MDT 2013

# Server Setup

Setup a Windows server OS as you normally do. Set the DVD drive to drive E:. Add a hard disk D: and format it with NTFS.

Start Powershell and type:

Add-WindowsFeature WDS

WDSutil /Initialize-Server /RemInst:D:\RemoteInstall /Authorize

WDSutil /Set-Server /AnswerClients:All

Download ADK 10 from: <https://developer.microsoft.com/nl-nl/windows/hardware/windows-assessment-deployment-kit>

Install ADK 10 with the following command:

Adksetup.exe /features OptionId.DeploymentTools OptionId.WindowsPreinstallationEnvironment OptionId.ImagingAndConfigurationDesigner OptionId.UserStateMigrationTool /norestart /ceip off /q

Download MDT from: <https://www.microsoft.com/download/details.aspx?id=54259>

Install MDT 2013 Update 2 with the following command:

MicrosoftDeploymentToolkit\_x64.msi /qb /norestart

# Reinstall and configure WDS

WDS can sometimes do crazy things and if you suspect WDS problems, you should first remove and reinstall WDS. Here are the PowerShell Commands for a WDS reinstall:

Remove-WindowsFeature WDS –Restart

RD D:\RemoteInstall –Recurse –Confirm:$False

Add-WindowsFeature WDS

WDSutil /Initialize-Server /RemInst:D:\RemoteInstall /Authorize

WDSutil /Set-Server /AnswerClients:All

WDSUtil /Add-Image /ImageFile:"D:\DeploymentShare\Boot\LiteTouchPE\_x86.wim" /ImageType:Boot

WDSUtil /Add-Image /ImageFile:"D:\DeploymentShare\Boot\LiteTouchPE\_x64.wim" /ImageType:Boot

# Create the Deployment share

Import-Module "C:\Program Files\Microsoft Deployment Toolkit\bin\MicrosoftDeploymentToolkit.psd1"

MD D:\DeploymentShare

new-PSDrive -Name "DS001" -PSProvider "MDTProvider" -Root "D:\DeploymentShare" -Description "MDT Deployment Share" -NetworkPath "\\$ENV:COMPUTERNAME\DeploymentShare$" -Verbose | add-MDTPersistentDrive –Verbose

Download MDTExitNameToGuid.vbs from: <http://blogs.technet.com/b/deploymentguys/archive/2010/04/18/banish-guids-from-customsettings-ini-in-mdt.aspx>

Extract the vbs script to the Control folder in the DeploymentShare.

If you are going to deploy multiple MDT installations in different Subnets with different gateways, change the Bootstrap.ini in the Control Folder in the following way:

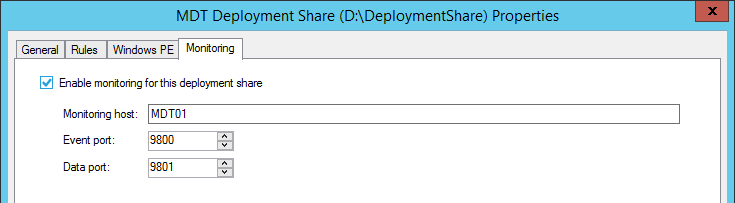
|  |
| --- |
| [Settings]  Priority=DefaultGateway, Default  [Default]  SkipBDDWelcome=Yes  Userid=Administrator  UserPassword=Administrator Password  Userdomain=Domain where MDT lives  [DefaultGateway]  192.168.1.1=Location A  192.168.2.1=Location B  [Location A]  Deployroot=\\MDT01\DeploymentShare$  [Location B]  Deployroot=\\MDT02\DeploymentShare$ |

If there is only one MDT server, this can be as simple as:

|  |
| --- |
| [Settings]  Priority=Default  [Default]  SkipBDDWelcome=Yes  Userid=Administrator  UserPassword=Administrator Password  Userdomain=Domain where MDT lives  DeployRoot=\\MDT01\DeploymentShare$ |

# Enable Monitoring

Right click on the DeploymentShare and check the box to monitor the deployments



# Bootfiles in WDS

After changing the Bootstrap.ini, the bootfiles can be created by the Update Deploymentshare command and imported in WDS.

update-MDTDeploymentShare -path "DS001:" -Force –Verbose

Import the image for the first time:

WDSUtil /Add-Image /ImageFile:"D:\DeploymentShare\Boot\LiteTouchPE\_x86.wim" /ImageType:Boot

WDSUtil /Add-Image /ImageFile:"D:\DeploymentShare\Boot\LiteTouchPE\_x64.wim" /ImageType:Boot

Replace the image:

WDSUtil /replace-image /image:"Lite Touch Windows PE (x86)" /imagetype:boot /Architecture:x86 /ReplacementImage /ImageFile:"D:\DeploymentShare\Boot\LiteTouchPE\_x86.wim"

WDSUtil /replace-image /image:"Lite Touch Windows PE (x64)" /imagetype:boot /Architecture:x64 /ReplacementImage /ImageFile:"D:\DeploymentShare\Boot\LiteTouchPE\_x64.wim"

You can also use the ISO files from the D:\DeploymentShare\Boot folder if booting from WDS is not possible.

