Active Directory Lab

Stavrianopoulou Maria

October 2023

Contents

1 Introduction

In this lab project, we will set up a virtual Active Directory (AD) environment, complete with a Domain Controller (DC) and a Window 10 client. Active Directory is a powerful technology that allows organizations to manage users, computers, and resources on a network in a centralized and efficient manner. Through this project, we gain hands-on experience in creating, configuring, and automating tasks within an AD environment.

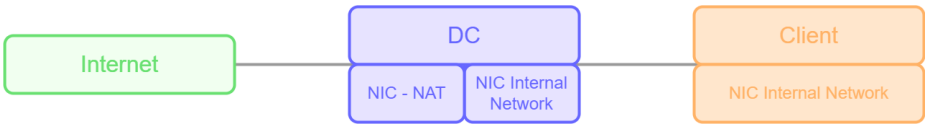
2 The Basics of Active Directory

Active Directory (AD) provides a structured and organized approach to handling resources, security, and user access. Here's a breakdown of some key terms and concepts:

- **Domain Controller (DC)**: A Domain Controller is a critical component of Active Directory. It serves as the central server responsible for authenticating users and granting them access to network resources. The DC also manages security policies, enforces access controls, and maintains a database containing information about all users, groups, and devices within the network. In essence, the DC is the authoritative source of identity and authentication within the domain.
- **Organizational Units (OUs)**: Organizational Units (OUs) are containers within the Active Directory hierarchy that enable the logical organization of network objects. OUs provide a way to group users, computers, and other resources based on administrative or functional criteria. This hierarchical structure enhances the efficiency of network management, as administrators can apply policies, permissions, and settings to specific OUs, tailoring the management process to suit the organization's needs.
- **User accounts**: User accounts are fundamental entities within Active Directory, representing individual users who access network resources. These accounts store information such as usernames, passwords, and attributes related to the user's role and permissions. By creating user accounts within the AD environment, administrators can control user access, manage authentication, and assign specific privileges based on job roles or departmental requirements.

3 Design and architecture

This lab will consist of two virtual machines: one acting as a Domain Controller (DC) and two others as Windows 10 clients.



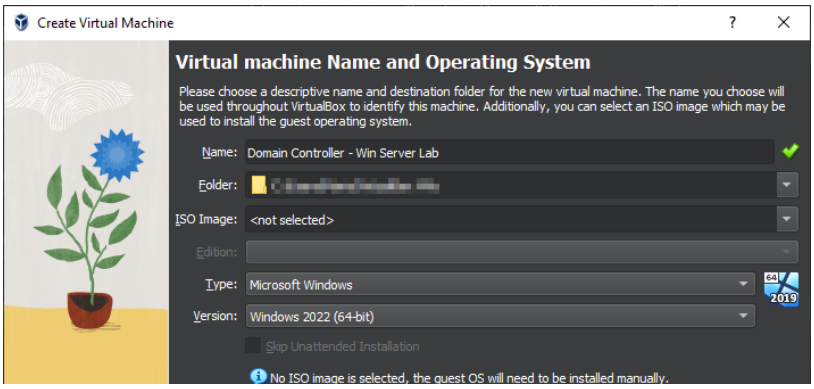
Virtual Machines:

- **Virtual Machines**:
  - **Domain Controller (DC)**: This virtual machine will serve as the base of the Active Directory environment, handling user authentication, security policies, and directory services.
  - **Client**: This virtual machine will function as a network client, accessing resources within the domain and interacting with the DC for authentication and authorization.
- **Network Adapters:**
  - **NAT Network Adapter**: This adapter will provide the Domain Controller with internet access.
  - **Internal Network Adapter**: This adapter will enable communication between the Domain Controller and the Windows 10 clients within the isolated internal network.

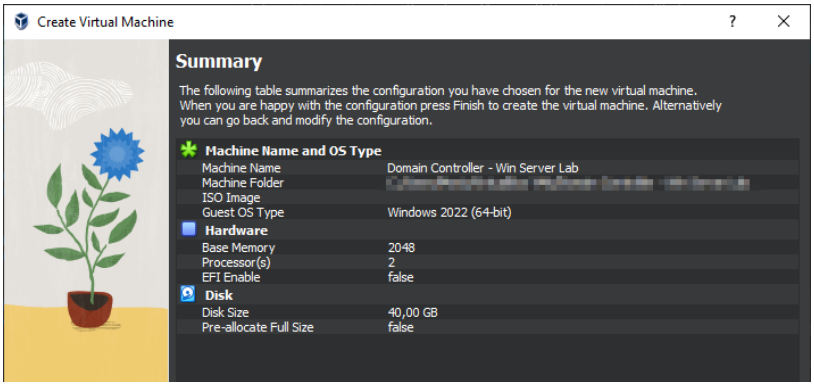
4 Domain Controller

4.1 Installation

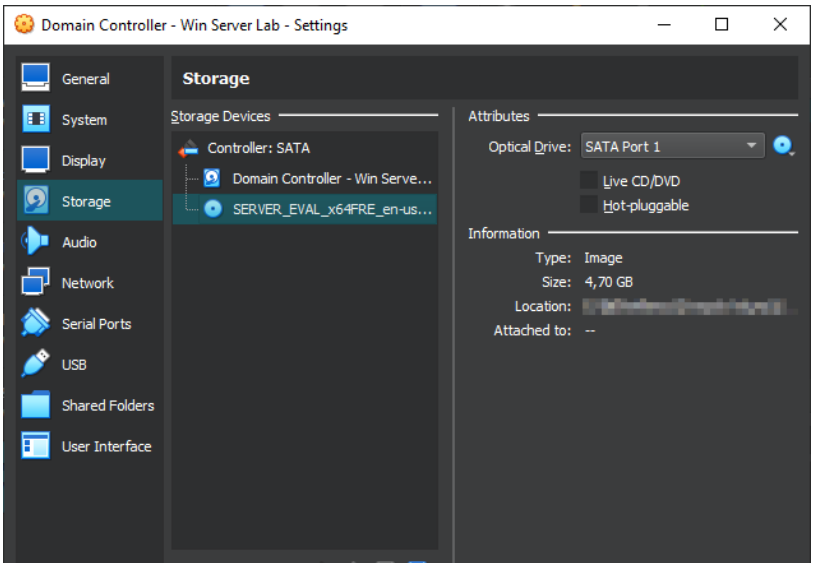
1. Download the Windows Server iso from here.
2. Open Virtualbox and click on Create a New Virtual Machine.
3. Define the name of the VM to Domain Controller - Win Server Lab.
4. Select Windows 2022 64 bit as the guest operating system.



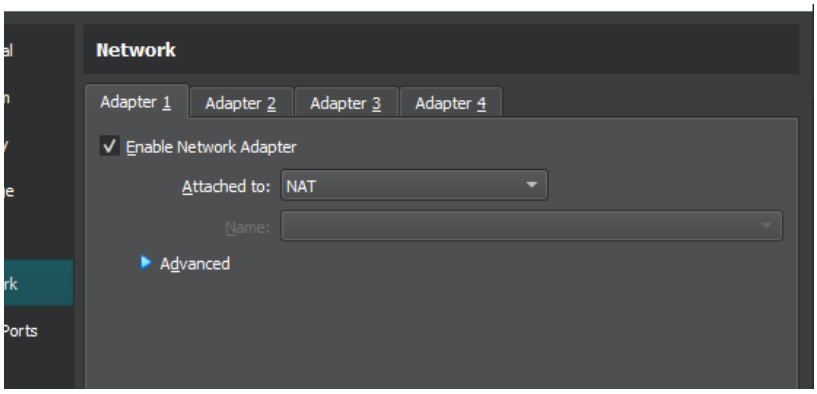
5. Configure the virtual hardware:
    - Set the disk size, in this implementation we use 40 GB.
    - Allocate 2 GB of RAM to the VM.



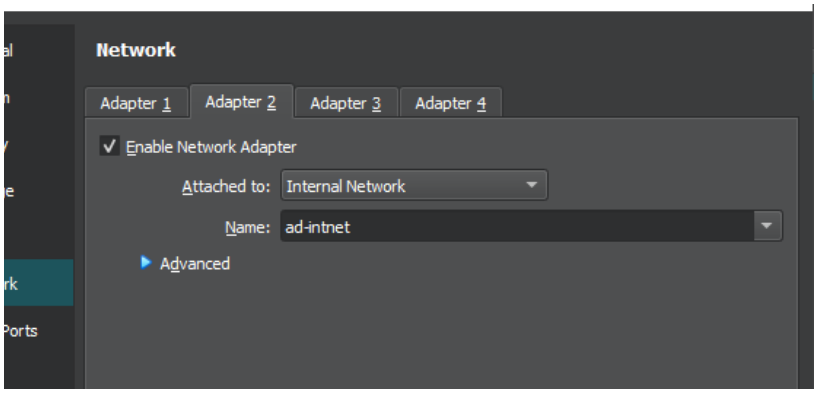
    - Add the ISO image that we downloaded earlier. To do so click Settings then Storage. Click on the Disk icon and click on Choose a disk file to attach the ISO file.



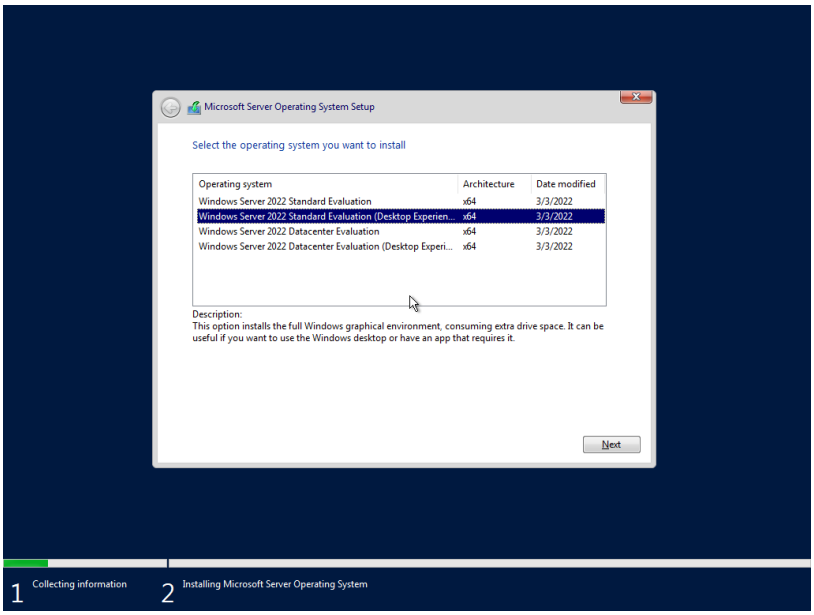
- Now we need to configure the VM's Network adapters in VirtualBox. To set them click Settings then Network.
  The lab's topology requires two network adapters:
  - **Internet Adapter (NAT Network)**: This adapter is used to connect the DC VM to the internet. To set it up go to Adapter 1 and set it as NAT.



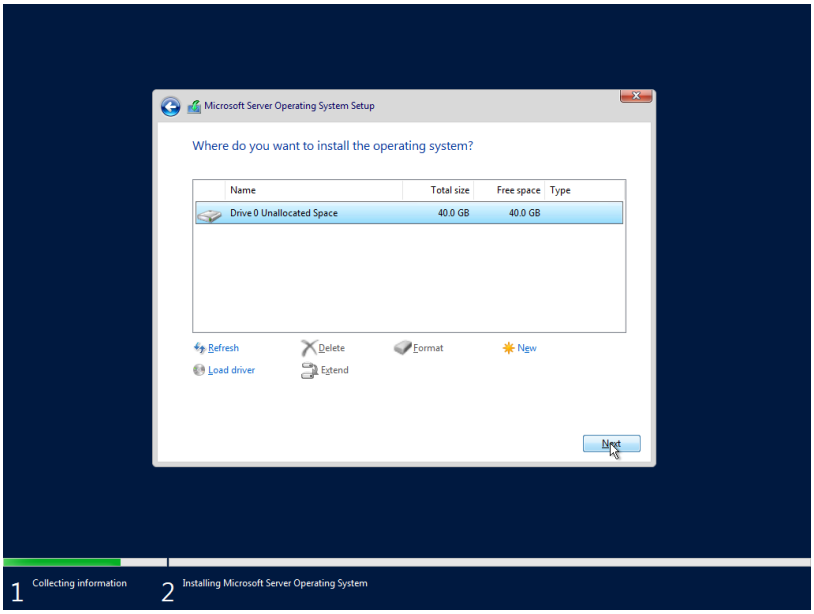
  - **Internal Network Adapter**: This adapter is used for the internal network of the DC with the Clients. Configure this adapter as an Internal Network with the name ad-intnet.

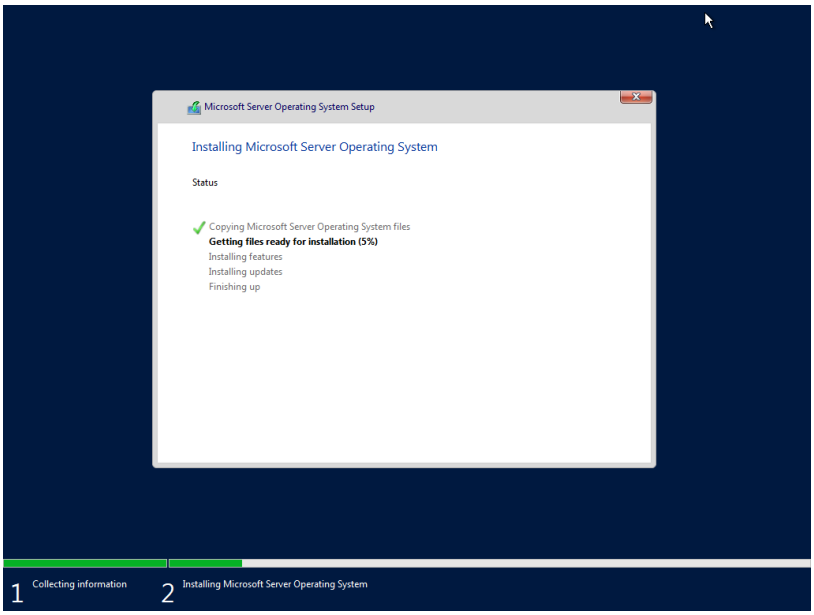


  The configuration of the VM is complete, we proceed to the installation of the Domain Controller.
- Boot the VM, on the startup screen choose the Standard Evaluation (Desktop Experience) and accept the licence.



