Contents

[Yleistä 2](#_Toc65323129)

[PowerShell ISE 3](#_Toc65323130)

[Execution Policy 4](#_Toc65323131)

[Peruskomentoja 4](#_Toc65323132)

[Syntaksin lukeminen 4](#_Toc65323133)

[Get-help 5](#_Toc65323134)

[TAB 6](#_Toc65323135)

[Putkitus 6](#_Toc65323136)

[Varmistus 8](#_Toc65323137)

[Vertailu 8](#_Toc65323138)

[get-Childitem 8](#_Toc65323139)

[Eventlog 9](#_Toc65323140)

[get-wmiobject 9](#_Toc65323141)

[get-computerinfo 10](#_Toc65323142)

[Powershell Web Access 10](#_Toc65323143)

[Where-Object 10](#_Toc65323144)

[Measure-command 11](#_Toc65323145)

[Clear-content 12](#_Toc65323146)

[Muuttujat 12](#_Toc65323147)

[new-pssession 12](#_Toc65323148)

[out-gridview 13](#_Toc65323149)

[Write-output 13](#_Toc65323150)

[Write-warning 13](#_Toc65323151)

[Get-Psdrive 13](#_Toc65323152)

[get-command 13](#_Toc65323153)

[Profiilit 14](#_Toc65323154)

[PowerShell kommentit 14](#_Toc65323155)

[Pikanäppäimiä 15](#_Toc65323156)

[Clipboard 15](#_Toc65323157)

[Palvelimen hallinta 15](#_Toc65323158)

[Rename, restart 15](#_Toc65323159)

[IP-asetusten vaihto 15](#_Toc65323160)

[Sammutus 16](#_Toc65323161)

[Palveluiden asentaminen 16](#_Toc65323162)

[Toimialueen hallinta 16](#_Toc65323163)

[OU 16](#_Toc65323164)

[Käyttäjät 16](#_Toc65323165)

[Käyttäjien hallinta: 17](#_Toc65323166)

[Ryhmät 17](#_Toc65323167)

[Tietokoneet 18](#_Toc65323168)

[Objektin siirtäminen 18](#_Toc65323169)

[AD roskakori 18](#_Toc65323170)

[DHCP 19](#_Toc65323171)

[Liikkuvat profiilit 19](#_Toc65323172)

[Kotikansiot 19](#_Toc65323173)

[Remote 19](#_Toc65323174)

[Group Policy 20](#_Toc65323175)

[Skriptaus PowerShellillä 20](#_Toc65323176)

# Yleistä

PowerShell komennot eli cmdletit käyttävät kaikki samaa perusrakennetta:

verbi-substantiivi. Hae-Suklaata

Esim.

Set-location – Aseta-Sijainti.



Verbejä on esimerkiksi:

Get – haetaan jotain.

Start – käynnistä jotain.

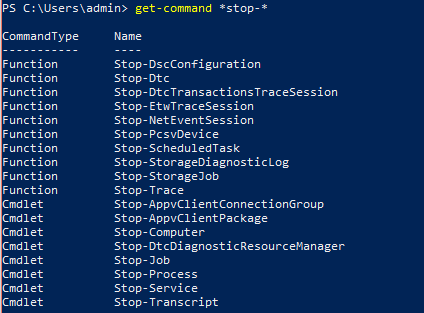
Out – Viedään ulos jotain (esim. tiedostoon)

Stop – pysäytetään jotain

Set – asetetaan jotain

New – luodaan jotain uutta

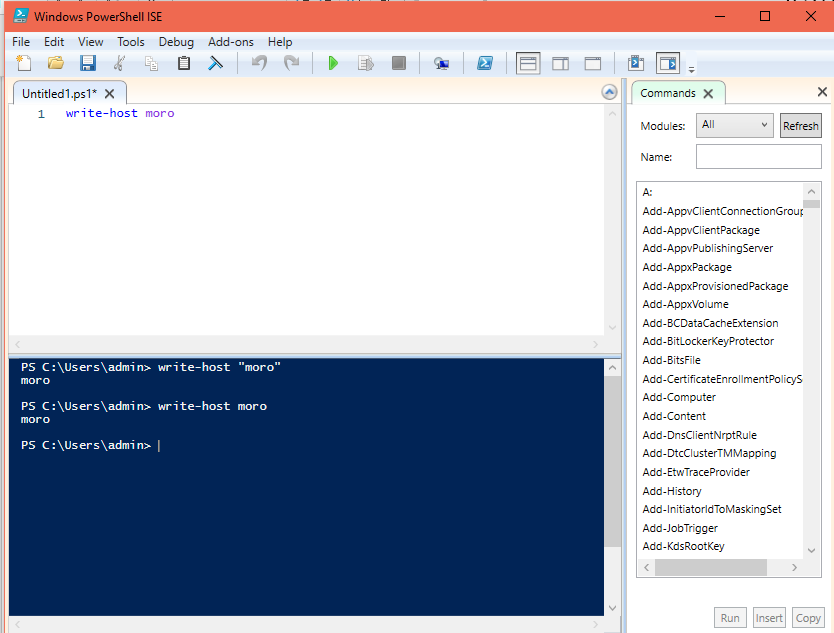
Tässä esimerkiksi kaikki komennot, joista löytyy Stop verbi:



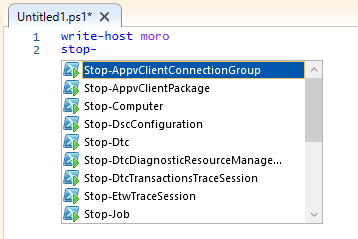
## PowerShell ISE

PowerShell ISE on editori, jolla voidaan luoda PowerShell skriptejä. Myös Visual Studio Codea voi käyttää.

