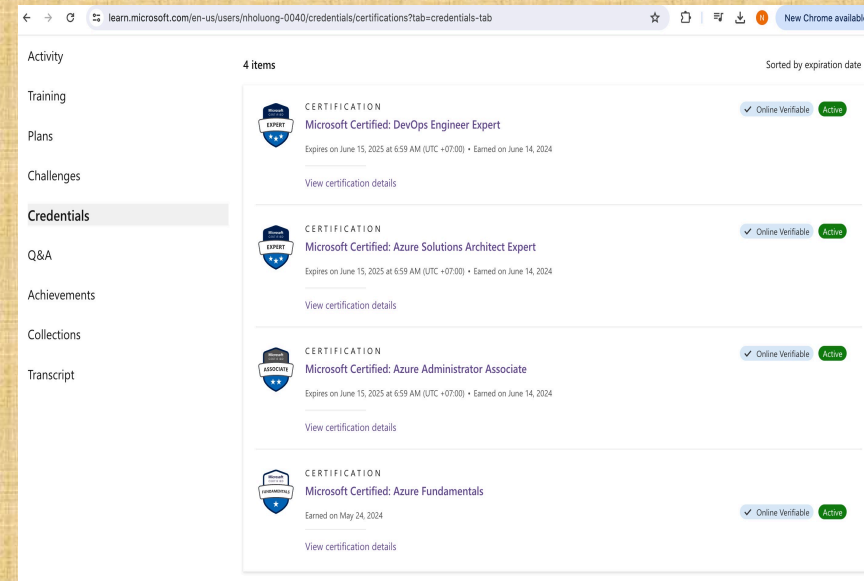
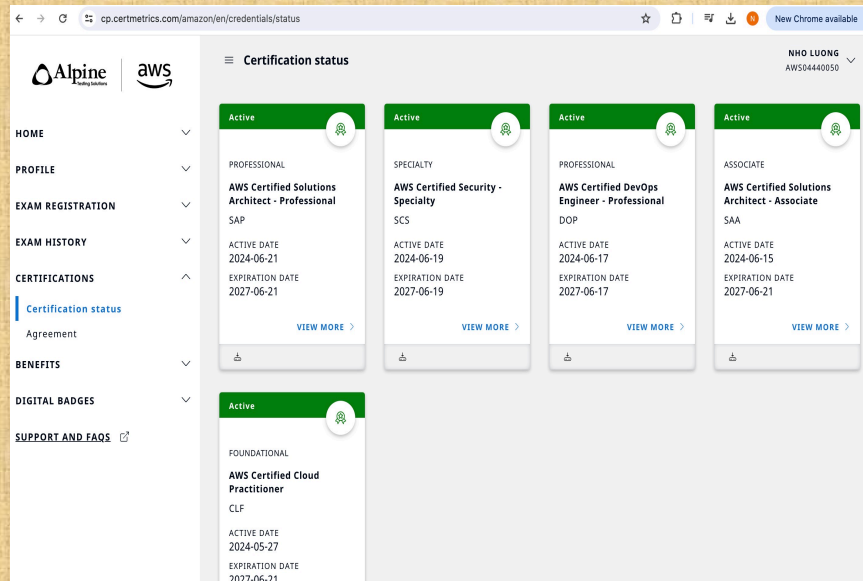


