

# Lesson 6: Configuring Servers for Remote Management

MOAC 70-410: Installing and Configuring  
Windows Server 2012

# Overview

- Exam Objective 2.3: Configure Servers for Remote Management
- Using Server Manager for Remote Management
- Using Remote Server Administration Tools
- Using Windows PowerShell Web Access
- Working with Remote Servers

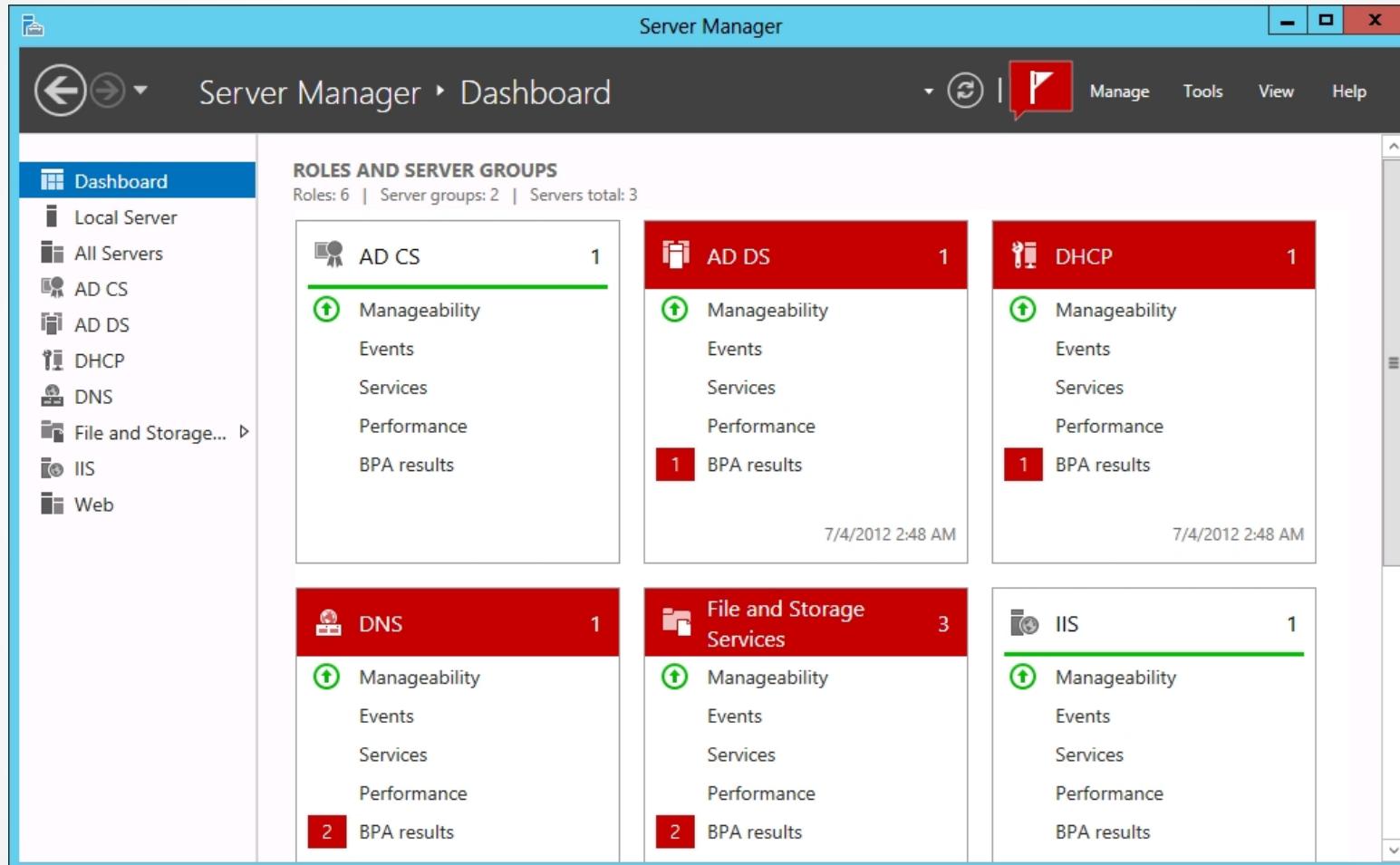
# Using Server Manager for Remote Management

Lesson 6: Configuring Servers for  
Remote Management

# Using Server Manager for Remote Management

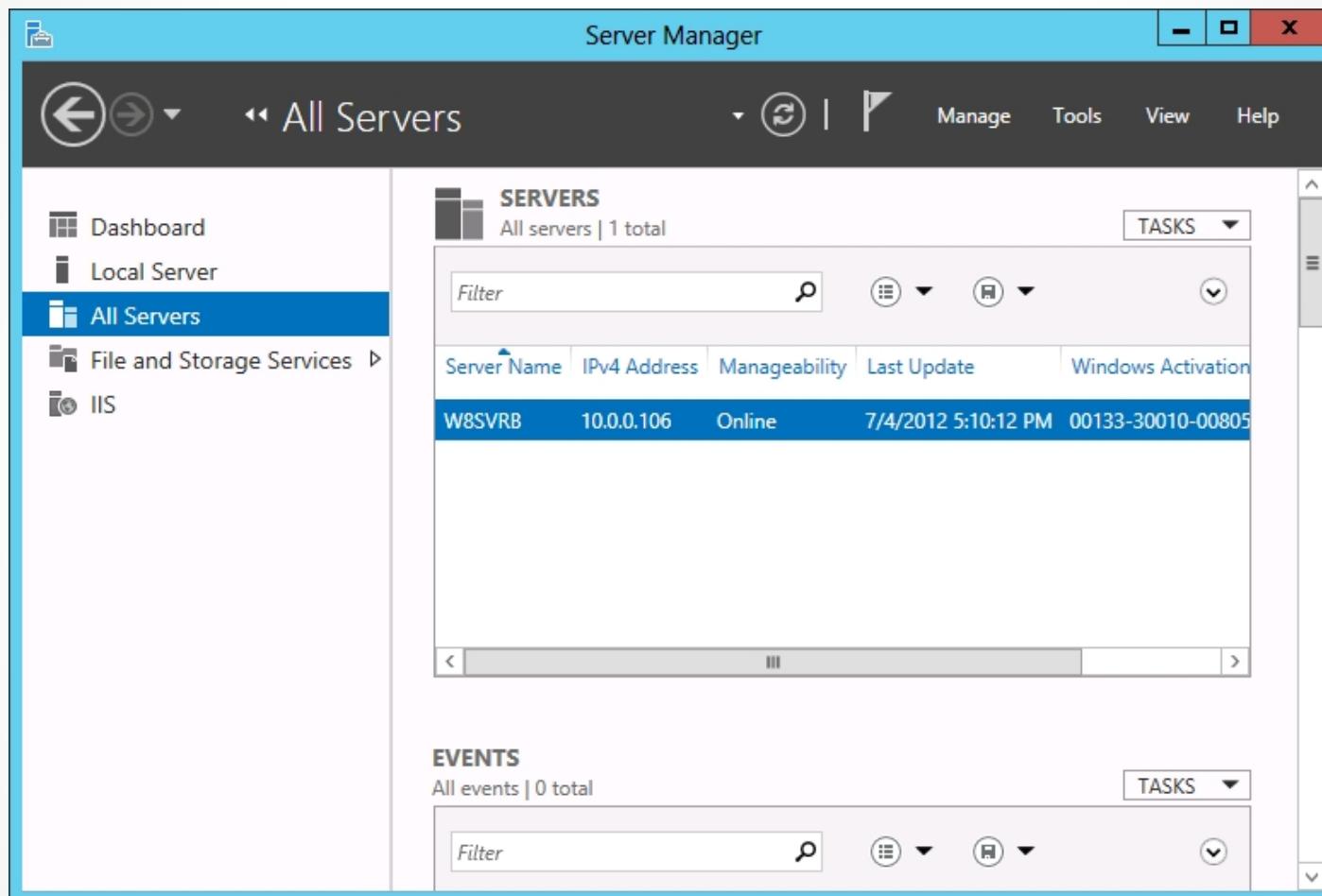
- In Windows Server 2012, Server Manager has been improved to include the ability to perform administrative tasks on remote servers as well as on the local system.
- Server Manager contains tiles that represent other views including a page for the Local Server and one for All Servers, and server groups and role groups.