# MDT PowerShell Module

All MDT PowerShell commands need the MDT module and a mapped DeploymentShare.

For the next commands, this should be entered first:

Import-Module "C:\Program Files\Microsoft Deployment Toolkit\bin\MicrosoftDeploymentToolkit.psd1"

New-PSDrive -Name "DS001" -PSProvider MDTProvider -Root "D:\DeploymentShare"

# Operating Systems

Now insert the DVD of the OS you would like to deploy in the DVD player. We use the Windows DVDs downloaded from the Microsoft Website.

import-mdtoperatingsystem -path "DS001:\Operating Systems" -SourcePath "E:\" -DestinationFolder "Windows 8.1 Enterprise x64 NL" –Verbose

Set-Location "DS001:\Operating Systems"

ren 'Windows 8.1 Enterprise in Windows 8.1 Enterprise x64 NL install.wim' 'Windows 8.1 Enterprise x64 NL'

import-mdtoperatingsystem -path "DS001:\Operating Systems" -SourcePath "E:\" -DestinationFolder "Windows 8.1 Enterprise x64 EN" –Verbose

Set-Location "DS001:\Operating Systems"

ren 'Windows 8.1 Enterprise in Windows 8.1 Enterprise x64 EN install.wim' 'Windows 8.1 Enterprise x64 EN'

import-mdtoperatingsystem -path "DS001:\Operating Systems" -SourcePath "E:\" -DestinationFolder "Windows 8.1 Enterprise x86 NL" –Verbose

Set-Location "DS001:\Operating Systems"

ren 'Windows 8.1 Enterprise in Windows 8.1 Enterprise x86 NL install.wim' 'Windows 8.1 Enterprise x86 NL'

import-mdtoperatingsystem -path "DS001:\Operating Systems" -SourcePath "E:\" -DestinationFolder "Windows 8.1 Enterprise x86 EN" –Verbose

Set-Location "DS001:\Operating Systems"

ren 'Windows 8.1 Enterprise in Windows 8.1 Enterprise x86 EN install.wim' 'Windows 8.1 Enterprise x86 EN'

import-mdtoperatingsystem -path "DS001:\Operating Systems" -SourcePath "E:\" -DestinationFolder "Windows 10 x64 NL" –Verbose

Set-Location "DS001:\Operating Systems"

ren 'Windows 10 Enterprise in Windows 10 x64 NL install.wim' 'Windows 10 Enterprise x64 NL'

import-mdtoperatingsystem -path "DS001:\Operating Systems" -SourcePath "E:\" -DestinationFolder "Windows 10 x64 EN" –Verbose

Set-Location "DS001:\Operating Systems"

ren 'Windows 10 Enterprise in Windows 10 x64 EN install.wim' 'Windows 10 Enterprise x64 EN'

import-mdtoperatingsystem -path "DS001:\Operating Systems" -SourcePath "E:\" -DestinationFolder "Windows 10 x86 NL" –Verbose

Set-Location "DS001:\Operating Systems"

ren 'Windows 10 Enterprise in Windows 10 x86 NL install.wim' 'Windows 10 Enterprise x86 NL'

import-mdtoperatingsystem -path "DS001:\Operating Systems" -SourcePath "E:\" -DestinationFolder "Windows 10 x86 EN" –Verbose

Set-Location "DS001:\Operating Systems"

ren 'Windows 10 Enterprise in Windows 10 x86 EN install.wim' 'Windows 10 Enterprise x86 EN'

import-mdtoperatingsystem -path "DS001:\Operating Systems" -SourcePath "E:\" -DestinationFolder "Windows Server 2012 R2" -Verbose

Set-Location "DS001:\Operating Systems"

ren 'Windows Server 2012 R2 SERVERDATACENTER in Windows Server 2012 R2 install.wim' 'Windows Server 2012 R2 DataCenter'

ren 'Windows Server 2012 R2 SERVERDATACENTERCORE in Windows Server 2012 R2 install.wim' 'Windows Server 2012 R2 DataCenterCore'

ren 'Windows Server 2012 R2 SERVERSTANDARD in Windows Server 2012 R2 install.wim' 'Windows Server 2012 R2 Standard'

ren 'Windows Server 2012 R2 SERVERSTANDARDCORE in Windows Server 2012 R2 install.wim' 'Windows Server 2012 R2 StandardCore'

