

# Powershell

## Change vswitch from internal to external

```
Set-VMSwitch -Name vSwitch -NetAdapterName bond0
```

If you have a duplicate

```
Get-VMNetworkAdapter -VMName svtest | ? MacAddress -eq '000000000000' | Remove-VMNetworkAdapter
```

## Roles&Features

### Install

List

```
Get-WindowsFeature
```

Install

```
Install-WindowsFeature -name <name>
```

Remove

```
Remove-WindowsFeature -name <name>
```

## Update OS

```
Install-Module PSWindowsUpdate  
Get-Command -module PSWindowsUpdate
```

Then you will need to register to use the Microsoft Update Service not just the default Windows Update Service.

```
Add-WUServiceManager -ServiceID 7971f918-a847-4430-9279-4a52d1efe18d
```

```
Get-WUInstall -MicrosoftUpdate -AcceptAll -AutoReboot
```

## List Updates

```
Get-WUInstall -MicrosoftUpdate -ListOnly
```

or

```
Get-WUList -MicrosoftUpdate
```

# Firewall

Disable unwanted ports

(Replace 1.1.1.1 with public IP)

TCP

```
New-NetFirewallRule -DisplayName "Block EXTERNAL AD PORTS TCP" -Action Block -Description  
"Block non secure ports" -Enabled True -LocalAddress 1.1.1.1/30 -Protocol TCP -LocalPort 53,  
135-139, 389, 445 -RemotePort Any -Program Any -RemoteAddress Any
```

UDP

```
New-NetFirewallRule -DisplayName "Block EXTERNAL AD PORTS UDP" -Action Block -Description  
"Block non secure ports" -Enabled True -LocalAddress 1.1.1.1/30 -Protocol UDP -LocalPort 53,  
135-139, 389, 445 -RemotePort Any -Program Any -RemoteAddress Any
```

# Active Directory DC

## Setup a primary domain controller

```
Rename-Computer -NewName "ad01"
```

```
Restart-Computer
```

```
Install-WindowsFeature -Name AD-Domain-Services -IncludeManagementTools
```

```
Install-ADDSForest -domainname testdomain.local -SafeModeAdministratorPassword (ConvertTo-  
SecureString "current-user-password" -AsPlainText -Force)
```

Restart-Computer

```
REG.exe ADD "HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters" /V DisabledComponents /t REG_DWORD /D 0xFFFFFFFF /F
REG.exe ADD "HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters" /V IPAutoconfigurationEnabled /t REG_DWORD /D 0 /F
REG.exe ADD "HKLM\Software\Policies\Microsoft\Windows NT\Terminal Services" /V fDisableCpm /t REG_DWORD /D 1 /F
```

Get-WuInstall -Install -AcceptAll -IgnoreReboot -Verbose

Restart-Computer

```
New-NetFirewallRule -DisplayName "Block EXTERNAL AD PORTS TCP" -Action Block -Description "Block non secure ports" -Enabled True -LocalAddress {WAN/SUBNET} -Protocol TCP -LocalPort 53, 135-139, 389, 636, 445 -RemotePort Any -Program Any -RemoteAddress Any
```

```
New-NetFirewallRule -DisplayName "Block EXTERNAL AD PORTS UDP" -Action Block -Description "Block non secure ports" -Enabled True -LocalAddress {WAN/SUBNET} -Protocol UDP -LocalPort 53, 135-139, 389, 636, 445 -RemotePort Any -Program Any -RemoteAddress Any
```

```
$DNS = get-dnsserversetting -all
$DNS.listeningIpAddress = @("{LAN-IP}")
Set-DnsServersetting -inputobject $DNS
Clear-DnsServerCache -force
```

```
Get-NetAdapter -Name "Ethernet 0" | Set-DnsClient -RegisterThisConnectionsAddress:$false
```

## Setup a secondary domain controller

Add role

```
Add-WindowsFeature AD-Domain-Services
```

Install and configure role

```
Install-ADDSDomainController -NoGlobalCatalog:$false -CreateDnsDelegation:$false -Credential (Get-Credential domain.local\Administrator) -CriticalReplicationOnly:$false -DatabasePath "C:\Windows\NTDS" -DomainName "domain.local" -InstallDns:$true -LogPath "C:\Windows\NTDS" -
```

```
NoRebootOnCompletion:$false -SiteName "Default-First-Site-Name" -SysvolPath  
"C:\Windows\SYSTEM32" -SafeModeAdministratorPassword (ConvertTo-SecureString 'DOM@INP@SSWORD' -  
AsPlainText -Force) -Force:$true
```

