**CloudPlatform**



CloudPlatform

How to automatically join a CloudPlatform

VM to an Active Directory domain after deployment

  
www.citrix.com

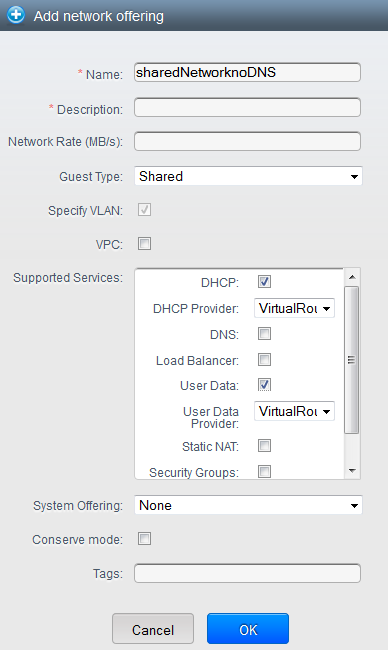
Objectives

Windows VMs are typically integrated in the domain when used. In classical deployments with CloudPlatform this has to be done manually after the deployment task. This document describes how virtual machines can be added to the domain automatically after a VM instance has been deployed. As part of the procedure this includes renaming the VM to the name the user specified during VM deployment wizard.

This documentation uses Windows 7 Enterprise 64bit as an example.

Creating new network offering

This section affects deployments where a virtual router is in place for the network the VM is connected to (e.g. isolated and shared networks in advanced zone). In the default configuration the CloudPlatform network offering for shared and isolated networks include DNS service for the router. As the DNS service does not forward tags required for AD authentication, AD integration won’t work using default network offerings. Hence a new network offering has to be created which does not include DNS service. All other services that are part of the default network offerings don’t matter. See here an example for a network offering for shared networks:

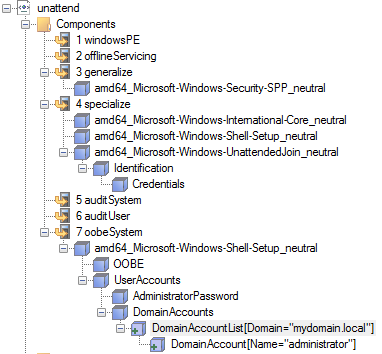


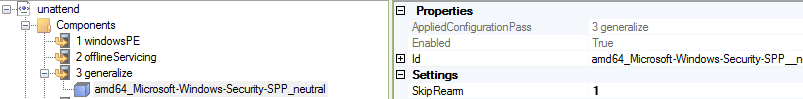
Integrating Sysprep in a Windows VM

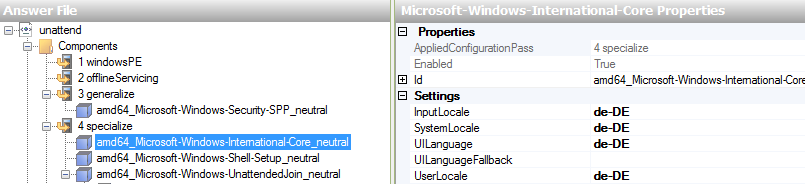
The prerequisite for this task is having a Windows VM instance which is prepared with the customer required applications and modifications. To generalize this virtual machine, Windows sysprep is used. The procedure how to do this is documented in the CloudPlatform Admin Guide (<http://support.citrix.com/article/CTX133615> ) page 97 “Sysprep for Windows Server 2008 R2).

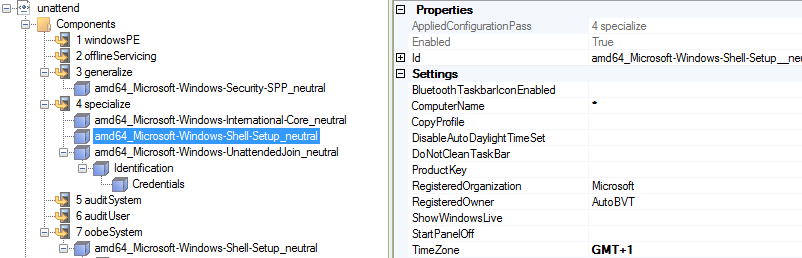
Follow this guideline until point 9. Don’t execute the sysprep.exe command yet.

