Assignment: Network Setup and Health Monitoring Task Scheduling

Objective:

Your task is to design and implement a complete network for health monitoring using **PowerShell**. This will involve configuring two servers and one client. You must demonstrate this assignment and be prepared to explain your script in class.

Requirements:

1. **Infrastructure Setup: (20 points)**

   o **Servers:** You will need two servers (**Server1** and **Server2**) running **Windows Server 2022 Core**.

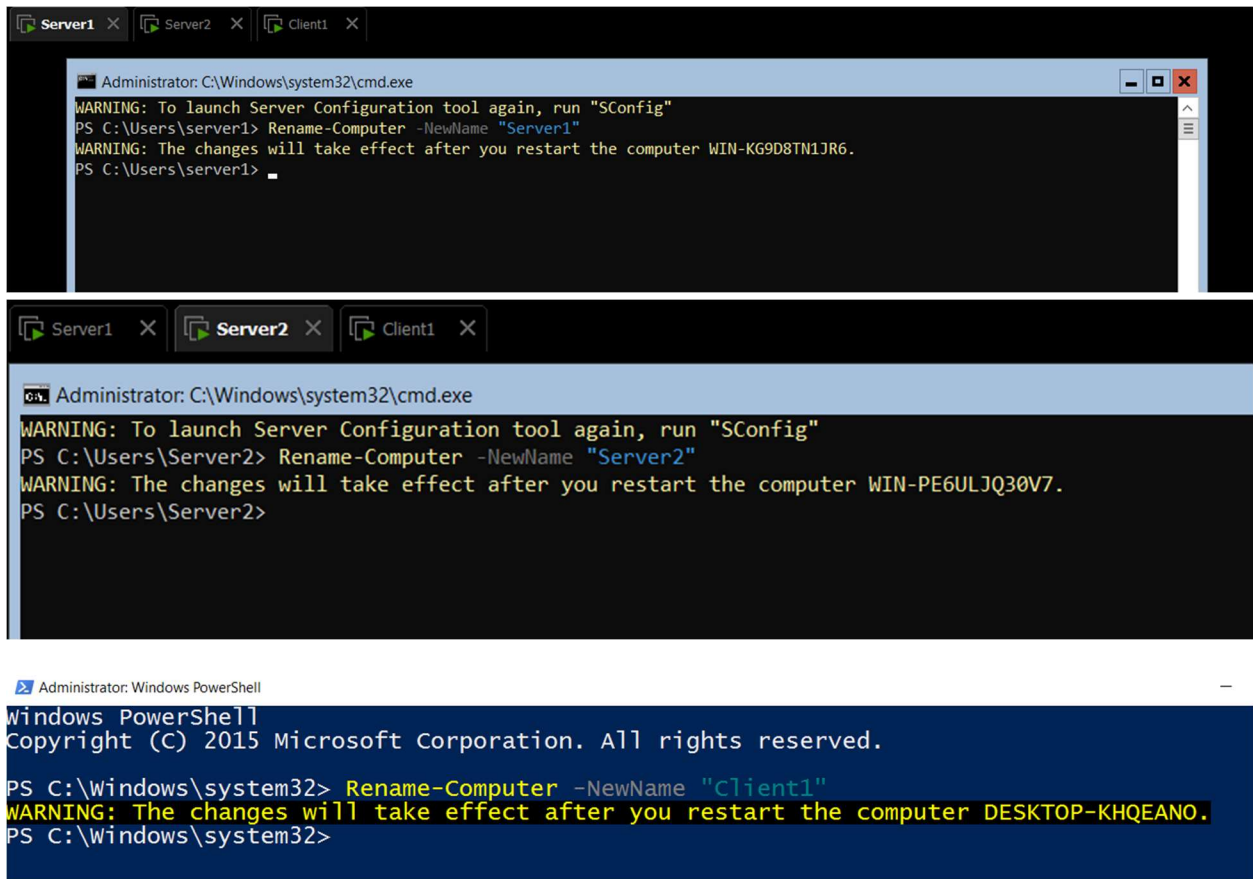   o **Client:** The client (**Client1**) can be either **Windows 10** or **Windows 11**.

   Paste screenshot of VMware displaying the running virtual machines below.
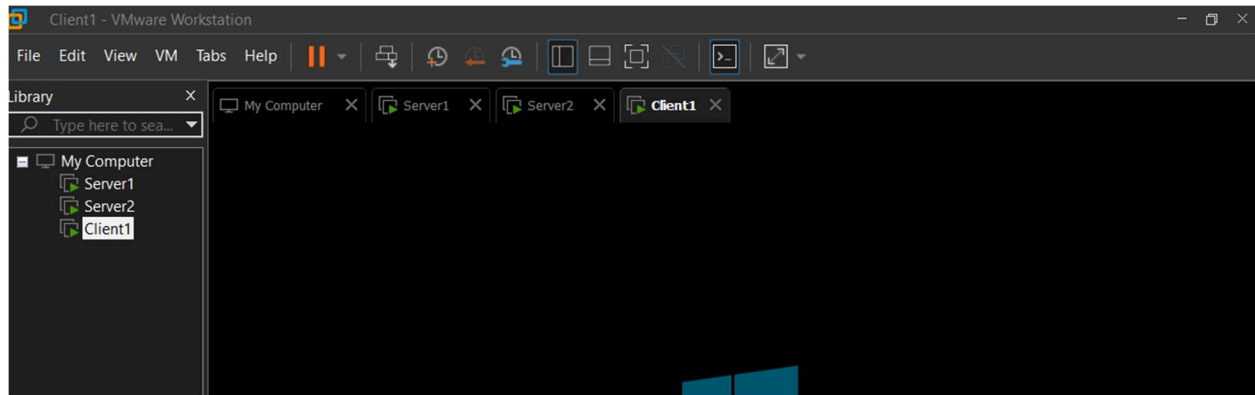
Assignment: Network Setup and Health Monitoring Task Scheduling

2. **Assign the following hostnames using PowerShell: (15 points)**

   o **Server1**

   o **Server2**

   o **Client1**

   o Paste screenshot showing the hostnames changed on each VM using PowerShell below.