Sync AD and check status

```
repadmin /syncall /A  
repadmin /replsummary
```

# IIS

List Domains / Bindings

```
Import-Module Webadministration  
Get-ChildItem -Path IIS:\Sites
```

# DFS

Create new replication group

```
DFSNew-DfsReplicationGroup -GroupName "Group-Name" -Description "Replication DFSR" -  
DomainName "AD-DOMAIN.com" -WhatIf
```

Add Members

```
Add-DfsrMember -GroupName "Group-Name" -ComputerName "SERVER1","SERVER2" -WhatIf
```

FolderName : List of folders to replicate

ContentPath : Local path of destination of the files that will be replicated

ComputerName : Primary Serve

```
Set-DfsrMembership -GroupName "Group-Name" -DomainName "AD-DOMAIN.com" -FolderName  
"List/Folder/To/Replicate" -ContentPath "C:\Local\Path\To\Replicate" -ComputerName "SERVER01"  
-PrimaryMember $True -Force -WhatIf | Format-Table *name,*path,primary* -auto -wrap
```

# Failover Cluster

List all VM's on a VLAN

```

$ClusterNodes = Get-Cluster | Get-ClusterNode
foreach ($ClusterNode in $ClusterNodes)
{
    [Get-VM -ComputerName $ClusterNode | Where {$_.NetworkAdapters.VlanSetting.AccessVlanId -eq
    "2603"}]
}

```

## WMI Test (Failover Cluster)

Script for cluster WMI status

```

Set-ExecutionPolicy unrestricted

cls

If(import-module failoverclusters)
{

    Write-Host "Imported Cluster module"

}

Write-Host "Getting the cluster nodes..." -NoNewline
$nodes = Get-ClusterNode
Write-host "Found the below nodes "
Write-host " "
$nodes
Write-host ""
Write-host "Running the WMI query..."
Write-host " "
ForEach ($Node in $nodes)
{
    Write-Host -NoNewline $node

    if($Node.State -eq "Down")
    {

        Write-Host -ForegroundColor White " : Node down skipping"

    }
}

```

```

else
{

    Try
    {
        #success

        $result = (get-wmiobject -class "MSCluster_CLUSTER" -namespace "root\MSCluster"
-authentication PacketPrivacy -computername $Node -erroraction stop).__SERVER
        Write-host -ForegroundColor Green " : WMI query succeeded "
    }
    Catch
    {
        #Failure

        Write-host -ForegroundColor Red -NoNewline " : WMI Query failed "
        Write-host "/"$_.Exception.Message
    }
}

}

```

### Allow WMI firewall

```

netsh advfirewall firewall set rule group="remote administration" new enable=yes
netsh advfirewall firewall set rule group="Windows Remote Management" new enable=yes
netsh advfirewall firewall set rule group="windows management instrumentation (wmi)" new
enable=yes

```

## Replication

### Check replication

```

foreach-object {Measure-VMReplication | Select VMName, ReplicationHealth, AvgReplSize,
PrimaryServerName, CurrentReplicaServerName | ft}

```

## Reboot in recovery mode

### Reboot in recovery mode

```
shutdown /r /o /f /t 00
```

# Networking Stuff

```
#Removing a bond
Get-NetLbfoTeam
Remove-NetLbfoTeam -name bond0

#Find Adpt Names
Get-NetAdapter

Remove-NetIPAddress -InterfaceAlias "OnBoard LAN 2"
Remove-NetIPAddress -InterfaceAlias "OnBoard LAN 1"
Remove-NetIPAddress -IPAddress 10.1.2.10

New-NetIPAddress -InterfaceAlias "OnBoard LAN 1" -IPAddress 10.1.2.120 -PrefixLength 24 -
DefaultGateway 10.1.2.10 -Type Unicast
```

# Check Hyper-V replication health

```
$ClusterName = "<Name of your cluster>"
Get-ClusterNode -Cluster $ClusterName
Get-ClusterNode -Cluster $ClusterName | foreach-object {Measure-VMReplication -ComputerName $_
| Select VMName, AvgReplSize, PrimaryServerName, CurrentReplicaServerName | ft}
```

# Convert VHD/VHDX

```
Convert-VHD -Path c:\VM\my-vhdx.vhdx -DestinationPath c:\New-VM\new-vhdx.vhds
```

or

```
Convert-VHD -Path c:\VM\my-vhdx.vhdx -DestinationPath c:\New-VM\new-vhdx.vhdx -VHDType Dynamic
```

# Repair WMI Repo

```
Stop-Service Winmgmt -Force; winmgmt /resetrepository
```

---

Revision #13

Created 29 October 2018 00:02:24 by Dave

Updated 21 January 2020 19:00:00 by Dave