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

# **Introduction**

The **xNetworking** module is a part of Windows PowerShell Desired State Configuration (DSC) Resource Kit, which is a collection of DSC Resources produced by the PowerShell Team. This module contains the **xIPAddress** and **xDnsServerAddress** resources. These DSC Resources allow configuration of a node’s IP Address and DNS Server Address.

**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 xNetworking 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 **xNetworking** module, 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 **xNetworking** module

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

**To confirm installation:**

* **Run Get-DSCResource to see that xIPAddress and xDnsServerAddress are 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). It also requires **<foo>** features. 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 **xNetworking** module contains two DSC Resources: **xIPAddress** and **xDnsServerAddress**. Instead of needing to know and remember the functionality and syntax for the IPAddress and DNS cmdlets, these DSC Resources allow you to easily configure and maintain your networking settings by writing simple configurations.

# Details

**xIPAddress** resource has following properties:

* **IPAddress:** The desired IP Address
* **InterfaceAlias:** Alias of the network interface for which IP Address is set
* **DefaultGateway:** Specifies the IP address of the default gateway for the host
* **SubnetMask:** Local subnet size using IP address format
* **AddressFamily:** IP address family - IPv4 or IPv6

**xDnsServerAddress** resource has following properties:

* **Address:** The desired DNS Server addresses
* **InterfaceAlias:** Alias of the network interface for which DNS Server Address is set
* **AddressFamily:** IP address family - IPv4 or IPv6

# **Example: Set IP Address on Ethernet NIC**

This configuration will set IP Address with some typical values for network interface alias = Ethernet

configuration Sample\_xIPAddress\_FixedValue

{

param

(

[string[]]$NodeName = 'localhost'

)

Import-DscResource -Module xNetworking

Node $NodeName

{

xIPAddress NewIPAddress

{

IPAddress = "2001:4898:200:7:6c71:a102:ebd8:f482"

InterfaceAlias = "Ethernet"

SubnetMask = 24

AddressFamily = "IPV6"

}

}

}

# **Example: Set IP Address with parameterized values**

This configuration will set IP Address along with default gateway on a network interface that is identified by its alias

configuration Sample\_xIPAddress\_Parameterized

{

param

(

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

[Parameter(Mandatory)]

[string]$IPAddress,

[Parameter(Mandatory)]

[string]$InterfaceAlias,

[Parameter(Mandatory)]

[string]$DefaultGateway,

[int]$SubnetMask = 16,

[ValidateSet("IPv4","IPv6")]

[string]$AddressFamily = 'IPv4'

)

Import-DscResource -Module xNetworking

Node $NodeName

{

xIPAddress NewIPAddress

{

IPAddress = $IPAddress

InterfaceAlias = $InterfaceAlias

DefaultGateway = $DefaultGateway

SubnetMask = $SubnetMask

AddressFamily = $AddressFamily

}

}

}

# **Example: Set DNS Server Address**

This configuration will set DNS Server Address on a network interface that is identified by its alias

configuration Sample\_xDnsServerAddress

{

param

(

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

[Parameter(Mandatory)]

[string]$DnsServerAddress,

[Parameter(Mandatory)]

[string]$InterfaceAlias,

[ValidateSet("IPv4","IPv6")]

[string]$AddressFamily = 'IPv4'

)

Import-DscResource -Module xNetworking

Node $NodeName

{

xDnsServerAddress DnsServerAddress

{

Address = $DnsServerAddress

InterfaceAlias = $InterfaceAlias

AddressFamily = $AddressFamily

}

}

}

# 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
  + xIPAddress
  + xDnsServerAddress