Assignment: Network Setup and Health Monitoring Task Scheduling

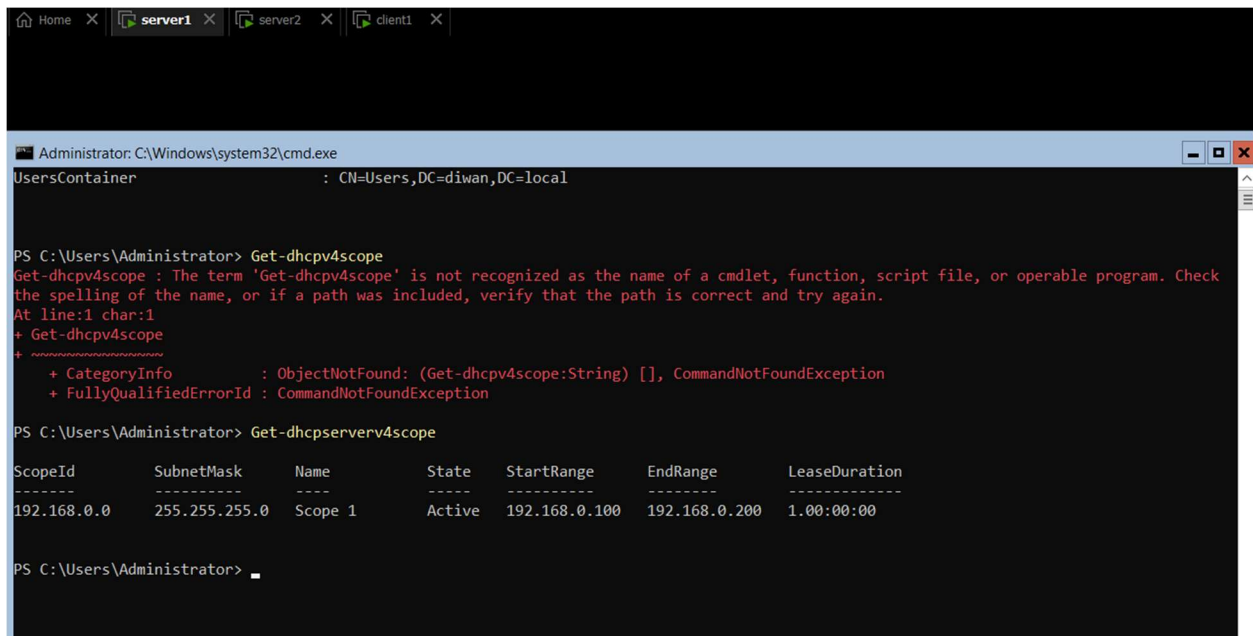3. **Domain Configuration using PowerShell: (25 points)**

   o Set up **Server1** as a **Domain Controller (DC)** with the domain name **"lastname.local"**.

   o Setup **DNS** and **DHCP** on **Server1** using PowerShell.

   o Make sure the **Server2** and **Client1** get IP Assignment via DHCP.

   o Join **Server2** and **Client1** to the domain using PowerShell.

   <mark>Provide screenshot demonstrating the setup of the domain, DNS, and DHCP, as well as joining a domain using PowerShell. Show that **Server2** and **Client1** successfully receive IP addresses assigned via DHCP configured on **Server1**.</mark>

```
PS C:\Users\Administrator> Get-ADDomain


AllowedDNSSuffixes              : {}
ChildDomains                    : {}
ComputersContainer              : CN=Computers,DC=diwan,DC=local
DeletedObjectsContainer         : CN=Deleted Objects,DC=diwan,DC=local
DistinguishedName               : DC=diwan,DC=local
DNSRoot                         : diwan.local
DomainControllersContainer      : OU=Domain Controllers,DC=diwan,DC=local
DomainMode                      : Windows2016Domain
DomainSID                       : S-1-5-21-2764522379-3804765367-4094371693
ForeignSecurityPrincipalsContainer : CN=ForeignSecurityPrincipals,DC=diwan,DC=local
Forest                          : diwan.local
InfrastructureMaster            : Server1.diwan.local
LastLogonReplicationInterval    :
LinkedGroupPolicyObjects        : {CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System,DC=diwan,DC=local}
LostAndFoundContainer           : CN=LostAndFound,DC=diwan,DC=local
ManagedBy                       :
Name                            : diwan
NetBIOSName                     : DIWAN
ObjectClass                     : domainDNS
ObjectGUID                      : b9db22f3-688c-45ef-b64a-4de4154529ba
ParentDomain                    :
PDCEmulator                     : Server1.diwan.local
PublicKeyRequiredPasswordRolling : True
QuotasContainer                 : CN=NTDS Quotas,DC=diwan,DC=local
ReadOnlyReplicaDirectoryServers : {}
ReplicaDirectoryServers         : {Server1.diwan.local}
```

Assignment: Network Setup and Health Monitoring Task Scheduling


```
Home    X    server1   X    server2   X    client1   X

Administrator: C:\Windows\system32\cmd.exe                                    _ □ ✕
UsersContainer                     : CN=Users,DC=diwan,DC=local


PS C:\Users\Administrator> Get-dhcpv4scope
Get-dhcpv4scope : The term 'Get-dhcpv4scope' is not recognized as the name of a cmdlet, function, script file, or operable program. Check
the spelling of the name, or if a path was included, verify that the path is correct and try again.
At line:1 char:1
+ Get-dhcpv4scope
+ ~~~~~~~~~~~~~~~
    + CategoryInfo          : ObjectNotFound: (Get-dhcpv4scope:String) [], CommandNotFoundException
    + FullyQualifiedErrorId : CommandNotFoundException

PS C:\Users\Administrator> Get-dhcpserverv4scope

ScopeId        SubnetMask      Name      State     StartRange       EndRange         LeaseDuration
-------        ----------      ----      -----     ----------       --------         -------------
192.168.0.0    255.255.255.0   Scope 1   Active    192.168.0.100    192.168.0.200    1.00:00:00


PS C:\Users\Administrator> _
```
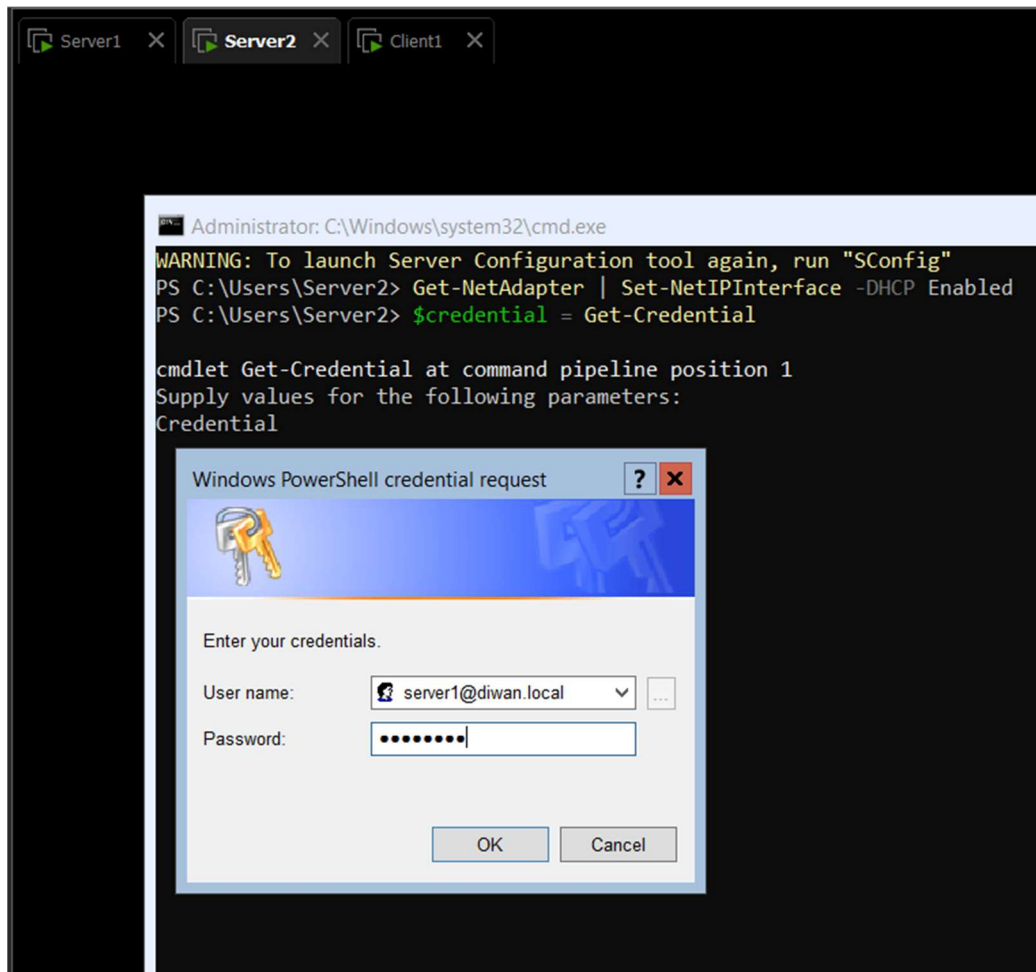


```
Server1    X    Server2    X    Client1    X

Administrator: C:\Windows\system32\cmd.exe
WARNING: To launch Server Configuration tool again, run "SConfig"
PS C:\Users\Server2> Get-NetAdapter | Set-NetIPInterface -DHCP Enabled
PS C:\Users\Server2> $credential = Get-Credential