Editorissa voi kirjoittaa koodia, ja sen voi samantien myös ajaa ISE ohjelmassa. Koodin voi tallentaa jonnekin muualle ja suorittaa skriptin. Commands valikon kautta löytyy lista kaikista käytettävistä komennoista.



Kun koodia kirjoittaa, osaa ISE myös ehdottaa vaihtoehtoja. Alla ehdotetaan mitä stop verbin jälkeen voisi tulla.



## Execution Policy

Execution Policyllä määritetään mitä skriptejä koneella saa ajaa. Oletuksena Execution Policy on asetettu Restricted tilaan, eli mitään skriptejä ei saa ajaa koneella.

## Peruskomentoja

clear-host - cls, tyhjennä ruutu

set-location - cd, vaihda kansiota

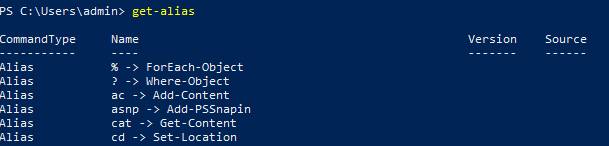
get-childitem - ls/dir, listaa sisältö

get-content - cat, lukee tiedoston sisällön

copy-item - cp, kopiointi

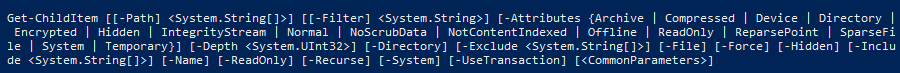
get-alias \*a - Kaikki alias komennot joissa on a kirjain ensimmäisen kirjaimen jälkeen

get-alias m\* - Kaikki alias komennot jotka alkavat m kirjaimella



## Syntaksin lukeminen

Esim Get-ChildItem[-Path] – tämä on valinnainen parametri, ei ole pakko käyttää



Get-childitem -path string[] – stringin sisältö on pakollinen koska [] ovat kiinni toisissa. Sen sisällä voi olla useampia parametrejä, kuten alla olevassa kuvassa:



get-childitem [[-path] <system.string[]>] - -path on kahden [ ] sulkumetrin välissä, joten myös se on valinnainen komento. get-childitem vain olettaa että tarkoitetaan -path parametriä, jos mitään ei tarkemmin kerrota, kuten alla oleva komento. Se tulostaa samat tiedot kuin ylempikin.



## Get-help

get-help \*ipaddr\* Etsii kaikki komennot joissa on sana ipaddr

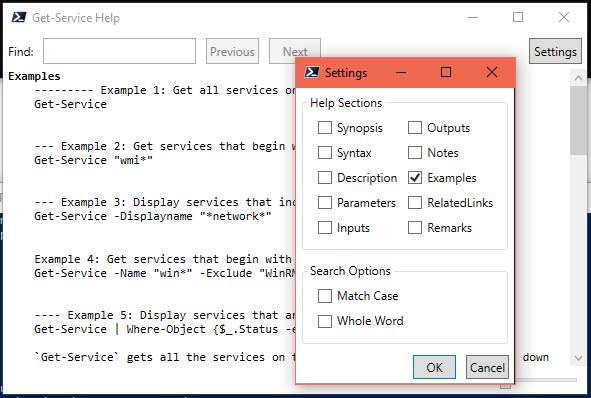
get-help komento Saa apuja halutusta komennosta

get-help get-service Antaa get-service avut

get-help -detailed komento Antaa enemmän apuja ja myös esimerkkejä

get-help komento -showwindow Avataan avut toiseen ikkunaan – helpottaa kopiointia

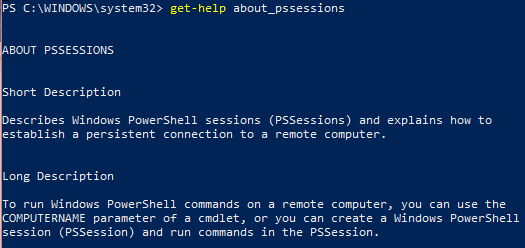
Ikkunan asetuksista voi määrittää mitä siinä näytetään, esim vain examples. Find kentästä voi etsiä ikkunan sisältöjä.



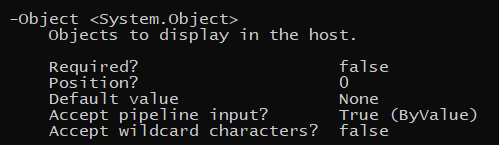
Jos parametri salli String[] niin sille voi antaa useita arvoja , erotettuina. Alla oleva komento antaisi kaikki servicet jotka alkavat d tai b kirjaimilla.

Esim. get-service -name d\*, b\*

get-help \*about\* löytää kaikki about sivut joissa on lisätietoja eri palveluista



get-help write-host -full – kertoo paljon lisätietoja komennon parametreista, mm voiko niitä putkittaa.



-online parametri avaa helpin selaimessa.