- Choose the custom installation option, bacause we are installing the Server from scratch





- Enter the password of the administrator's account *(choose a simple password for the purposes of this lab)*.
- The system will restart in order to complete the installation.

4.2 Configuration

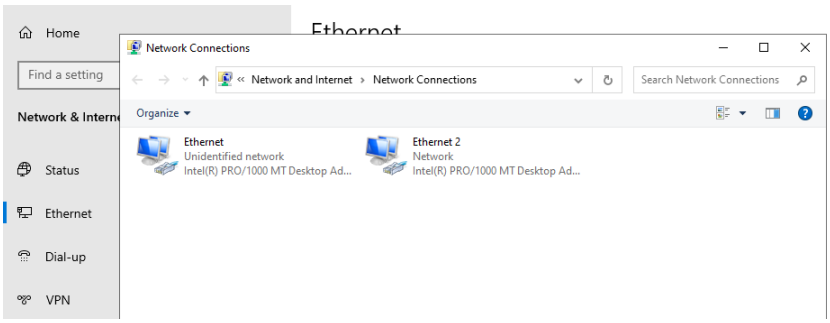
4.2.1 System Name

1. Navigate to the Settings menu, choose System and then select About and then click on Rename this PC. Rename the VM to "DC" (Domain Controller) and restart the system.

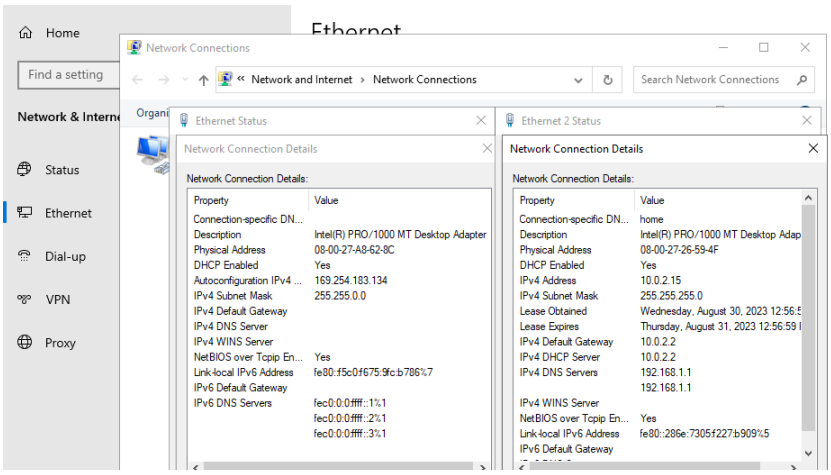
4.2.2 Configuration of the Network Cards

This section outlines the process of configuring the network cards to align with the lab's previously described topology. The following steps will guide you through the network card configuration:

1. Navigate to the Settings menu, choose Network & Internet and then select Ethernet and then click on Change Adapter Options. A new window will appear that contains the two adapters.



2. Looking at the details of each adapter:



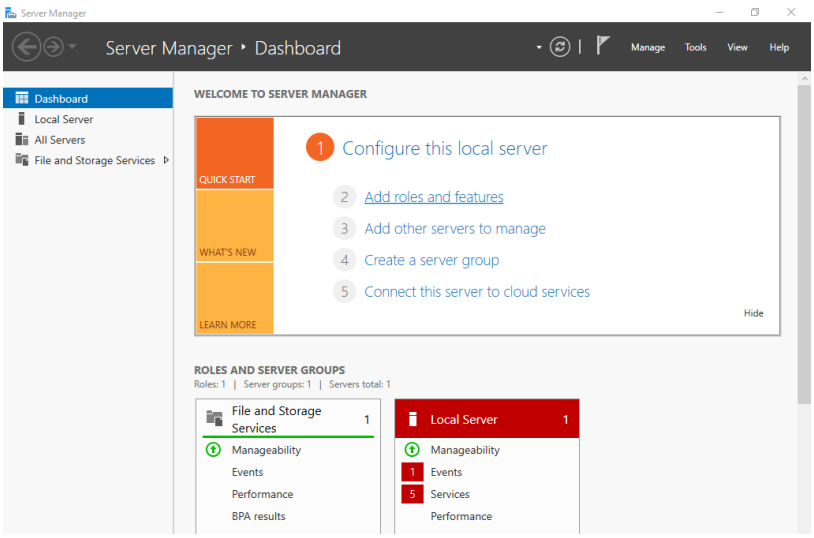
- **Ethernet 2** with the IP address 10.0.2.15 is the NAT (Network Address Translation) Adapter. NAT (Network Address Translation) adapters are usually assigned IP addresses in the 10.0.2.x range by default.
- **Ethernet** is the Internal Network Adapter.

3. Based on this observation, we rename ”Ethernet” and ”Ethernet 2” to ”INTERNAL_NET” and ”INTERNET” respectively.
4. On the ”INTERNAL_NET” network adapter right-click and select Properties.
    - Set the IP of the adapter to 192.168.24.1
    - Set the subnet mask to 255.255.255.0 (/24 subnet mask).
    - Set the DNS to 127.0.0.1, the Domain Controller itself will be used as the DNS server.

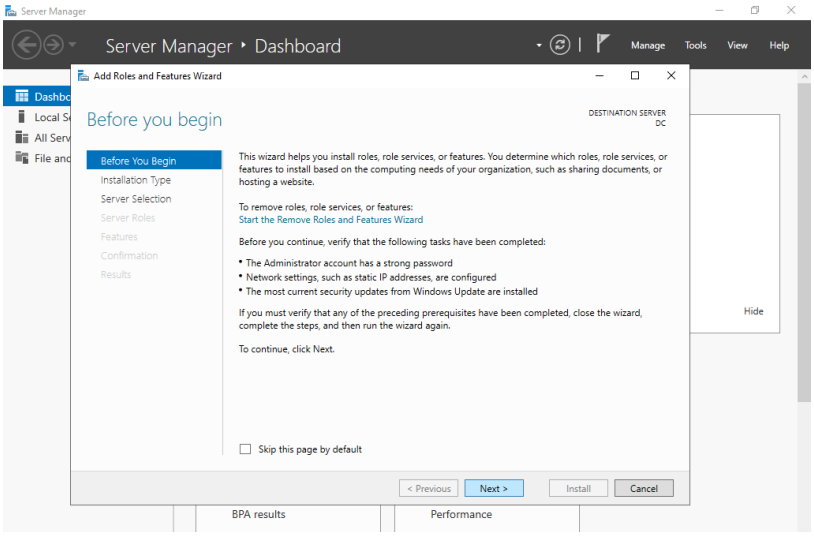
4.2.3 Install AD DS

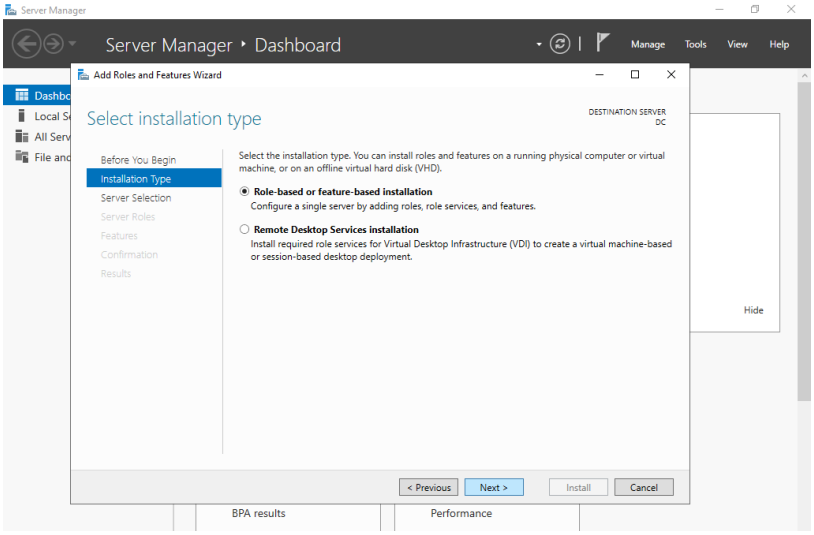
In this section we install Active Directory Domain Services (AD DS) and set up our domain.

1. On the server manager window, select Add roles and features.

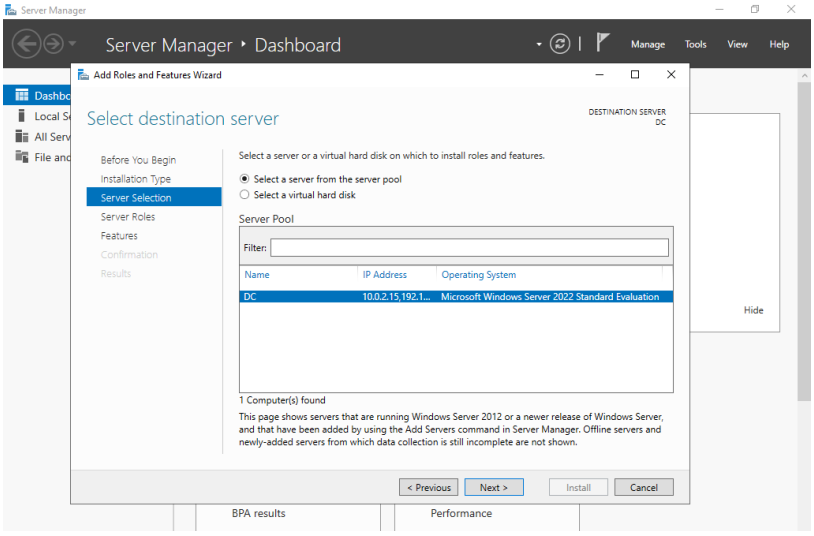


2. On the setup wizard select the default options.

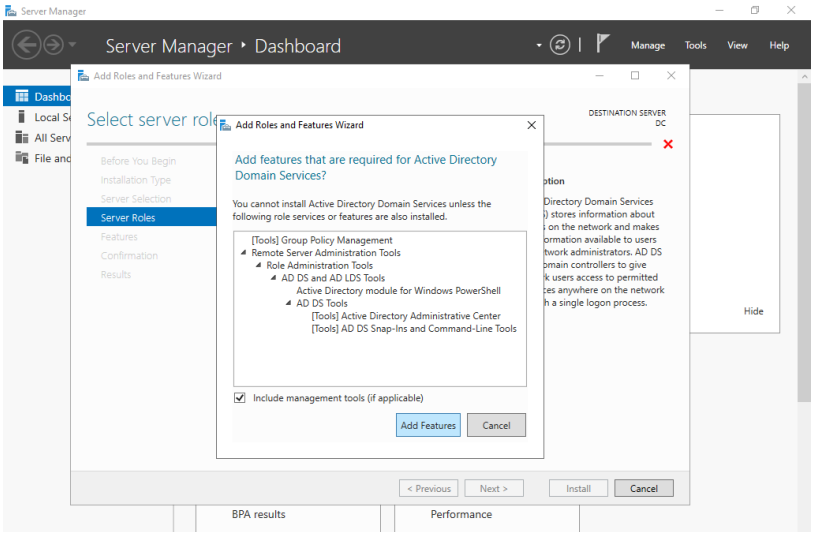




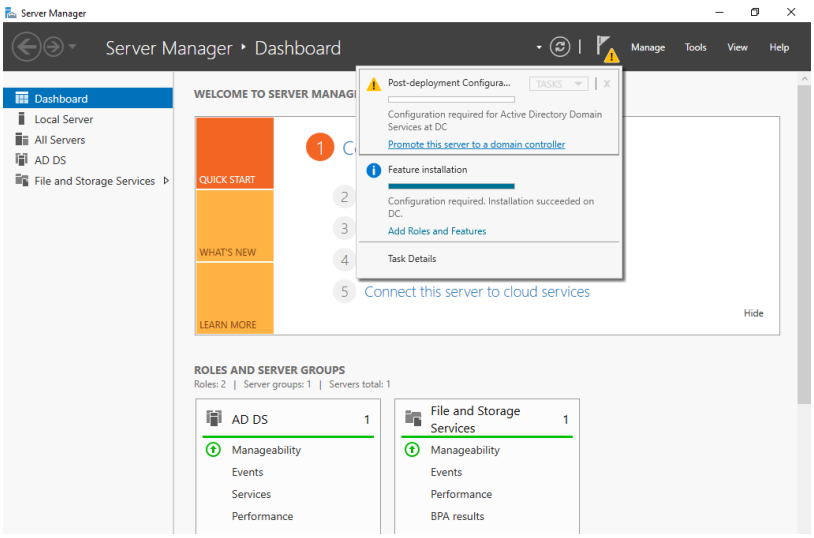
3. On the server selection section we select our domain controller, in this case DC.



4. Next, on the server roles section of the wizard we select the roles that we want to install Active Directory Domain Services. We choose the default options and finish the installation.

