# Module 9: Automating Integration with Azure Resources

# Lab: Automating the Creation of Azure Assets using PowerShell and xPlat CLI

### Scenario

Now that you have created many of the resources that you will use in your Azure application, you have decided to automate the creation of your assets in Azure. Some of your administrators are Windows experts and would prefer to automate using PowerShell while others use Linux and would prefer to automate from the command line. Due to this requirement, you will try and implement automation using PowerShell and separately using a cross-platform CLI interface.

### Objectives

After you complete this lab, you will be able to:

* Authenticate to Azure by using xPlat CLI
* Create a Website instance by using xPlat CLI.
* Remove a Website instance by using xPlat CLI.
* Authenticate to Azure by using PowerShell.
* Create a resource group from a template by using PowerShell.
* Remove a resource group by using PowerShell.

### Lab Setup

Estimated Time: 45 minutes

For this lab, you will use the available host machine. Before you begin this lab, you must complete the following steps:

Prior to starting this lab, you must have completed the lab in **Module 2**.

1. On the host computer, click **Start**, type **Remote**, and then click **Remote Desktop Connection**.
2. In Remote Desktop Connection, provide the name of your virtual machine or IP Address in the **Computer** box by using the following format:

* **[Your VM IP Address]:[*Your VM RDP Port*]**
* **Note:** The name and port for your virtual machine might be saved in the Computer drop-down list. If this is the case, use this value instead of typing it in manually. If you are unsure about your virtual machine’s RDP port, use either of the Azure portals to find your virtual machine’s endpoints. The endpoint with the name **Remote Desktop** is the correct port for RDP. This port is randomized to protect your virtual machine from unauthorized access.

1. In Remote Desktop Connection, click **Connect**. Wait until the RDP client accesses the virtual machine.
2. If necessary, sign in by using the following credentials:

* User name: **Student**
* Password: **AzurePa$$w0rd**

1. Verify that you received the credentials to sign in to the Azure portal from your training provider. You will use these credentials and the Azure account throughout the labs in this course.

## Exercise 1: Use xPlat CLI to Create and Manage and Azure Web App

### Scenario

In this exercise, you will:

* Install the Azure xPlat CLI package.
* Authenticate to Azure using the xPlat CLI.
* Create a Web App using the xPlat CLI.
* Remove the Web App using xPlat CLI

The main tasks for this exercise are as follows:

1. Install xPlat CLI.
2. Authenticate to Azure by using xPlat CLI.
3. Create a new Website instance by using xPlat CLI.
4. Remove a Website instance by using xPlat CLI.

#### Task 1: Preparing the xPlat CLI Environment

1. Open the **Microsoft Web Platform Installer** application.
2. Locate and install the **Microsoft Azure Cross-platform Command Line Tools** product.
3. Open the **Command Prompt** console window.
4. Validate that the **azure** CLI command is available using the following command:

* azure --help

1. Switch to the **ARM** mode for the CLI using the following command:

* azure config mode arm

#### Task 2: Login to Your Azure Subscription Using Interactive Login

1. Start the interactive login by using the following command:

* azure login

1. Sign in to the Azure Device Login page (<https://aka.ms/devicelogin>) using the code provided by the CLI command.
2. Return to the **Command Prompt** application.
3. View your active account details:

* azure account show
* **Note:** If you have multiple Azure subscriptions, you can use **azure account set [subscription-name]** to select the appropriate subscription. If you are unsure about your subscription's name, you can simply use **azure account list**. If your Azure subscription is provided by a Learning Partner, it might be an auto-generated account without any details. In such a case, this command will return an empty result.

**Results:** After completing this exercise, you will have set up a development environment to use xPlat CLI for Azure service automation.