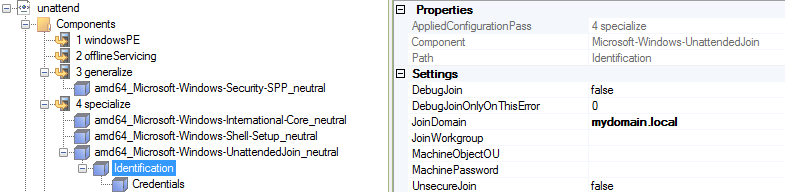
To enable domain join for sysprep you have to add additional parameters to your unattend.xml file using Windows System Image Manager. See an example of the parameters required below:

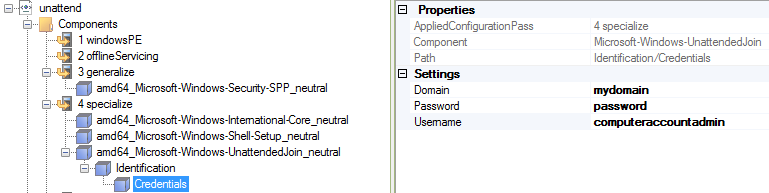


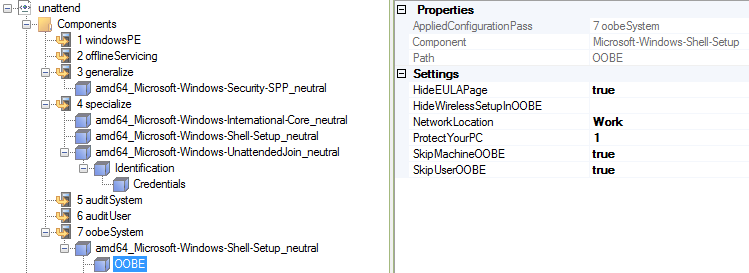


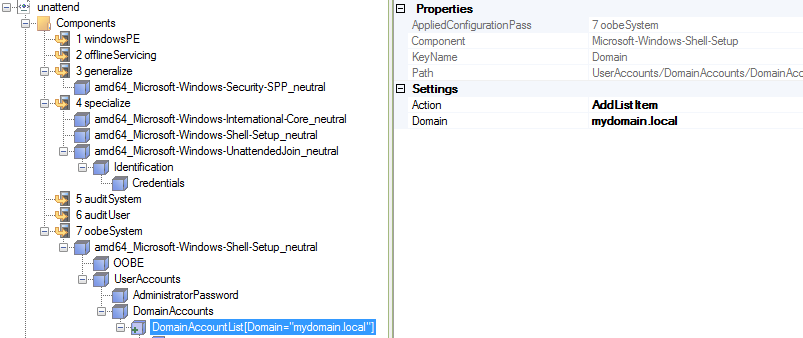


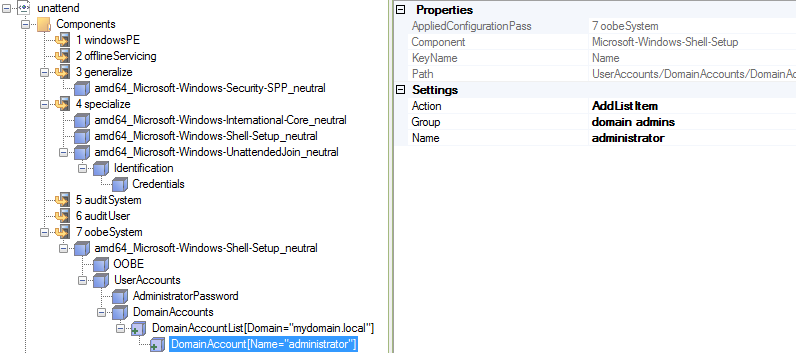












Appendix A includes an example unattend.xml file as well. If you don’t want to run through Windows System Image Manager you could use this file as an example, modify the relevant parts and copy it into your deployment.

Integrating powershell script for changing computer name (version 1)

*This version uses WMI commands only, no additional components to be installed on VM required. The downside is that this script won’t work if the computer account of the VM already exists in AD.*

1.) Create directory c:\scripts in the VM.