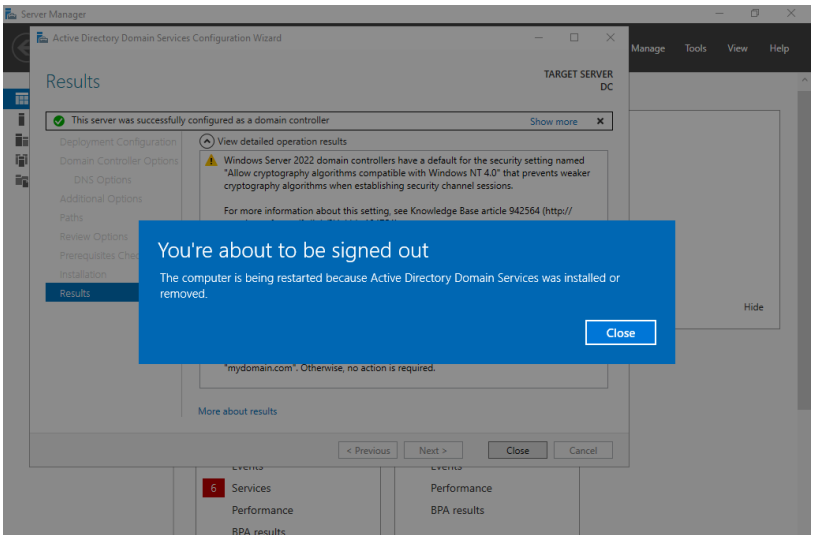


5. The installation is completed and now we to do a post-deployment configuration to create the domain. A notification indicator appears on the screen. To create the domain, we click on it and we select the Promote this server to a domain controller option.



6. In the deployment and configuration window that appears on the screen we select the Add a new forest option. This option is suitable in our case because we start from scratch, we have do not an existing AD domain, and we want to create an entirely new AD forest.

7. Continue with the default options (database paths etc) and finish the installation.

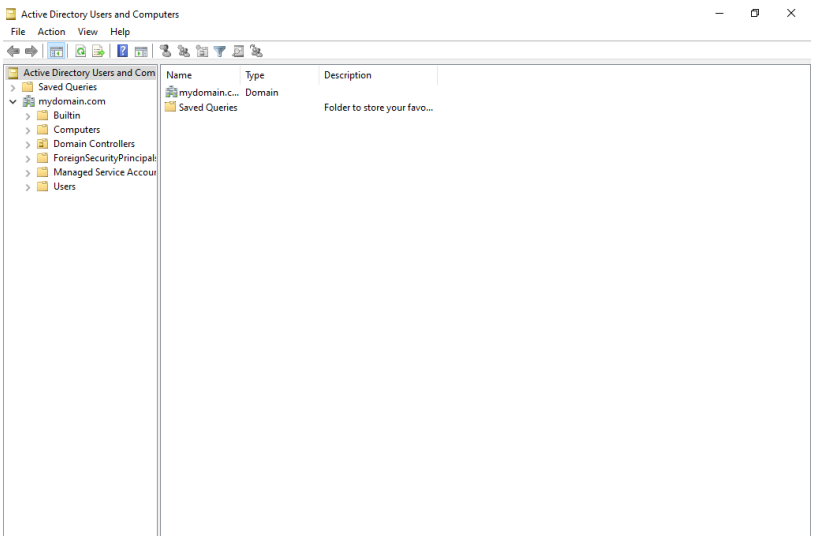


Once the installation is complete we restart our system.

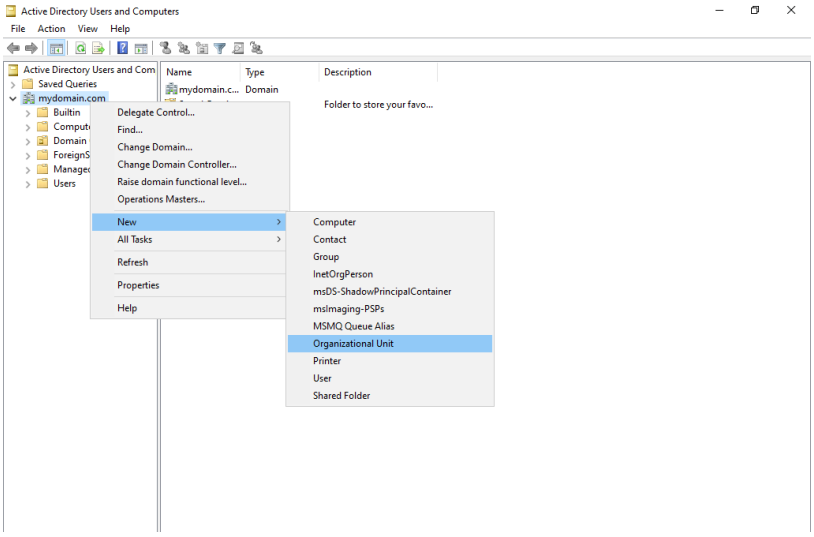
4.2.4 Administrator Account Setup

Setting up an Administrator account on AD is a fundamental step that grants us full administrative control over the AD environment. This account serves as the primary means for managing, configuring, and maintaining the Active Directory domain and its associated resources. To create this account execute the following steps:

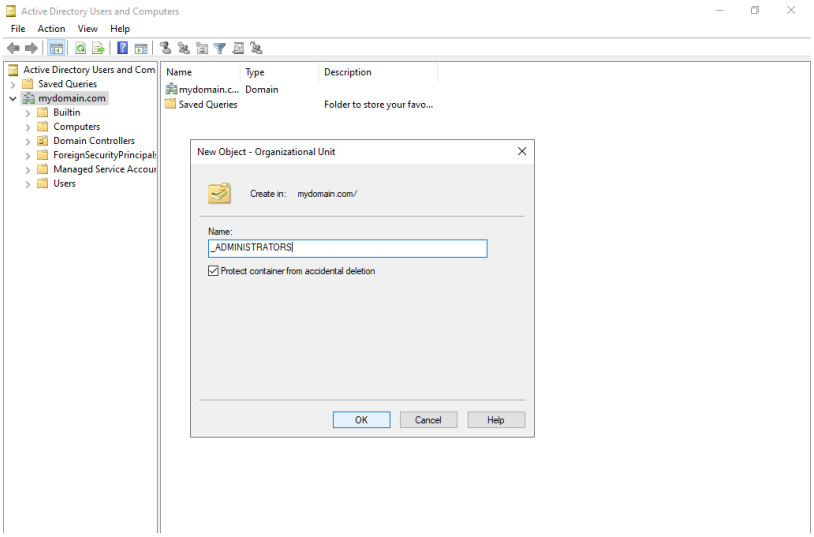
1. On the start menu select the Active Directory Users and Computers.



2. First, we set up an Organizational Unit for our administrators. Setting up an ”Admin” OU in your Active Directory environment to store accounts is a good practice that offers several benefits in terms of security, organization, and management. To create the OU right-click on the domain name (mydomain.com) and select New > Organizational Unit.

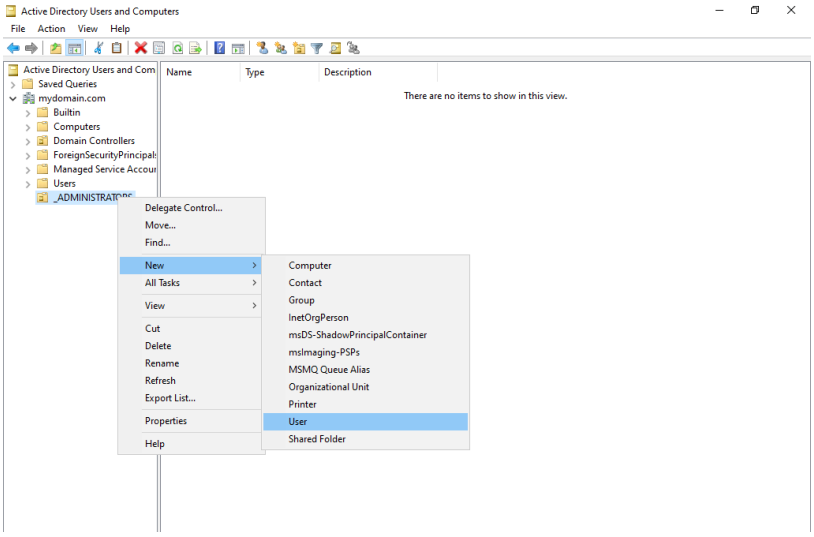


3. On the new window that appears we name the OU to _ADMINISTRATORS.

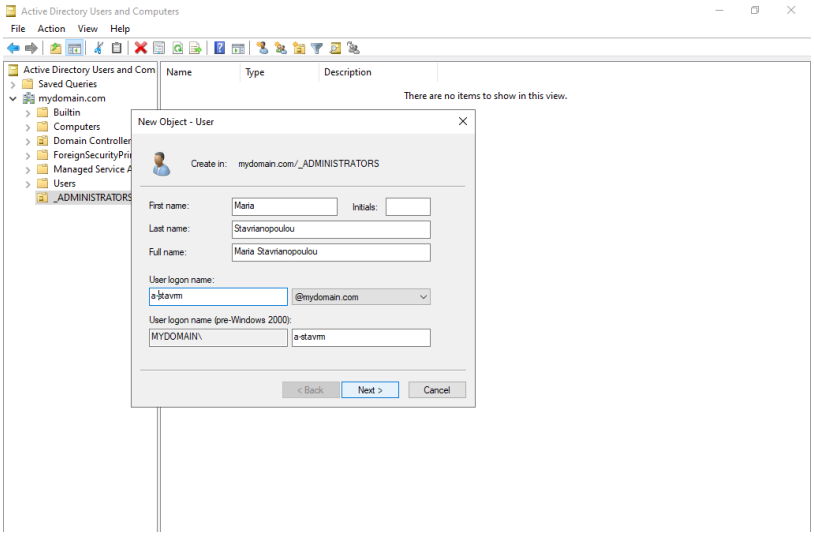


Note: we check the "Protect container from accidental deletion" option. It is a good practice in AD to prevent potential data loss, disruptions, and security risks.

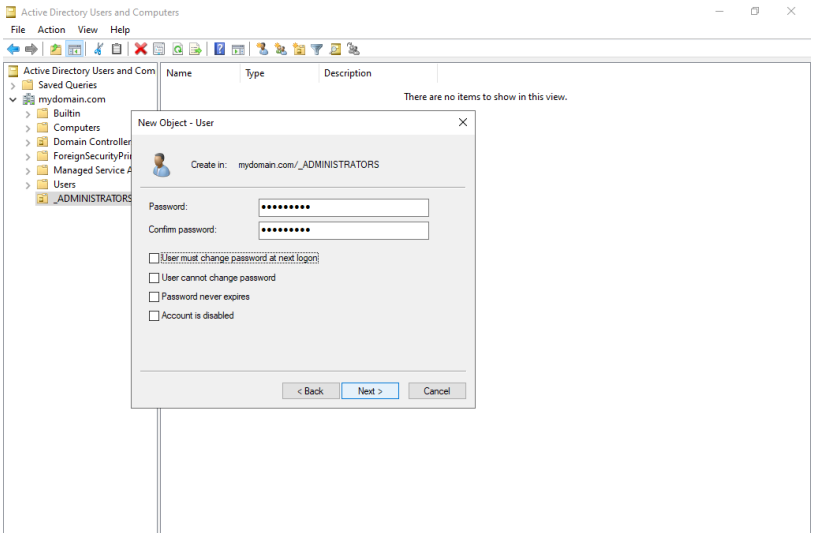
4. Once the _ADMINISTRATORS OU is created, we proceed to the creation of the administrator account. To do so we right-click on the _ADMINISTRATORS OU and select New > User.



5. On the new window that appears we enter the administrator's credentials.



6. Then we enter the password for this account. We also unchecked the "User must change the password at next log on" option and we finish the setup.



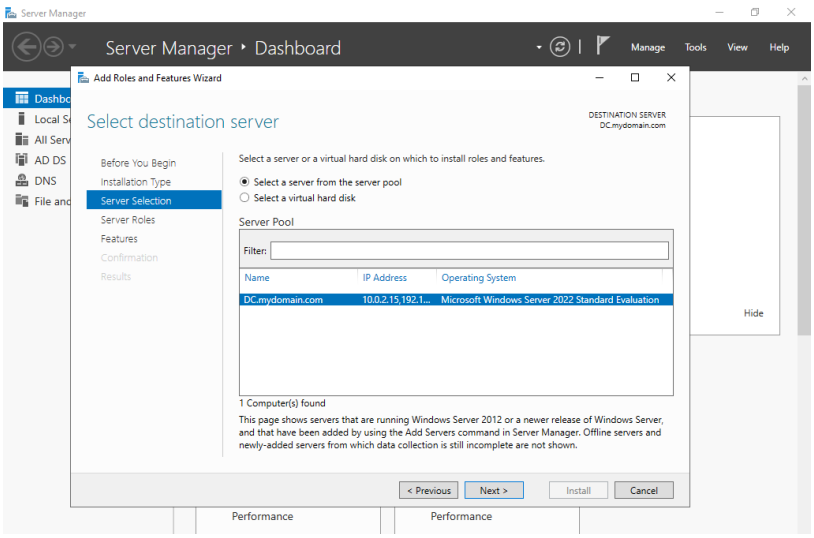
Note: enforcing the "User must change the password at next log on" option is a se-

curity measure that ensures immediate password modification, reducing the risk of unauthorized access due to initial passwords being known or compromised. This practice promotes strong security by preventing prolonged use of default or temporary passwords. We unchecked this option just for this lab's purposes, in a real-world environment it should be enabled.