#### Task 3: Create a new Website instance by using xPlat CLI

1. Create a new Resource Group by using the name **rga20532** and the **West US** location:

* azure group create --name rga20532 --location "West US"

1. View the list of Resource Groups for your subscription by using the following command:

* azure group list

1. Create a new Web App by using a unique name and the **East US** location:

* azure resource create --name [Unique Name] --resource-type "Microsoft.Web/Sites" --api-version "2014-06-01" --location "East US" --resource-group rga20532

1. View the list of Resources in the **rga20532** Resource Group by using the following command:

* azure resource list --resource-group rga20532

1. View the details for your newly created Website by using the following command:

* azure resource show --name [Unique Name] --resource-type "Microsoft.Web/Sites" --resource-group rga20532 --api-version "2014-06-01" --json

1. Record the **defaultHostName** and **sku** values from your new Web App's properties.
2. View the newly created Website by using the following command:

* explorer “http://[Host Name]”

1. Close Internet Explorer.
2. Switch to the **Command Prompt** console window.

#### Task 4: Remove a Website instance by using xPlat CLI

1. Remove your newly created Web App instance by using the following command:

* azure resource delete --name [Unique Name] --resource-type "Microsoft.Web/Sites" --resource-group rga20532 --api-version "2014-06-01" --json

1. Remove your newly created Resource Group by using the following command:

* azure group delete --name rga20532

1. Validate that the Resource Group has been successfully removed using the following command:

* azure group show --name rga20532

**Results:** After completing this exercise, you will have used xPlat CLI to manage instances of an Azure service.

## Exercise 2: Use PowerShell to Create and Manage a Resource Group

### Scenario

In this exercise, you will:

* Authenticate to Azure in Windows PowerShell by using Azure Active Directory.
* Switch the Azure modules in use by using Windows PowerShell.
* Create a resource group from a template by using Windows PowerShell.
* Delete a resource group by using Windows PowerShell.

The main tasks for this exercise are as follows:

1. Switch to the Azure Resource Manager PowerShell Modules.
2. Create a resource group by using Windows PowerShell.
3. Remove the resource group by using Windows PowerShell.

#### Task 1: Open the Azure PowerShell Console

1. Open the **Windows PowerShell** console window.

#### Task 2: Authenticate to Azure Resource Manager Using PowerShell

1. Switch to the **AzureResourceManager** mode by using the following command:

* Login-AzureRmAccount

1. Authenticate to Azure by signing in to your Azure Active Directory account:
2. To view your current Azure PowerShell session context type the following command in the console, and then press Enter:

* Get-AzureRmContext
* **Note:** If you have multiple Azure subscriptions, you can use **Select-AzureRmSubscription** to select the appropriate subscription.

#### Task 3: Create a resource group by using PowerShell

1. Create a new instance of the **Resource group** by using following command and details:

* New-AzureRmResourceGroup -Name rgb20532 -Location "West US"   
  New-AzureRmResourceGroupDeployment -ResourceGroupName rgb20532 -TemplateFile "F:\Mod09\Labfiles\azuredeploy.json"
  + administratorLogin: **testuser**
  + administratorLoginPassword: **TestPa$$w0rd**
* **Note:** Wait for the provisioning process to complete. Status messages will be displayed periodically and the final message with the details will be displayed when provisioning is complete.

1. View the details for your newly created resource group by using the following command:

* Get-AzureRmResourceGroup –ResourceGroupName rgb20532

1. View the details for your newly created Web App in your resource group by using the following command:

* (Get-AzureRmResource -IsCollection -ResourceGroupName rgb20532 -ResourceType "Microsoft.Web/sites" -ApiVersion 2015-08-01).Properties

1. Record the value of the **hostNames** property from your new Web App's properties.
2. View the newly created Website by using the following command:

* explorer “http://[Host Name]”

1. Close Internet Explorer.
2. Switch to the **Windows PowerShell** console window.

#### Task 4: Remove the resource group by using Windows PowerShell

1. Remove your newly created resource group by using the following command:

* Remove-AzureRmResourceGroup –Name rgb20532
* **Note:** Because removing a resource group involves removing multiple resources, this process can take 5 to 15 minutes on average.

**Results:** After completing this exercise, you will have used Azure to interact with the Resource Manager, resource groups, and resource group templates.

©2016 Microsoft Corporation. All rights reserved. The text in this document is available under the [Creative Commons Attribution 3.0 License](https://creativecommons.org/licenses/by/3.0/legalcode), additional terms may apply. All other content contained in this document (including, without limitation, trademarks, logos, images, etc.) are ***not*** included within the Creative Commons license grant. This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

This document is provided "as-is." Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it. Some examples are for illustration only and are fictitious. No real association is intended or inferred. Microsoft makes no warranties, express or implied, with respect to the information provided here.