cmdlet Get-Credential at command pipeline position 1
Supply values for the following parameters:
Credential
```
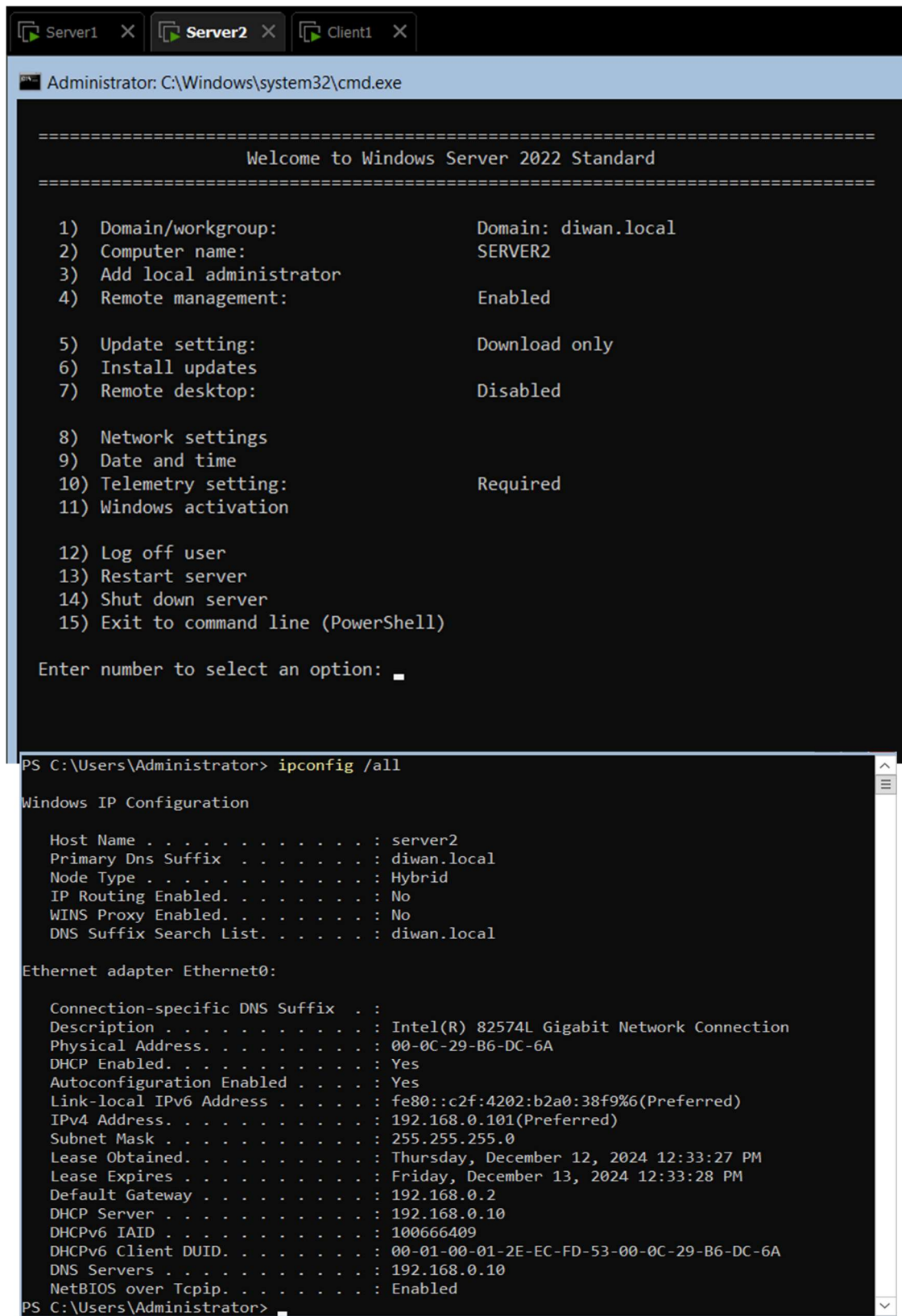
Windows PowerShell credential request

Enter your credentials.

User name: server1@diwan.local

Password: ••••••••

OK   Cancel

Assignment: Network Setup and Health Monitoring Task Scheduling

Assignment: Network Setup and Health Monitoring Task Scheduling

Assignment: Network Setup and Health Monitoring Task Scheduling

4. **Organizational Units: (20 points)**

   o Create a **Staff OU**.

   o Inside the Staff OU, create the following nested OUs:

      o **HR**

      o **Admin**

      o **IT**

   o Create **six domain users** using csv and distribute them into the respective nested OUs.