# Using Server Manager for Remote Management



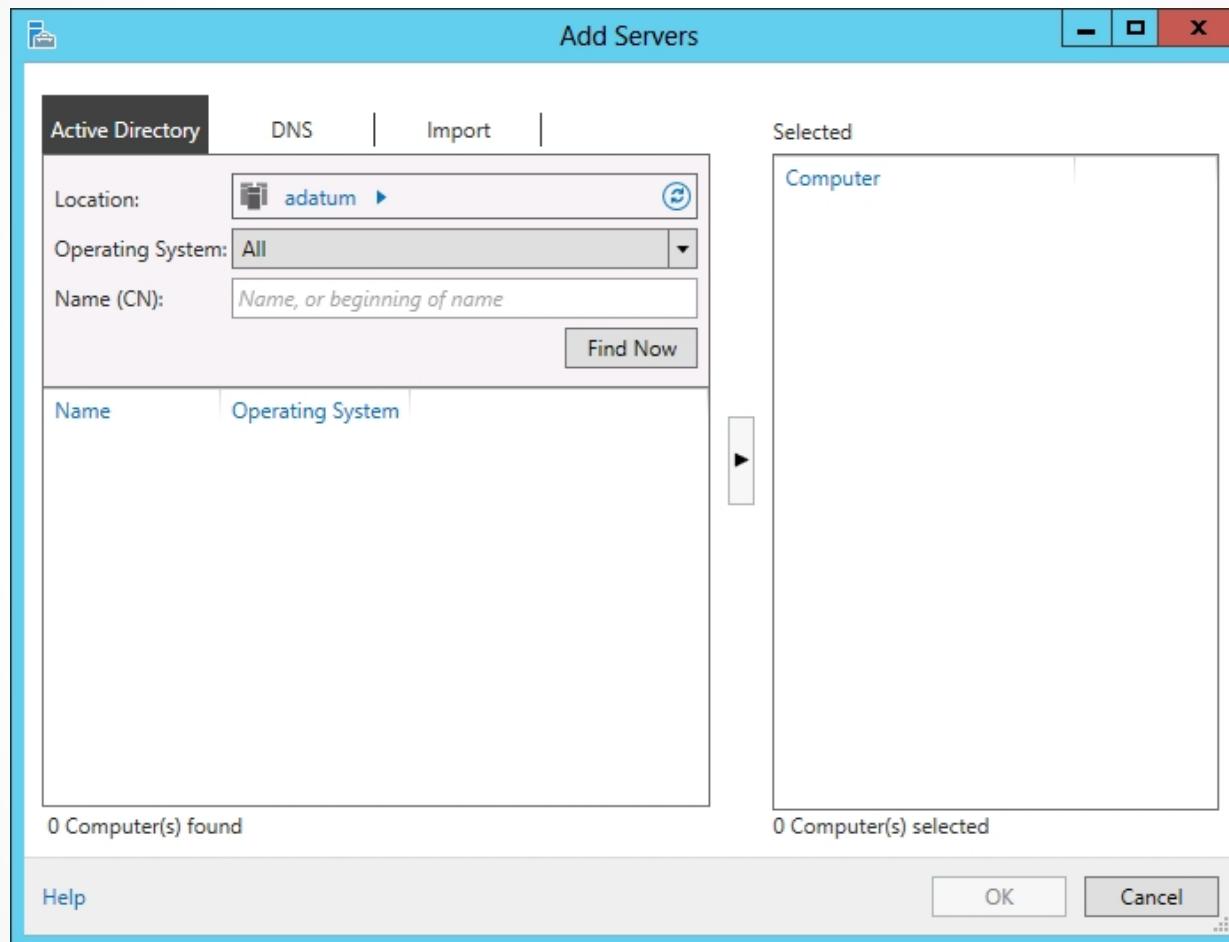
Dashboard thumbnails in Server Manager

# Adding Servers



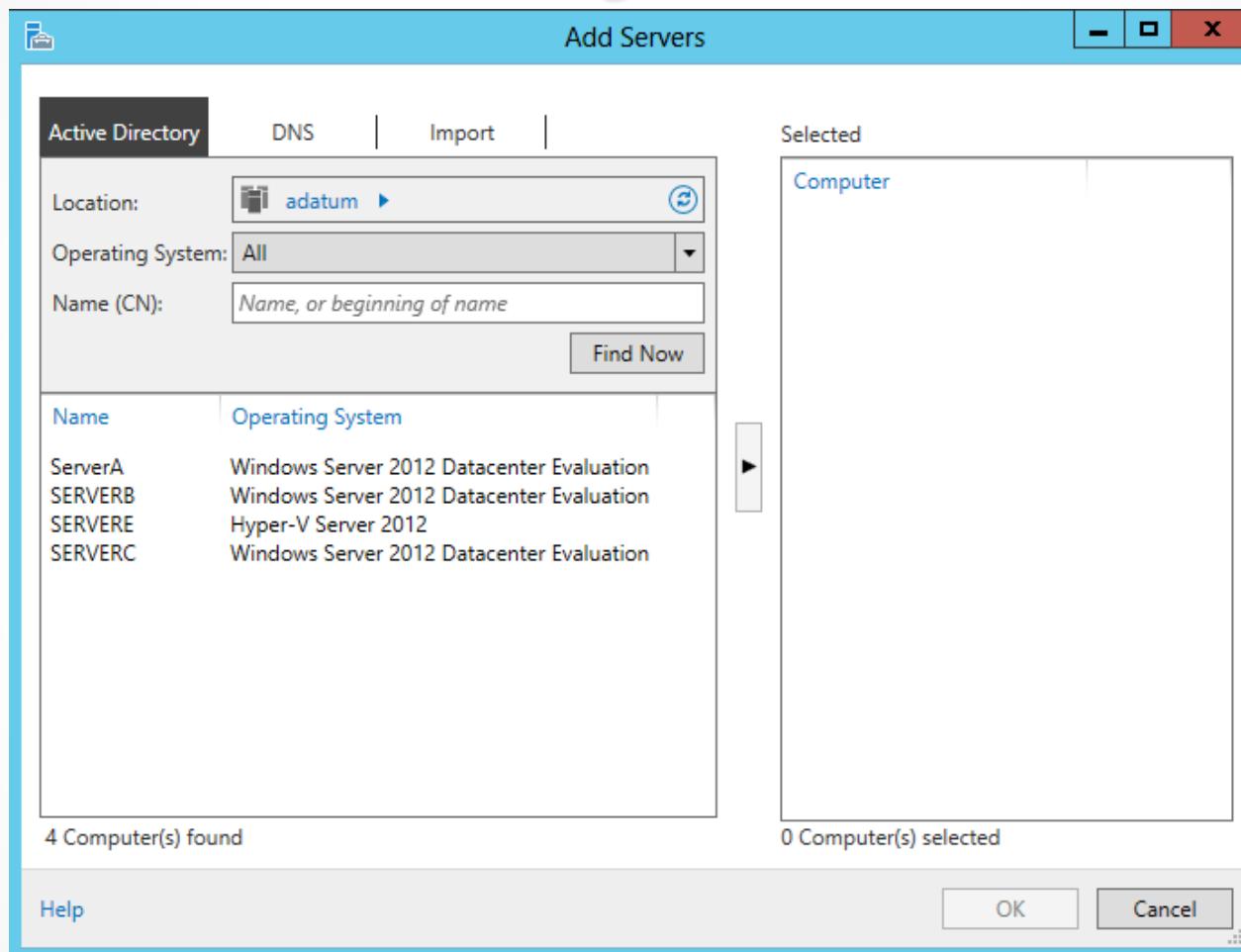
The All Servers homepage in Server Manager

