



Huge List Of PowerShell Commands for Active Directory, Office 365 and more

By Robert Allen (https://activedirectorypro.com/author/mug/) | January 13, 2019 |
27 🗨️ (https://activedirectorypro.com/powershell-commands/#comments)

This is the ultimate collection of PowerShell commands for Active Directory, Office 365, Windows Server and more.

These commands will help with numerous tasks and make your life easier.

Table of Contents:

- Active Directory Commands
- Office 365 Commands
- Windows Server & Client Commands
- Basic PowerShell Commands

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
```

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
```

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"
```

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
```

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:\Output\Groupmembers.csv
```

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-ADGroupMembers -Members $members -Group group-name
```

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, operatin
```

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 operatingSy
```

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
```

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.LastBo
```

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:\windows\system32\logfiles\firewall\pfirewall.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
```

Recommended Tool: SolarWinds Server & Application Monitor

(https://www.solarwinds.com/topics/active-directory-monitor?CMP=BIZ-HAD-ADP-SW_WW_X_PP_PPD_LD_EN_HLTH_SW-SAM-X_X_Tools_List_X-ADHC_Topics)

This utility was designed to Monitor Active Directory



(https://www.solarwinds.com/topics/active-directory-monitor?CMP=BIZ-HAD-ADP-SW_WW_X_PP_PPD_LD_EN_HLTH_SW-SAM-X_X_Tools_List_X-ADHC_Topics) and other critical services like DNS & DHCP. It will quickly spot domain controller issues, prevent replication failures, track failed logon attempts and much more.

What I like best about SAM is it's easy to use dashboard and alerting features. It also has the ability to monitor virtual machines and storage.

Download Your Free Trial Here

(http://www.solarwinds.com/register/registrationb.aspx?program=857&c=701500000012m8b&CMP=BIZ-TAD-ADP-SW_WW_X_PP_PPD_LD_EN_APPERF_SW-SAM-X_X_X_X-Q417)

Posted in Powershell (<https://activedirectorypro.com/powershell/>)

27 Comments



Josh on January 29, 2019 at 10:13 pm

Thanks for this awesome list!

Reply



Robert Allen on January 29, 2019 at 10:59 pm

You're welcome. I'll be adding more, stay tuned.

Reply



Joe Watson on March 1, 2019 at 2:23 pm

Thanks for the list, I'm looking for a script to show logged in users per server showing user name, computer name and IP address. You have anything like that yet?

Reply



Robert Allen on March 1, 2019 at 9:22 pm

This should get you the who is logged into a remote computer.

```
invoke-command -computername -scriptblock { qwinsta }
```

For something more complex that can query multiple computers check out this article <https://sid-500.com/2018/02/28/powershell-get-all-logged-on-users-per-computer-ou-domain-get-userlogon/comment-page-1/#comments>

(<https://sid-500.com/2018/02/28/powershell-get-all-logged-on-users-per-computer-ou-domain-get-userlogon/comment-page-1/#comments>)

Reply



Ms.Tina on July 2, 2019 at 4:35 pm

Omg thank you so much; Now all I need to do is figure out how to get power shell to grab history and list every change its ever made.. if its possible. I hope it can. I'm trying to look for tampering on my laptop by someone using PS. I know it's been done but how, where, the extents – hell even my registry has become questionable.

Reply



Mani on July 14, 2019 at 1:10 pm

This is very useful, am trying to get the users of an OU, with Name, Lastlogged on to, Email memberships, Group policy

Reply



NavaL on July 29, 2019 at 2:17 pm

Its awesome

[Reply](#)

Egyir on August 28, 2019 at 1:08 pm

Great job and thanks for the list, I'm searching for a script to extend the expiration date for a User Account NOT TO SET TO PASSWORDNOTREQUIRED OR PASSWORDNEVEREXPIRE . Kindly help me out?

THANK YOU.

[Reply](#)

Dustin on September 10, 2019 at 8:42 am

Thanks!!! Love this

[Reply](#)

JSchauer on September 19, 2019 at 9:11 pm

This list will go a long way in helping me get comfortable with PowerShell. Thanks for taking the time to create this list.

[Reply](#)



Robert Allen on September 19, 2019 at 11:32 pm

No problem

Reply

Alexey on September 26, 2019 at 8:14 am

Thank you very much, this list will be very useful for me and my team!

Reply

boima (<http://none>) on October 18, 2019 at 9:09 am

Hello Sir

Thank you for this amazing list.

I am having problems finding the size of all folders on a user C drive as well as the size of all folders on a users desktop. I have until now tried several commands but yet to receive the desired result.

For exampe to display the folders on the C drive used this: `get-childitem -force | select fullname`

To display the folders size :

