**NETW 2500 Network Operating System Administration**

Active Directory Management, Group Policy, and System Maintenance

**Demonstrated Skills:**

* *Active Directory user creation via CSV scripting*
* *Active Directory group creation and management scripting*
* *Group Policy management*
* *User home drive creation*
* *AD object configuration and permissions*
* *Client domain join*
* *Client file management scripting*

**NETW 2500**

**Assignment 4**

**Creating and Managing Users, Active Directory and Group Policy and System Maintenance**

**Jeff Guitard**

**November 7, 2022**

**Table of Contents**

[TranscriptA4.txt Converted to Text 4](#_Toc130975250)

[NETWUserList.csv Converted to Text 7](#_Toc130975251)

[Command to copy script to PDC c:\dcscripts 7](#_Toc130975252)

[Successful run of Execution Policy and Script 7](#_Toc130975253)

[NETWaddusers\_YourInitials.ps1 Script converted to text 8](#_Toc130975254)

[NETWgroupadd.ps1 Script converted to text 9](#_Toc130975255)

[Successful Execution of Sales Password Policy 10](#_Toc130975256)

[Steps to create Home Drive for all users 10](#_Toc130975257)

[Delegate Control of Computer Objects settings 11](#_Toc130975258)

[Permissions/General settings 11](#_Toc130975259)

[Steps to add Windows 10 Workstation to Domain 12](#_Toc130975260)

[NETWcleantemp.ps1 converted to text 14](#_Toc130975261)

[Path location of script file in SYSVOL 14](#_Toc130975262)

# TranscriptA4.txt Converted to Text

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Windows PowerShell transcript start

Start time: 20221107104106

Username: jdg\Administrator

RunAs User: jdg\Administrator

Configuration Name: Microsoft.PowerShell

Machine: DC01A2U26 (Microsoft Windows NT 10.0.20348.0)

Host Application: C:\Windows\system32\wsmprovhost.exe -Embedding

Process ID: 680

PSVersion: 5.1.20348.859

PSEdition: Desktop

PSCompatibleVersions: 1.0, 2.0, 3.0, 4.0, 5.0, 5.1.20348.859

BuildVersion: 10.0.20348.859

CLRVersion: 4.0.30319.42000

WSManStackVersion: 3.0

PSRemotingProtocolVersion: 2.3

SerializationVersion: 1.1.0.1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Transcript started, output file is C:\PDCtranscriptA4.txt

PS C:\Users\Administrator\Documents> CommandInvocation(New-ADOrganizationalUnit): "New-ADOrganizationalUnit"

>> ParameterBinding(New-ADOrganizationalUnit): name="Name"; value="Staff"

>> ParameterBinding(New-ADOrganizationalUnit): name="Path"; value="DC=jdg,DC=netw2500,DC=ca"

>> ParameterBinding(New-ADOrganizationalUnit): name="ProtectedFromAccidentalDeletion"; value="True"

PS C:\Users\Administrator\Documents> CommandInvocation(Get-ADOrganizationalUnit): "Get-ADOrganizationalUnit"

>> ParameterBinding(Get-ADOrganizationalUnit): name="Identity"; value="OU=staff,DC=jdg,DC=netw2500,DC=ca"

>> ParameterBinding(Get-ADOrganizationalUnit): name="Properties"; value="\*"

CanonicalName : jdg.netw2500.ca/Staff

City :

CN :

Country :

Created : 2022-11-07 10:41:20 AM

createTimeStamp : 2022-11-07 10:41:20 AM

Deleted :

Description :

DisplayName :

DistinguishedName : OU=Staff,DC=jdg,DC=netw2500,DC=ca

dSCorePropagationData : {2022-11-07 10:41:20 AM, 1600-12-31 8:00:00 PM}

instanceType : 4

isDeleted :

LastKnownParent :

LinkedGroupPolicyObjects : {}

ManagedBy :

Modified : 2022-11-07 10:41:20 AM

modifyTimeStamp : 2022-11-07 10:41:20 AM

Name : Staff

nTSecurityDescriptor : System.DirectoryServices.ActiveDirectorySecurity

ObjectCategory : CN=Organizational-Unit,CN=Schema,CN=Configuration,DC=jdg,DC=netw2500,DC=ca