## TAB

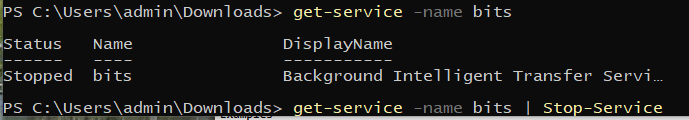
tabulaattori toimii

; merkillä saadaan ajettua kaksi tai useampia komentoja toistensa jälkeen

ipconfig ; hostname

## Putkitus

Ekan komennon tiedot annetaan toiselle komennolle



Alla putkitetaan kaikki palvelut csv muotoon haluttuun polkuun.

get-service | Export-Csv -path c:\testi.csv

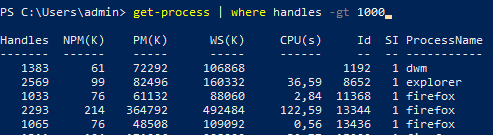
get-service sisältö tehdään html, nimi ja status ja sitten tallennetaan joo.html tiedostoon.



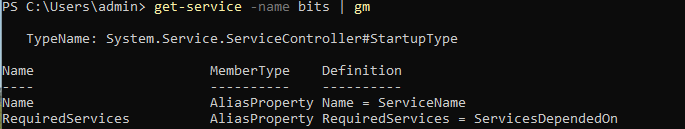
-whatif parametrilla komentoa ei ajeta, vaan powershell kertoo mitä tapahtuisi JOS ajaisit sen.



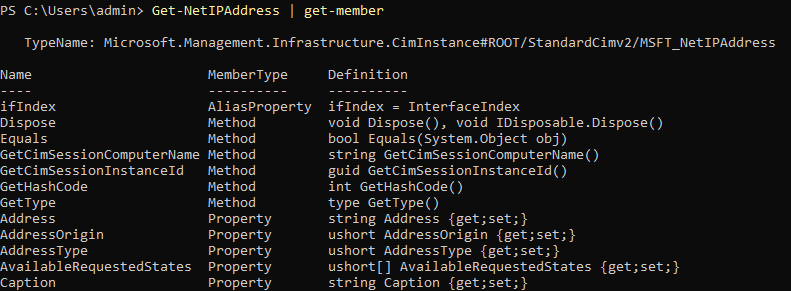
where parametrilla voidaan kanssa etsiä jotain tiettyä asiaa. Kuvassa on etsitty get-process tulosteesta handles kohdasta, kohdat jotka ovat suurempia kuin (gt, greater than) 1000.



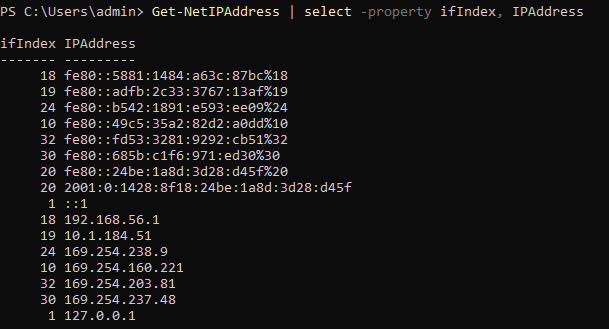
gm = get-member. Saadaan selville objektin metodit.



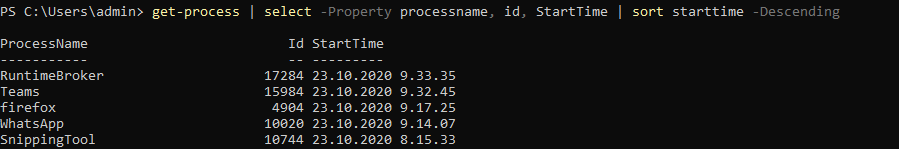
Esim. selvitetään mitkä ovat get-netipaddress komennon asetukset (property). Kun nämä tiedetään niitä voidaan kysellä helpommin.



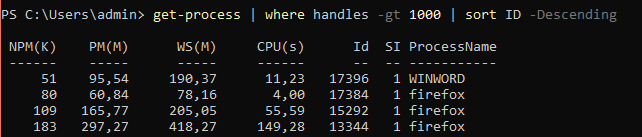
Valitaan asetukset ifindex ja ipaddress Select valitsimella.



Haetaan prosessit, vain processname, id ja starttime kohdat, sortataan starttimen perusteella



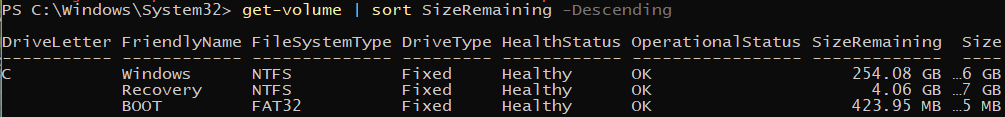
Get-process joiden handle on suurempi kuin 1000, sortataan ID mukaan, laskevassa järjestyksessä.



Muuten sama kuin yllä, mutta sortataan startimen mukaan alaspäin, konvertoidaan html tiedostoksi ja tallennetaan se C: asemalle

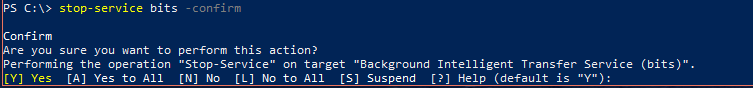


Haetaan kiintolevyt, sortataan jäljellä olevan koon mukaan laskevassa järjestyksessä.



## Varmistus

Ennen kuin komento ajetaan pitää varmistaa, että se halutaan suorittaa.