# Drivers

Put all the drivers that you need in a folder like D:\Drivers

import-mdtdriver -path "DS001:\Out-of-Box Drivers" -SourcePath "D:\Drivers" -Verbose

# Task Sequences

import-mdttasksequence -path "DS001:\Task Sequences" -Name "Windows 8 x64 EN" -Template "Client.xml" -ID "W8x64EN" -Version "1.0" -OperatingSystemPath "DS001:\Operating Systems\Windows 8.1 Enterprise x64 EN"

import-mdttasksequence -path "DS001:\Task Sequences" -Name "Windows 8 x64 NL" -Template "Client.xml" -ID "W8x64NL" -Version "1.0" -OperatingSystemPath "DS001:\Operating Systems\Windows 8.1 Enterprise x64 NL"

import-mdttasksequence -path "DS001:\Task Sequences" -Name "Windows 8 x86 EN" -Template "Client.xml" -ID "W8x86EN" -Version "1.0" -OperatingSystemPath "DS001:\Operating Systems\Windows 8.1 Enterprise x86 EN"

import-mdttasksequence -path "DS001:\Task Sequences" -Name "Windows 8 x86 NL" -Template "Client.xml" -ID "W8x86NL" -Version "1.0" -OperatingSystemPath "DS001:\Operating Systems\Windows 8.1 Enterprise x86 NL"

import-mdttasksequence -path "DS001:\Task Sequences" -Name "Windows 10 x64 EN" -Template "Client.xml" -ID "W10x64EN" -Version "1.0" -OperatingSystemPath "DS001:\Operating Systems\Windows 10 Enterprise x64 EN"

import-mdttasksequence -path "DS001:\Task Sequences" -Name "Windows 10 x64 NL" -Template "Client.xml" -ID "W10x64NL" -Version "1.0" -OperatingSystemPath "DS001:\Operating Systems\Windows 10 Enterprise x64 NL"

import-mdttasksequence -path "DS001:\Task Sequences" -Name "Windows 10 x86 EN" -Template "Client.xml" -ID "W10x86EN" -Version "1.0" -OperatingSystemPath "DS001:\Operating Systems\Windows 10 Enterprise x86 EN"

import-mdttasksequence -path "DS001:\Task Sequences" -Name "Windows 10 x86 NL" -Template "Client.xml" -ID "W10x86NL" -Version "1.0" -OperatingSystemPath "DS001:\Operating Systems\Windows 10 Enterprise x86 NL"

import-mdttasksequence -path "DS001:\Task Sequences" -Name "Windows 2012 DataCenter" -Template "Server.xml" -ID "W2012DC" -Version "1.0" -OperatingSystemPath "DS001:\Operating Systems\Windows Server 2012 R2 DATACENTER"

import-mdttasksequence -path "DS001:\Task Sequences" -Name "Windows 2012 Standard" -Template "Server.xml" -ID "W2012STD" -Version "1.0" -OperatingSystemPath "DS001:\Operating Systems\Windows Server 2012 R2 STANDARD"

# Import Applications

$SourcePath = "D:\Software\"

import-MDTApplication -path "DS001:\Applications" -enable "True" -Name "Adobe Shockwave Player" -ShortName "Adobe Shockwave Player" -Version "" -Publisher "" -Language "" -CommandLine "wscript install.vbs" -WorkingDirectory ".\Applications\Adobe Shockwave Player" -ApplicationSourcePath $SourcePath"Adobe Shockwave Player" -DestinationFolder "Adobe Shockwave Player"