   o Install **RSAT** Tools on Windows 10/11 and access the OU Structure.

Assignment: Network Setup and Health Monitoring Task Scheduling

Paste screenshot of creating a parent Organizational Unit (OU) and a nested OU. Provide screenshot showing the script running and successfully creating six users along with the screenshot of the script and the csv file. Provide screenshot of installing RSAT tools on Windows 10 using PowerShell
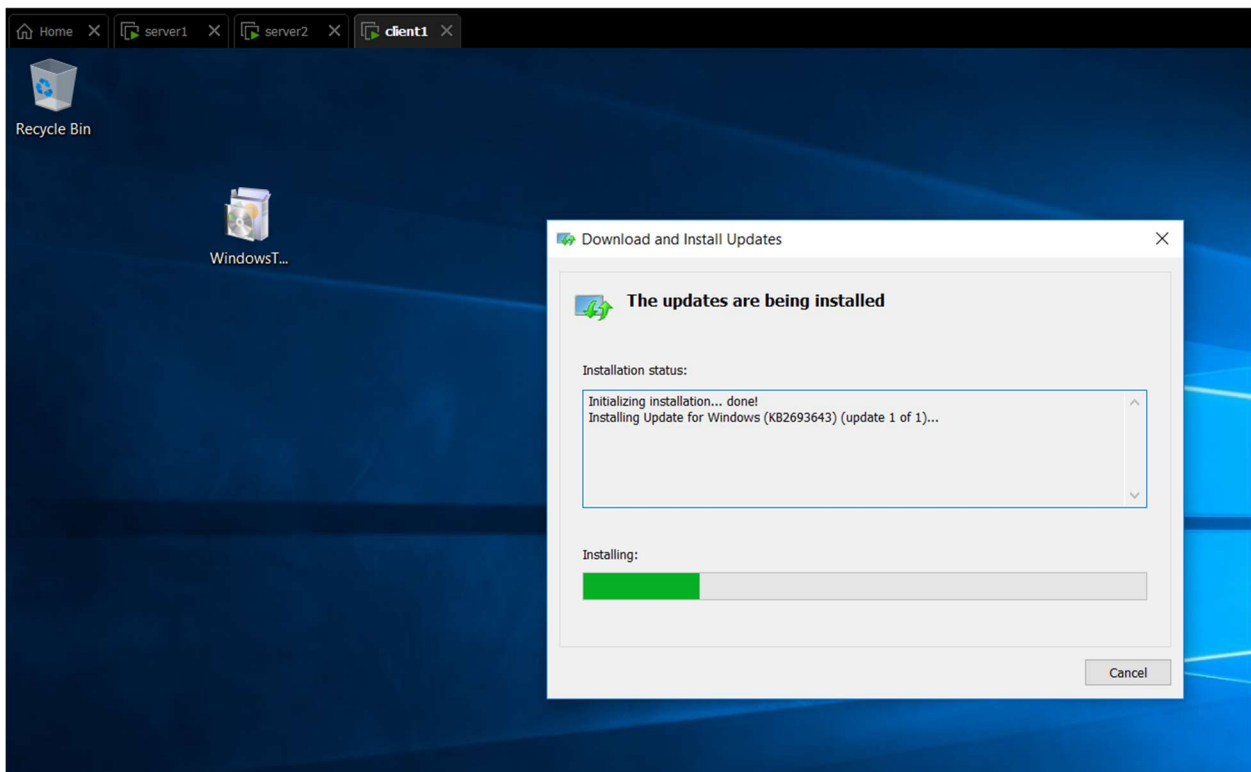
Assignment: Network Setup and Health Monitoring Task Scheduling
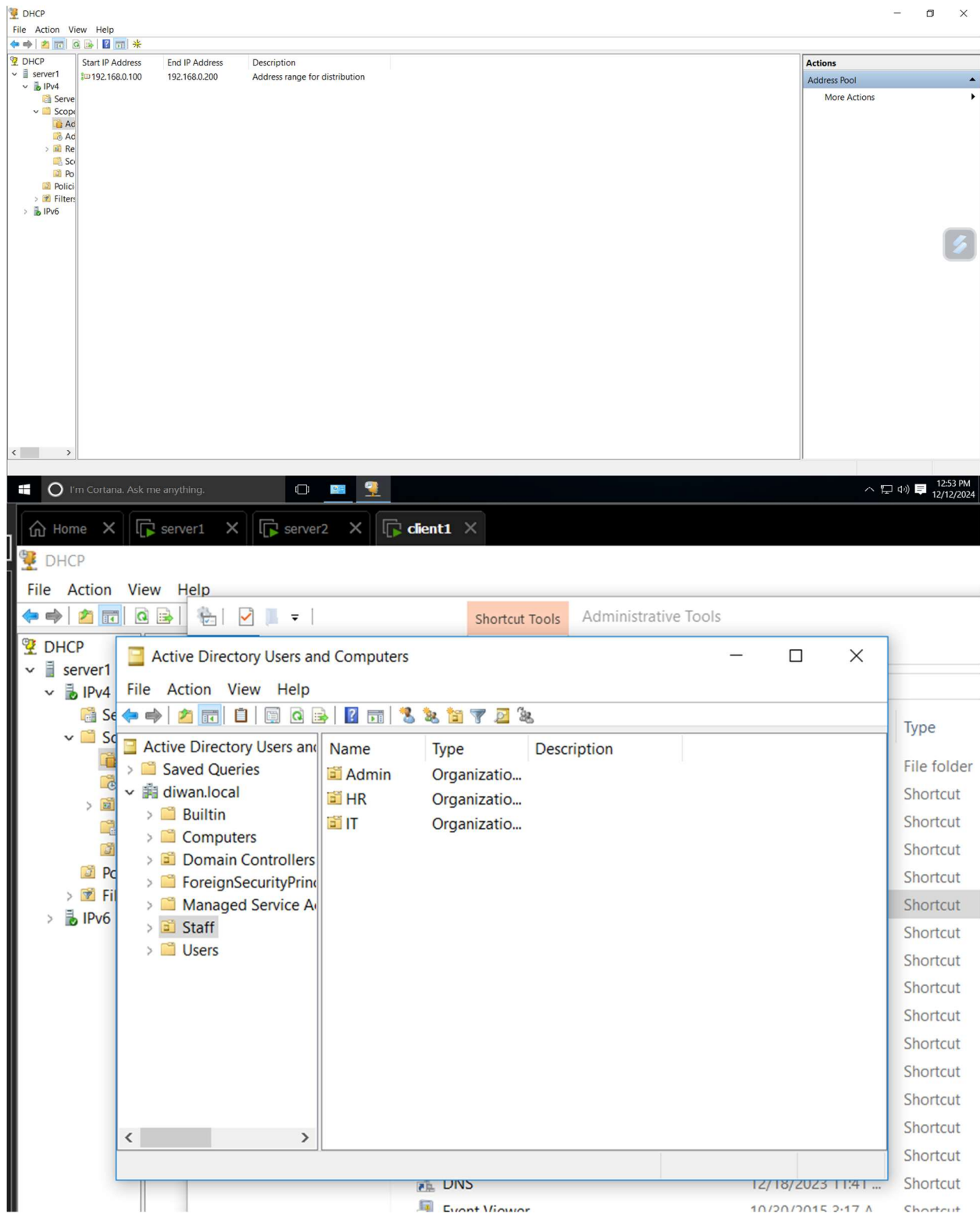
Assignment: Network Setup and Health Monitoring Task Scheduling

User.csv - Notepad

File Edit Format View Help