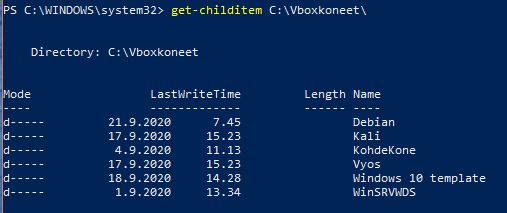
## Vertailu

Vertaillaan ovatko eka.txt ja toka.txt samat, tai mitä eroja niissä on.

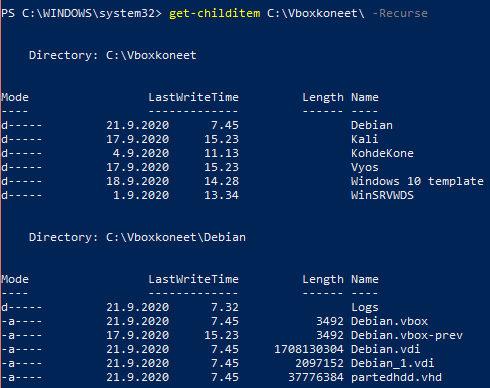


## get-Childitem

get-childitem näyttää mitä kansion sisällä on, samoin kuin esim ls /var/



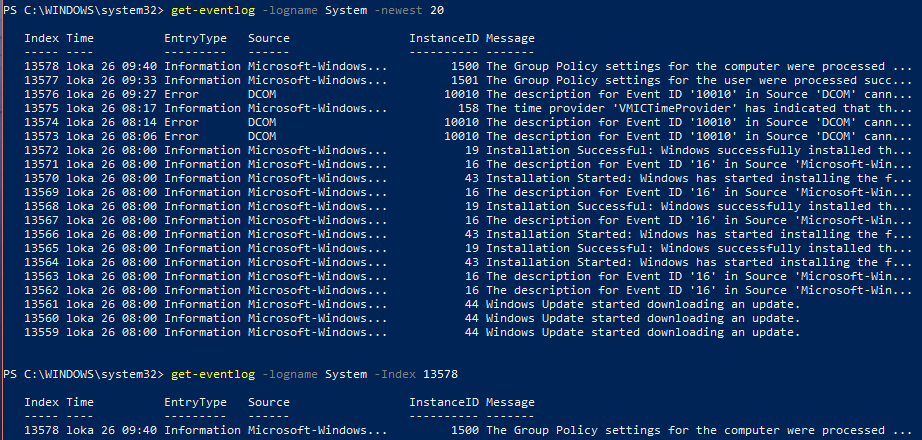
-recurse näyttää myös mitä kaikkien alikansioiden sisällä on.



## Eventlog

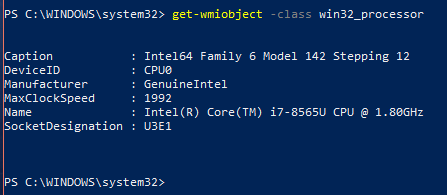
get-eventlog -logname valitaa mistä lokista -newest määritetään uusimmat

-index voidaan valita haluttu event



## get-wmiobject

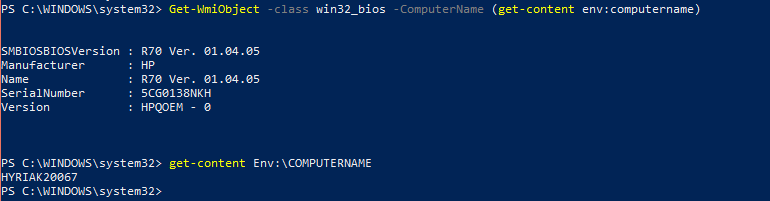
wmiobject voi hakea erilaisia tietoja Windows järjestelmästä kuten CPU, RAM, HDD ja näyttis.



Alla olevalla komennolla saadaan html tiedosto joka sisältää kaikki nämä win32\_ wmi objektit.

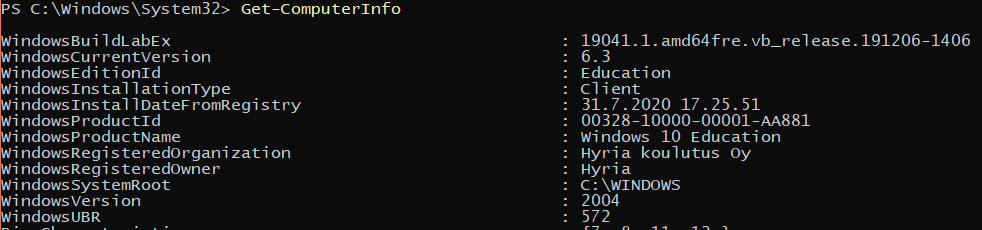


get-wmiobject hakee biosin tiedot tietokoneelle, jonka nimi saadaan get-content env:computernamelta.



## get-computerinfo

Antaa ison kasan hyödyllistä infoa



## Powershell Web Access

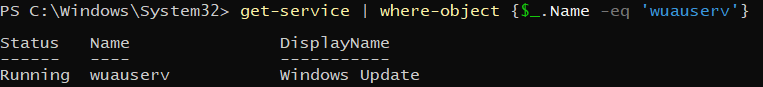
<https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/hh831611(v=ws.11)>

Tällä toiminnolla saat selaimen kautta muodostettua PS yhteyden laitteisiin, se toimii kaikilla selaimilla, käyttöjärjestelmillä ja mobiililaitteilla.