# What is Desired State Configuration how does it help me?

Author: Nho Luong

Skill: DevOps Engineer Lead



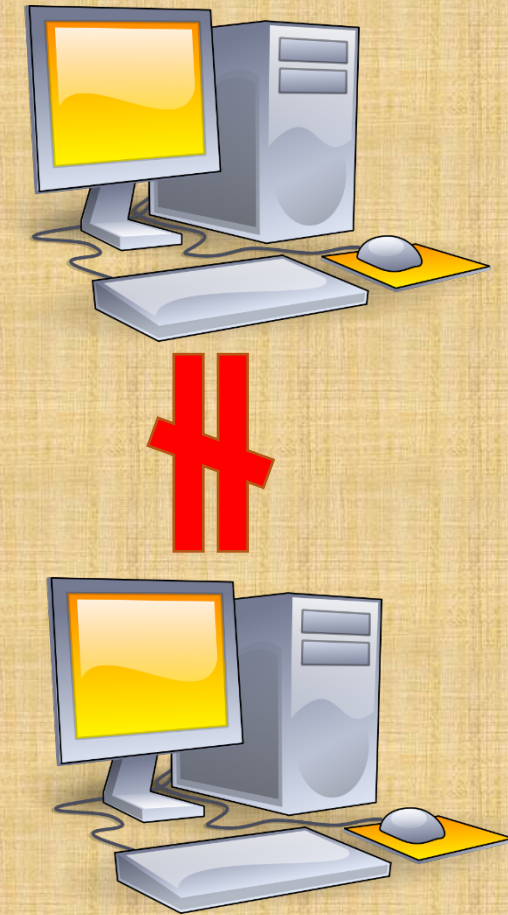
# We all have an install process

## Setup process

1. Find the installation media
2. Install the operating system with standard settings
3. Run the installer
4. Do next step.....



Follow the process



# Robustness as Consistency

*"Taguchi argues that quality robustness stems from consistency. Even if it means a product falls outside standard tolerances, as long as it does so consistently, then it is much easier to adjust machinery to attain precise targets."*

# So what is Desired State Configuration?

- Microsoft's implementation of the "Management Object Format," as developed by the Distributed Management Task Force (DMTF).
- Provides configuration as code
- Part of PowerShell 4.0
- Aims for 'Star Trek' deployment

# Operating Systems Support

- By default
  - Windows 8.1
  - Windows Server 2012 R2
- Install Windows Management Framework 4.0
  - Windows® 7 with Service Pack 1
  - Windows Server® 2008 R2 with Service Pack 1



# Demo – DSC 101

# Sample Configuration

```
$MyData =
@{
    AllNodes =
    @(
        @{
            NodeName          = "*"
            LogPath            = "C:\Logs"
        },
        @{
            NodeName = "VM-1";
            Role      = "WebServer"
            SiteContents = "C:\Site1"
            SiteName = "Website1"
        },
        @{
            NodeName = "VM-2";
            Role      = "SQLServer"
        },
        @{
            NodeName = "VM-3";
            Role      = "WebServer";
            SiteContents = "C:\Site2"
            SiteName = "Website3"
        }
    );
    NonNodeData =
    @{
        ConfigFileContents = (Get-Content C:\Template\Config.xml)
    }
}
```