# Adding Servers



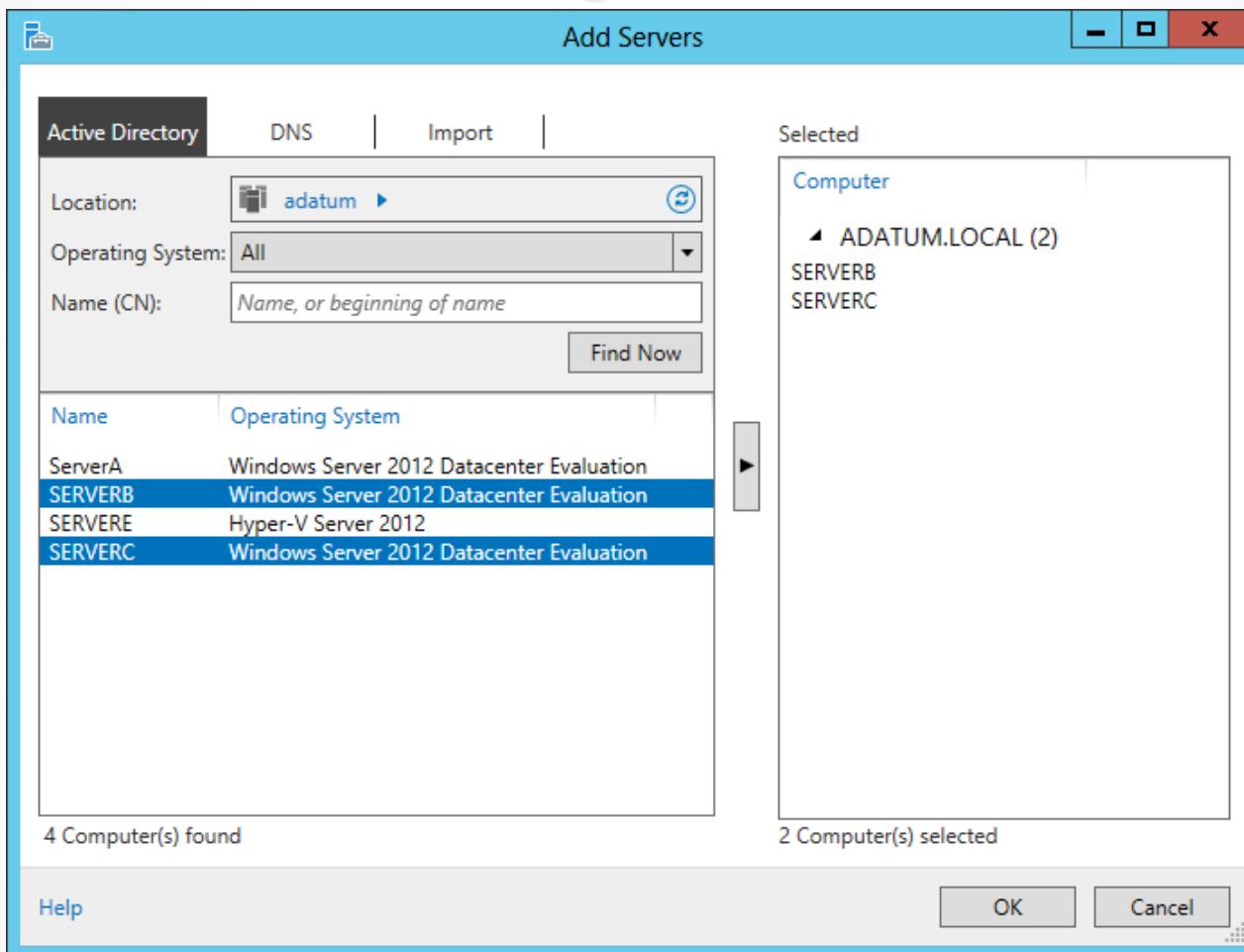
The Add Servers dialog box in Server Manager

# Adding Servers



Searching for servers in Server Manager

# Adding Servers



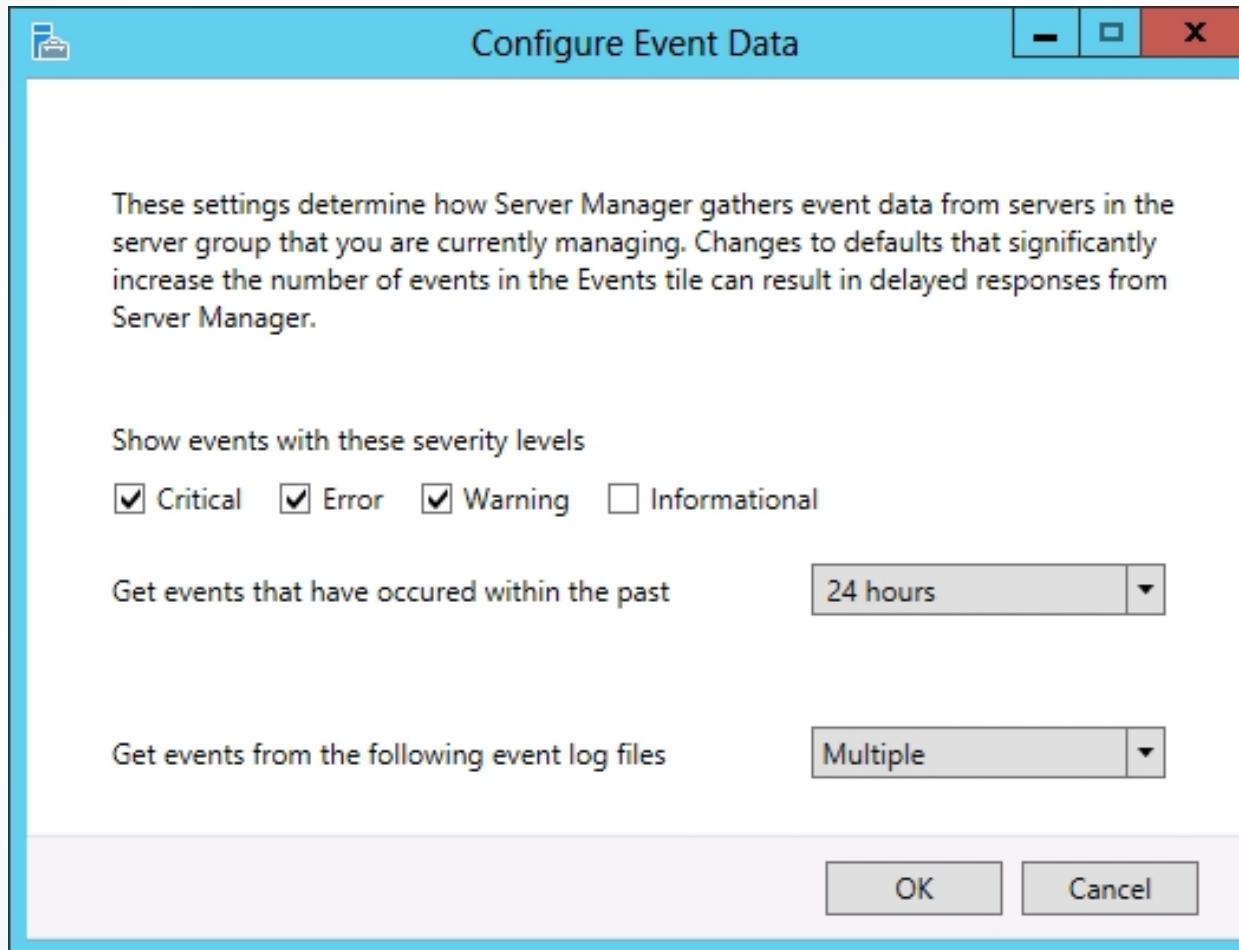
Selecting servers in Server Manager

# Adding Workgroup Servers

- To remotely manage a server that is part of a workgroup, you must add the name of the workgroup server to the TrustedHosts list on the computer running Server Manager.
- PowerShell command:

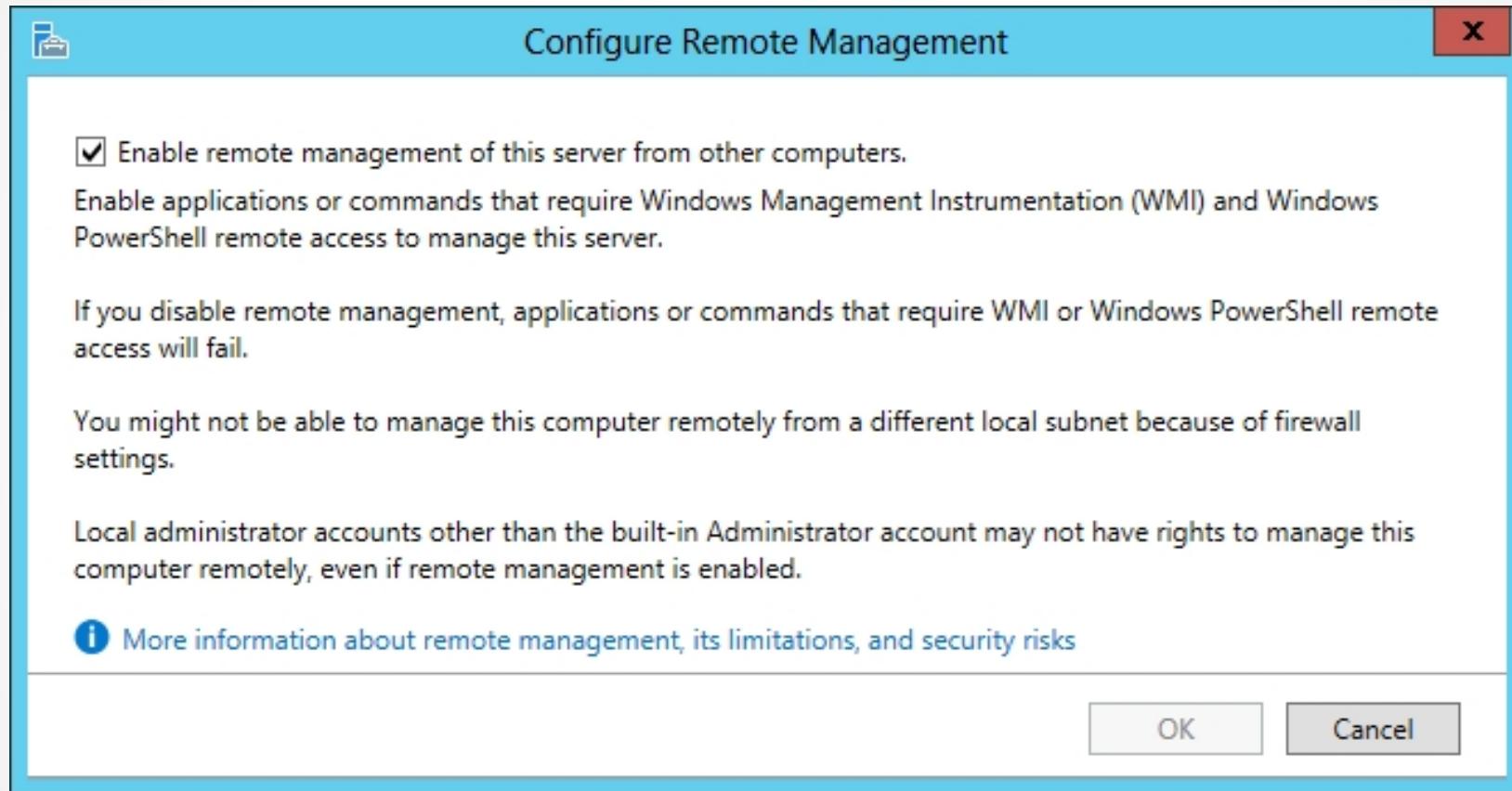
```
Set-Item wsman:\localhost\Client\TrustedHosts  
    <servername> -Concatenate -Force
```

