testnet,DC=com".

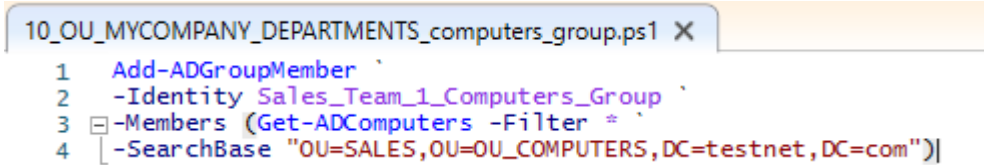


```
9_OU_MYCOMPANY_DEPARTMENTS_computer.ps1 X
1  Add-Computer `
2  -DomainName "TESTNET.COM" `
3  -OUPath "OU=SALES,OU=OU_COMPUTERS,DC=testnet,DC=com"
```

Figure 7: Create and join computer into the domain

5.2 Joining Computer Account into Specific Computer Group

From the code listing shown in Figure 8, the computer accounts inside organizational unit "OU=SALES,OU=OU_COMPUTERS,DC=testnet,DC=com" are being joined into a computer group named Sales_Team_1_Computers_Group.

A screenshot of a PowerShell console window. The title bar reads "10_OU_MYCOMPANY_DEPARTMENTS_computers_group.ps1 X". The script content is as follows:

```
1 Add-ADGroupMember `
2   -Identity Sales_Team_1_Computers_Group `
3   -Members (Get-ADComputers -Filter * `
4   [-SearchBase "OU=SALES,OU=OU_COMPUTERS,DC=testnet,DC=com"])
```

Figure 8: Join computer accounts into a group