ObjectClass : organizationalUnit

ObjectGUID : 3efbfca3-bf93-46d5-9c45-f850c73fd87e

ou : {Staff}

PostalCode :

ProtectedFromAccidentalDeletion : True

sDRightsEffective : 15

State :

StreetAddress :

uSNChanged : 155831

uSNCreated : 155830

whenChanged : 2022-11-07 10:41:20 AM

whenCreated : 2022-11-07 10:41:20 AM

PS C:\Users\Administrator\Documents> CommandInvocation(Stop-Transcript): "Stop-Transcript"

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Windows PowerShell transcript end

End time: 20221107104214

# NETWUserList.csv Converted to Text

FirstName,MiddleInitial,LastName,FirstInitial,Department,Office,Title

Charlotte,H,Anichka,C,Exec,Executive offices,VP

Clare,L,Cade,C,Exec,Executive offices,VP

Jeff,D,Guitard,J,Exec,Executive offices,CEO

Myra,Y,Dante,M,Admin,General cubicle space,Admin

Jaymz,A,Dinara,J,Admin,General cubicle space,Admin

Brock,R,Finn,B,Admin,General cubicle space,Admin

Troy,R,Macey,T,Admin,General cubicle space,Admin

Zia,I,Oleg,Z,Admin,General cubicle space,Admin

Francis,R,Uta,F,Admin,General cubicle space,Admin

Carlos,A,Yen,C,Sales,Sales offices,Rep

Bruce,R,Orca,B,Sales,Sales offices,Rep

Nina,I,Yuli,N,Sales,Sales offices,Rep

# Command to copy script to PDC c:\dcscripts

Text

Description automatically generated

Screenshot of script copied to c:\dcscripts on PDC

# Successful run of Execution Policy and Script

Text

Description automatically generated

Screenshot of Execution Policy and Script

# NETWaddusers\_YourInitials.ps1 Script converted to text

#Author: Jeff Guitard

#Creation Date: 11/7/2022

#Script to add new users to Active Directory. Script must be ran as an administrator.

$Users = Import-Csv -Path \\MS01A2U28\C$\Scripts\NETWUserList.csv

#This creates a $Users variable and imports the users list to make all the users information available.

Foreach ($User in $Users)

#This ensures all users in the CSV file are processed

{

$SamName = $User.FirstInitial + $User.LastName

#This creates the login name using the user's first inital and last name.

$FullName = $User.FirstName + " " + $User.MiddleInitial + " " + $User.LastName

#This creates the user's full name in the format first name then space middle initial then space last name. (Jeff D Guitard for example)

$Display = $User.LastName + ", " + $User.FirstName

#This creates the user's full name in the format last name comma then space then first name. (Guitard, Jeff for example)

$Dept = $User.Department

#This adds the department that the user is employed in to their profile.

$Office = $User.Office

#This adds the office that the user is employed in to their profile.

$Title = $User.Title

#This adds the title of the user to their profile.

New-ADUser -SamAccountName $SamName -DisplayName $Display -AccountPassword (ConvertTo-SecureString -AsPlainText "Netw2500@2022" -Force) -Name $FullName -ChangePasswordAtLogon $True -Path "OU=Staff,DC=jdg,DC=netw2500,DC=ca" -Department $Dept -Office $Office -Title $Title

}

#This is the final output creating the new local user account in the format: the SAMAccountName, their last name then first name, their password, the full employee name including middle initial, the password is changed at first login, the organizational unit, the department, the office that they work in and the employee's title

# NETWgroupadd.ps1 Script converted to text

#Author: Jeff Guitard

#Creation Date: 11/8/2022

#Script to create new groups and add existing AD users to correct group. Script must be ran as an administrator.

#This will create a "General\_group" group for the users in the OU=Staff,DC=jdg,DC=netw2500,DC=ca distinguished name path with the "global" group scope

New-ADGroup General\_group -Path "OU=Staff,DC=jdg,DC=netw2500,DC=ca" -GroupScope Global

#This will search for users in the OU=Staff,DC=jdg,DC=netw2500,DC=ca distinguished name path then filter by department name "Admin" and add the AD group member to the "General\_group" group using the SamAccountName

Get-ADUser -SearchBase "OU=Staff,DC=jdg,DC=netw2500,DC=ca" -filter {Department -eq "Admin"} | %{Add-ADGroupMember "General\_group" $\_.SamAccountName}