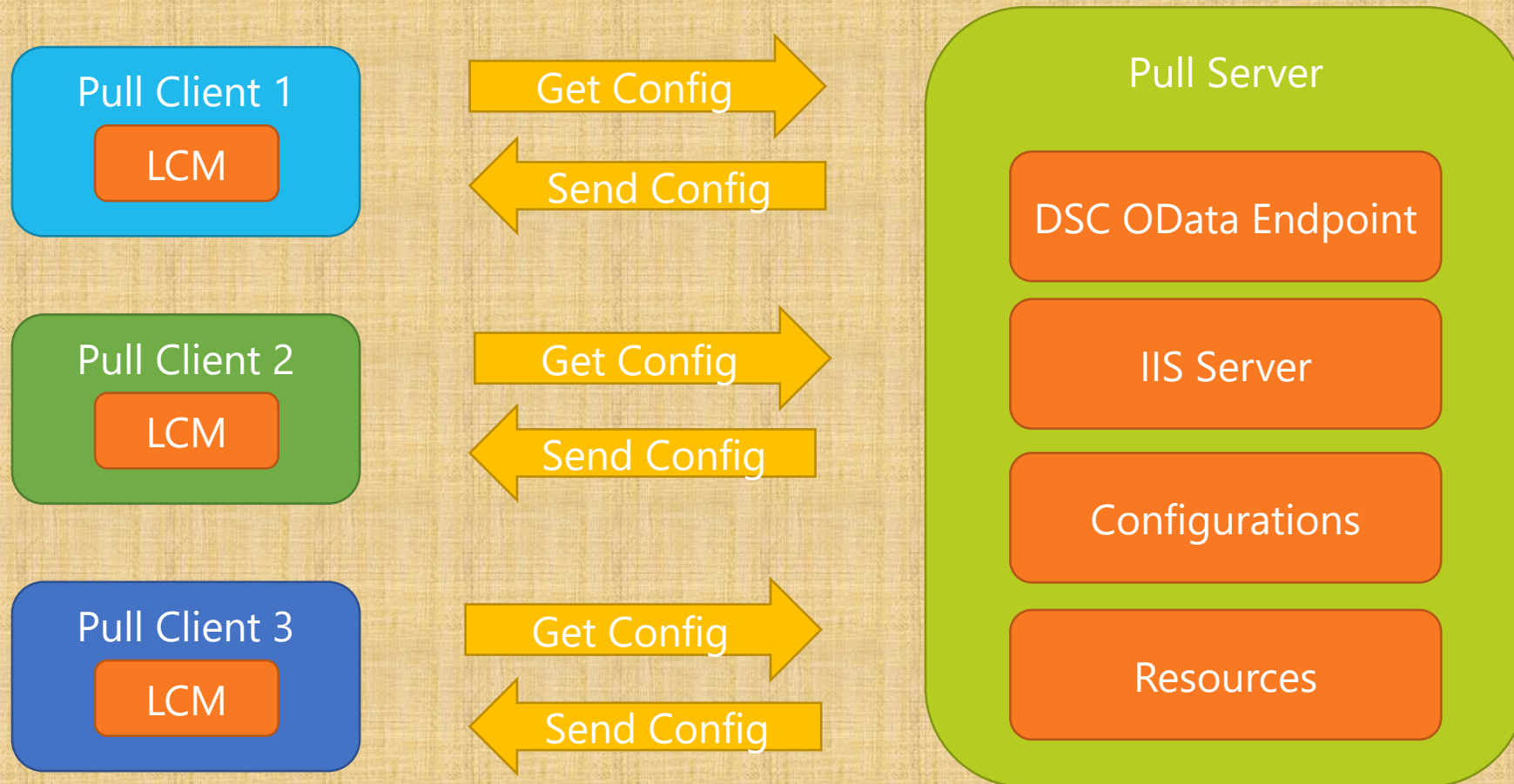
```
configuration MyConfiguration
{
    Import-DscResource -ModuleName xWebAdministration
                        -Name MSFT_xWebsite

    node $AllNodes.Where{$_Role -eq "WebServer"}.NodeName
    {
        xWebsite Site
        {
            Name          = $Node.SiteName
            PhysicalPath = $Node.SiteContents
            Ensure         = "Present"
        }

        File ConfigFile
        {
            DestinationPath = $Node.SiteContents + "\\config.xml"
            Contents =
                $ConfigurationData.NonNodeData.ConfigFileContents
        }
    }
}

MyConfiguration -ConfigurationData $MyData
```