# Calibrating Server Manager Performance



The Configure Event Data dialog box in Server Manager

# Configuring WinRM

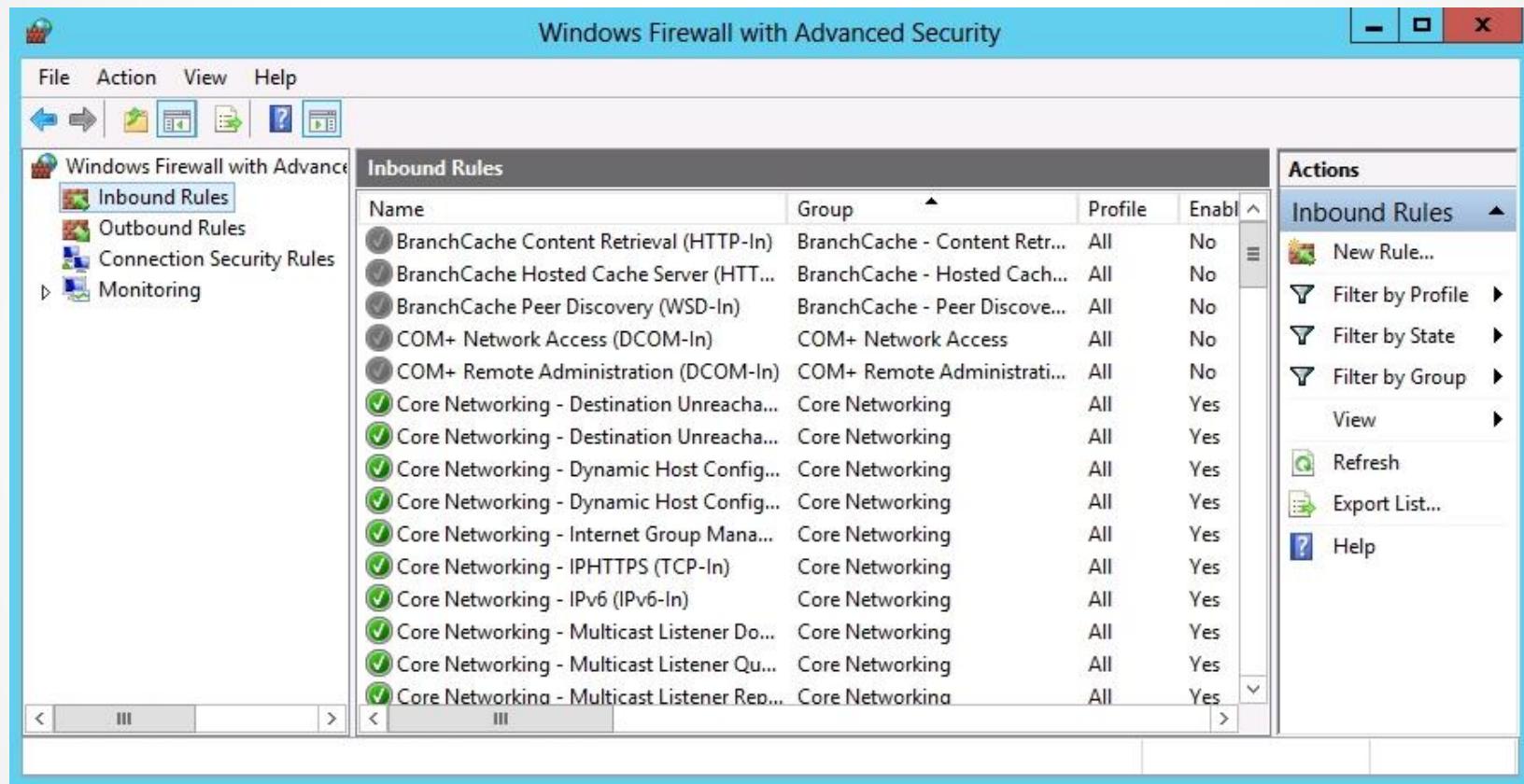


The Configure Remote Management dialog box

# Configuring Windows Firewall

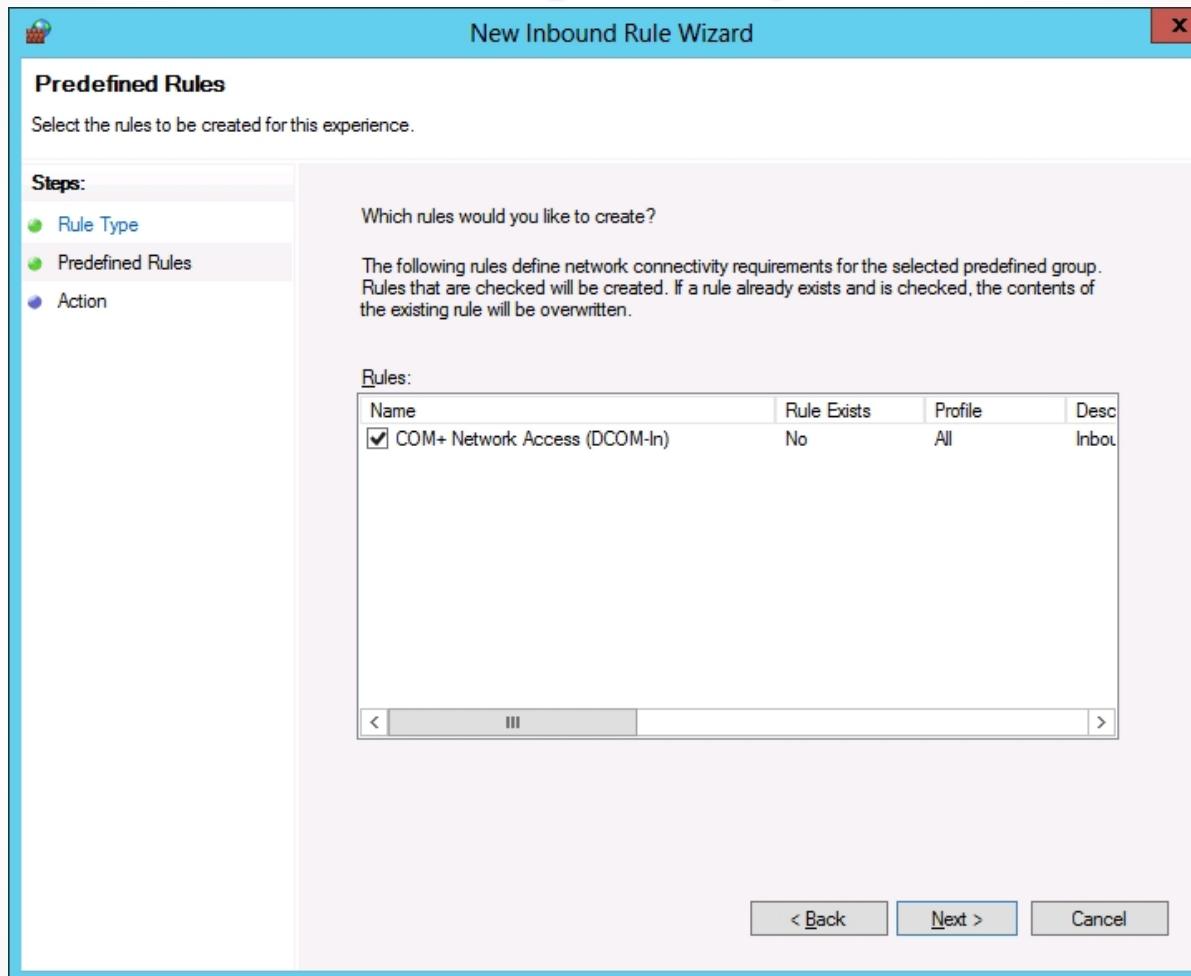
- If you use MMC snap-ins targeting a remote server, Windows Firewall default settings will block the communications.
- Inbound Firewall rules must be managed:
  - COM+ Network Access (DCOM-In)
  - Remote Event Log Management (NP-In)
  - Remote Event Log Management (RPC)
  - Remote Event Log Management (RPC-EPMAP)