```
Name,SamAccountName,Department,Password
John Doe,jdoe,HR,P@ssw0rd1
Jane Smith,jsmith,Admin,P@ssw0rd2
Bob Johnson,bjohnson,IT,P@ssw0rd3
Alice Brown,abrown,HR,P@ssw0rd4
Charlie Davis,cdavis,Admin,P@ssw0rd5
Eve Wilson,ewilson,IT,P@ssw0rd6
```

importusers.ps1*

```powershell
 1   # Import the CSV file
 2   $users = Import-Csv "C:\Users\administrator\Desktop\User.csv"
 3
 4   # Define the OUs
 5   $ous = @(
 6       "OU=HR,OU=Staff,DC=diwan,DC=local",
 7       "OU=Admin,OU=Staff,DC=diwan,DC=local",
 8       "OU=IT,OU=Staff,DC=diwan,DC=local"
 9   )
10
11   # Loop through each user in the CSV
12   foreach ($user in $users) {
13       # Convert the password to a secure string
14       $password = ConvertTo-SecureString $user.Password -AsPlainText -Force
15
16       # Select the OU based on the user's department
17       $ou = $ous | Where-Object { $_ -like "*$($user.Department)*" }
18
19       # Create the user
20       New-ADUser -Name $user.Name `
21                   -SamAccountName $user.SamAccountName `|
22                   -UserPrincipalName "$($user.SamAccountName)@diwan.local" `
23                   -Path $ou `
24                   -AccountPassword $password `
25                   -Enabled $true `
26                   -ChangePasswordAtLogon $true
27
28       Write-Host "Created user $($user.Name) in $ou"
29   }
30
```

Assignment: Network Setup and Health Monitoring Task Scheduling
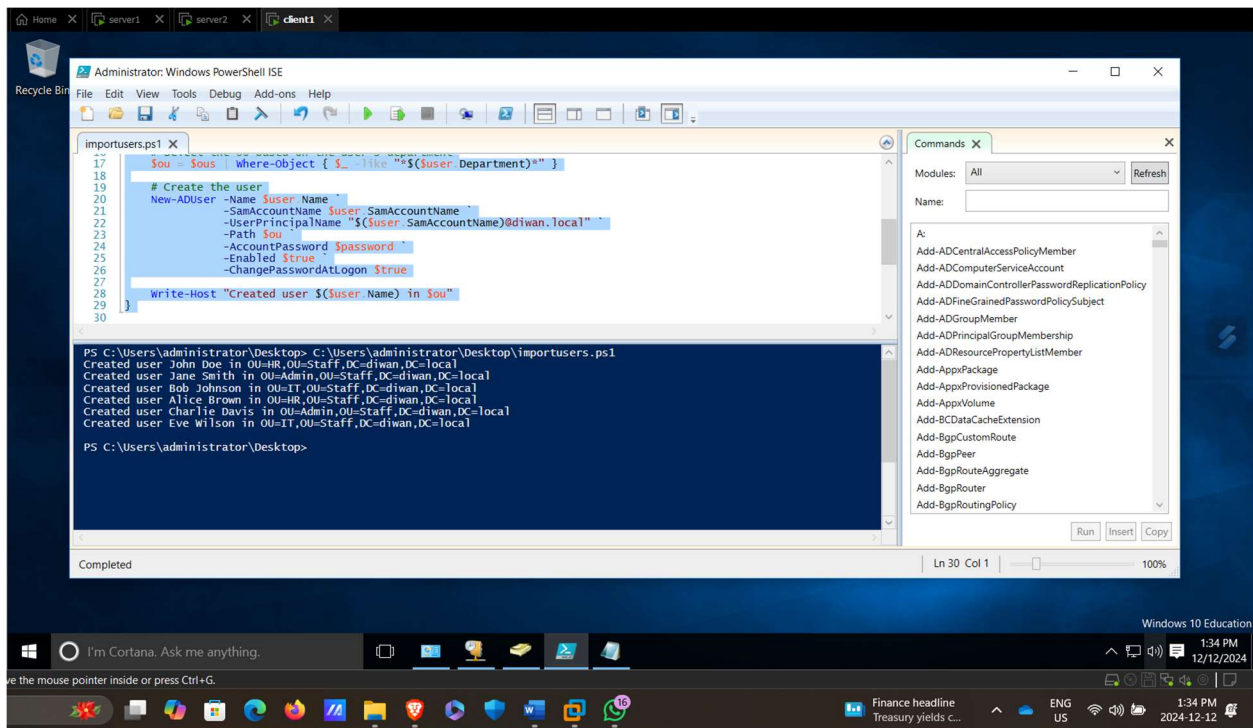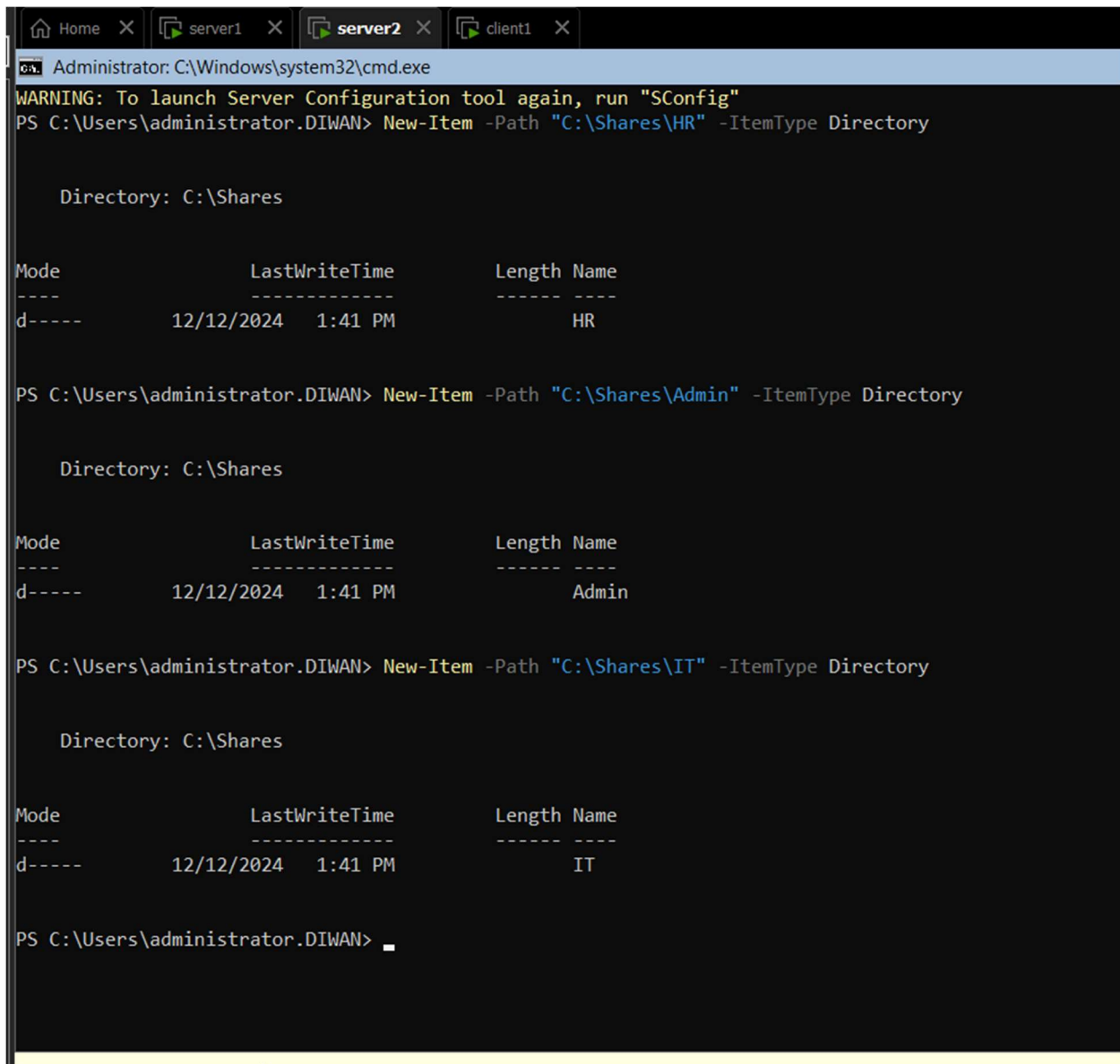
5. **File Sharing Setup: (20 Points)**

- o Set up file sharing on **Server2**.

- o Create folders for **HR**, **Admin**, and **IT** to mirror the OU structure.

- o **Create User Groups:**

  - Establish groups: **HR_group**, **Admin_group**, and **IT_group**.
  - Randomly assign users to these groups.

- o Set Access Permissions:
- o Ensure each group has access only to their respective folder:

  - **HR_group** → HR share
  - **Admin_group** → Admin share
  - **IT_group** → IT share

Provide screenshots of creating folders and groups, assigning users to groups, creating shared resources, and assigning permissions using PowerShell below.

Assignment: Network Setup and Health Monitoring Task Scheduling


```
Home   X    server1   X    server2   X    client1   X

Administrator: C:\Windows\system32\cmd.exe
WARNING: To launch Server Configuration tool again, run "SConfig"
PS C:\Users\administrator.DIWAN> New-Item -Path "C:\Shares\HR" -ItemType Directory


    Directory: C:\Shares


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        12/12/2024    1:41 PM               HR


PS C:\Users\administrator.DIWAN> New-Item -Path "C:\Shares\Admin" -ItemType Directory


    Directory: C:\Shares


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        12/12/2024    1:41 PM               Admin


PS C:\Users\administrator.DIWAN> New-Item -Path "C:\Shares\IT" -ItemType Directory