Example script to import applications

|  |
| --- |
| $DeploymentShare = "D:\DeploymentShare"  $Source = "\\server01\Dropbox$\Software"  $Import = $false # or $true  If (!([Security.Principal.WindowsPrincipal][Security.Principal.WindowsIdentity]::GetCurrent()).IsInRole([Security.Principal.WindowsBuiltInRole] "Administrator")) {Start-Process powershell.exe "-NoProfile -ExecutionPolicy Bypass -File `"$PSCommandPath`"" -Verb RunAs; exit}  #######  # De regels hieronder bepalen de logica van het script  #######  Import-Module "C:\Program Files\Microsoft Deployment Toolkit\bin\MicrosoftDeploymentToolkit.psd1"  New-PSDrive -Name "DS001" -PSProvider MDTProvider -Root $DeploymentShare  Function ImportApp(){  param([Parameter(Mandatory=$true)][string]$Command,[Parameter(Mandatory=$true)][string]$Naam,[Parameter(Mandatory=$False)]$Reboot)  $location = Get-Location  Set-Location "DS001:\Applications"  If (Test-Path $Naam){If ($Naam -notmatch "VDA" -and $Naam -notmatch "XenConvert"){Remove-Item $Naam -Recurse -Force}}  If ($Import) {  Import-MDTApplication -Name $Naam -ShortName $Naam -enable "True" -CommandLine $Command -WorkingDirectory ".\Applications\$Naam" -ApplicationSourcePath (Join-Path $Source $Naam) -DestinationFolder $Naam  } ELSE {  If ($Command -Match " ") {$CMD = $Command.Split(" "); $C = $CMD[1]; $Command = $CMD[0]+" """+(Join-Path $Source $Naam\$C)+""""} ELSE {$Command = """"+(Join-Path $Source $Naam\$Command)+""""}  Import-MDTApplication -Name $Naam -ShortName $Naam -enable "True" -CommandLine $Command -NoSource  }  If ($Reboot){Set-ItemProperty -path "DS001:\Applications\$Naam" -Name Reboot -Value True}  While (Test-Path -Path $DeploymentShare\Control\Applications.LOCK){Start-Sleep -Milliseconds 10}  Set-Location $location  }  #######  # Voeg nieuwe applicaties hieronder toe  # ImportApp("commandline") ("App naam/gelijk aan de mapnaam op $Source share") [("True voor reboot")]  #######  del "DS001:\Applications\\*"  ForEach ($Folder in Get-Childitem -Path $Source) {  If (!(Test-Path ($Folder.FullName + "\ExcludeMDT.txt"))) {  If (Test-Path ($Folder.FullName + "\install.bat")) {  If (Test-Path ($Folder.FullName + "\reboot.txt")) {  ImportApp("install.bat") ($Folder.Name) ("True")  } Else {  ImportApp("install.bat") ($Folder.Name)  }  }  If (Test-Path ($Folder.FullName + "\install.vbs")) {  If (Test-Path ($Folder.FullName + "\reboot.txt")) {  ImportApp("wscript install.vbs") ($Folder.Name) ("True")  } Else {  ImportApp("wscript install.vbs") ($Folder.Name)  }  }  }  }  #######  # Voeg de nieuwe applicatie toe in de installatie volgorde voor XenApp  #######  # $Order = $Order + (Get-ItemProperty -Path "DS001:\Applications\AppName").Guid  #$Order = @()  #Set-ItemProperty -path "DS001:\Applications\VDA" -Name Dependency -Value $Order |

# CustomSettings.ini

This is an Example CustomSettings.ini

|  |
| --- |
| [Settings]  Priority=Default, NameToGuid  Properties=ConvertList  [Default]  OSInstall=YES  \_SMSTSORGNAME=Displaytext  EventService=http://MDT01:9800  DoNotCreateExtraPartition=YES  FinishAction=REBOOT  SkipAdminPassword=YES  AdminPassword=Password  SkipBitLocker=YES  SkipBitLockerDetails=YES  SkipBuild=NO  SkipDeploymentType=YES  ;DeploymentType=NEWCOMPUTER  SkipLocaleSelection=YES  AreaCode=010  Language=nl-NL  UILanguage=nl-NL  SystemLocale=nl-NL  UserLocale=nl-NL  InputLocale=0413:00020409  ;KeyboardLocale=nl-NL  KeyboardLocale=0413:00020409  SkipPackageDisplay=YES  SkipProductKey=YES  SkipUserData=YES  UserDataLocation=NONE  SkipFinalSummary=YES  SkipSummary=YES  SkipTimeZone=YES  TimeZoneName=W. Europe Standard Time  TimeZone=110  SkipCapture=YES  SkipDomainMembership=YES  JoinDomain=Domain.lan  DomainAdmin=DomainJoinAccount  DomainAdminDomain=Domain.lan  DomainAdminPassword=Password  MachineObjectOU=DC=Domain,DC=lan  SkipTaskSequence=YES  TaskSequenceID=  SkipComputerName=YES  OSDComputername=Templatename  OSDAdapterCount=1  OSDAdapter0EnableDHCP=FALSE  OSDAdapter0IPAddressList=192.168.1.49  OSDAdapter0SubnetMask=255.255.255.0  OSDAdapter0Gateways=192.168.254.254  OSDAdapter0GatewayCostMetrics=Automatic  OSDAdapter0DNSServerList=192.168.1.1,192.168.1.2  SkipApplications=YES  SkipAppsOnUpgrade=YES  [NameToGuid]  UserExit=MDTExitNameToGuid.vbs  Applications001=Adobe Shockwave Player  Applications002=Java 8  Applications003=Adobe Reader 11  Applications004=MS Office Professional Plus 2013  Applications005=VMware Tools  Applications006=Create D Partition  Applications007=Finish Installation  Applications008=Regfiles  ConvertList=#ConvertNameToGUIDInList("Applications", "Applications")# |