# Configuring Windows Firewall



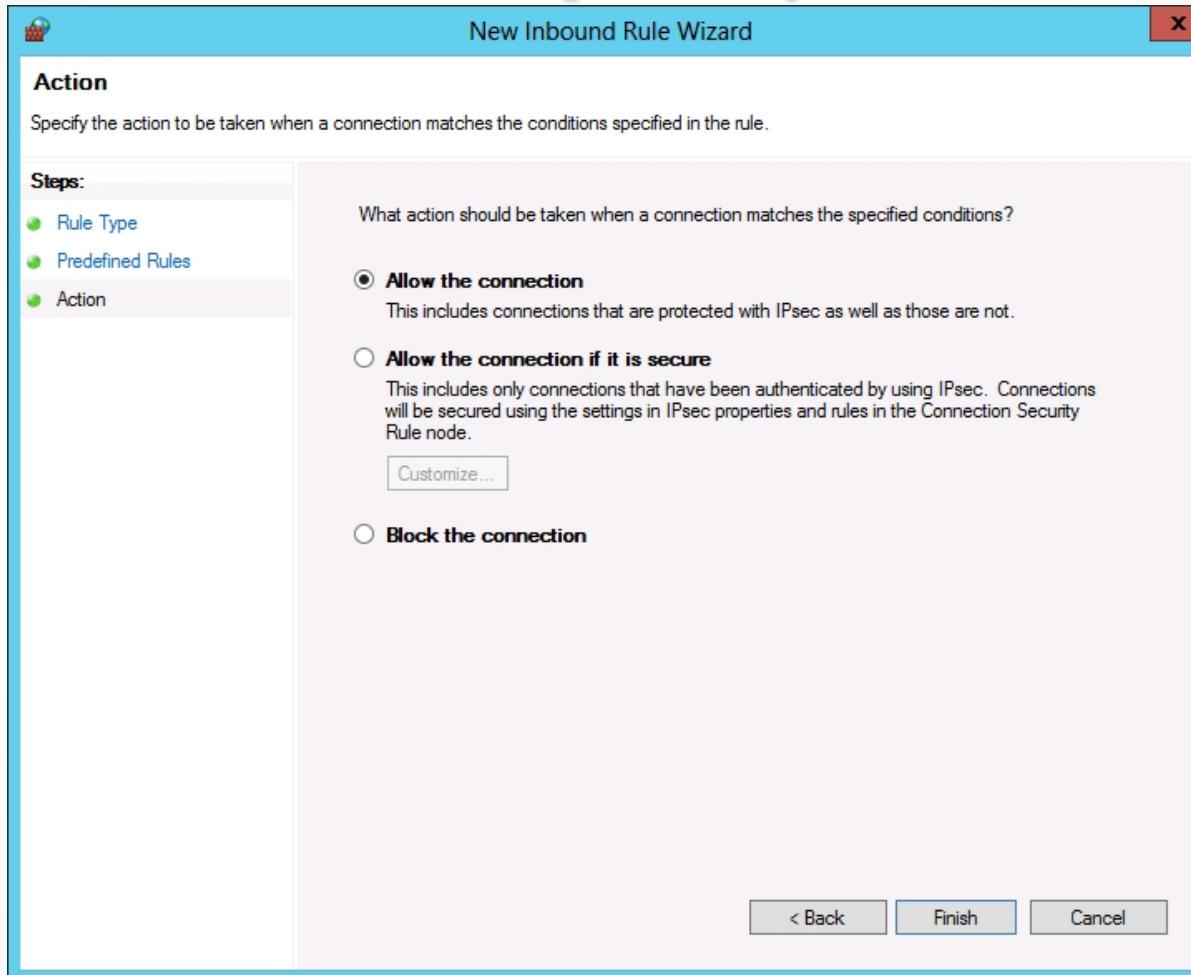
The Windows Firewall with Advanced Security snap-in

# Configure Windows Firewall with Group Policy



The Predefined Rules page of the New Inbound Rule Wizard

# Configure Windows Firewall with Group Policy



The Action page of the New Inbound Rule Wizard

# Managing Down-Level Servers

- Earlier versions of Windows Server lack the WinRM support needed for them to be remotely managed by Server Manager
- Windows Server 2008 and 2008 R2 must have the following updates downloaded and installed:
  - .NET Framework 4.0
  - Windows Management Framework 3.0

# Managing Down-Level Servers

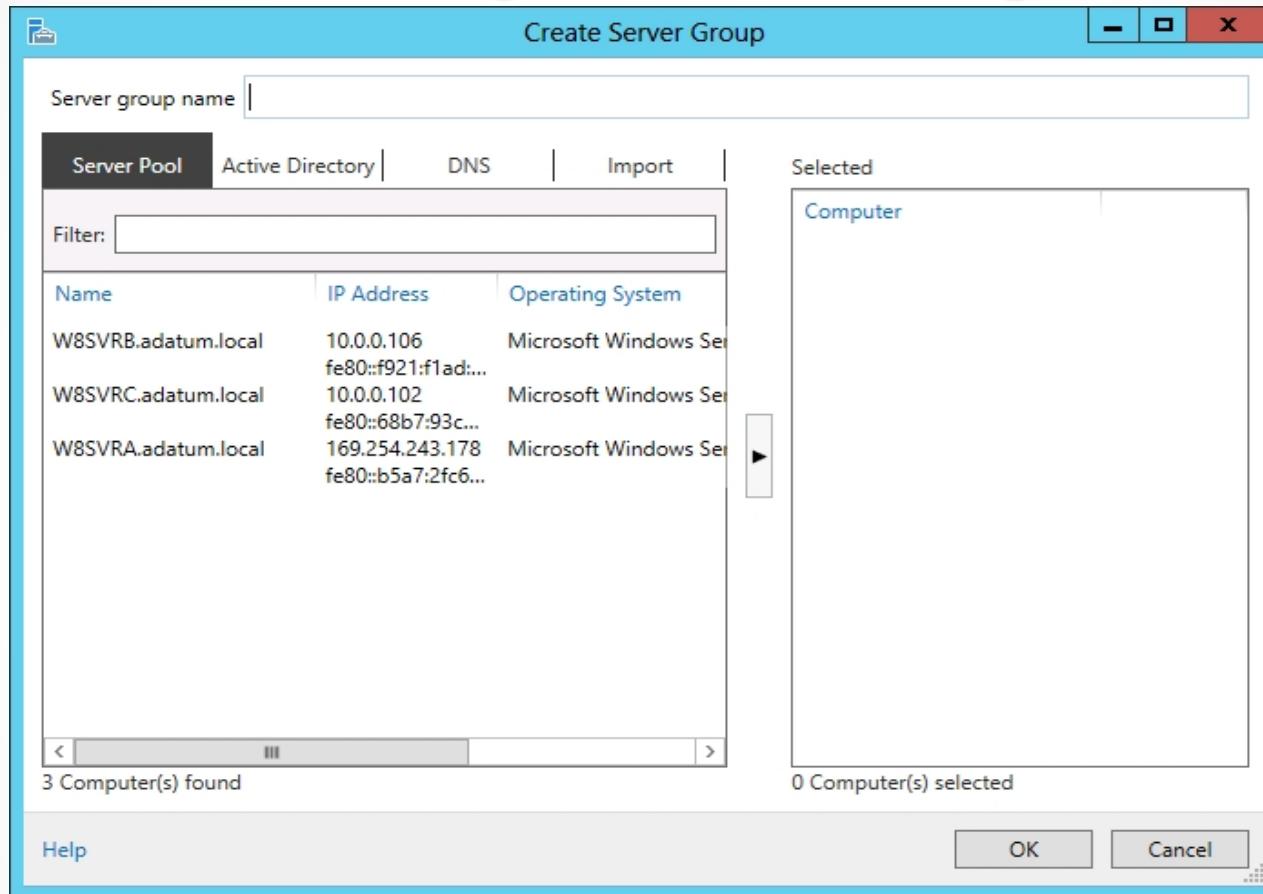
After the updates are installed, the system automatically starts the Windows Remote Management Service, but there are still tasks that must be completed on the remote server:

- Enable the Windows Remote Management (HTTP-In) rules in Windows Firewall.
- Create a WinRM listener by running the **winrm quickconfig** command at a command prompt with administrative privileges.
- Enable the COM+ Network Access and Remote Event Log Management rules in Windows Firewall.

# Creating Server Groups

- Server groups can be used to simplify administration of several servers.
- Groups can be based on server locations, functions, or any other organizational paradigm.
- Once created, it appears as an icon in the navigational pane and you can manage all the servers in the group, just like the All Servers group.