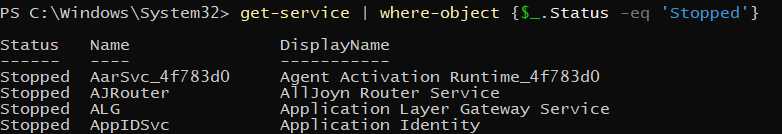
## Where-Object

Alias: Where

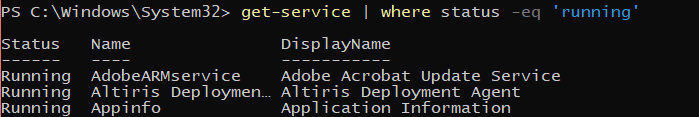
Komennolla voidaan hakea syötteestä jotakin haluttua. Alla haetaan objektia jonka nimi on wuauserv, eli windows update



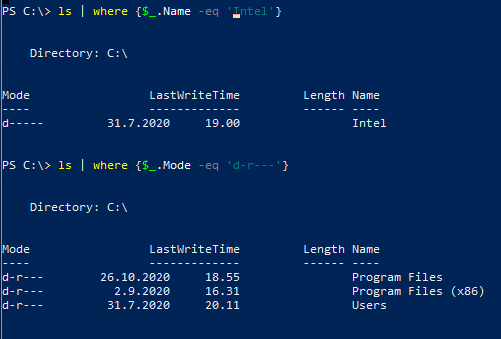
Haetaan kaikki palvelut jotka on pysäytetty



Yksinkertaistettu syntaksi:

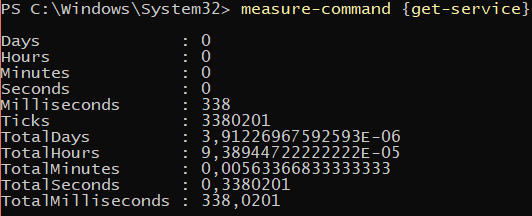


Haetaan ls komennon syötteestä nimet jotka = Intel. Alla haetaan kansiot joissa on vain lukuoikeus.



## Measure-command

Voidaan mitata kuinka kauan komennon suorittaminen kestää.



## Clear-content

Tyhjennetään tiedoston koko sisältö, mutta tiedosto säilyy.

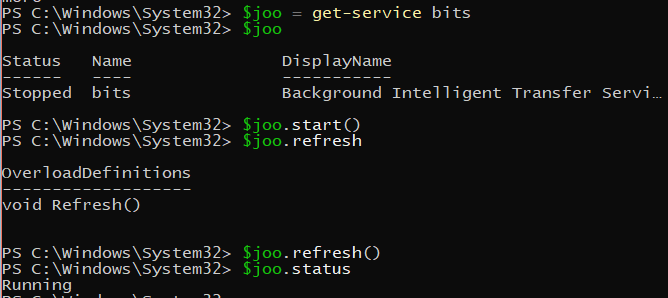


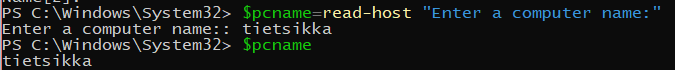
## Muuttujat

$muuttuja=”moro”



Toimii myös ikään kuin alias komento.





## new-pssession

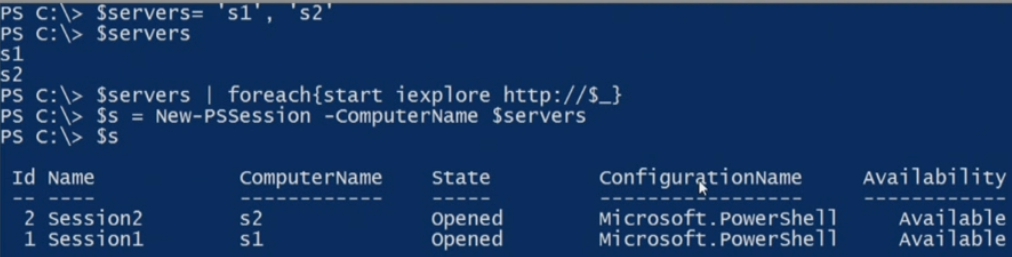
Luodaan muuttuja jolla saadaan syötettyä komentoja halutulle etälaitteelle.



Suoritetaan get-netadapter komento muuttujan mukaisella koneella.



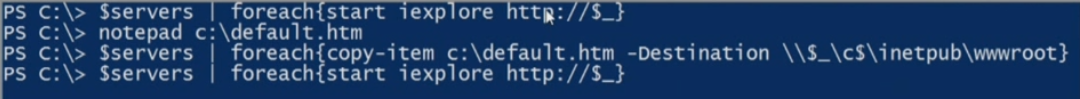
Muuttujassa voi olla useita arvoja, näin saadaan aloitettua etäyhteys useampaan koneeseen kerralla.



Asennetaan iis palvelu muuttujan mukaisille koneille.



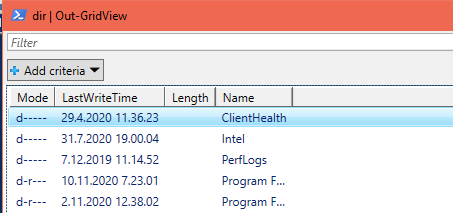
Luodaan isäntäkoneella default.htm tiedosto. Sitten se siirretään copy-item komennolla jokaiselle muuttujan mukaiselle etäkoneelle, niiden inetpub\wwwroot kansioon.