# You don't have to push





# How do you create a DSC Pull Server?

You use DSC of course!

```
Configuration NewPullServer
{
    param
    (
        [string[]]$ComputerName = 'localhost'
    )

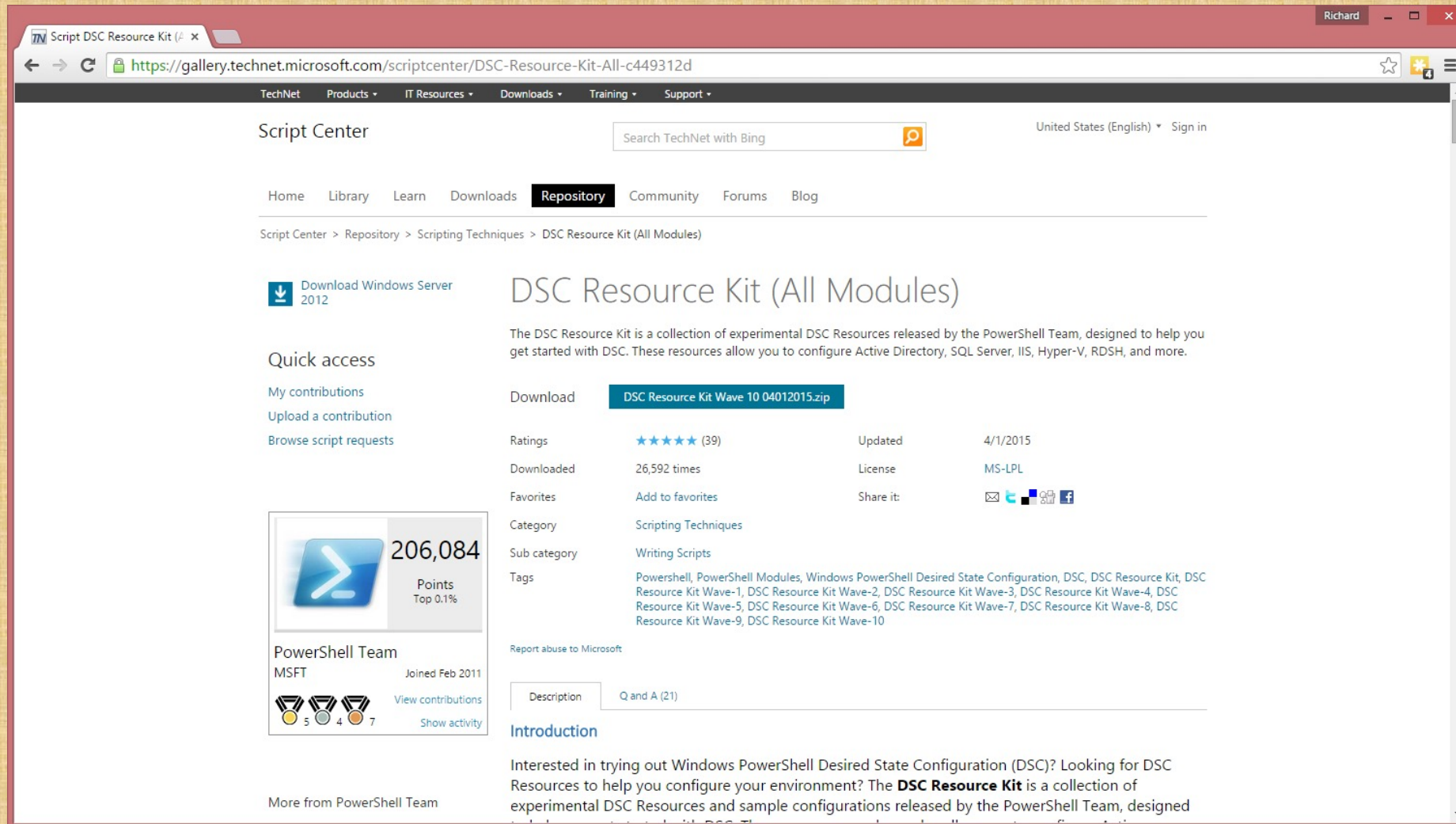
    Import-DSCResource -ModuleName xPSDesiredStateConfiguration

    Node $ComputerName
    {
        WindowsFeature DSCServiceFeature
        {
            Ensure = "Present"
            Name    = "DSC-Service"
        }

        xDscWebService PSDSCPullServer
        {
            Ensure = "Present"
        }
    }
}
```

. . .

# Where to find module resources - today



The screenshot shows the Microsoft Script Center page for the DSC Resource Kit (All Modules). The page is viewed in a browser window with the URL <https://gallery.technet.microsoft.com/scriptcenter/DSC-Resource-Kit-All-c449312d>. The page features a navigation bar with links to TechNet, Products, IT Resources, Downloads, Training, and Support. The main content area includes a search bar, a breadcrumb trail (Script Center > Repository > Scripting Techniques > DSC Resource Kit (All Modules)), and a download button for the DSC Resource Kit Wave 10 04012015.zip. The page also displays ratings (4.5 stars), download count (26,592 times), and a list of tags including Powershell, PowerShell Modules, and Windows PowerShell Desired State Configuration. A sidebar on the left shows the PowerShell Team's profile, including their points (206,084) and a list of contributions.