# Creating a Server Group



The Create Server Group dialog box in Server Manager

# Using Remote Server Administration Tools

Lesson 6: Configuring Servers for  
Remote Management

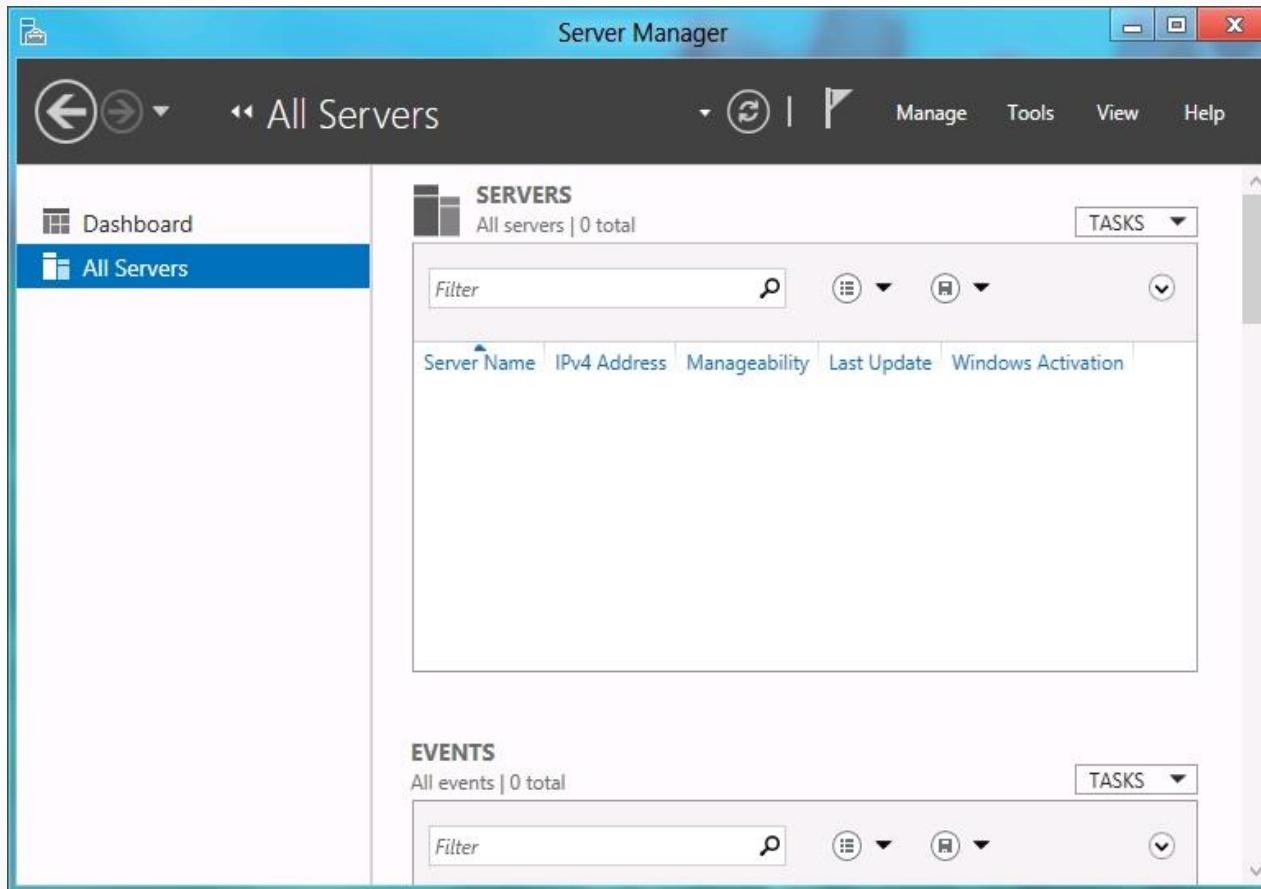
# Using Remote Server Administration Tools

- You can manage remote servers from any computer running Windows Server 2012.
- All the required tools are installed by default.
- The new administrative method that Microsoft is promoting urges administrators to keep servers locked away and use a workstation to manage servers from a remote location.
- To manage Windows servers from a workstation, you must download and install the Remote Server Administration Tools package.

# Using Remote Server Administration Tools

- When you install RSAT on a workstation running Windows 8, all the tools are activated by default.
- When you launch Server Manager on a Windows workstation, there is no local server, and there are no remote servers to manage until you add some.
- Your access to the servers you add depends on the account you use to log on to the workstation.
- You can connect to the server using another account by right-clicking it and, from the context menu, selecting Manage As to display a standard Windows Security dialog box, in which you can supply alternative credentials.

# Using Remote Server Administration Tools



Server Manager on a Windows workstation

# Using Windows PowerShell Web Access

Lesson 6: Configuring Servers for  
Remote Management

# Using Windows PowerShell Web Access

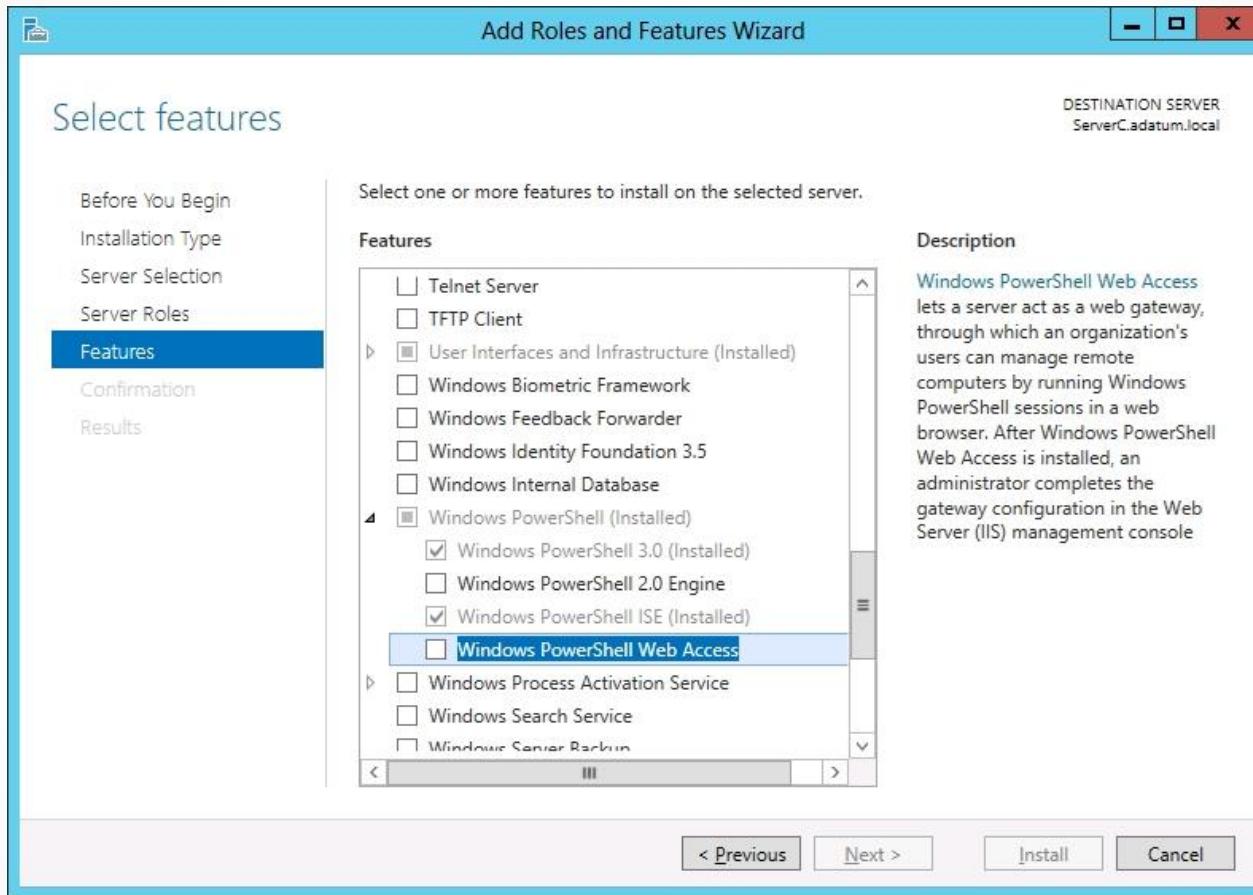
- A web gateway hosted by Internet Information Services (IIS) on the server to be managed, which enables an administrator to execute PowerShell commands on the server using a standard web browser.
- The big advantage is that the gateway is implemented entirely on the remote server being managed.
- The only software required on the client is a web browser that supports JavaScript and can retain cookies.
- The Administrator can execute PowerShell commands on a remote server using any computer, or even a smartphone or tablet.

