## **Active Directory PowerShell Commands**

### View all Active Directory commands

get-command -Module ActiveDirectory

### Display Basic Domain Information

Get-ADDomain

### Get all Domain Controllers by Hostname and Operating

Get-ADDomainController -filter \* | select hostname, operatingsystem

### Get all Fine Grained Password Policies

Get-ADFineGrainedPasswordPolicy -filter \*

### Get Domain Default Password Policy

Gets the password policy from the logged in domain

Get-ADDefaultDomainPasswordPolicy

### Backup Active Directory System State Remotely

This will back up the domain controllers system state data. Change DC-Name to your server name and change the Backup-Path. The backup path can be a local disk or a UNC path

invoke-command -ComputerName DC-Name -scriptblock {wbadmin start systemstateback up -backupTarget:"Backup-Path" -quiet}

## **AD User PowerShell Commands**

This section is all Active Directory user commands

#### Get User and List All Properties (attributes)

Change username to the samAccountName of the account

Get-ADUser username -Properties \*

#### Get User and List Specific Properties

Just add whatever you want to display after select

Get-ADUser username -Properties \* | Select name, department, title

### Get All Active Directory Users in Domain

Get-ADUser -Filter \*

### Get All Users From a Specific  OU

OU = the distinguished path of the OU

Get-ADUser -SearchBase “OU=ADPRO Users,dc=ad,dc=activedirectorypro.com” -Filter \*

### Get AD Users by Name

This command will find all users that have the word robert in the name. Just change robert to the word you want to search for.

get-Aduser -Filter {name -like "\*robert\*"}

### Get All Disable User Accounts

Search-ADAccount -AccountDisabled | select name

### Disable User Account

Disable-ADAccount -Identity rallen

### Enable User Account

Enable-ADAccount -Identity rallen

### Get All Accounts with Password Set to Never Expire

get-aduser -filter \* -properties Name, PasswordNeverExpires | where {$\_.passwordNeverExpires -eq "true" } | Select-Object DistinguishedName,Name,Enabled

### Find All Locked User Accounts

Search-ADAccount -LockedOut

### Unlock User Account

Unlock-ADAccount –Identity john.smith

### List all Disabled User Accounts

Search-ADAccount -AccountDisabled

### Force Password Change at Next Login

Set-ADUser -Identity username -ChangePasswordAtLogon $true

### Move a Single User to a New OU

You will need the distinguishedName of the user and the target OU

Move-ADObject -Identity "CN=Test User (0001),OU=ADPRO Users,DC=ad,DC=activedirectorypro,DC=com" -TargetPath "OU=HR,OU=ADPRO Users,DC=ad,DC=activedirectorypro,DC=com"

### Move Users to an OU from a CSV

Setup a csv with a name field and a list of the users sAmAccountNames. Then just change the target OU path.

# Specify target OU. $TargetOU = "OU=HR,OU=ADPRO Users,DC=ad,DC=activedirectorypro,DC=com" # Read user sAMAccountNames from csv file (field labeled "Name"). Import-Csv -Path Users.csv | ForEach-Object { # Retrieve DN of User. $UserDN = (Get-ADUser -Identity $\_.Name).distinguishedName # Move user to target OU. Move-ADObject -Identity $UserDN -TargetPath $TargetOU }

## **AD Group Commands**

### Get All members Of A Security group

Get-ADGroupMember -identity “HR Full”

### Get All Security Groups

This will list all security groups in a domain

Get-ADGroup -filter \*

### Add User to Group

Change group-name to the AD group you want to add users to

Add-ADGroupMember -Identity group-name -Members Sser1, user2

### Export Users From a Group

This will export group members to a CSV, change group-name to the group you want to export.

Get-ADGroupMember -identity “Group-name” | select name | Export-csv -path C:OutputGroupmembers.csv -NoTypeInformation

### Get Group by keyword

Find a group by keyword. Helpful if you are not sure of the name, change group-name.

get-adgroup -filter \* | Where-Object {$\_.name -like "\*group-name\*"}

### Import a List of Users to a Group

$members = Import-CSV c:itadd-to-group.csv | Select-Object -ExpandProperty samaccountname Add-ADGroupMember -Identity hr-n-drive-rw -Members $members

## **AD Computer Commands**

### Get All Computers

This will list all computers in the domain

Get-AdComputer -filter \*

### Get All Computers by Name

This will list all the computers in the domain and only display the hostname

Get-ADComputer -filter \* | select name

### Get All Computers from an OU

Get-ADComputer -SearchBase "OU=DN" -Filter \*

### Get a Count of All Computers in Domain

Get-ADComputer -filter \* | measure

### Get all Windows 10 Computers

Change Windows 10 to any OS you want to search for

Get-ADComputer -filter {OperatingSystem -Like '\*Windows 10\*'} -property \* | select name, operatingsystem

### Get a Count of All computers by Operating System

This will provide a count of all computers and group them by the operating system. A great command to give you a quick inventory of computers in AD.