2.) Create file c:\scripts\change-computername.ps1

3.) Edit change-computername.ps1 and insert below code

$ErrorActionPreference = "SilentlyContinue"

$DHCPServers = Get-WmiObject Win32\_NetworkAdapterConfiguration -Filter "DHCPEnabled=TRUE AND DHCPServer IS NOT NULL" -Property DHCPServer

ForEach ($DHCPServer in $DHCPServers.DHCPServer) {

$URL = "http://$DHCPServer/latest/meta-data/local-hostname"

$metadatarequest=[System.Net.WebRequest]::Create($URL)

$resp=$metadatarequest.GetResponse()

$reqstream=$resp.GetResponseStream()

$sr=new-object System.IO.StreamReader $reqstream

$newhostname=$sr.ReadToEnd()

Write-Host $newhostname

Get-WmiObject Win32\_ComputerSystem

If ($newhostname.Length -gt 0) {$(Get-WmiObject Win32\_ComputerSystem).Rename($newhostname,'password','mydomain\computeraccountadmin')}

}

4.) Replace the red marked sections by custom values from your deployment  
‘mydomain\computeraccountadmin’ = user account that has permission to create and change computer accounts in active directory

‘password’ = Password of above mentioned account

5.) Save the file

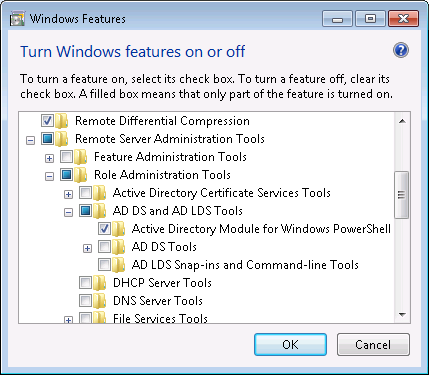
Integrating powershell script for changing computer name (version 2)

*This version uses AD powershell plugin in addition to WMI. It automatically overwrites the object in AD if the computername already exists.*

1.) Install Remote-Server-Administration-Tools (RSAT) on Windows 7.  
Download from: <http://www.microsoft.com/en-us/download/details.aspx?id=7887>

*Note: This requires Service Pack 1 to be installed on Windows 7. If you are running a Server OS these tools are part of the OS install and required components can be activated by adding features.*

2.) Activate “Active Directory Module for Windows PowerShell” (Windows 7)  
Control Panel 🡪 Programs 🡪 Turn Windows Features on and off



3.) Create directory c:\scripts in the VM.

4.) Create file c:\scripts\change-computername.ps1

5.) Edit change-computername.ps1 and insert below code

Import-Module ActiveDirectory

$username = 'mydomain\computeraccountadmin’

$password = 'password'

$cred = New-Object System.Management.Automation.PSCredential -ArgumentList @($username,(ConvertTo-SecureString -String $password -AsPlainText -Force))

$ErrorActionPreference = "SilentlyContinue"

$DHCPServers = Get-WmiObject Win32\_NetworkAdapterConfiguration -Filter "DHCPEnabled=TRUE AND DHCPServer IS NOT NULL" -Property DHCPServer

ForEach ($DHCPServer in $DHCPServers.DHCPServer) {

$URL = "http://$DHCPServer/latest/meta-data/local-hostname"

$metadatarequest=[System.Net.WebRequest]::Create($URL)

$resp=$metadatarequest.GetResponse()

$reqstream=$resp.GetResponseStream()

$sr=new-object System.IO.StreamReader $reqstream

$newhostname=$sr.ReadToEnd()

Write-Host $newhostname

$compaccount = Get-ADcomputer $newhostname -Credential $cred

Write-Host $compaccount

if ($compaccount)

{

Write-Host "Computeraccount exists"

Remove-ADComputer -identity $newhostname -Credential $cred -confirm:$false

}

else

{

Write-Host "Computeraccount does not exist"

}

Get-WmiObject Win32\_ComputerSystem

If ($newhostname.Length -gt 0) {$(Get-WmiObject Win32\_ComputerSystem).Rename($newhostname,$password,$username)}

}

6.) Replace the red marked sections by custom values from your deployment  
‘mydomain\computeraccountadmin’ = user account that has permission to create and change computer accounts in active directory

‘password’ = Password of above mentioned account

7.) Save the file

Configuring powershell script for autorun

1.) Create the file SetupComplete.cmd under c:\Windows\Setup\scripts   
*(you might need to create the sub dir “scripts” as it likely is not present)*

2.) Edit SetupComplete.cmd and insert below code

powershell Set-ExecutionPolicy RemoteSigned

powershell c:\scripts\change-computername.ps1

shutdown /r

3.) Save the file

Running Sysprep

When having done all preparation steps you are ready to run sysprep as explained in section 9, page 100 of the admin guide (the point we stopped above).

c:\windows\system32\sysprep\sysprep.exe /oobe/generalize /shutdown

Now you can create a template from the VM. After deployment the new instance will use the computer name in Windows which was defined in CloudPlatform instance wizard. In addition the Windows VM will be part of the domain specified in the unattend.xml file.