# Using Windows PowerShell Web Access

The gateway server setup process includes the following steps:

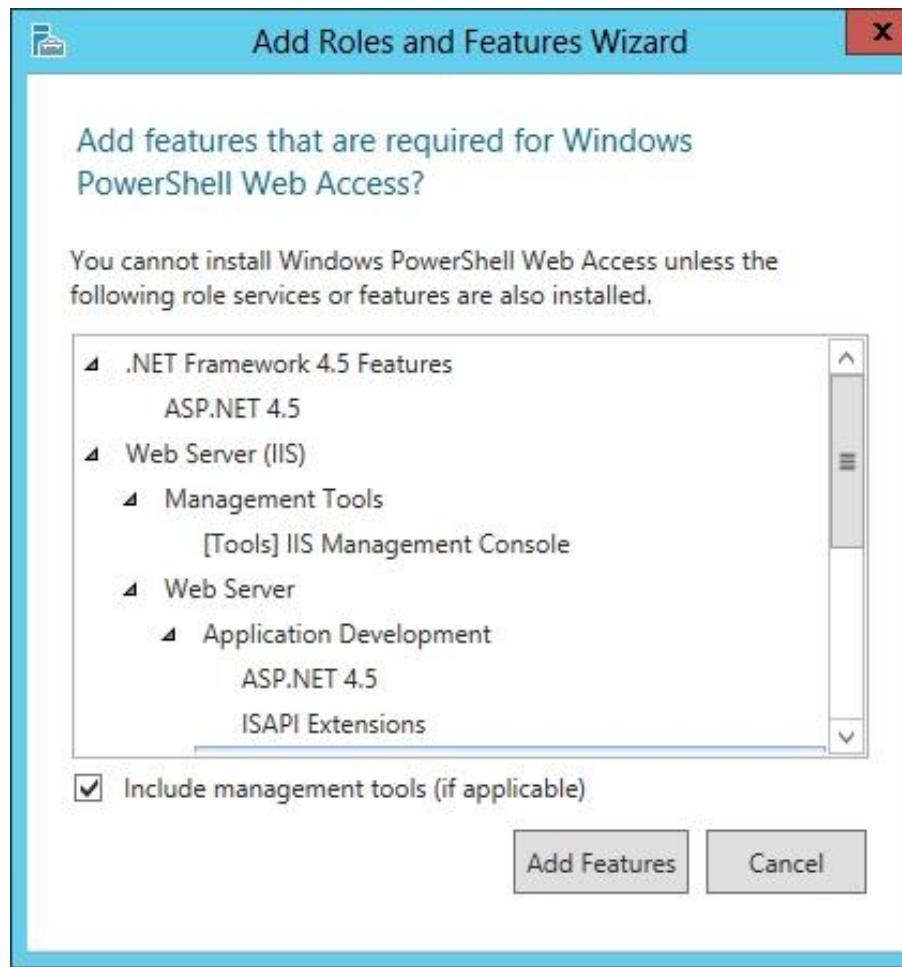
- Install the Windows PowerShell Web Access feature.
- Configure the IIS gateway.
- Create Authorization rules.

# Installing Windows PowerShell Web Access



The Windows PowerShell Web Access feature in the Add Roles and Features Wizard

# Installing Windows PowerShell Web Access



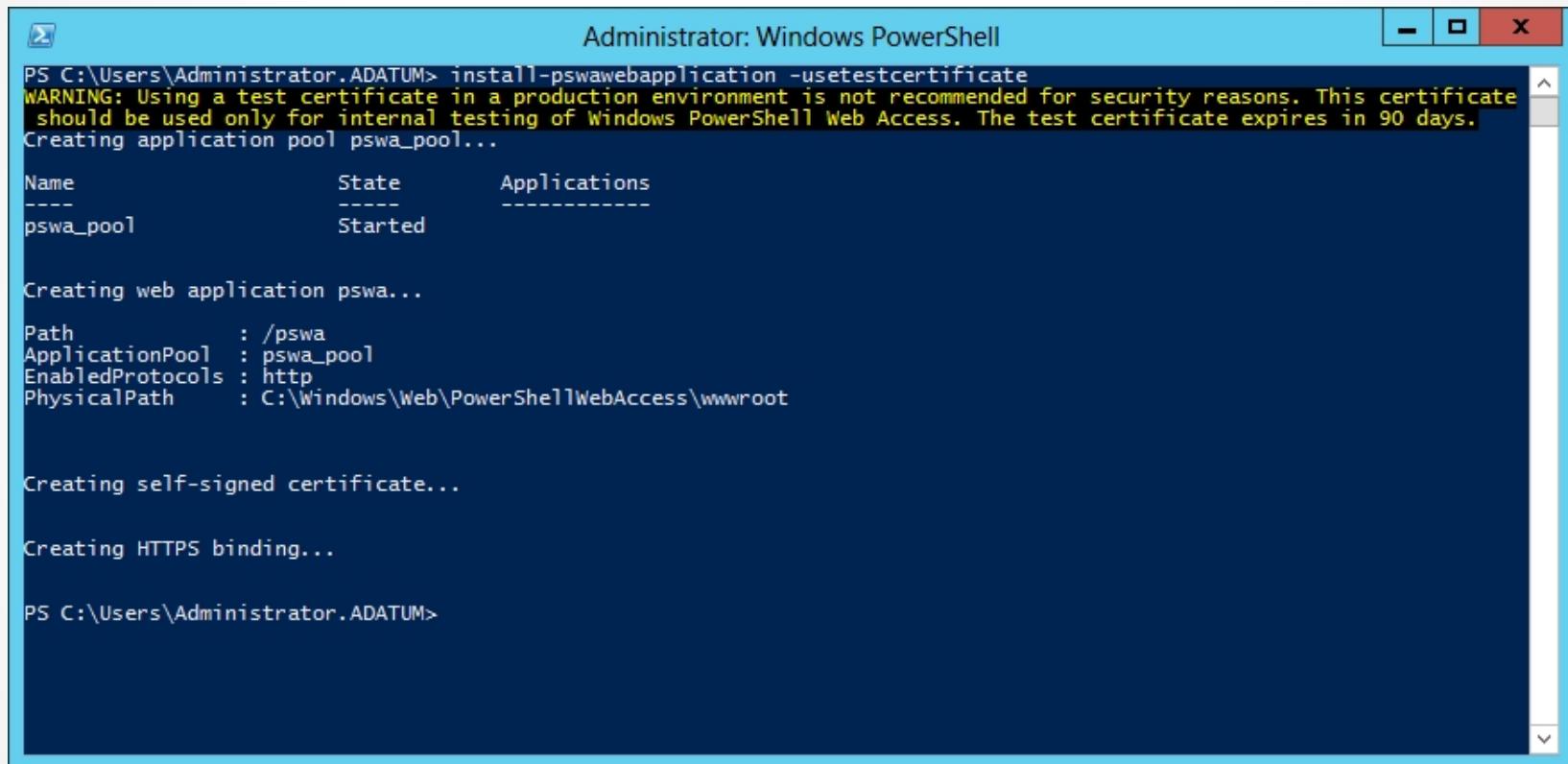
The Add Features that are required for Windows PowerShell Web Access dialog box

# Configuring PowerShell Web Access Gateway

The gateway configuration process consists of the following IIS tasks:

- Create an application pool for the **pswa** web application.
- Associate the application pool with a website.
- Configure the website to use the path to the gateway site files.
- Configure the website to use an https binding.
- Specify an SSL certificate for the website to use.

# Configuring a Test Installation



The screenshot shows an Administrator Windows PowerShell window with the title "Administrator: Windows PowerShell". The command entered is "install-pswawebapplication -usetestcertificate". A warning message follows: "WARNING: Using a test certificate in a production environment is not recommended for security reasons. This certificate should be used only for internal testing of Windows PowerShell Web Access. The test certificate expires in 90 days." The output shows the creation of an application pool named "pswa\_pool" which is started. It then creates a web application named "pswa" under this pool. The application's path is "/pswa", it uses the "pswa\_pool" application pool, has "http" as its enabled protocol, and its physical path is "C:\Windows\Web\PowerShellWebAccess\wwwroot". Next, it creates a self-signed certificate. Finally, it creates an HTTPS binding for the web application. The session ends with "PS C:\Users\Administrator.ADATUM>".

```
Administrator: Windows PowerShell
PS C:\Users\Administrator.ADATUM> install-pswawebapplication -usetestcertificate
WARNING: Using a test certificate in a production environment is not recommended for security reasons. This certificate
should be used only for internal testing of Windows PowerShell Web Access. The test certificate expires in 90 days.
Creating application pool pswa_pool...
Name          State      Applications
----          -----      -----
pswa_pool     Started

Creating web application pswa...
Path          : /pswa
ApplicationPool : pswa_pool
EnabledProtocols : http
PhysicalPath   : C:\Windows\Web\PowerShellWebAccess\wwwroot

Creating self-signed certificate...

Creating HTTPS binding...

PS C:\Users\Administrator.ADATUM>
```

Configuring the PowerShell Web Gateway with the  
default settings

# Customizing a Gateway Installation

The syntax of the cmdlet, with its main parameters:

```
Install-PswaWebApplication [-WebApplicationName <app name>]  
[-WebSiteName <site name>] [-UseTestCertificate]
```

The functions of the parameters:

**-WebApplicationName**: Enables you to specify an alternative to the default application name, which is pswa.

**-WebSiteName**: Enables you to specify an alternative to the default site in which the cmdlet installs the gateway application.