Get-ADComputer -Filter "name -like '\*'" -Properties operatingSystem | group -Property operatingSystem | Select Name,Count

### Delete a single Computer

Remove-ADComputer -Identity "USER04-SRV4"

### Delete a List of Computer Accounts

Add the hostnames to a text file and run the command below.

Get-Content -Path C:ComputerList.txt | Remove-ADComputer

### Delete Computers From an OU

Get-ADComputer -SearchBase "OU=DN" -Filter \* | Remote-ADComputer

## **Group Policy Section**

### Get all GPO related commands

get-command -Module grouppolicy

### Get all GPOs by status

get-GPO -all | select DisplayName, gpostatus

### Backup all GPOs in the Domain

Backup-Gpo -All -Path E:GPObackup

## **Office 365 PowerShell Commands**

### Connect To Exchange Online

This will pop up and ask for credentials

$UserCredential = Get-Credential $Session = New-PSSession -ConfigurationName Microsoft.Exchange -ConnectionUri https://ps.outlook.com/powershell/ -Credential $LiveCred -Authentication Basic -AllowRedirection Import-PSSession $Session

### Force Azure Sync

This is for the azure ad sync client.

Force delta sync (only sync changes

Start-ADSyncSyncCycle -PolicyType Delta Force a full sync Start-ADSyncSyncCycle -PolicyType Initial

### Get A List of All Office 365 Users

Get-MsolUser | Select DisplayName, City, Department, ObjectID

### Get Full mailbox details

Get-Mailbox email-address | fl

### Get Calendar Permissions

Get-MailboxFolderPermission username:calendar

### Enable Remote Mailbox (Hybrid Environment)

Use this command if you have an existing on-premise user that needs an office 365 mailbox. There are other ways to do this but this creates all the attributes in the AD account.

Replace the username and the tenant fields

Enable-RemoteMailbox username -RemoteRoutingAddress "username@tenant.mail.onmicrosoft.com"

## **Windows Server & Client Commands**

### Get all Services

get-service

### Get all Processes

get-process

### Display Network Adapters

Gets detailed about the network adapter installed such as name,  status, speed and mac address.

get-netadapater

### Restart Remote Computers

Restart-Computer -ComputerName "Server01", "Server02", "localhost"

### Get Last Boot Time

This takes a few lines

$os = Get-WmiObject win32\_operatingsystem $uptime = (Get-Date) - $os.ConvertToDateTime($os.LastBootUpTime) Write-Output ("Last boot: " + $os.ConvertToDateTime($os.LastBootUpTime))

You can also run this single line to get last boot time

systeminfo | more

### Start a Remote Session

Use this to start an interactive session with a remote computer

Enter-PSSession -ComputerName

### Read the Content of a File (Open a file)

This example shows how to read the content of the windows firewall log file

Get-Content -Path "c:windowssystem32logfilesfirewallpfirewall.log"

### Copy Files & Folders

Use this command to copy an entire folder to another folder. This will copy the folder and all the sub folder/files. The -verbose command will display the results to the console.

copy-item E:\WindowsImageBackup\exchange -destination \\server1\Backups\Exchange -recurse -verbose

## **Basic PowerShell Commands**

### Get Execution Policy

get-executionpolicy

### Set Execution Policy to Unrestricted

set-executionpolicy unrestricted

### Show PowerShell Version

$PSVersionTable

### Get help for a command

Use this to get the help information for a command

get-help command-name

### Search Get Help

Use this to search the help files. This is useful if you don’t know the command or want to see if one exists.

get-help \*keyword\*

### Get Installed Modules

Use this command to display all the installed modules on a computer

get-installedmodule

### List All Available Modules

This will list all available modules on the computer.

Get-Module -ListAvailable

### Exporting results to CSV

Add export-csv to the end of commands

Get-ADUser username -Properties \* | Select name, department, title | export-csv c:user.csv

### Display available commands

This will display all commands that are available based on the modules that are loaded.

get-command

### Find New Modules

Replace \*ntfs\* with the keyword you want to search for. This searches modules at https://www.powershellgallery.com/

Find-Module \*ntfs\*

### Install a New Module

Installs modules from https://www.powershellgallery.com/

I found a module called NTFSSecurity, to install it I run this command

install-module NTFSSecurity