## out-gridview

Vaatii PowerShell ISE:n





## Write-output

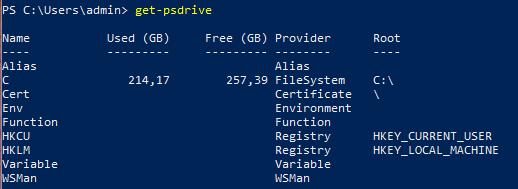
Käytä write-output älä write-host, koska sillä on metodeita.

## Write-warning



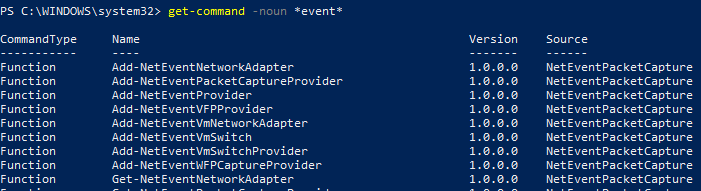
## Get-Psdrive

Näyttää kaikki asemat jotka ovat käytettävissä. Myös certit ja rekisterit

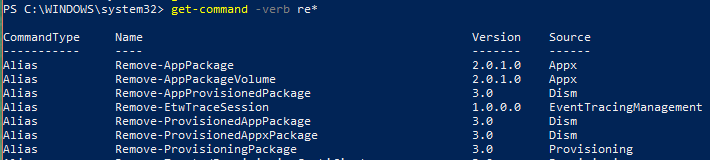


## get-command

Voidaan etsiä komentoja ja filteröidä niitä



Haetaan kaikki komennot jotka alkaa re kirjaimilla, esim restore ja remove



## Profiilit

PowerShelliin voi luoda profiileja eri käyttäjille. Oma profiili löytyy \user\documents\WindowsPowerShell\profile.ps1 tiedostosta

## PowerShell kommentit

Block kommenttien sekaan voi laittaa lisäinfoa .merkillä. Kuten alla on .synopsis. Get-help komennolla saisi lisäinfoa tästä skriptistä ja se kertoisi nämä .kohdat.

<#

.SYNOPSIS

This script/function does - What?

.DESCRIPTION

A more detailed description of why and how the function works.

.PARAMETER *DemoParam1*

The parameter DemoParam1 is used to define the value of blah and also blah.

.PARAMETER *DemoParam2*

The parameter DemoParam2 is used to define the value of blah and also blah.

.EXAMPLE

The example below does blah

PS C:\> *Example*

.EXAMPLE

Another example

.NOTES

Author: *Name*

Last Edit: yyyy-mm-dd

Version 1.0 - initial release of blah

Version 1.1 - update for blah

#>

## Pikanäppäimiä

Ctrl+End Delete all the characters in the line after the cursor.

Ctrl+L.Arrow Move left one word at a time.

Ctrl+R.Arrow Move right one word at a time.

Ctrl+S Pause or resume the display of output.

Delete Delete the character under the cursor

Esc Clear the current line.

Home Move to the beginning of the line.

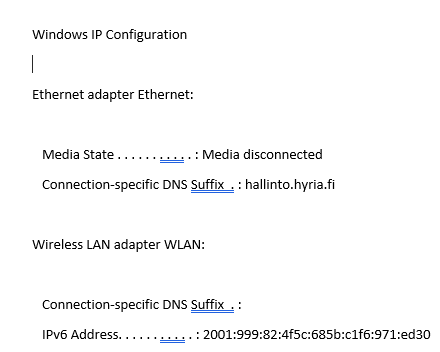
End Move to the end of the line.

## Clipboard

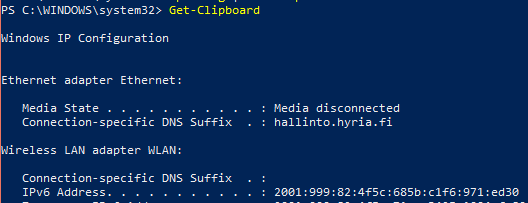
Lähetetään ipconfig tulosta leikepöydälle set-clipboard komennolla.



Leikepöydältä tiedot voidaan kopioida minne tahansa, kuten dokumenttiin tai PowerShelliin.



Get-clipboard voidaan hakea leikepöydältä.



# Palvelimen hallinta

## Rename, restart

**Rename-computer -newname WIN-DC1**

**Restart-computer**

## IP-asetusten vaihto

Kun halutaan antaa ENSIMMÄISTÄ kertaa IP-osoite, tulee se antaa New-netipaddress komennolla. Jos verkkokortilla on jo asetukset, niitä voi muokata set-netipaddress – default gateway ei voi muokata

**New-NetIPAddress -InterfaceIndex 8 -AddressFamily IPv4 -IPAddress 192.168.0.10 -PrefixLength 26 -DefaultGateway 192.168.0.1**

Asetetaan DNS osoite verkkokortille:

**Set-DnsClientServerAddress -InterfaceIndex 8 -ServerAddresses 8.8.8.8**

Haetaan DNS palvelimen osoite:

**Get-DnsClientServerAddress**

## Sammutus

**Stop-computer**

## Palveluiden asentaminen

**Get-WindowsFeature**

Asennetaan AD DS palvelu ja management toolsit, sen jälkeen asennetaan AD metsä ja annetaan domainin nimeksi hamkahot.local, asennetaan myös DNS