**-UseTestCertificate**: This parameter causes the server to create a self-signed certificate and bind it to the website.

# Creating Authorization Rules

When the gateway is properly configured, there are four layers of security that users must go through before they can execute commands on a server:

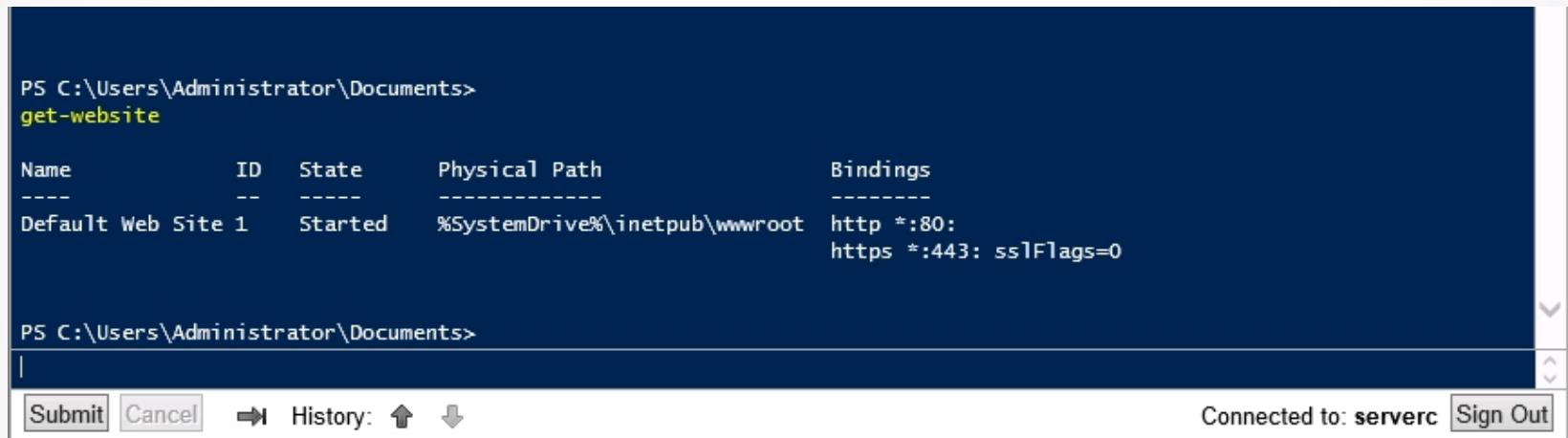
- IIS certificate authentication
- Windows PowerShell Web Access Gateway authentication
- Windows PowerShell Web authorization rules
- Target server authentication and authorization

# Creating Authorization Rules

To create and manage authorization rules, you use the following PowerShell cmdlets:

- Get-PswaAuthorizationRule
- Test-PswaAuthorizationRule
- Add-PswaAuthorizationRule
- Remove-PswaAuthorizationRule

# Creating Authorization Rules



The screenshot shows a Windows PowerShell Web Gateway session. The command `get-website` is run, displaying the configuration for the Default Web Site 1. The output is as follows:

Name	ID	State	Physical Path	Bindings
Default Web Site 1	--	Started	%SystemDrive%\inetpub\wwwroot	http *:80: https *:443: sslFlags=0

At the bottom of the window, there are buttons for **Submit**, **Cancel**, **History** (with up and down arrows), and status information **Connected to: serverc** and **Sign Out**.

An active Windows PowerShell Web Gateway session

# Working with Remote Servers

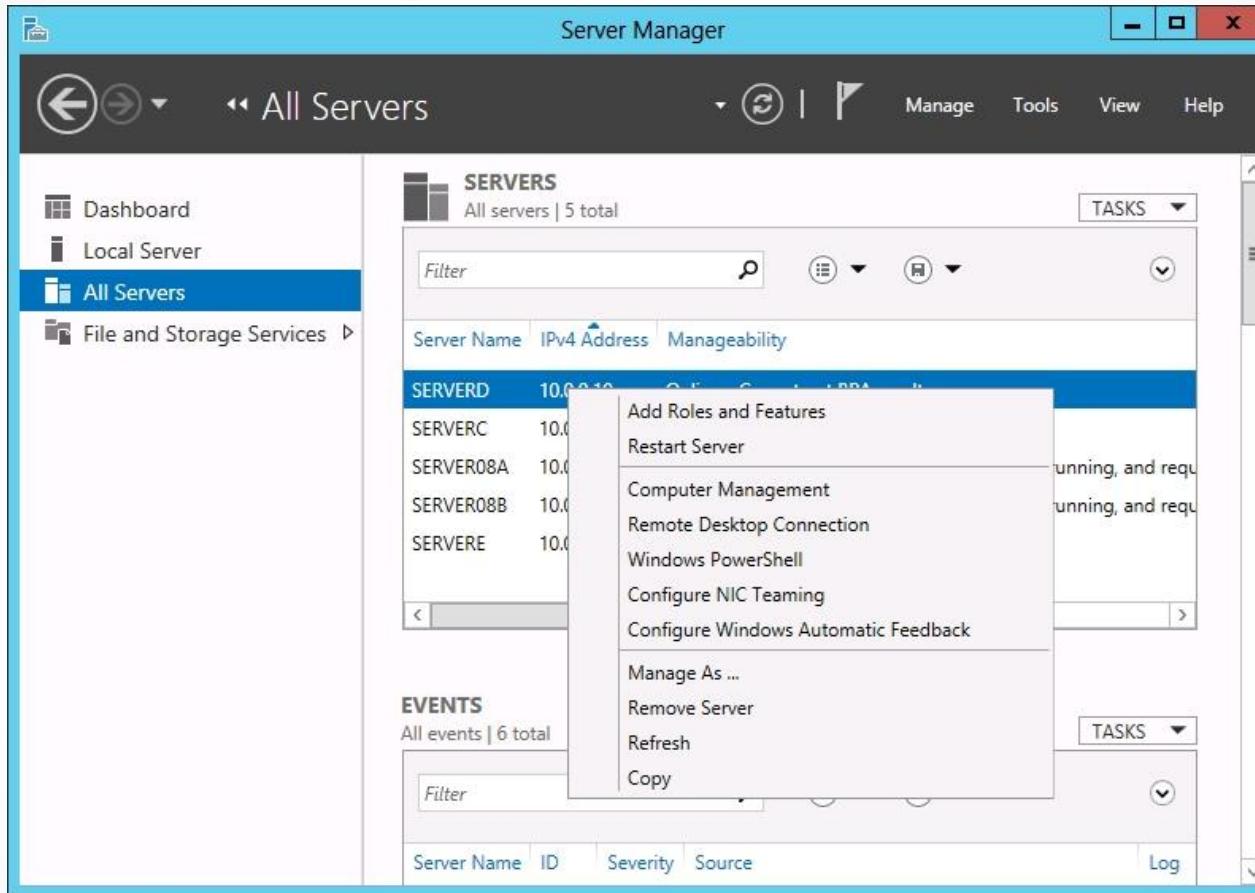
Lesson 6: Configuring Servers for  
Remote Management

# Working with Remote Servers

Server Manager provides three basic methods for addressing remote servers:

- **Contextual tasks:** When you right-click a server in a Servers tile, anywhere in Server Manager, you see a context menu that provides access to tools and commands pointed at the selected server.
- **Non-contextual tasks:** The menu bar at the top of the Server Manager console provides access to internal tasks.
- **Non-contextual tools:** The console's Tools menu provides access to external programs.

# Working with Remote Servers



Contextual tasks in Server Manager

# Lesson Summary

- Windows Server 2012 facilitates remote server management, so that administrators rarely have to work directly at the server console. This conserves server resources that can better be devoted to applications.
- The primary difference between the Windows Server 2012 Server Manager and previous versions is the ability to add and manage multiple servers at once.
- Server Manager has been tested with as many as 100 servers added to the interface. However, the tool's performance is based on a number of factors, including the hardware resources of the computer running Server Manager and the amount of data the remote servers are transmitting to Server Manager over the network.
- When you add servers running Windows Server 2012 to Server Manager, you can immediately begin using the Add Roles and Features Wizard to install roles and features on any of the servers you have added.

**Copyright 2013 John Wiley & Sons, Inc.**

All rights reserved. Reproduction or translation of this work beyond that named in Section 117 of the 1976 United States Copyright Act without the express written consent of the copyright owner is unlawful. Requests for further information should be addressed to the Permissions Department, John Wiley & Sons, Inc. The purchaser may make back-up copies for his/her own use only and not for distribution or resale. The Publisher assumes no responsibility for errors, omissions, or damages, caused by the use of these programs or from the use of the information contained herein.