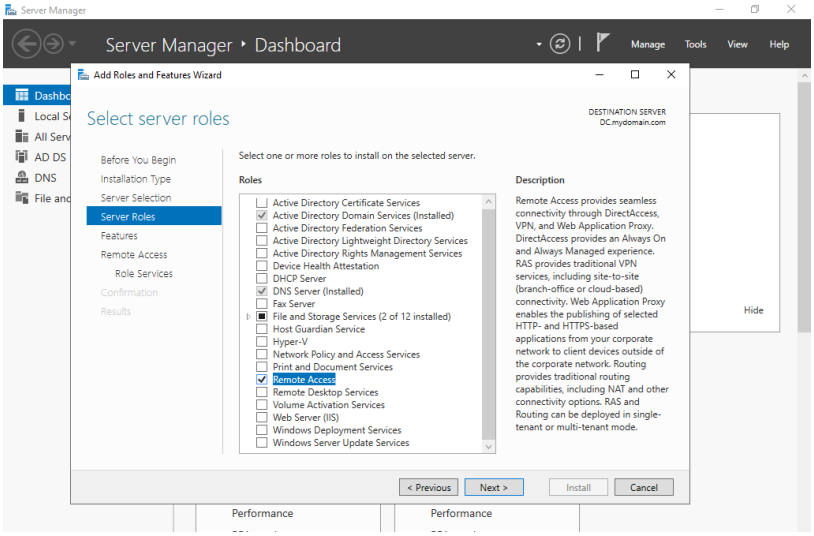
4.2.5 RAS Configuration

In a real-world environment, we want the clients of our AD to be able to use the internet. In this section we configure the RAS (Remote Access Control) to allow the clients (VMs) to connect to the internet via our DC, given that they use the internal network adapters.

1. In Service Manager select Add roles and features.
2. In the Server Selection section we select our DC.



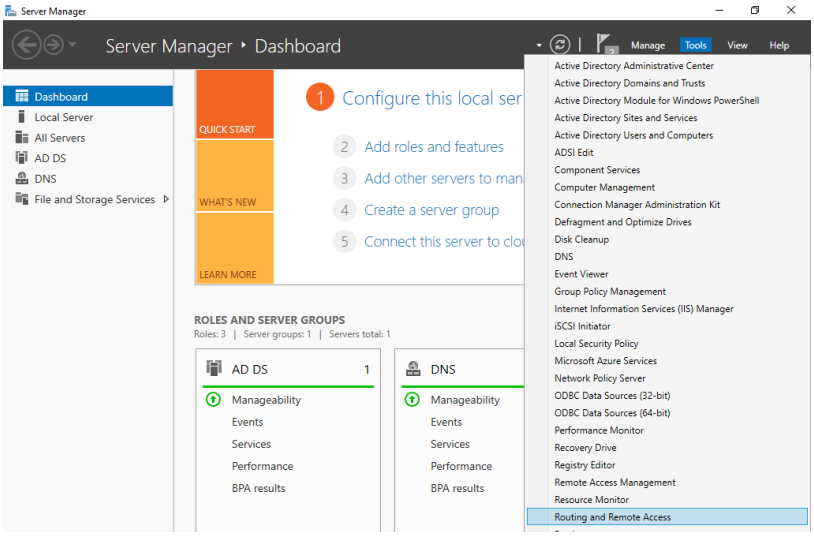
3. In the Server Roles we select to install the Remote Access. Once it's selected we continue with the defaults.



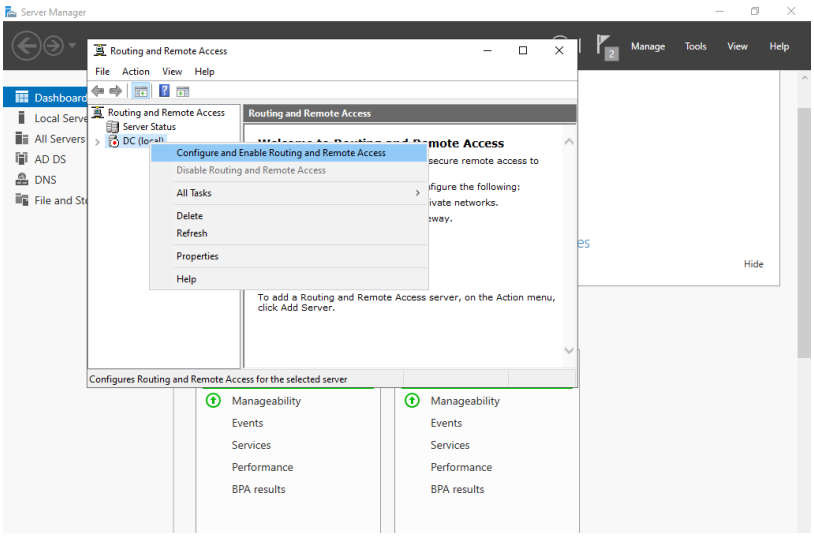
4. In the Role Services we check the Direct Access and VPN (RAS) and Routing. The combination of these two options enhances remote access by providing secure and encrypted connections for off-site users, enabling seamless and controlled network

access while maintaining data privacy. Add both of these features and proceed to the next steps of the installation using the default options.

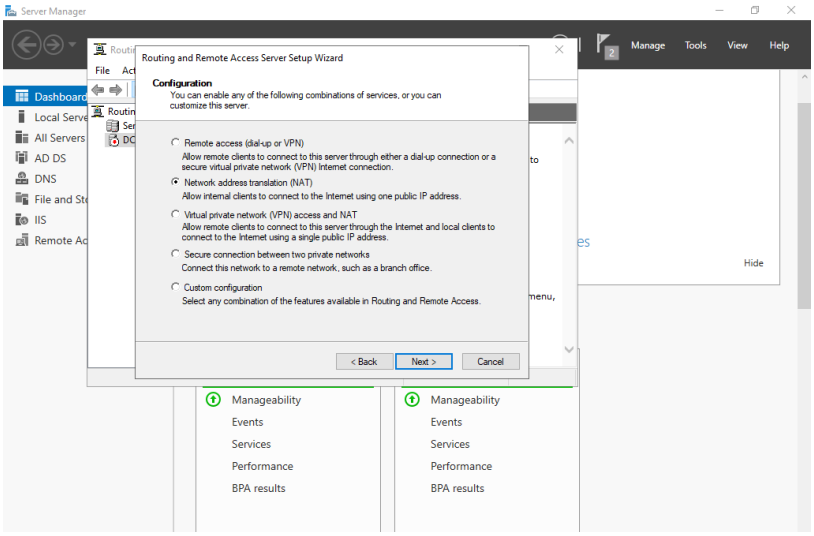
5. Once the installation is complete navigate to Tools > Routing and Remote Access.



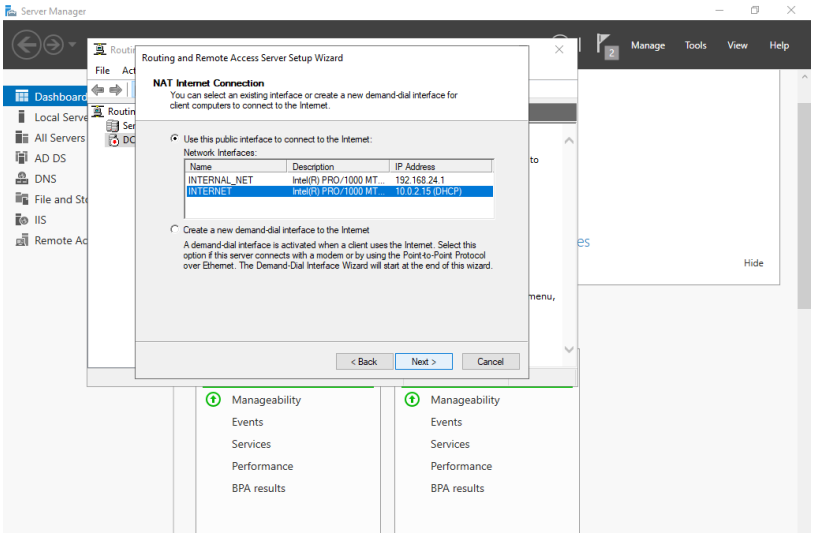
6. On the Routing and Remote Access window do the main configuration. We right-click on Domain and select Configure and Enable Routing and Remote Access to open up the setup wizard.



7. We select NAT in order to allow the clients of the AD to use the internet.



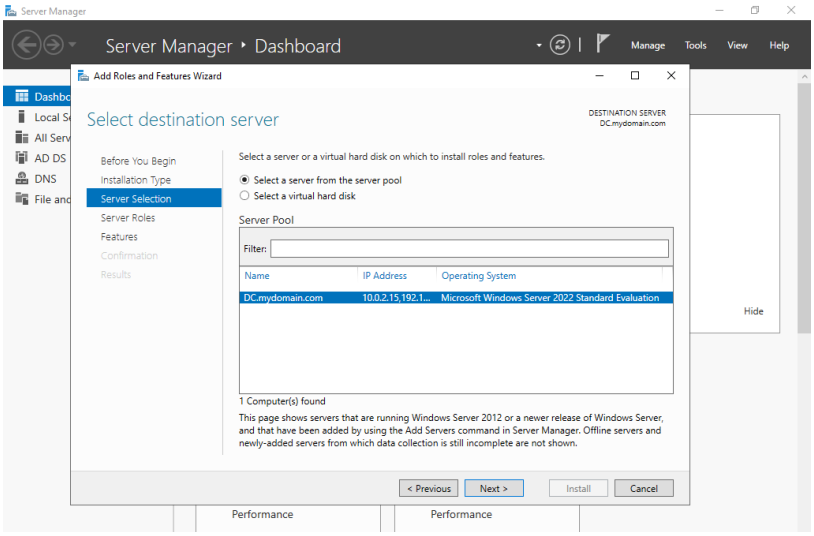
8. Next we select the network adapter that is used to connect to the internet. In this case we select the INTERNET interface as configured previously in the Configuration of the Network Cards section.



4.2.6 DHCP Configuration

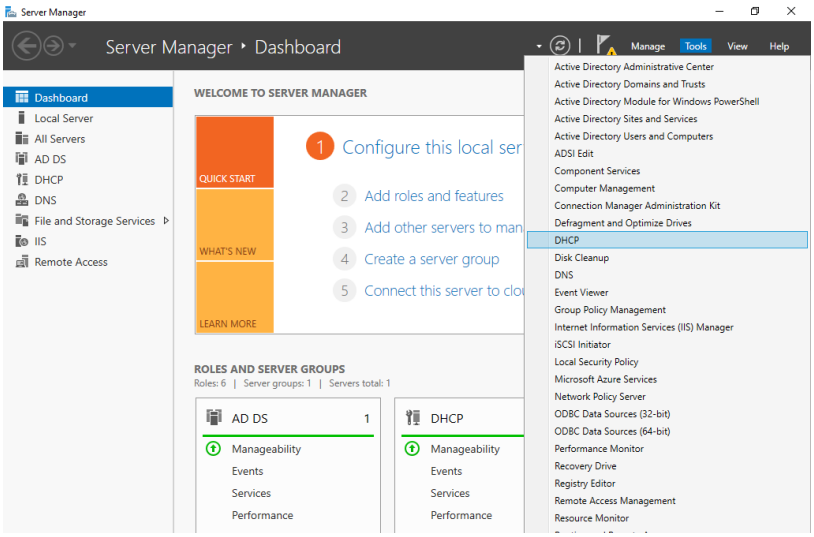
Configuring DHCP on your Active Directory (AD) server simplifies network management by automatically assigning IP addresses to clients, ensuring seamless network connectivity and centralizing IP assignment within the AD environment. To configure DHCP we follow the steps below:

1. In Service Manager select Add roles and features.
2. In the Server Selection section we select our DC.

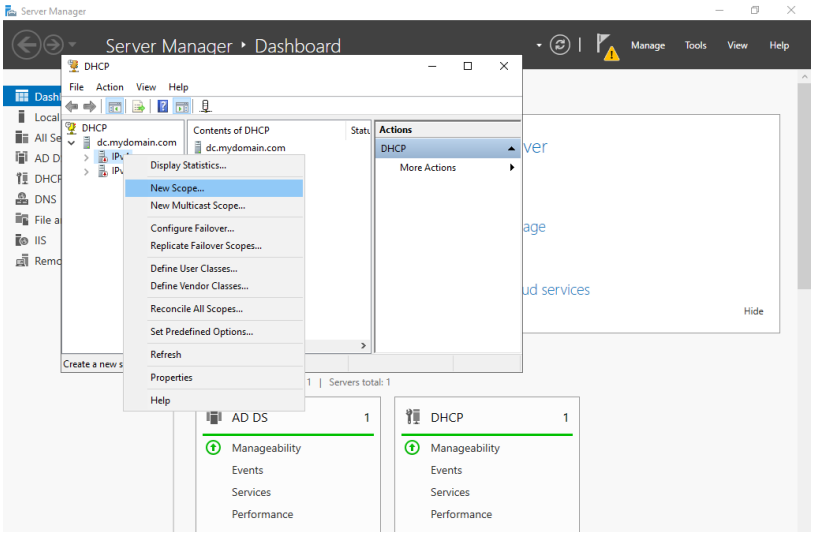


3. In the Server Roles we select to install the DHCP. Once it's selected we continue with the defaults.

4. Once the installation is complete navigate to Tools > DHCP.

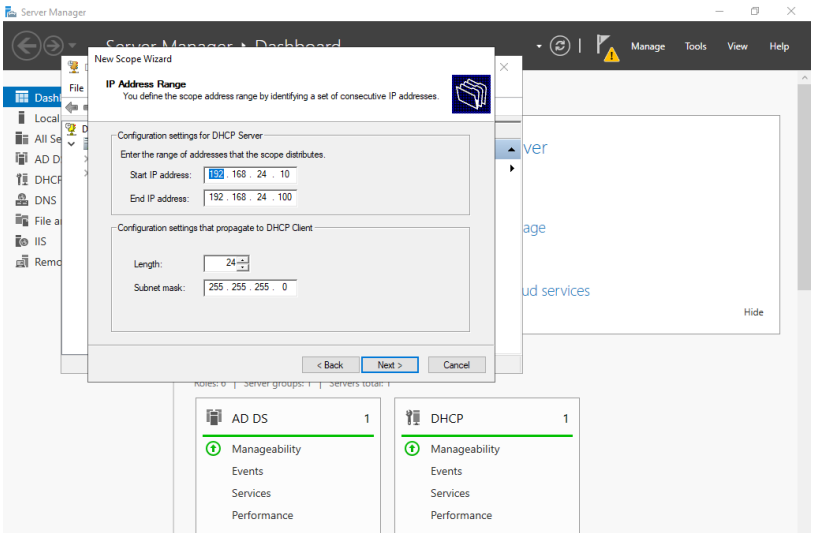


5. On the DHCP window we right-click on mydomain.com and select New Scope to setup the scope that is used to distribute the IPs on the network.