#This will create a "Sales\_group" group for the users in the OU=Staff,DC=jdg,DC=netw2500,DC=ca distinguished name path with the "global" group scope

New-ADGroup Sales\_group -Path "OU=Staff,DC=jdg,DC=netw2500,DC=ca" -GroupScope Global

#This will search for users in the OU=Staff,DC=jdg,DC=netw2500,DC=ca distinguished name path then filter by department name "Sales" and add the AD group member to the "Sales\_group" group using the SamAccountName

Get-ADUser -SearchBase "OU=Staff,DC=jdg,DC=netw2500,DC=ca" -filter {Department -eq "Sales"} | %{Add-ADGroupMember "Sales\_group" $\_.SamAccountName}

#This will create an "Exec\_group" group for the users in the OU=Staff,DC=jdg,DC=netw2500,DC=ca distinguished name path with the "global" group scope

New-ADGroup Exec\_group -Path "OU=Staff,DC=jdg,DC=netw2500,DC=ca" -GroupScope Global

#This will search for users in the OU=Staff,DC=jdg,DC=netw2500,DC=ca distinguished name path then filter by department name "Exec" and add the AD group member to the "Exec\_group" group using the SamAccountName

Get-ADUser -SearchBase "OU=Staff,DC=jdg,DC=netw2500,DC=ca" -filter {Department -eq "Exec"} | %{Add-ADGroupMember "Exec\_group" $\_.SamAccountName}

# Successful Execution of Sales Password Policy

Graphical user interface, text, application

Description automatically generated

Screenshot of successful execution of sales password policy

# Steps to create Home Drive for all users

1) Go to *Active Directory Users and Computers*

2) Select all required Users (at the same time) for new Home$ share

3) Go to **Properties**

4) Click **Profile**

5) Check the **Home folder** box

6) Connect **H:** to [**\\SD01A1U30\Home$\%username%**](file:///\\SD01A1U30\Home$\%25username%25)

# Delegate Control of Computer Objects settings

Graphical user interface, text, application

Description automatically generated

Screenshot of Delegate Control of Computer Objects settings

# Permissions/General settings

Graphical user interface, text, application

Description automatically generated Graphical user interface, text, application

Description automatically generated

Permissions/General settings screenshots

# Steps to add Windows 10 Workstation to Domain

Graphical user interface

Description automatically generated

Step 1: Set DNS server screenshot

Graphical user interface, text, application, email

Description automatically generated

Step 2: Domain join client and change computer name screenshot

Graphical user interface, application

Description automatically generated

Step 3: Join domain using CTechBB account screenshot

Graphical user interface, application

Description automatically generated

Step 4: Confirm client has joined domain screenshot

Graphical user interface, application

Description automatically generated

Step 5: Log in to client as user Bruce Orca screenshot

# NETWcleantemp.ps1 converted to text

#Author: Jeff Guitard

#Creation Date: 11/17/2022

$tempfiles = @("c:\Users\\*\Appdata\Local\Temp\\*", "$env:windir\temp\\*")

#Assigned variable for the directories to remove items from.

$folders = @("c:\Users\\*\Appdata\Local\Temp\", "$env:windir\temp\")

#Assigned variable for the directories to create new items in.

Remove-Item -Path $tempfiles -Recurse -Force -ErrorAction Ignore

#Remove the items from the locations in the $tempfiles variable, -Recurse deletes the items in the specified

#locations and in all child items of the locations, -Force forces the removal of items that cannot otherwise be

#changed, such as hidden or read-only files for example, -ErrorAction means the script will continue processing

#without notifying the user that an action has occurred

New-Item -Path $folders -ItemType File -Name cleanlog.txt -Value (Get-Date)

#Creates a new item in the directories specified in the $folders variable, the item is a file named cleanlog.txt

#that will have the date and time that the script ran in the file contents.

# Path location of script file in SYSVOL

[**\\jdg.netw2500.ca\SysVol\jdg.netw2500.ca\Policies\{CF92B01C-8731-4450-8B0B-57A5251DFD83}\Machine\Scripts\Startup**](file:///\\jdg.netw2500.ca\SysVol\jdg.netw2500.ca\Policies\%7bCF92B01C-8731-4450-8B0B-57A5251DFD83%7d\Machine\Scripts\Startup)