**xComputerManagement Module – Windows PowerShell Desired State Configuration Resource Kit**

# **Introduction**

The **xComputerManagement** module is a part of the Windows PowerShell Desired State Configuration (DSC) Resource Kit, which is a collection of DSC Resources produced by the PowerShell Team. This module contains the **xComputer** resource. This DSC Resource allows you to rename a computer and add it to a domain or workgroup.

**All of the resources in the DSC Resource Kit are provided AS IS, and are not supported through any Microsoft standard support program or service. The “x” in xComputerManagement stands for experimental**, which means that these resources will be **fix forward** and monitored by the module owner(s).

Please leave comments, feature requests, and bug reports in the Q & A tab for this module.

If you would like to modify **xComputerManagement,** feel free. When modifying, please update the module name, resource friendly name, and MOF class name (instructions below). As specified in the license, you may copy or modify this resource as long as they are used on the Windows Platform.

For more information about Windows PowerShell Desired State Configuration, check out the blog posts on the [PowerShell Blog](http://blogs.msdn.com/b/powershell/) ([this](http://blogs.msdn.com/b/powershell/archive/2013/11/01/configuration-in-a-devops-world-windows-powershell-desired-state-configuration.aspx) is a good starting point). There are also great community resources, such as [PowerShell.org](http://powershell.org/wp/tag/dsc/), or [PowerShell Magazine](http://www.powershellmagazine.com/tag/dsc/). For more information on the DSC Resource Kit, check out [this blog post](http://go.microsoft.com/fwlink/?LinkID=389546).

# Installation

To install **xComputerManagement** module

* Unzip the content under $env:ProgramFiles\WindowsPowerShell\Modules folder

**To confirm installation:**

* **Run Get-DSCResource to see that xComputer is among the DSC Resources listed**

# **Requirements**

This module requires the latest version of PowerShell (v4.0, which ships in Windows 8.1 or Windows Server 2012R2). To easily use PowerShell 4.0 on older operating systems, [install WMF 4.0](http://www.microsoft.com/en-us/download/details.aspx?id=40855). Please read the installation instructions that are present on both the download page and the release notes for WMF 4.0.

# **Description**

The **xComputerManagement** module contains the **xComputer** DSC Resource. This DSC Resource allows you to configure a computer by changing its name and modifying its domain or workgroup.

# Details

**xComputer** resource has following properties:

* **Name**: The desired computer name
* **DomainName**: The name of the domain to join
* **WorkGroupName**: The name of the workgroup
* **Credential**: Credential to be used to join or leave domain

# **Example: Change the Name and the Workgroup Name**

This configuration will set a machine name and changes the WorkGroup it is in.

configuration Sample\_xComputer\_ChangeNameAndWorkGroup

{

param

(

[string[]]$NodeName="localhost",

[Parameter(Mandatory)]

[string]$MachineName,

[Parameter(Mandatory)]

[string]$WorkGroupName

)

#Import the required DSC Resources

Import-DscResource -Module xComputerManagement

Node $NodeName

{

xComputer NewNameAndWorkgroup

{

Name = $MachineName

WorkGroupName = $WorkGroupName

}

}

}

# **Example: Switch from a Workgroup to a Domain**

This configuration sets the machine name and joins a Domain.

*Note: this requires a credential.*

configuration Sample\_xComputer\_WorkgroupToDomain

{

param

(

[string[]]$NodeName="localhost",

[Parameter(Mandatory)]

[string]$MachineName,

[Parameter(Mandatory)]

[string]$Domain,

[Parameter(Mandatory)]

[pscredential]$Credential

)

#Import the required DSC Resources

Import-DscResource -Module xComputerManagement

Node $NodeName

{

xComputer JoinDomain

{

Name = $MachineName

DomainName = $Domain

Credential = $Credential # Credential to join to domain

}

}

}

<#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

To save the credential in plain-text in the mof file, use the following configuration data

$ConfigData = @{

AllNodes = @(

@{

NodeName = "localhost";

# Allows credential to be saved in plain-text in the the \*.mof instance document.

PSDscAllowPlainTextPassword = $true;

};

);

}

Sample\_xComputer\_WorkgroupToDomain -ConfigurationData $ConfigData -MachineName <machineName> -credential (Get-Credential) -Domain <domainName>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*#>

# Example: Change the Name while staying on the Domain

This example will change the machines name while remaining on the domain.

*Note: this requires a credential.*

configuration Sample\_xComputer\_ChangeNameInDomain

{

param

(

[string[]]$NodeName="localhost",

[Parameter(Mandatory)]

[string]$MachineName,

[Parameter(Mandatory)]

[pscredential]$Credential

)

#Import the required DSC Resources

Import-DscResource -Module xComputerManagement

Node $NodeName

{

xComputer NewName

{

Name = $MachineName

Credential = $Credential # Domain credential

}

}

}

<#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

To save the credential in plain-text in the mof file, use the following configuration data

$ConfigData = @{

AllNodes = @(

@{

NodeName = "localhost";

# Allows credential to be saved in plain-text in the the \*.mof instance document.

PSDscAllowPlainTextPassword = $true;

};

);

}

Sample\_xComputer\_ChangeNameInDomain -ConfigurationData $ConfigData -MachineName <machineName> -Credential (Get-Credential)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*#>

# Example: Change the Name while staying on the Workgroup

This example will change the machines name while remaining on the workgroup.

configuration Sample\_xComputer\_ChangeNameInWorkgroup

{

param

(

[string[]]$NodeName="localhost",

[Parameter(Mandatory)]

[string]$MachineName

)

#Import the required DSC Resources

Import-DscResource -Module xComputerManagement

Node $NodeName

{

xComputer NewName

{

Name = $MachineName

}

}

}

# Example: Switch from a Domain to a Workgroup

This example switches the computer from a Domain to a Workgroup.

*Note: this requires a credential*.

configuration Sample\_xComputer\_DomainToWorkgroup

{

param

(

[string[]]$NodeName="localhost",

[Parameter(Mandatory)]

[string]$MachineName,

[Parameter(Mandatory)]

[string]$WorkGroup,

[Parameter(Mandatory)]

[pscredential]$Credential

)

#Import the required DSC Resources

Import-DscResource -Module xComputerManagement

Node $NodeName

{

xComputer JoinWorkgroup

{

Name = $MachineName

WorkGroupName = $WorkGroup

Credential = $Credential # Credential to unjoin from domain

}

}

}

<#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

To save the credential in plain-text in the mof file, use the following configuration data

$ConfigData = @{

AllNodes = @(

@{

NodeName = "localhost";

# Allows credential to be saved in plain-text in the the \*.mof instance document.

PSDscAllowPlainTextPassword = $true;

};

);

}

Sample\_xComputer\_DomainToWorkgroup -ConfigurationData $ConfigData -MachineName <machineName> -credential (Get-Credential) -WorkGroup <workgroupName>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*#>

# Renaming Requirements

1. Update the following names by replacing MSFT with your company/community name and replace the “x” with your own prefix (e.g. the resource name should change from MSFT\_xComputer to Contoso\_myComputer):

* **Module name**
* **Resource Name**
* **Resource Friendly Name**
* **MOF class name**
* **Filename for the <resource>.schema.mof**

1. Update module and metadata information in the module manifest
2. Update any configuration that use these resources

# Versions

1.0.0.0

* Initial release with the following resources
  + xComputer