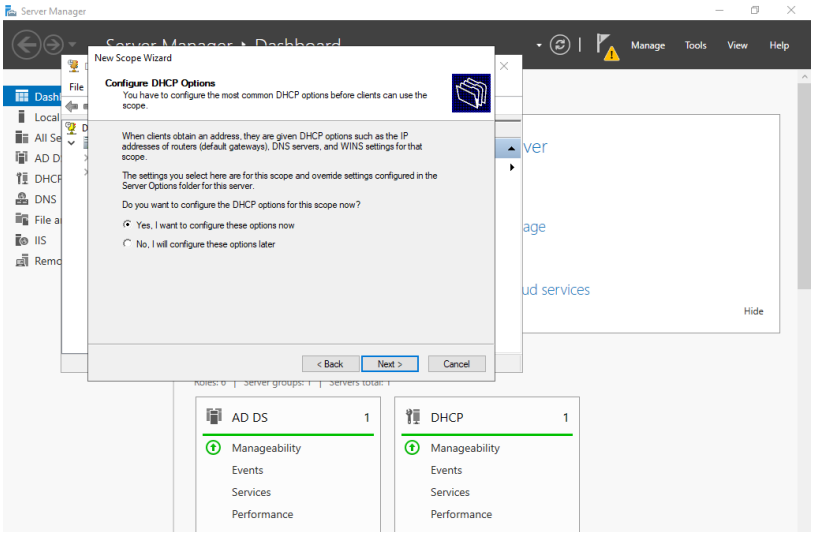
6. Based on the configuration of the network adapters as discussed previously on the Configuration of the Network Cards section, set up the scope as follows:

   - We name the scope as 192.168.24.10-100.
   - Setup the start address as 192.168.24.10 and the end address as 192.168.24.100 and the subnet mask as 255.255.255.0 (/24 subnet mask).

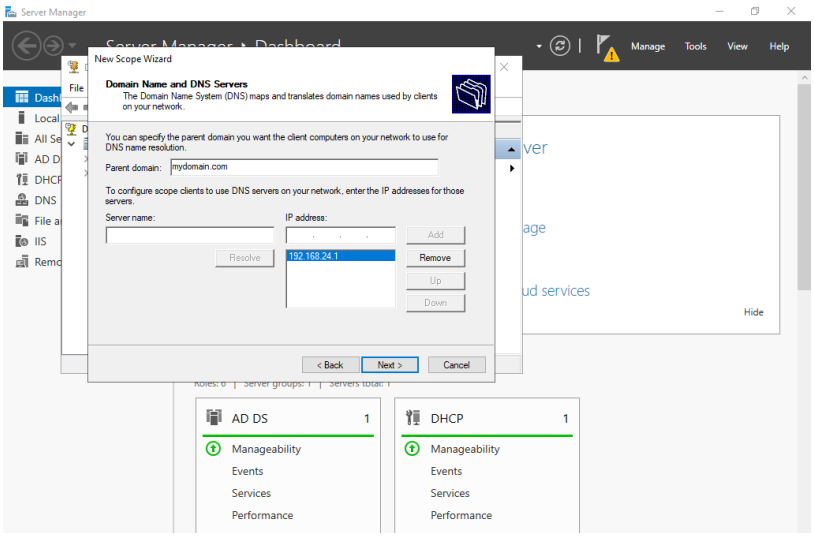


7. We continue with the default options on Lease Duration.
   Note: Configuring a Lease Duration on DHCP controls how long a client can use an IP address. It optimizes IP address utilization and allows network resources to be efficiently managed by automatically reclaiming and reallocating addresses after the lease period expires.

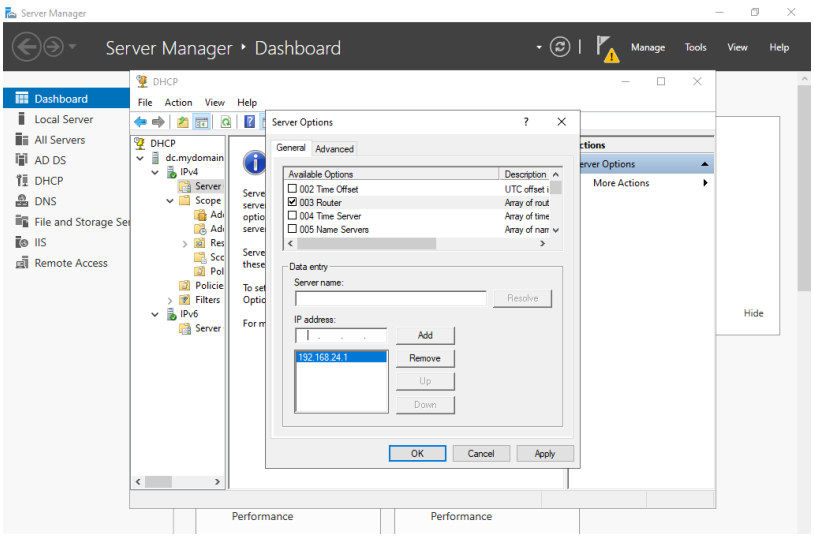
8. On the new window that appears we select that we do want to configure DHCP options.



9. Setup the default gateway as 192.168.24.1, which is the DCs IPv4 address, and click Add and we configure the DNS Server shown in the image below.



10. In the final step, we right click on IPv4 → Server Options and select Configure Options. On the new window that appears we check the Router tab with the DCs IPv4 address.

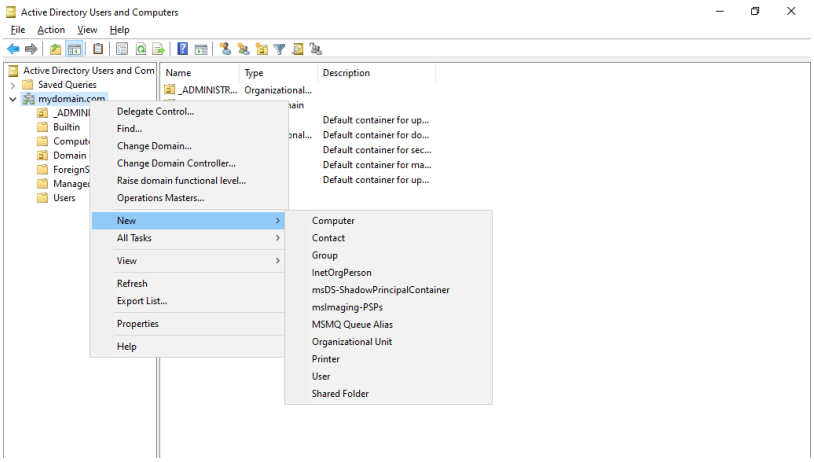


4.2.7 Configuration of OUs

In an Active Directory (AD) environment, Organizational Units (OUs) serve as containers for organizing and managing objects like users, groups, and computers. Group Policy Objects (GPOs), on the other hand, are a collection of settings that define what a system will look like and how it will behave for a defined group of users.

1. To create Organizational Units (OUs) using the GUI in Active Directory we follow these steps:

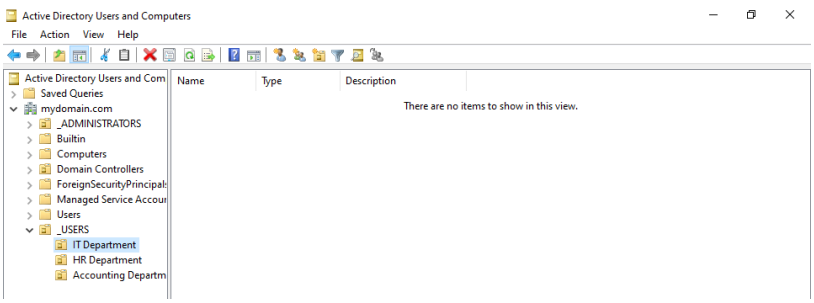
    - Open Active Directory Users and Computers from Start → Active Directory Users and Computers.
    - In the left pane, right-click on our domain, click New → Organizational Unit.



2. We create four Organisational Units (OUs):

    - **_USERS OU**: This OU serves as the parent OU for user-related OUs.
    - **IT Department OU**: This OU is dedicated to IT-related user accounts and resources and it is nested within the _USERS OU.
    - **HR Department OU**: This OU is dedicated to HR-related user accounts and resources.
    - **Accounting Department OU**: This OU is dedicated to Accounting-related user accounts and resources.

    Finally, the OU structure looks like this:



4.2.8 Configuration of GPOs

1. Once OUs are established, specific policies can be applied through GPOs to manage the behavior and security settings of systems within these OUs. We can include

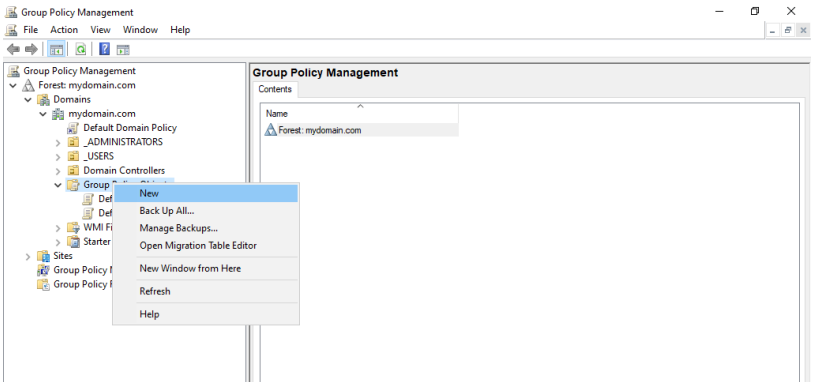
computer and user settings in the same GPO, but splitting these policies into separate objects makes maintenance and problem-solving a lot easier. We define a main GPO called Main GPO the will be linked with the _USERS OU. Based on the structure of OUs the children of the _USERS OU, inherit the main GPO.

- **USERS** (Main GPO)
  - **HR Department**
  - **IT Department**
  - **Accounting Department**

2. To create the main GPO for the _USERS OU in Active Directory, in the Server Manager Dashboard, we click on Tools and select Group Policy Management.

3. In the Group Policy Management Console, we expand Forest and then Domains. We locate and select the domain where the USERS OU is present (mydomain.com). We expand the domain and navigate to the Group Policy Objects.

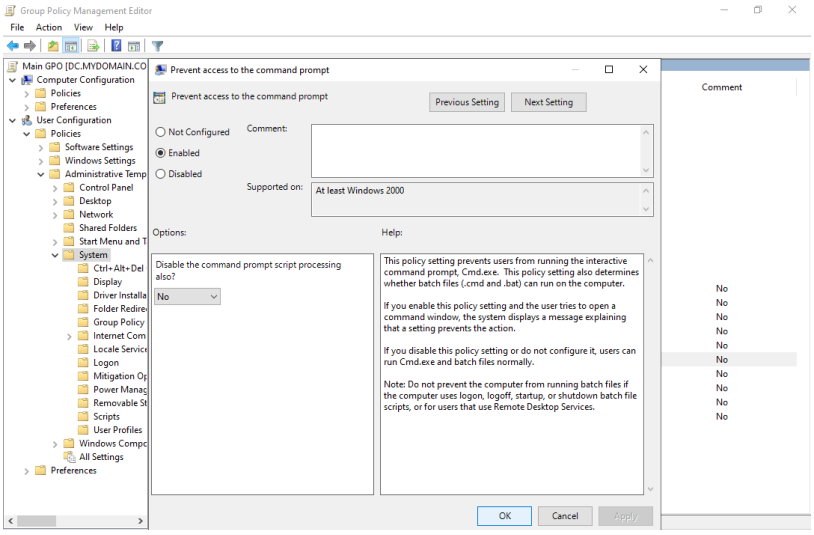
4. We right-click and select New and name our new GPO to Main GPO.



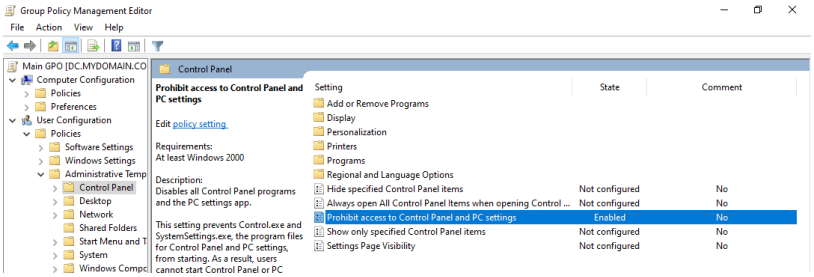
5. After creating the new GPO, right-click on it and choose Edit.

6. In the opened window, the Group Policy Management Editor window, we configure various settings for the GPO. For the purposes of this lab, we focus on some fundamental yet crucial configurations.

   - **Restrict access to the CMD and PowerShell**: Disabling command-line tools reduces the risk of unintended system changes, malicious scripts, and potential security breaches. This restriction is especially important in environments where users have limited administrative privileges.
     - In the Group Policy Management Editor, expand User Configuration.
     - Navigate to Policies → Administrative Templates → System.
     - Set the following policy to "Enabled": Prevent access to the command prompt.