**Install-WindowsFeature -name AD-Domain-Services -IncludeManagementTools**

**Install-ADDSForest -DomainName hamkahot.local -InstallDns**

Otetaan käyttöön AD moduuli:

**import-module ActiveDirectory**

# Toimialueen hallinta

## OU

Tehdään uusi OU:

**New-ADOrganizationalUnit -name HAMKAHOT -Path "DC=hamkahot,dc=local"**

Ali-OU:

**New-ADOrganizationalUnit -name KONEET -Path "OU=TIETOHALLINTO,OU=HAMKAHOT,DC=hamkahot,dc=local"**

Oletuksena OU ei voi poistaa vaan tulee asetaa Accidental Deletion pois päältä:

**Set-ADObject -Identity "OU=poistoon,OU=HAMKAHOT,dc=hamkahot,dc=local" -ProtectedFromAccidentalDeletion $false**

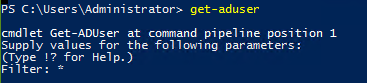
Sitten sen voi poistaa:

**Remove-ADOrganizationalUnit -Identity "OU=poistoon,OU=HAMKAHOT,dc=hamkahot,dc=local"**

## Käyttäjät

Käyttäjän poistaminen:

Get-aduser saa kätevästi selville Distinguished namen.



**get-aduser -Identity TestiKayttaja -Properties \***

**Remove-ADUser -Identity "CN=TestiKayttaja,OU=HAMKAHOT,DC=hamkahot,DC=local"**

Toinen tosi kätevä get-aduser:

Haetaan kaikki käyttäjät jotka löytyvät tietohallinto OU alta

**get-aduser -Filter \* -SearchBase "ou=tietohallinto,ou=hamkahot,dc=hamkahot,dc=local"**

Käyttäjän lisääminen:

Read-host kohta kysyy käyttäjältä mikä salasana asetetaan.

**New-ADUser -Name "TestiKayttaja" -GivenName "TestiKayttaja" -Surname "TK" -SamAccountName** **"TestiKayttaja" -UserPrincipalName "TestiKayttaja@hamkahot.local" -Path "OU=Tietohallinto,OU=hamkahot,dc=hamkahot,DC=local" -AccountPassword(Read-Host -AsSecureString "User Password") -Enabled $true**

## Käyttäjien hallinta:

Tilin disablointi ja enablointi

**Disable-ADAccount -Identity testi1**

**Enable-ADAccount -Identity testi1**

Käyttäjän poistaminen:

**Remove-ADUser -Identity testi1**

Salasanan nollaus ja vaihtaminen, tämän jälkeen kehote pyytää antamaan uuden salasanan.

**Set-ADAccountPassword -Identity testi2 -Reset**

Tilin vanhenemisen asettaminen:

**Set-ADAccountExpiration -Identity testi3 -DateTime "10/10/2020"**

Ei koskaan vanhene:

**Clear-ADAccountExpiration -Identity testi3**

## Ryhmät

Luodaan uusi ryhmä HR OU:n alle, se on global ja tyypiltään security:

**New-ADGroup -Name HRryhmä -Path "OU=HR,OU=hamkahot,DC=hamkahot,dc=local" -GroupScope Global -GroupCategory Security**

Ryhmien poistaminen:

**Remove-ADGroup -Identity "CN=delry,OU=markkinointi,OU=hamkahot,DC=hamkahot,dc=local"**

Tehdään ryhmästä suojattu:

**Set-ADObject -Identity "CN=suojary,OU=markkinointi,OU=hamkahot,DC=hamkahot,dc=local" -ProtectedFromAccidentalDeletion $true**

Käyttäjien lisääminen ryhmiin:

Lisätään muutama käyttäjä HRryhmä ryhmään:

**Add-ADGroupMember -Identity HRryhmä -Members testikayttaja, helpdesk, thpomo**

Tiedot ryhmästä eli näytetään mitä käyttäjiä ryhmässä on

**Get-ADGroupMember -Identity hrryhmä**

Tällä saadaan lisätietoja itse ryhmän tiedoista.

**Get-ADGroup -Identity hrryhmä**

Poistaminen ryhmästä:

**Remove-ADGroupMember -Identity hrryhmä -Members testikayttaja, thpomo, helpdesk**

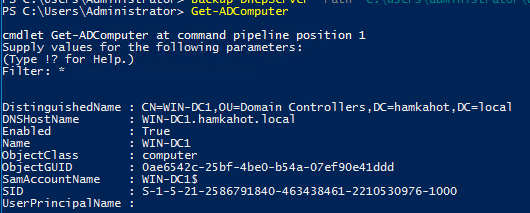
Itse ryhmän poistaminen:

**Remove-ADGroup -Identity feikkiry**

## Tietokoneet

Listan tietokoneista saa kätevästi:

Get-adcomputer ja \* filtteri



## Objektin siirtäminen

Objektin kuten käyttäjä, ryhmä tai OU (ota huomioon acc protection) voi siirtää toiseen kohteeseen. HUOM. Targetpathissa ei ole CN enää, koska eihän objektin nimi muuta, vain sen sijainit.