    Directory: C:\Shares


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        12/12/2024    1:41 PM               IT


PS C:\Users\administrator.DIWAN>
```
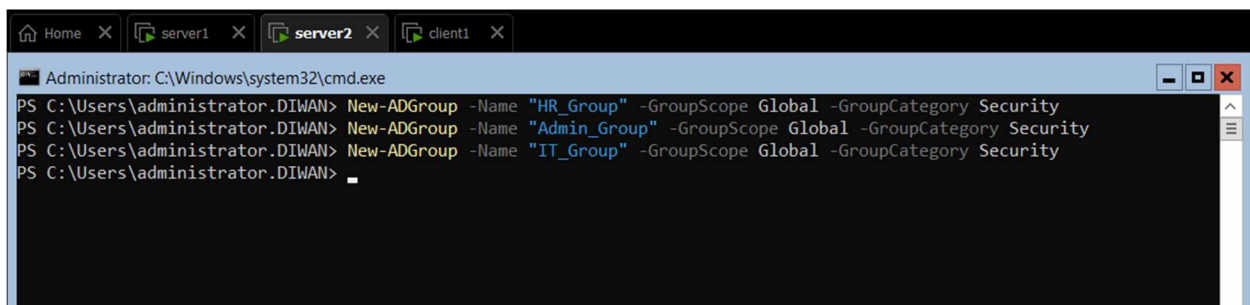


```
Home   X    server1   X    server2   X    client1   X

Administrator: C:\Windows\system32\cmd.exe
PS C:\Users\administrator.DIWAN> New-ADGroup -Name "HR_Group" -GroupScope Global -GroupCategory Security
PS C:\Users\administrator.DIWAN> New-ADGroup -Name "Admin_Group" -GroupScope Global -GroupCategory Security
PS C:\Users\administrator.DIWAN> New-ADGroup -Name "IT_Group" -GroupScope Global -GroupCategory Security
PS C:\Users\administrator.DIWAN>
```

Assignment: Network Setup and Health Monitoring Task Scheduling


```
PS C:\Users\administrator.DIWAN> # Add HR users to HR_Group
PS C:\Users\administrator.DIWAN> Get-ADUser -Filter * -SearchBase "OU=HR,OU=Staff,DC=diwan,DC=local" | ForEach-Object {
>>      Add-ADGroupMember -Identity "HR_Group" -Members $_
>> }
PS C:\Users\administrator.DIWAN>
```



```
PS C:\Users\administrator.DIWAN> # Add HR users to HR_Group
PS C:\Users\administrator.DIWAN> Get-ADUser -Filter * -SearchBase "OU=HR,OU=Staff,DC=diwan,DC=local" | ForEach-Object {
>>      Add-ADGroupMember -Identity "HR_Group" -Members $_
>> }
PS C:\Users\administrator.DIWAN> # Add IT users to IT_Group
PS C:\Users\administrator.DIWAN> Get-ADUser -Filter * -SearchBase "OU=IT,OU=Staff,DC=diwan,DC=local" | ForEach-Object {
>>      Add-ADGroupMember -Identity "IT_Group" -Members $_
>> }
PS C:\Users\administrator.DIWAN>
```



```
PS C:\Users\administrator.DIWAN> # Add Admin users to Admin_Group
PS C:\Users\administrator.DIWAN> Get-ADUser -Filter * -SearchBase "OU=Admin,OU=Staff,DC=diwan,DC=local" | ForEach-Object
 {
>>      Add-ADGroupMember -Identity "Admin_Group" -Members $_
>> }
PS C:\Users\administrator.DIWAN>
```



```
PS C:\Users\administrator.DIWAN> # Create and share the HR folder
PS C:\Users\administrator.DIWAN> New-SmbShare -Name "HR" -Path "C:\Shares\HR" -FullAccess "HR_Group"

Name ScopeName Path           Description
---- --------- ----           -----------
HR   *         C:\Shares\HR


PS C:\Users\administrator.DIWAN> Grant-SmbShareAccess -Name "HR" -AccountName "HR_Group" -AccessRight Full -Force

Name ScopeName AccountName    AccessControlType AccessRight
---- --------- -----------    ----------------- -----------
HR   *         DIWAN\HR_Group Allow             Full


PS C:\Users\administrator.DIWAN>
```

Assignment: Network Setup and Health Monitoring Task Scheduling

```
Home    X    server1    X    server2    X    client1    X

Administrator: C:\Windows\system32\cmd.exe                              _  □  X

HR    *         DIWAN\HR_Group Allow            Full


PS C:\Users\administrator.DIWAN> New-SmbShare -Name "Admin" -Path "C:\Shares\Admin" -FullAccess "Admin_Group"

Name   ScopeName Path             Description
----   --------- ----             -----------
Admin  *         C:\Shares\Admin


PS C:\Users\administrator.DIWAN> Grant-SmbShareAccess -Name "Admin" -AccountName "Admin_Group" -AccessRight Full -Force

Name   ScopeName AccountName       AccessControlType AccessRight
----   --------- -----------       ----------------- -----------
Admin  *         DIWAN\Admin_Group Allow             Full


PS C:\Users\administrator.DIWAN>
```

```
Home    X    server1    X    server2    X    client1    X

Administrator: C:\Windows\system32\cmd.exe                              _  □  X

PS C:\Users\administrator.DIWAN> New-SmbShare -Name "IT" -Path "C:\Shares\IT" -FullAccess "IT_Group"

Name ScopeName Path           Description
---- --------- ----           -----------
IT   *         C:\Shares\IT


PS C:\Users\administrator.DIWAN> Grant-SmbShareAccess -Name "IT" -AccountName "IT_Group" -AccessRight Full -Force

Name ScopeName AccountName       AccessControlType AccessRight
---- --------- -----------       ----------------- -----------
IT   *         DIWAN\IT_Group Allow                Full