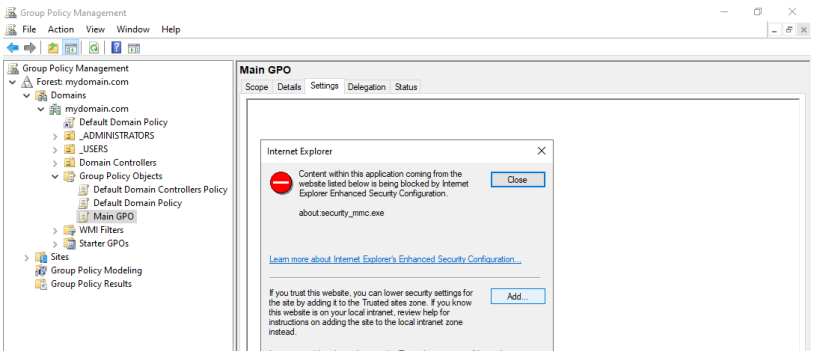
- **Limiting access to Control Panel options**: By restricting access to Control Panel features, administrators can prevent users from making unintended system changes, enhance security by minimizing the risk of unauthorized modifications, and ensure compliance with organizational policies.
  - In the Group Policy Management Editor, navigate to User Configuration → Policies → Administrative Templates → Control Panel.
  - Set the following policy to ”Enabled”: Prohibit access to the Control Panel and PC Settings.



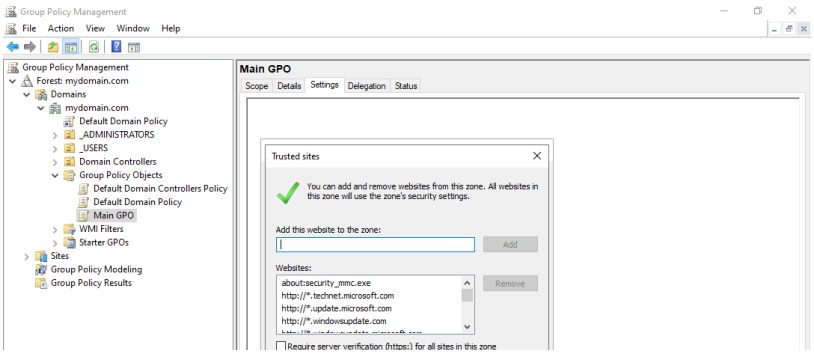
- **Limiting access to Removable Media**: This setting is crucial for several reasons, primarily focusing on security and data integrity. Allowing users to freely connect USB drives, external hard drives, or insert CDs/DVDs poses potential risks to your network.
  - In the Group Policy Management Editor, navigate to User Configuration → Policies → Administrative Templates → System → Removable Storage Access.
  - Set the following policy to ”Enabled”: All Removable Storage classes: Deny all access.
- **Limit Access to the Registry**: Altering registry settings can have significant consequences for system stability and security. Unauthorized changes might lead to system malfunctions, security vulnerabilities, or unintended behavior. Limiting access to the registry is crucial for maintaining a secure and stable computing environment, especially for user accounts where users should not have the privilege to modify critical system settings.
  - In the Group Policy Management Editor, navigate to User Configuration → Administrative Templates → System.
  - Set the following policy to ”Enabled”: Prevent access to registry editing tools.

7. To review the configured settings of the Group Policy Object (GPO) that we created, the Main GPO, with the GPO Editor open, we click on the Settings tab.

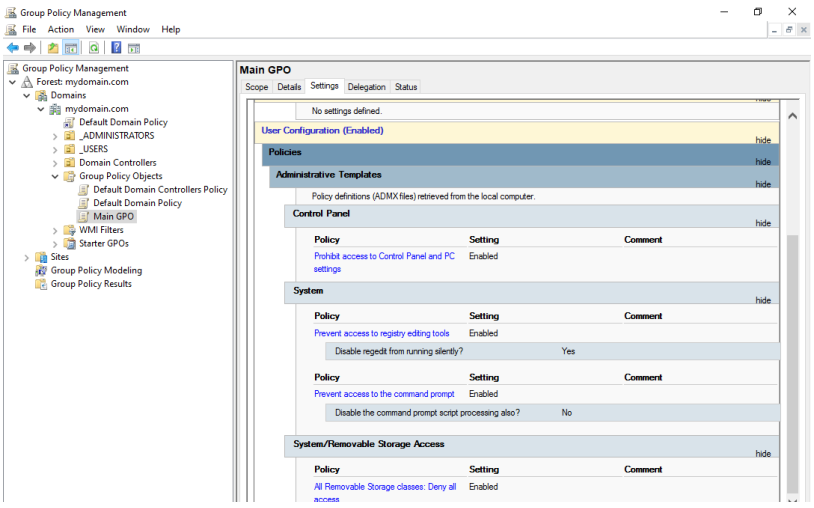
   - We select Add on the new window that appears.



   - We select Add and Close in order to add the website to the zone.

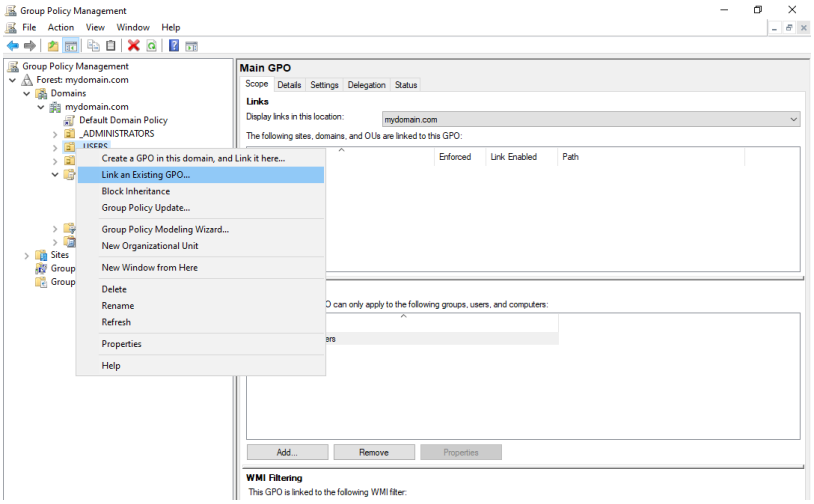


   - Finally, we can see the report of our GPO in the settings tab.

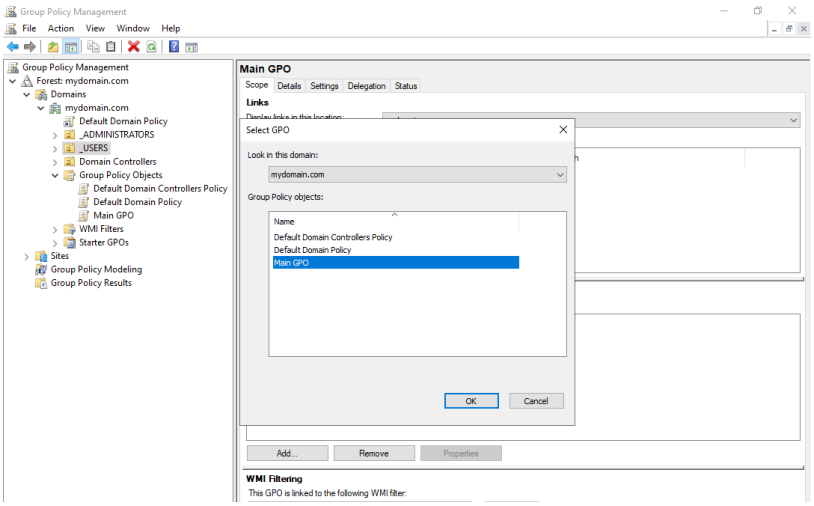


8. Now, we have to link this GPO with the USERS OU, to do that we execute the following steps:

   - On Group Policy Management window, we right-click on our _USERS OU and select Link an Existing GPO.

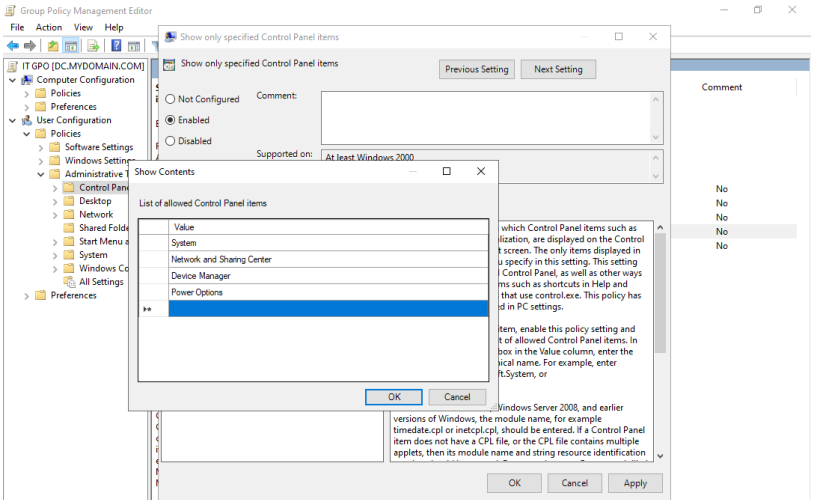


- On the new window that appears, we select the Main GPO object in order to link with the OU.



9. The GPO for the IT department should balance security restrictions with the operational needs of IT professionals. While it's essential to maintain a secure environment, IT staff often require elevated privileges for troubleshooting and system administration. We create a Group Policy Object (GPO), called IT GPO.

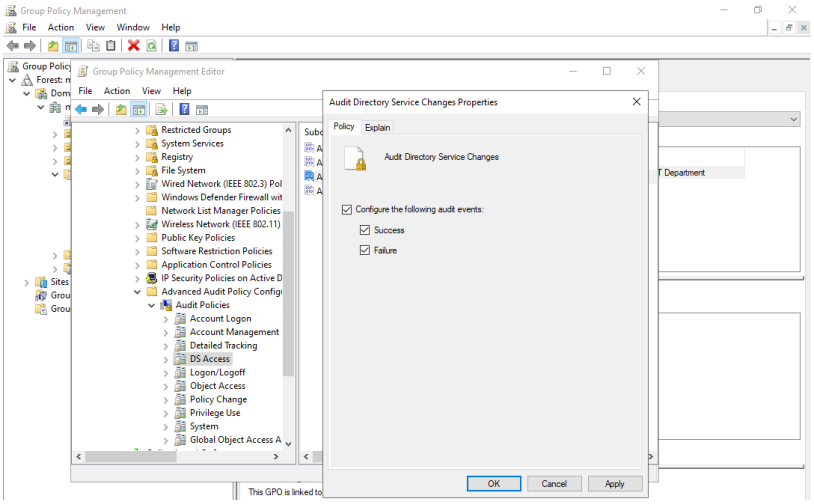
   - **Allow Access to CMD and PowerShell**: We do this in order to facilitate troubleshooting, scripting, and system administration tasks. To apply this we navigate to User Configuration → Policies → Administrative Templates → System and we set the Restrict access to the CMD and PowerShell as "Disabled".
   - **Allow Access to certain types of Removable Media**: We can configure this based on IT department needs. If we want to allow all removable storage classes to the IT department, we navigate to User Configuration → Policies → Administrative Templates → System → Removable Storage Access and we set the All Removable Storage classes: Deny all access as "Enabled".
   - **Allow Access to specified Control Panel Items**: We can configure this based on IT department needs. We navigate to User Configuration → Policies → Administrative Templates → Control Panel and select Show Only Specified Control Panel Items. We click on "Enabled" and we add the following Control Panel Items: System, Network and Sharing Center, Device Manager and Power Options. We click Apply and OK.



- Finally we link the IT GPO to the IT Department OU just like we did for the Main GPO before.

10. While we have discussed GPOs for user accounts and specific departments, it's equally important to implement GPOs for computers. This ensures a comprehensive security framework covering aspects such as audit logs, password policies, account lockout policies, software installation control, and more. Some GPOs for Computers are:

    - **Password Policy**: Strengthen password security on computers. In the GPO, we can navigate to Computer Configuration → Policies → Windows Settings → Security Settings → Account Policies and set the Password Policy parameters. For example, we can set the minimum password length to 12.
    - **Software Installation Control**: Regulate software installation on computers. To configure this setting we can navigate to Computer Configuration → Windows Settings → Security Settings → Local Policies → Security Options and set the Prohibit User Installs to "Enabled".
    - **Prevent Storing LAN Manager Hash**: Enhance password security by preventing the storage of LAN Manager (LM) hashes. We configure this in Computer Configuration → Windows Settings → Security Settings → Local Policies → Security Options.
    - **Enable Audit Logs**: The objective of enabling audit logs for directory service changes and file system events is to capture detailed information about security events on computers. By implementing auditing, organizations can monitor and track changes within the Active Directory (AD) environment and file system. This enhanced visibility is critical for security, compliance, and troubleshooting purposes.
        - To enable this setting we navigate to Computer Configuration → Windows Settings → Security Settings → Advanced Audit Policy Configuration → Audit Policies → DS Access, set the Audit Directory Service Changes and check the boxes labeled Configure the following audit events, Success, and Failure.



This setting captures events related to changes in the Active Directory structure, such as additions, modifications, or deletions of objects like users, groups, or organizational units.

- Do the same in Computer Configuration → Windows Settings → Security Settings → Advanced Audit Policy Configuration → Audit Policies → Object Access for the Audit File System.
  This setting captures events related to file and folder access, modifications, or deletions within the file system.

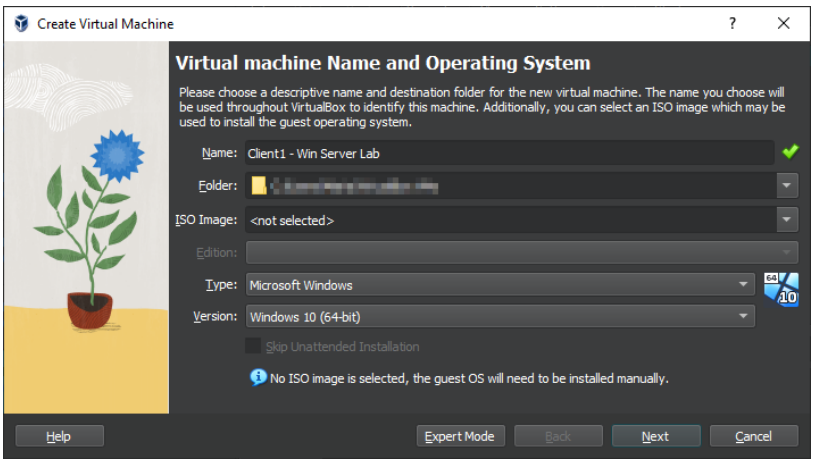
11. If the new Group Policy settings are not applied, in the Group Policy Management Console we right-click the OU in which the GPO was linked, and click Group Policy Update.