```
get-childitem -force | select fullname | select @{l='Size'; e={
{$fso.GetFolder($_.FullName).Size}},FullName `
| sort Size -Descending `
| ft @{l='Size [MB]'; e='{o:N2} ' -f ($_.Size / 1MB)}};FullName
```

It displays the folders and also show the sizes unfortunately, it does not show all folder sizes only the first two folders.

I will be very happy if you would look at the code and please assist me with it.

(Powershell)

Best regards

Boima

Reply

Reshma on November 20, 2019 at 7:43 am

Thanks for the amazing list.all the commands are in 1 place

Reply

UVNAIDU on February 18, 2020 at 1:26 pm

Thanks for providing valuable information and Please provide exchange commands also.

Reply

John Hughes on February 20, 2020 at 2:00 pm

A better job needs to be done in the formatting. For example

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

should be shown as

```
Get-ADUser [username] -Properties * | Select name, department, title
```

So a person knows to put in a specific name in place of [username] otherwise people think it's part of the command string.

Reply



Robert Allen on February 22, 2020 at 2:54 pm

Good tips John.

Thanks

Reply

Aro on February 24, 2020 at 1:52 am

Awesome

Reply

Jenn on March 26, 2020 at 3:21 pm

I don't know who wrote this awesome list of PS scripts but they deserve a RAISE, a toast of glasses, a high-five with elbows (during Covid-19) a standing ovation, and a big bear hug!!! Thank you so much from the bottom of my heart ❤️!!!! You ROCK, John.

Reply



Robert Allen on November 7, 2020 at 10:55 pm

Thanks Jenn!

Reply

secuo4 on August 29, 2020 at 2:07 pm

Very good commands. I am looking for a command that lists the logon history of all users who opened their windows session. i have active directory 2008. i dont have third party tools.

Reply



Robert Allen on September 20, 2020 at 1:37 pm

What do you mean opened their windows session?

Getting logon history is difficult because the real logon event is stored on the local computer, not in Active Directory.

Reply

daylon on September 9, 2020 at 2:03 pm

Hi Guys

I am new to powershell and trying to do basic things.

Please can you help, I will greatly appreciate it

How would I push an address say for example the address is " 10 xyz street" across to an entire OU, lets call the OU Test?

I need to push different addresses across multiple OUS

Regards

Daylon

Reply



Robert Allen on November 7, 2020 at 10:30 pm

Hi Daylon,

This can definitely be done with PowerShell but can get a bit complex. For that reason, I created an easy to use GUI tool called the AD Bulk User Updater (<https://activedirectorypro.com/bulk-user-update-tool/>).

If you want to try the PowerShell option check out my post on updated ProxyAddress.

<https://activedirectorypro.com/how-to-bulk-update-proxyaddresses-attribute/> (<https://activedirectorypro.com/how-to-bulk-update-proxyaddresses-attribute/>)

Reply

Santosh on September 22, 2020 at 10:05 pm

Really helpful. Thanks for this!

Reply



Robert Allen on September 26, 2020 at 1:13 pm



Reply

Tom Cipriano (<http://www.tomcipriano.com>) on November 7, 2020 at

1:40 am

Thanks for sharing. Very helpful scripts indeed!

Reply

Leave a Comment

Comment

Name (required)

Email (will not be published) (required)

Website

Submit Comment

© 2020 Active Directory Pro, All rights reserved

About (<https://activedirectorypro.com/about/>) | Contact (<https://activedirectorypro.com/contact/>) | EULA
(<https://activedirectorypro.com/eula/>) | Privacy Policy (<https://activedirectorypro.com/privacy-policy/>) | YouTube
(<http://youtube.com/c/ActiveDirectoryPro>)