PS C:\Users\administrator.DIWAN> _
```

```
Home    X    server1    X    server2    X    client1    X

Administrator: C:\Windows\system32\cmd.exe                              _  □  X

Name     ScopeName AccountName          AccessControlType AccessRight
----     --------- -----------          ----------------- -----------
Admin    *         DIWAN\Admin_Group    Allow             Full
HR       *         DIWAN\HR_Group       Allow             Full
IT       *         DIWAN\IT_Group       Allow             Full
C$       *         BUILTIN\Administrators    Allow         Full
C$       *         BUILTIN\Backup Operators Allow          Full
C$       *         NT AUTHORITY\INTERACTIVE Allow          Full
ADMIN$   *         BUILTIN\Administrators    Allow         Full
ADMIN$   *         BUILTIN\Backup Operators Allow          Full
ADMIN$   *         NT AUTHORITY\INTERACTIVE Allow          Full
IPC$     *         BUILTIN\Administrators    Allow         Full
IPC$     *         BUILTIN\Backup Operators Allow          Full
IPC$     *         NT AUTHORITY\INTERACTIVE Allow          Full
```

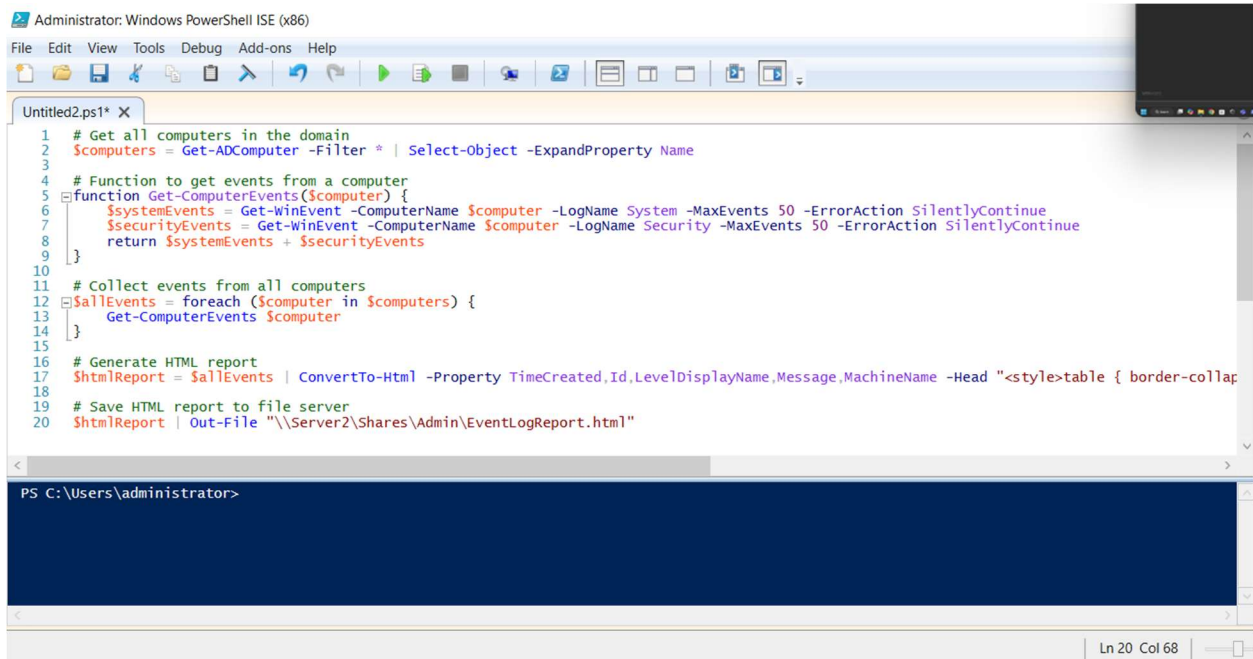# Assignment: Network Setup and Health Monitoring Task Scheduling
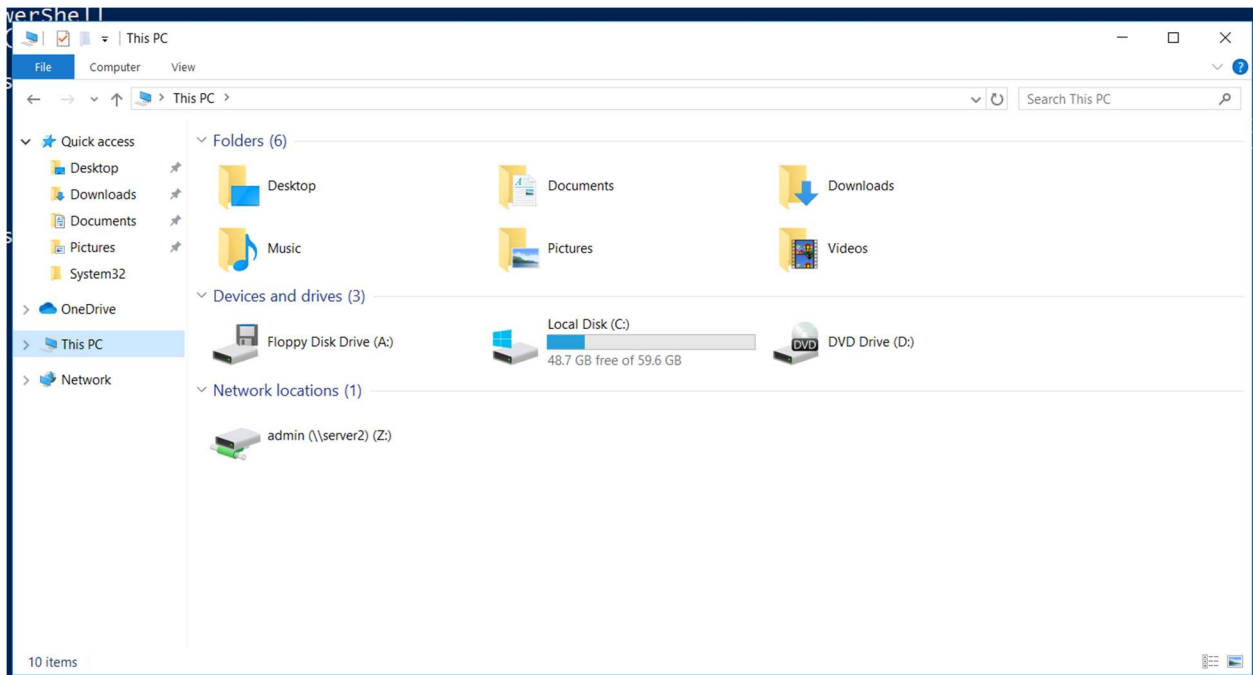
6. **Event Log Monitoring Script: (20 points)**

   o Develop a PowerShell script on your Windows 10/11 client that:

   - Connects to all servers and clients in the domain.

   - Retrieves the latest **50 system and security events** from the event logs.

   - Publishes these events to an **HTML file** on the file server.

   - Schedule this script to run **once a day** using **Task Scheduler**.

   Provide a screenshot of the script that retrieves the event logs, generates an HTML file, and copies the HTML file to the shared location. Additionally, include a screenshot of the process to create a Task Scheduler entry using PowerShell, and demonstrate the task's execution.

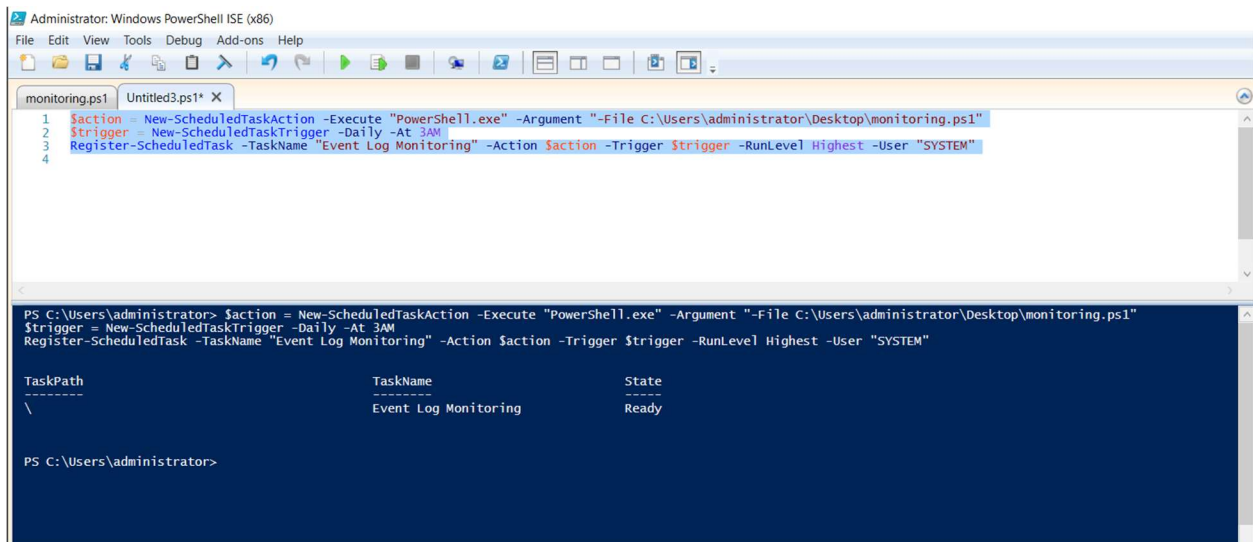# Assignment: Network Setup and Health Monitoring Task Scheduling