5 Client

In this section we proceed with the setup of the Windows 10 VM for the client.

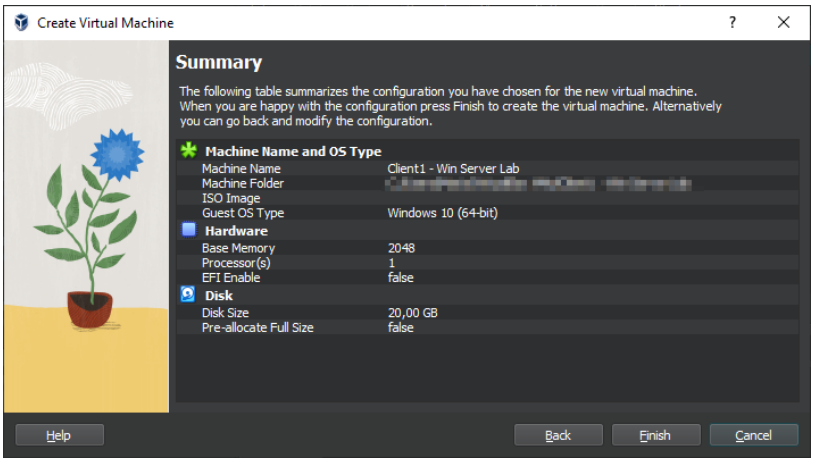
5.1 Installation

1. Download the Windows 10 iso from here.
2. Open Virtualbox and click on Create a New Virtual Machine.
3. Define the name of the VM to Domain Controller - Win Server Lab.
4. Select Windows 2022 64 bit as the guest operating system.

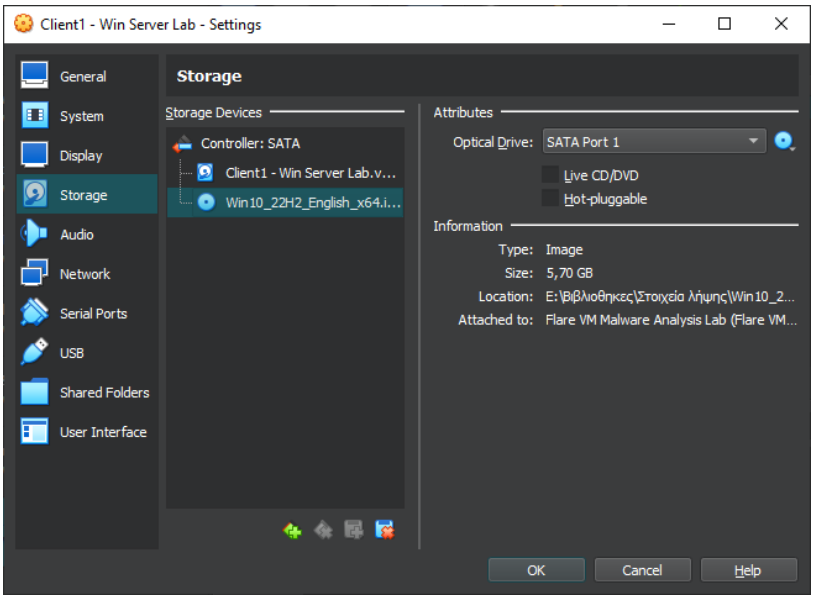


5. Configure the virtual hardware:

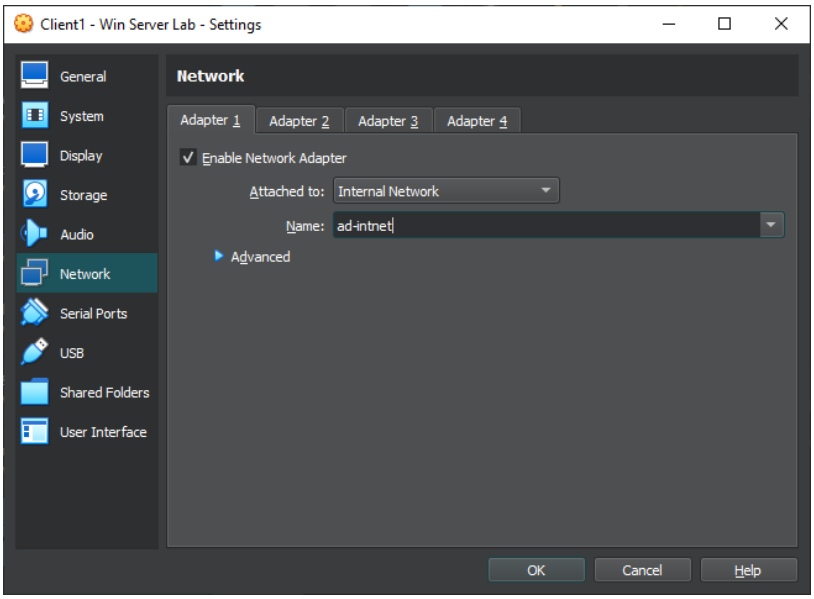
   - Set the disk size, in this implementation we use 40 GB.
   - Allocate 2 GB of RAM to the VM.



   - Add the ISO image that we downloaded earlier. To do so click Settings then Storage. Click on the Disk icon and click on Choose a disk file to attach the ISO file.



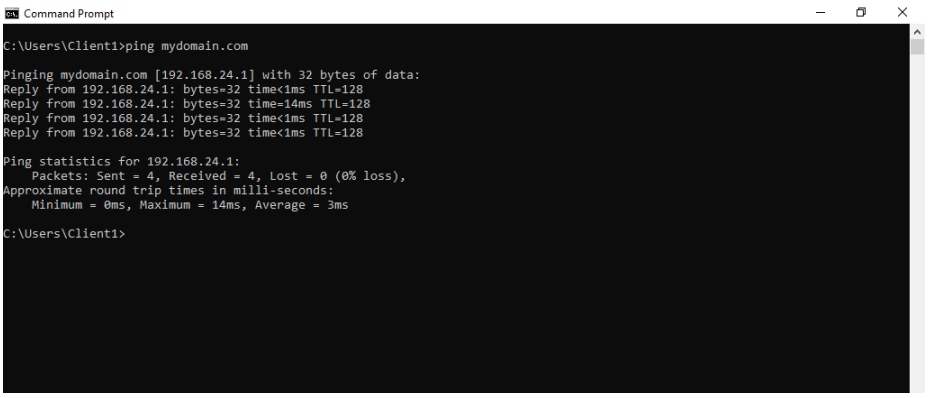
   - Now we need to configure the VM's Network adapters in VirtualBox. To set them click Settings then Network. We choose the Internal Network option and select the ad-inet.



6. Install Windows 10: Follow the on-screen instructions to install Windows 10.
    - When prompted, choose your language, time, and keyboard input preferences.
    - Enter ”Client1” as the username when prompted for user information.
7. Continue with the Windows setup, including setting a password for the Client1 account. Install VirtualBox Guest Additions (Optional):
8. Install VirtualBox Guest Additions (Optional):
    - After Windows 10 is installed, you can install VirtualBox Guest Additions from the VirtualBox menu to enhance integration and performance.

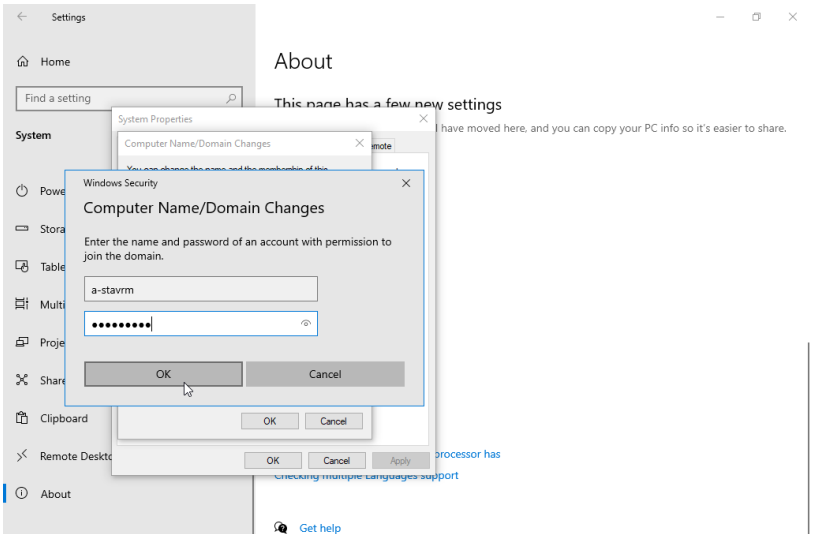
5.2 Configuration & Testing

1. Boot your Windows 10 VM.
2. Log in using the Client1 username and the password you set during the Windows setup.
3. Go to start and choose Command Prompt from the menu. To test the connection to your domain controller use the following command: `ping mydomain.com`.

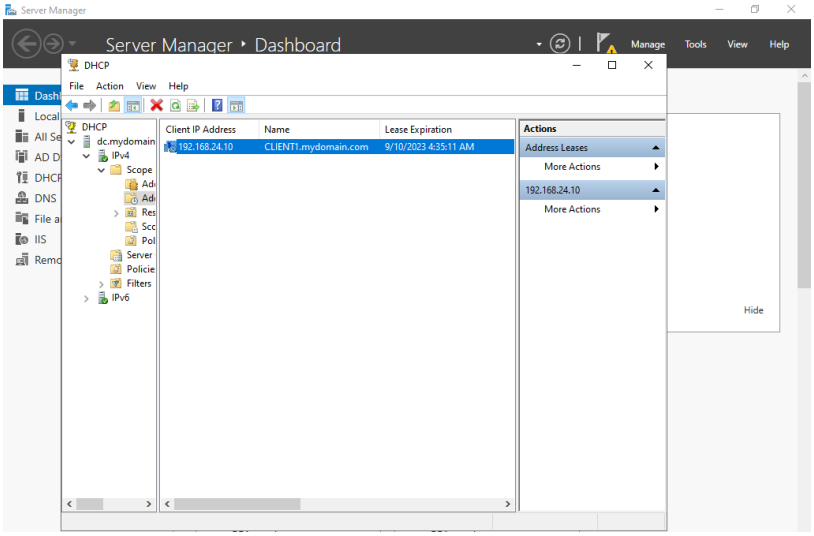


4. Click on Settings in the Start menu. This will open the Windows Settings app.
5. In the Windows Settings window, click on the System category.

6. In the About section, under the Related Settings heading, click on the Rename this PC (advanced) button.
7. Enter CLIENT1 as the new name for your PC.
8. Enter mydomain.com as the domain.
9. In the Computer Name/Domain Changes we enter the name and password of the account that will join the domain.



10. On the AD server's VM open the DHCP from the Tools option in the Server Manager.
11. In the DHCP window select the IPv4 and the client appears.



6 Powershell Automation

6.1 Creating Users

Our PowerShell script takes advantage of the Active Directory module to automate the user account creation process. The script utilizes a CSV file that serves as a data source for user account creation. The CSV file includes columns that map to essential user attributes. Each row in the CSV file represents a user account to be created.

To create this script we execute the following steps:

1. To create the CSV file for user account creation, we include the following attributes for each user:
    - **SamAccountName**: The username used for logging into the domain.
    - **FirstName**: The user's first name.
    - **LastName**: The user's last name.
    - **DisplayName**: The full name or display name of the user.
    - **Department**: The user's department.
    - **JobTitle**: The job title of the user.
2. In order to write the script we open up the Powershell ISE. Click on the Start button in the Windows taskbar to open the Start Menu, type and click on PowerShell ISE.
3. We write the script as follows:
    - **Import Required Module**: It starts by importing the Active Directory module, which provides the necessary cmdlets for managing Active Directory objects.

      ```
      Import-Module ActiveDirectory
      ```

    - **Import User Data**: The script imports user data from a CSV file named `users_data.csv`. This CSV file contains user information in a structured format, as described previously.

      ```
      $UserData = Import-Csv "users_data.csv"
      ```

    - **User Creation Loop**: The script enters a loop to process each user's data one by one. Inside the loop, the script extracts user attributes from the CSV data.

      ```
      foreach ($User in $UserData) {
          # Extract user attributes from CSV data
          $FirstName = $User.FirstName
          $LastName = $User.LastName
          $JobTitle = $User.JobTitle
          $SamAccountName = $User.SamAccountName
          $DisplayName = $User.DisplayName
          $Department = $User.Department
          ...
      }
      ```

    - **Check User Existence**: Before creating a new user, the script checks whether a user with the same SAM account name already exists in Active Directory. If the user exists, it logs a warning and skips the user creation.