Script Center

Search TechNet with Bing

United States (English) Sign in

Home Library Learn Downloads **Repository** Community Forums Blog

Script Center > Repository > Scripting Techniques > DSC Resource Kit (All Modules)

Download Windows Server 2012

Quick access

My contributions

Upload a contribution

Browse script requests

206,084 Points Top 0.1%

PowerShell Team MSFT Joined Feb 2011

View contributions

Show activity





Download DSC Resource Kit Wave 10 04012015.zip

Ratings ★★★★★ (39)

Downloaded 26,592 times

Updated 4/1/2015

License MS-LPL

Share it:    

Category Scripting Techniques

Sub category Writing Scripts

Tags Powershell, PowerShell Modules, Windows PowerShell Desired State Configuration, DSC, DSC Resource Kit, DSC Resource Kit Wave-1, DSC Resource Kit Wave-2, DSC Resource Kit Wave-3, DSC Resource Kit Wave-4, DSC Resource Kit Wave-5, DSC Resource Kit Wave-6, DSC Resource Kit Wave-7, DSC Resource Kit Wave-8, DSC Resource Kit Wave-9, DSC Resource Kit Wave-10

Report abuse to Microsoft

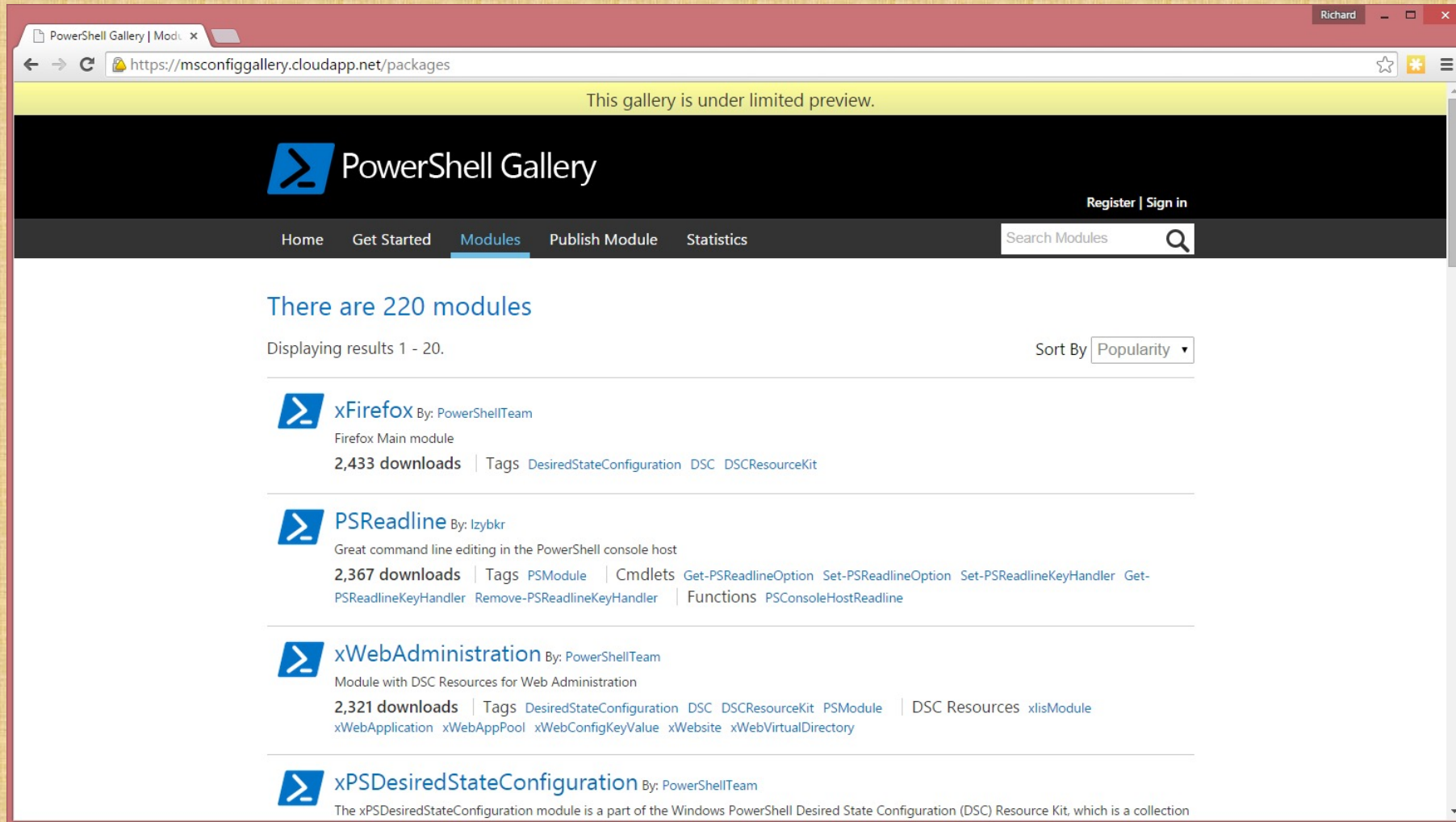
Description Q and A (21)