```powershell
# Get all computers in the domain
$computers = Get-ADComputer -Filter * | Select-Object -ExpandProperty Name

# Function to get events from a computer
function Get-ComputerEvents($computer) {
    $systemEvents = Get-WinEvent -ComputerName $computer -LogName System -MaxEvents 50 -ErrorAction SilentlyContinue
    $securityEvents = Get-WinEvent -ComputerName $computer -LogName Security -MaxEvents 50 -ErrorAction SilentlyContinue
    return $systemEvents + $securityEvents
}

# Collect events from all computers
$allEvents = foreach ($computer in $computers) {
    Get-ComputerEvents $computer
}

# Generate HTML report
$htmlReport = $allEvents | ConvertTo-Html -Property TimeCreated,Id,LevelDisplayName,Message,MachineName -Head "<style>table { border-collap

# Save HTML report to file server
$htmlReport | Out-File "\\Server2\Shares\Admin\EventLogReport.html"
```



```powershell
$action = New-ScheduledTaskAction -Execute "PowerShell.exe" -Argument "-File C:\Users\administrator\Desktop\monitoring.ps1"
$trigger = New-ScheduledTaskTrigger -Daily -At 3AM
Register-ScheduledTask -TaskName "Event Log Monitoring" -Action $action -Trigger $trigger -RunLevel Highest -User "SYSTEM"
```

```
PS C:\Users\administrator> $action = New-ScheduledTaskAction -Execute "PowerShell.exe" -Argument "-File C:\Users\administrator\Desktop\monitoring.ps1"
$trigger = New-ScheduledTaskTrigger -Daily -At 3AM
Register-ScheduledTask -TaskName "Event Log Monitoring" -Action $action -Trigger $trigger -RunLevel Highest -User "SYSTEM"

TaskPath                    TaskName                    State
--------                    --------                    -----
\                           Event Log Monitoring        Ready


PS C:\Users\administrator>
```

Assignment: Network Setup and Health Monitoring Task Scheduling
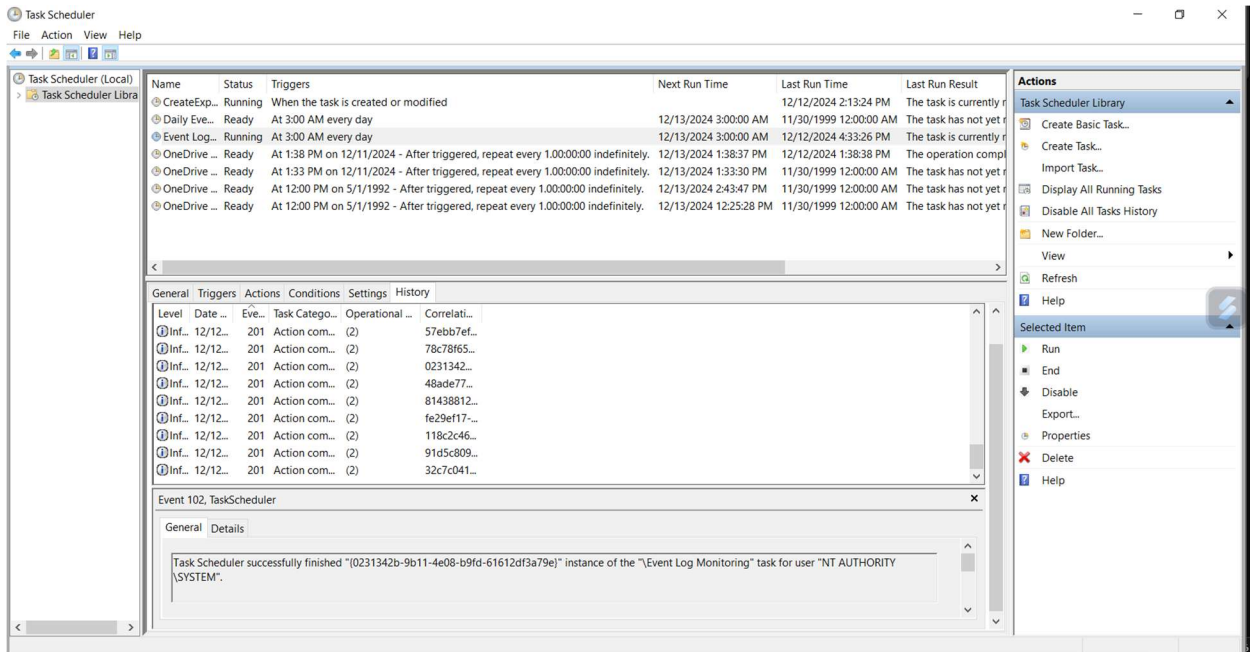
7. Active Directory Management: (15 points)

o Create a PowerShell script that prompts the user to select from the following **Active Directory** tasks:

- Reset a user password
- Disable a user account
- Enable a user account
- Unlock a user account
- Delete a user account

o The script should then prompt for the username and perform the selected task accordingly. The script should be run from Client1.

Paste screenshot showing the script running, prompting for inputs, and performing tasks as expected. Ensure the changes made are demonstrated clearly in the process.

Assignment: Network Setup and Health Monitoring Task Scheduling

```powershell
1  function Reset-UserPassword($username) {
2      $newPassword = Read-Host "Enter new password" -AsSecureString
3      Set-ADAccountPassword -Identity $username -NewPassword $newPassword -Reset
4      Write-Host "Password reset for $username"
5  }
6
7  function Disable-UserAccount($username) {
8      Disable-ADAccount -Identity $username
9      Write-Host "Account disabled for $username"
10 }
11
12 function Enable-UserAccount($username) {
13     Enable-ADAccount -Identity $username
14     Write-Host "Account enabled for $username"
15 }
16
17 function Unlock-UserAccount($username) {
18     Unlock-ADAccount -Identity $username
19     Write-Host "Account unlocked for $username"
20 }
21
22 function Delete-UserAccount($username) {
23     Remove-ADUser -Identity $username -Confirm:$false
24     Write-Host "Account deleted for $username"
25 }
26
27 # Main script
28 Write-Host "Select an action:"
29 Write-Host "1. Reset user password"
30 Write-Host "2. Disable user account"
31 Write-Host "3. Enable user account"
32 Write-Host "4. Unlock user account"
33 Write-Host "5. Delete user account"
34
```

```powershell
33 Write-Host "5. Delete user account"
34
35 $choice = Read-Host "Enter your choice (1-5)"
36 $username = Read-Host "Enter the username"
37
38 switch ($choice) {
39     1 { Reset-UserPassword $username }
40     2 { Disable-UserAccount $username }
41     3 { Enable-UserAccount $username }
42     4 { Unlock-UserAccount $username }
43     5 { Delete-UserAccount $username }
44     default { Write-Host "Invalid choice" }
45 }
46
```

```
Select an action:
1. Reset user password
2. Disable user account
3. Enable user account
4. Unlock user account
5. Delete user account
Enter your choice (1-5): 1
Enter the username: jdoe
Password reset for jdoe

PS C:\Users\administrator> |
```