      ```
      $Exists = Get-ADUser -Filter {SamAccountName -eq
         $SamAccountName}
      if ($Exists) {
          Write-Warning "User $SamAccountName already exists in
         Active Directory."
          continue
      }
      ```

    - **User Object Creation**: The script constructs a hashtable (`$NewUser`) to store user attributes required for creating a new AD user account.

```
$NewUser = @{
    SamAccountName = $SamAccountName
    GivenName = $FirstName
    Surname = $LastName
    Name = "$FirstName $LastName"
    DisplayName = $DisplayName
    Title = $JobTitle
    Department = $Department
    Path = $OUPath
    AccountPassword = (ConvertTo-SecureString "P@ssw0rd" -
   AsPlainText -Force)
    Enabled = $true
    ChangePasswordAtLogon = $true
}
```

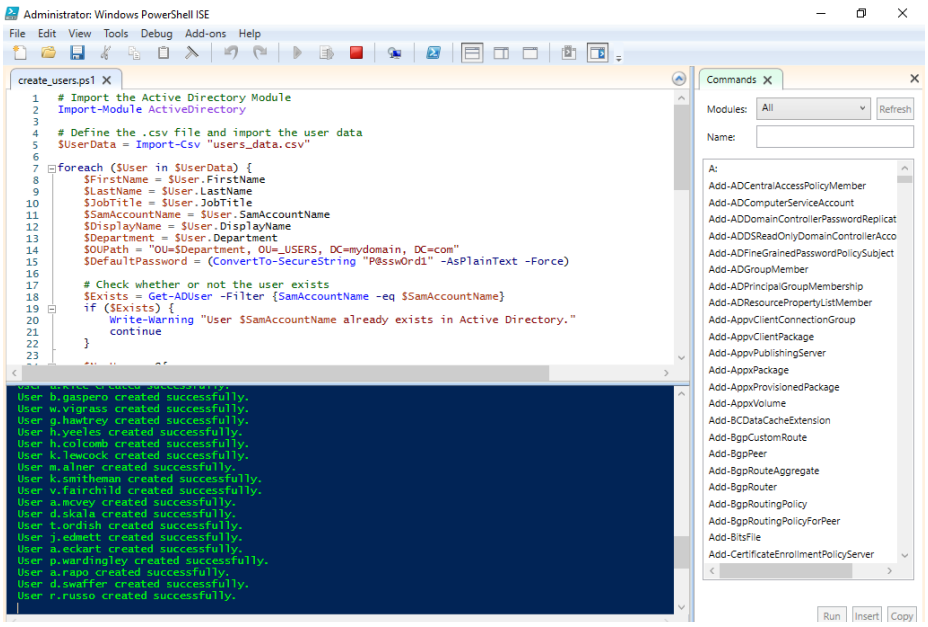
- **User Creation**: The script attempts to create a new AD user using the New-ADUser cmdlet, providing the attributes stored in the `$NewUser` hashtable.

```
        New-ADUser @NewUser
```

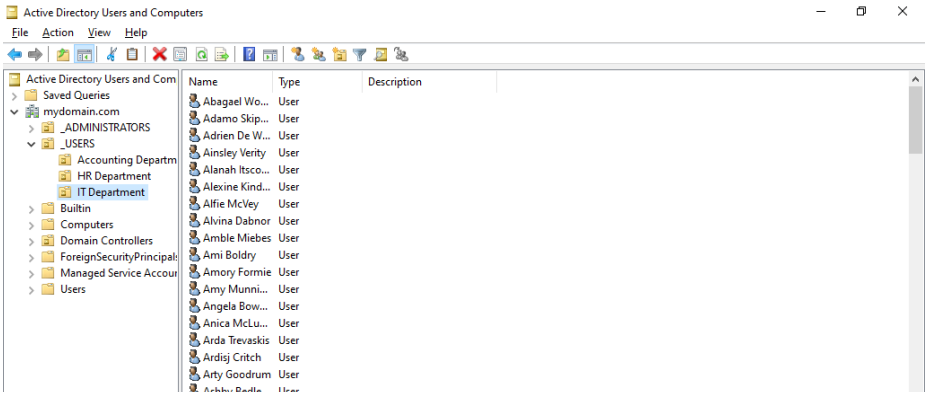
- **Logging and Error Handling**: The script logs the success or failure of user creation and provides appropriate feedback.

```
try {
    # Create a new AD user
    New-ADUser @NewUser
    Write-Host "User $SamAccountName created successfully
   ." -ForegroundColor Green
} catch {
    Write-Warning "Failed to create user $SamAccountName."
}
```

4. We run our script in PowerShell ISE by clicking on the Run button. As we can see in the image below, the users are being created successfully.



5. Now, we can verify this by going again to the Active Directory Users and Computers. Each department now has the corresponding users from the CSV file.



6.2 Disabling Inactive User Accounts

Disabling inactive user accounts in Active Directory is an important security practice to moderate potential security risks. Automating this task can help ensure that user accounts are promptly disabled when they are no longer in use.

1. We the create the `_INACTIVE` Organisational Unit (OUs). This OU contains the inactive users in the AD. In order to create it we execute the following steps:
   - Open Active Directory Users and Computers from Start → Active Directory Users and Computers.
   - In the left pane, right-click on our `_USERS` OU, click New → Organizational Unit.
2. To create this script we follow the steps below:
   - **Import Required Module**: Import the AD module, which provides the necessary cmdlets for managing Active Directory objects.

     ```
     Import-Module ActiveDirectory
     ```

   - **Define the inactivity threshold** for user accounts based on the configured `$InactiveDaysOld` and `$InactiveDaysNew`.

     ```
     $DaysInactiveThresholdOld = (Get-Date).AddDays(-
        $InactiveDaysOld)
     $DaysInactiveThresholdNew = (Get-Date).AddDays(-
        $InactiveDaysNew)
     ```

   - **Configure the search parameters** for identifying inactive user accounts in the specified `$UsersOU`.

     ```
     $InactiveUsersParam = @{
          UsersOnly = $true
          SearchBase = $UsersOU
          AccountInactive = $true
     }
     ```

   - **Find and separate** inactive old and new users based on the defined thresholds.

     ```
     $InactiveOldUsers = Search-ADAccount @InactiveUsersParam |
         Where-Object { $_.LastLogonDate -ne $null -and $_.
        LastLogonDate -lt $DaysInactiveThresholdOld }
     $InactiveNewUsers = Search-ADAccount @InactiveUsersParam |
         Where-Object { $_.LastLogonDate -eq $null -and $_.
        WhenCreated -lt $DaysInactiveThresholdNew }
     ```

- **Disable** the inactive user accounts and move them to the `_INACTIVE` OU.

```
$InactiveOldUsers | Disable-ADAccount -PassThru
$InactiveOldUsers | Move-ADObject -TargetPath $InactiveOU
$InactiveNewUsers | Disable-ADAccount -PassThru
$InactiveNewUsers | Move-ADObject -TargetPath $InactiveOU
```

- **Define new search parameters** for users to be removed from the `_INACTIVE` OU based on the specified `$InactiveDays`.

```
$SearchParam = @{
     UsersOnly = $true
     SearchBase = $InactiveOU
     AccountInactive = $true
     TimeSpan = $InactiveDays
}
```

- **Remove user accounts** from the `_INACTIVE` OU if they have been inactive for more than `$InactiveDays`.

```
$InactiveUsers = Search-ADAccount @SearchParam
$InactiveUsers | Remove-ADUser
```

6.3 Creating User Reports

Creating a Reporting and Auditing automation script for Active Directory (AD) involves generating reports that provide insights into your AD environment's status, user account information, and more. In this lab, we create a user report automation script using Powershell.

1. We create the script by executing the following steps.

   - **Import Required Module**: Import the AD module, which provides the necessary cmdlets for managing Active Directory objects.

```
Import-Module ActiveDirectory
```

   - **Define the OU** to the Users Organizational Unit (OU) in Active Directory where the user search will be performed.

```
# Specify OU and file path
$UsersOU = 'OU=_USERS,DC=mydomain,DC=com'
```

   - **Retrieve the current date** and time and format it as a string in the `dd-MM-yyyy_HH-mm-ss` format. This formatted date will be used in the filename of the exported CSV report.

```
$CurrentDate = Get-Date -Format 'dd-MM-yyyy_HH-mm-ss'
```

   - **Specify the file path** where the user report will be saved. The filename includes the current date and time, ensuring that each report has a unique name.

```
$FilePath = "C:\ADReports\User-Reports\
   User_Report_$CurrentDate.csv"
```

   - **Define the attributes** you want to include in the user report. `$Attributes` is an array containing the attribute names. `$DepartmentFilter` allows you to specify a filter for the Department attribute during the search. By default, it matches any Department using the regular expression `.*`.

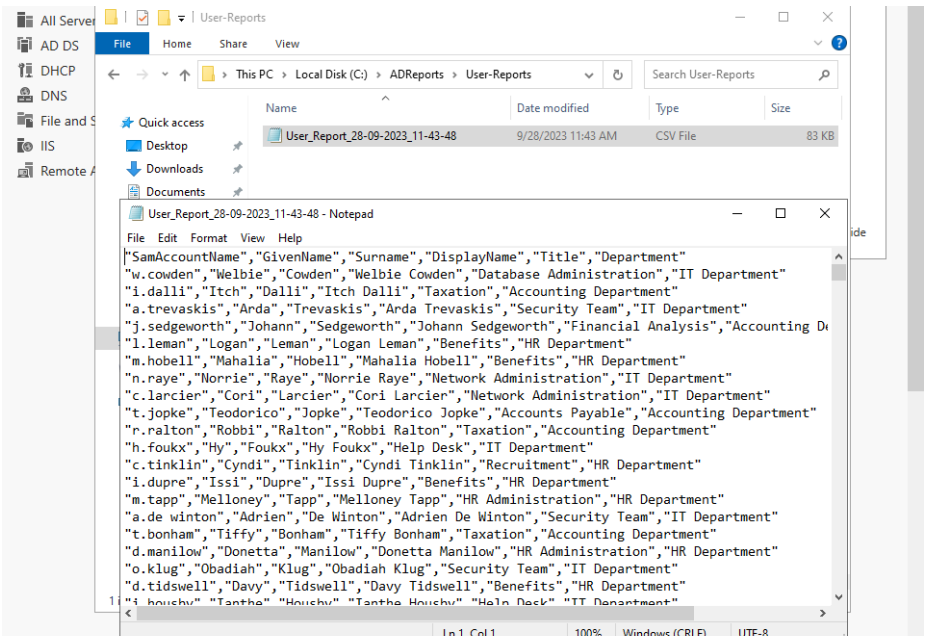
```
# Specify attributes and the filters to include in the
    search
$Attributes = 'SamAccountName', 'GivenName', 'Surname', '
    DisplayName', 'Title', 'Department'
$DepartmentFilter = ".*"
```

- The section below performs the following tasks:

```
# Create a user report
Get-ADUser -Filter * -SearchBase $UsersOU -Properties
    $Attributes |
Where-Object { $_.Department -match $DepartmentFilter } |
Select-Object -Property $Attributes |
Export-Csv $FilePath -NoTypeInformation
```

  - `Get-ADUser` retrieves user objects from Active Directory based on the specified filter and search base. It also loads the specified user attributes using the -Properties parameter.
  - `Where-Object` filters the retrieved user objects based on the Department attribute matching the specified regular expression filter.
  - `Select-Object` selects the specified attributes to include in the final report.
  - `Export-Csv` exports the filtered and selected user data to a CSV file located at the specified file path. The -NoTypeInformation parameter omits the type information from the CSV file.

- Once we run the script we can see the new report file in the path that we specified earlier. It contains the user data as we intended.

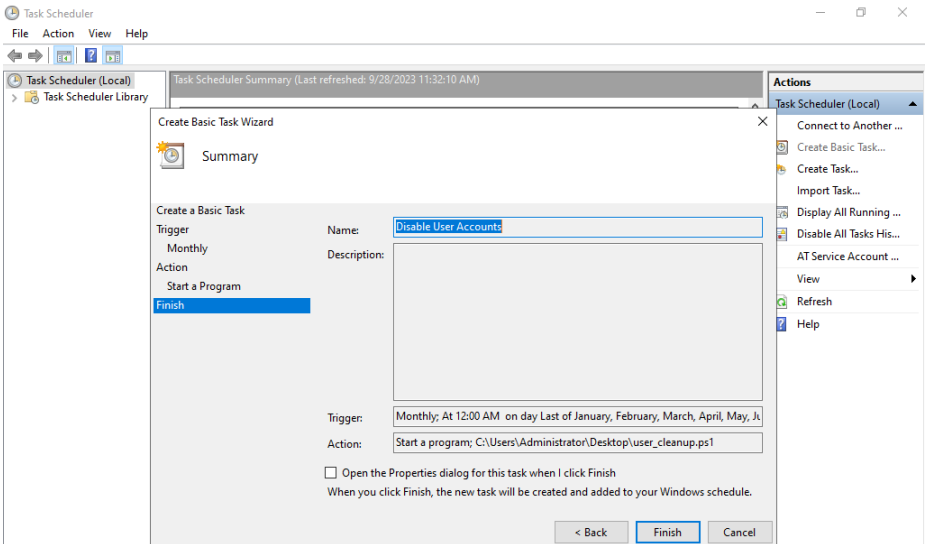


6.4 Automating Tasks

Running a script automatically at specific time intervals is a common practice for tasks like monitoring, reporting, or maintenance. In this section, we demonstrate the automated execution of the `user_cleanup.ps1` script. Here are the steps to schedule your PowerShell script(s):

1. Open Task Scheduler from Start → Task Scheduler.

2. Click on Create Basic Task in the Actions pane on the right.
3. Enter a name and description for your task. In our implementation is the name ”Disable User Accounts” and we leave the description empty, then click Next.
4. We set the trigger depending on our preference, we choose Monthly in this implementation.
5. Select Start a program, we browse for the script and select it.



In addition to handling user-related tasks, similar scripts exist to manage computers within the Active Directory environment.

7 Conclusions

In conclusion, in this lab, we successfully implemented an Active Directory Domain Controller (AD DC), configured Remote Access Service (RAS) and DHCP settings, established an administrative account, and deployed a new client virtual machine. Additionally, we created and linked Group Policy Objects (GPOs) to specific Organizational Units (OUs) within our Active Directory structure. These GPOs, tailored for various OUs, encompassed security configurations, user access restrictions, and other policies vital for maintaining a secure and well-organized IT infrastructure. Furthermore, we automated user management tasks using PowerShell scripts, ensuring streamlined processes for user creation, inactive user identification, and the generation of user reports in CSV format. By using the Task Scheduler to automate scripts, we improved efficiency and set the stage for future scalability and maintenance of our Active Directory environment.