Appendix A

If you want to get up and running quickly with a default configuration, find a working unattend.xml file here (for Windows 7 64bit). The local Administrator password is “Pa$$w0rd”. Simply copy the code into an unattend.xml file and store the file under c:\windows\system32\sysprep\. Make sure you adapt at least the red marked sections to match your deployment.

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <unattend xmlns="urn:schemas-microsoft-com:unattend">  <settings pass="specialize">  <component name="Microsoft-Windows-International-Core" processorArchitecture="amd64" publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS" xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">  <InputLocale>de-DE</InputLocale>  <SystemLocale>de-DE</SystemLocale>  <UILanguage>de-DE</UILanguage>  <UserLocale>de-DE</UserLocale>  </component>  <component name="Microsoft-Windows-Shell-Setup" processorArchitecture="amd64" publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS" xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">  <ComputerName>\*</ComputerName>  <TimeZone>GMT+1</TimeZone>  </component>  <component name="Microsoft-Windows-UnattendedJoin" processorArchitecture="amd64" publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS" xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">  <Identification>  <Credentials>  <Domain>mydomain</Domain>  <Password>password</Password>  <Username>comuteraccountadmin</Username>  </Credentials>  <JoinDomain>mydomain.local</JoinDomain>  </Identification>  </component>  </settings>  <settings pass="oobeSystem">  <component name="Microsoft-Windows-Shell-Setup" processorArchitecture="amd64" publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS" xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">  <OOBE>  <HideEULAPage>true</HideEULAPage>  <NetworkLocation>Work</NetworkLocation>  <ProtectYourPC>1</ProtectYourPC>  <SkipUserOOBE>true</SkipUserOOBE>  <SkipMachineOOBE>true</SkipMachineOOBE>  </OOBE>  <UserAccounts>  <AdministratorPassword>  <Value>UABhACQAJAB3ADAAcgBkAEEAZABtAGkAbgBpAHMAdAByAGEAdABvAHIAUABhAHMAcwB3AG8AcgBkAA==</Value>  <PlainText>false</PlainText>  </AdministratorPassword>  <DomainAccounts>  <DomainAccountList wcm:action="add">  <Domain>space.local</Domain>  <DomainAccount wcm:action="add">  <Group>domain admins</Group>  <Name>administrator</Name>  </DomainAccount>  </DomainAccountList>  </DomainAccounts>  </UserAccounts>  <TimeZone>GMT+1</TimeZone>  </component>  </settings>  <settings pass="generalize">  <component name="Microsoft-Windows-Security-SPP" processorArchitecture="amd64" publicKeyToken="31bf3856ad364e35" language="neutral" versionScope="nonSxS" xmlns:wcm="http://schemas.microsoft.com/WMIConfig/2002/State" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">  <SkipRearm>1</SkipRearm>  </component>  </settings>  <cpi:offlineImage cpi:source="wim:c:/unattend/install.wim#Windows 7 ENTERPRISE" xmlns:cpi="urn:schemas-microsoft-com:cpi" />  </unattend> |

**About Citrix**

Citrix Systems, Inc. (NASDAQ:CTXS) is the leading provider of virtualization, networking and software as a service technologies for more than 230,000 organizations worldwide. Its Citrix Delivery Center, Citrix Cloud Center (C3) and Citrix Online Services product families radically simplify computing for millions of users, delivering applications as an on-demand service to any user, in any location on any device. Citrix customers include the world’s largest Internet companies, 99 percent of Fortune Global 500 enterprises, and hundreds of thousands of small businesses and prosumers worldwide. Citrix partners with over 10,000 companies worldwide in more than 100 countries. Founded in 1989, annual revenue in 2008 was $1.6 billion.

©2009 Citrix Systems, Inc. All rights reserved. Citrix®, Access Gateway™, Branch Repeater™, Citrix Repeater™, HDX™, XenServer™, XenApp™, XenDesktop™ and Citrix Delivery Center™ are trademarks of Citrix Systems, Inc. and/or one or more of its subsidiaries, and may be registered in the United States Patent and Trademark Office and in other countries. All other trademarks and registered trademarks are property of their respective owners.