Introduction

Interested in trying out Windows PowerShell Desired State Configuration (DSC)? Looking for DSC Resources to help you configure your environment? The **DSC Resource Kit** is a collection of experimental DSC Resources and sample configurations released by the PowerShell Team, designed to help you get started with DSC. These resources allow you to configure Active Directory, SQL Server, IIS, Hyper-V, RDSH, and more.

More from PowerShell Team

# Where to find module resources - future



# Writing Modules & Resources

# Writing a DSC Resources

Get-TargetResource

Set-TargetResource

Test-TargetResource



# The Future



# Open Management Infrastructure

- Provides standards-based management for
  - Cloud management
  - Storage management
  - Server hardware management
  - Device management
  - Network management
- <https://collaboration.opengroup.org/omi/>

# DSC on Linux

- Uses OMI as opposed to WMI
  - So can use WS-Management protocol
  - Uses Python under the covers
  - Source is all on GitHub
- 
- It is not going to convert the Linux world
  - But can extend the reach of DSC for the Windows admin

# Is DSC a competitor to Chef, Puppet etc?

- No
- Puppet & Chef each provide a complete solution stack
- These tools work really well, but came from Unix world where configuration is 'easier'
- DSC only provides the platform-level implementation
- But Chef/Puppet can call DSC, they can play nice together, so it can extend their reach on Windows

# DSC at the core of Microsoft's DevOps

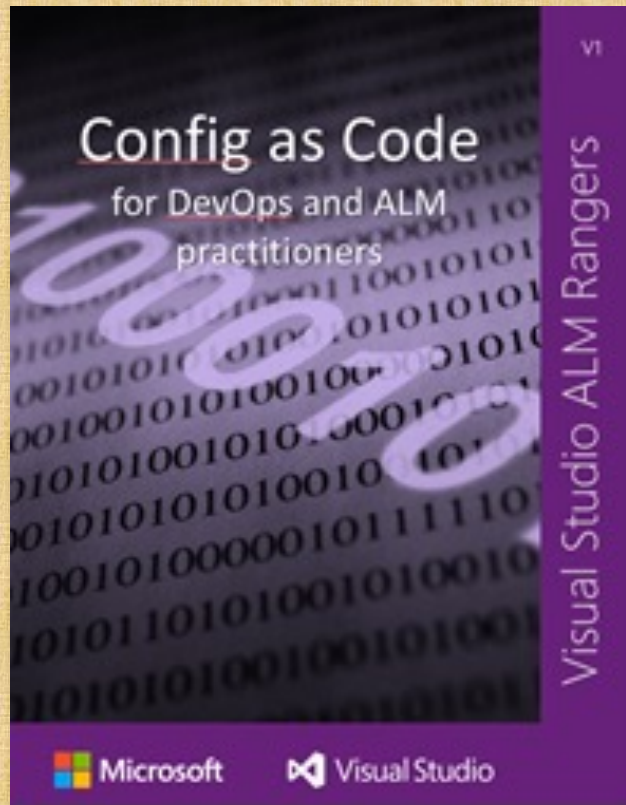
- DSC modules appearing for most Windows systems, including Azure
- But it is a new technology, can be 'flaky'
- A technology to keep an eye on...

# Resources

- TechEd 2014, A Practical Overview of Desired State Configuration
  - <http://channel9.msdn.com/Events/TechEd/NorthAmerica/2014/DCIM-B417>
- PowerShell DSC Forum
  - <http://powershell.org/wp/forums/forum/dsc-desired-state-configuration>
- Windows PowerShell Blog
  - <http://blogs.msdn.com/b/powershell/>
- DSC Resource Kit (Wave 6)
  - <http://gallery.technet.microsoft.com/DSC-Resource-Kit-All-c449312d>
- Community Resources
  - <https://github.com/powershellorg/dsc>
- ALM Rangers 'Configuration as Code'
  - <https://vsardevops.codeplex.com/>



# ALM Rangers







# Thank You