**Move-ADObject -Identity "CN=siirtoon,OU=HR,OU=HAMKAHOT,DC=hamkahot,DC=local" -TargetPath "OU=TIETOHALLINTO,OU=HAMKAHOT,DC=hamkahot,DC=local"**

## AD roskakori

AD roskakorin saa käyttöön seuraavasti:

Enable-ADOptionalFeature -Identity 'Recycle Bin Feature' -Scope ForestOrConfigurationSet -Target hamkahot.local

Tietokoneen liittäminen toimialueeseen:

**add-computer –domainname hamkahot.local -Credential hamkahot.local\helpdesk**

## DHCP

Asennetaan DHCP:

**Add-WindowsFeature dhcp -IncludeManagementTools**

Autorisoidaan se toimialueelle:

**Add-DhcpServerInDC -DnsName "win-dc1.hamkahot.local" -IPAddress 192.168.0.10**

Lisätään skooppi:

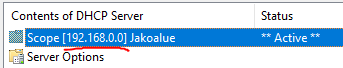
**Add-DhcpServerv4Scope -name Jakoalue -StartRange 192.168.0.30 -EndRange 192.168.0.62 -SubnetMask 255.255.255.192**

Ja lisäasetukset:

**Set-DhcpServerv4OptionValue -DnsServer 192.168.0.10 -DnsDomain hamkahot.local -Router 192.168.0.1**

Vuokra aika:

ScopeID tulee kohdasta:



**Set-DhcpServerv4Scope -ScopeId 192.168.0.0 -LeaseDuration 2.00:00:00**

DHCP backup:

**Backup-DhcpServer -Path "c:\users\administrator\desktop\backup"**

## Liikkuvat profiilit

Yleistä verkkojakojen testausta:

**Set-location \\win-dc1\profiilit**

Tehdään kansio, joka laitetaan jakoon:

**new-item -ItemType Directory -Name PROFIILIT -Path C:\**

**New-SmbShare -Path C:\PROFIILIT\ -Name PROFIILIT**

Annetaan jaolle oikeudet:

**Grant-SmbShareAccess -Name PROFIILIT -AccountName everyone -AccessRight full**

## Kotikansiot

Muuten sama kuin liikkuva profiili, mutta ota huomioon kansion/jaon oikeudet ja määritä kotikansio alla olevalla komennolla:

**Set-ADUser -Identity mark1 -HomeDrive U: -HomeDirectory \\win-dc1\KOTIKANSIOT\%username%**

## Remote

PowerShell session toiselle koneelle:

**Enter-PSSession -ComputerName win-wks1**

Käynnistetään yksi kone uudestaan palvelimelta käsin:

**Invoke-Command -ScriptBlock { Restart-Computer -Force } -ComputerName win-wks1**

Haetaan kahden koneen hostnamet:

**Invoke-Command -ScriptBlock { hostname } -ComputerName win-wks1, win-hd**

Gpupdate päivitys usealla koneelle etänä

**Invoke-Command -ScriptBlock { gpupdate /force } -ComputerName win-wks1, win-hd**

## Group Policy

Luodaan uusi ryhmäkäytäntö

**New-GPO -Name YleinenVJako -Comment "Yleinen verkkojako PowerShell tehty"**

Linkitetään se haluttuun paikkaan

**New-GPLink -Name YleinenVJako -Target "ou=hamkahot,dc=hamkahot,dc=local" -LinkEnabled yes**

Haetaan tietoja ryhmäkäytännöstä

**get-gpo YleinenVJako**

Gpupdate /force powershellillä, voi myös etänä suorittaa koneille

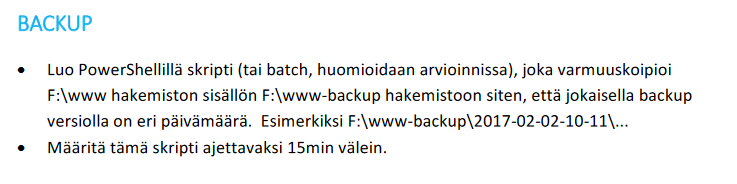
**Invoke-GPUpdate -Computer win-dc1 -Force**

# Skriptaus PowerShellillä

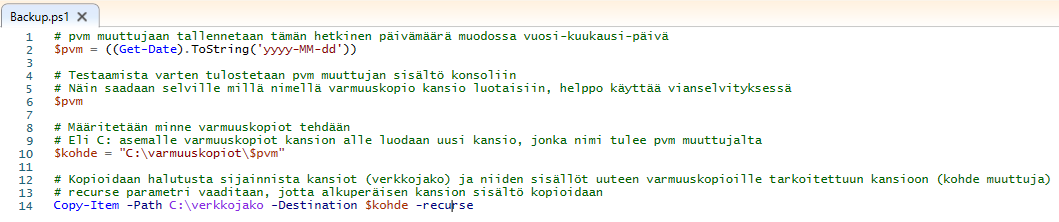
PowerShell skriptejä saa helposti tehtyä PowerShell ISE -ohjelmalla.

## Backup skripti

Taitaja tehtävissä on ollut alla oleva PowerShell-skripti useamman kerran.



Alla on yksi esimerkkiratkaisu sen tekemiseksi. Kuvassa on komennot kommentoituna. **Jos alla olevassa kuvassa teksti ei näy kunnolla. Avaa tämä 014 Powershell tiedosto Wordin työpöytäversiossa.**



Tämä skripti voitaisiin sitten ajastaa käyttämällä Windows palvelimen Task Scheduler -työkalua.

## Käyttäjien haku skripti

Skripti, jolla haetaan toimialueen vanhentuneet ja käytöstä poistetut käyttäjätilit sekä tallennetaan ne html tiedostoon.

Tällä haetaan kaikki disabloidut ja vanhentuneet käyttäjätilit:





Paranneltu versio:

