Module 11 - Lab 2: Use a Windows VM system-assigned managed identity to access Resource Manager

This lab shows you how to access the Azure Resource Manager API using a Windows virtual machine with system-assigned managed identity enabled. Managed identities for Azure resources are automatically managed by Azure and enable you to authenticate to services that support Azure AD authentication without needing to insert credentials into your code. You learn how to:

- Grant your VM access to a Resource Group in Azure Resource Manager
- Get an access token using the VM identity and use it to call Azure Resource Manager

Task 1: Grant your VM access to a resource group in Resource Manager

Using managed identities for Azure resources, your code can get access tokens to authenticate to resources that support Azure AD authentication. The Azure Resource Manager supports Azure AD authentication. First, we need to grant this VM's system-assigned managed identity access to a resource in Resource Manager, in this case the Resource Group in which the VM is contained.

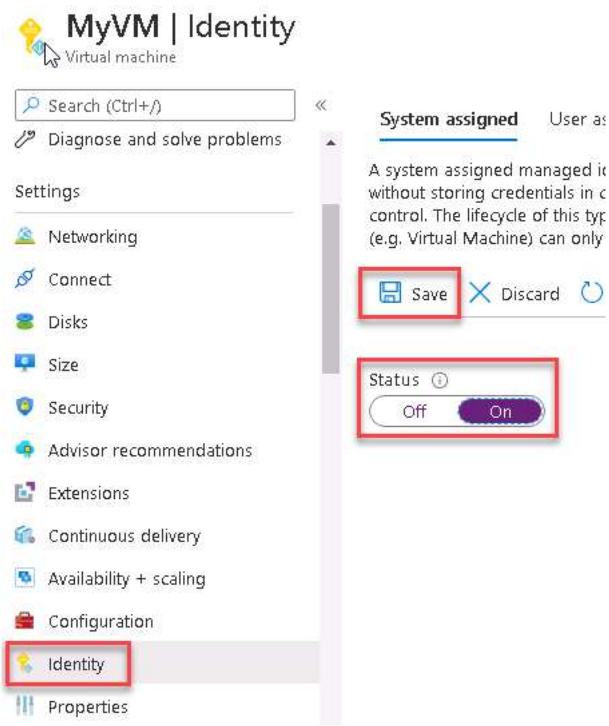
☐ 1. Login to the **Azure Portal** with the username **sheikhnasir7YCAV@gdcs1.com** and password **6d0x82Cy3Qa2CV3I**

☐ 2. Select **Virtual Machines**.

☐ 3. Select your **myVM** Virtual Machine.

☐ 4. In the **Settings** section select **identity**.

☐ 5. Change the System assigned identity to **On** and click **Save** > **Yes**.

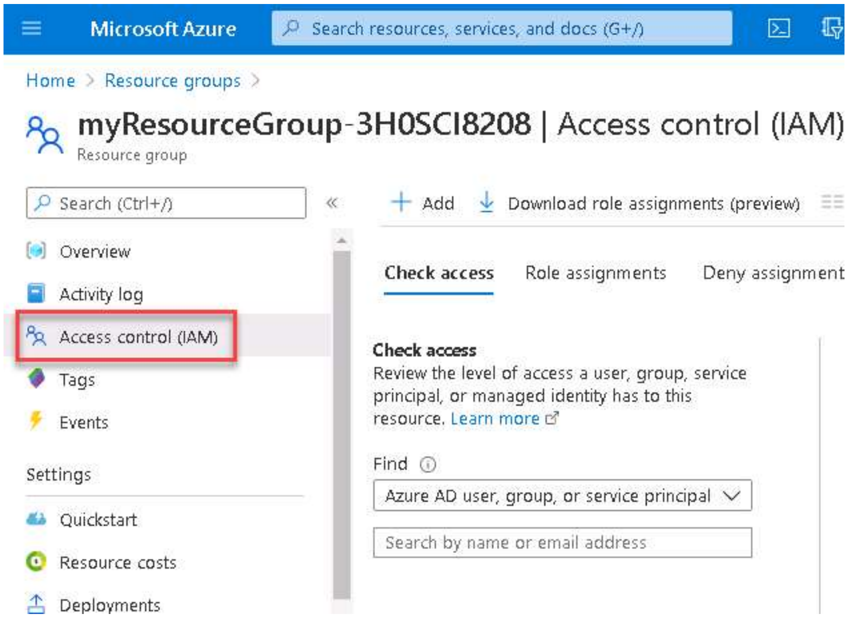


A system assigned managed i
without storing credentials in c
control. The lifecycle of this typ
(e.g. Virtual Machine) can only

☐ 6. Navigate to **Resource Groups**.

☐ 7. Select the **myResourceGroup** which contains a VM created for you called **myVM**.

☐ 8. Go to **Access control (IAM)** in the left panel.



Home > Resource groups >

myResourceGroup-3H0SCI8208 | Access control (IAM)

Resource group

Check access

Review the level of access a user, group, service principal, or managed identity has to this resource. Learn more

Find: Azure AD user, group, or service principal

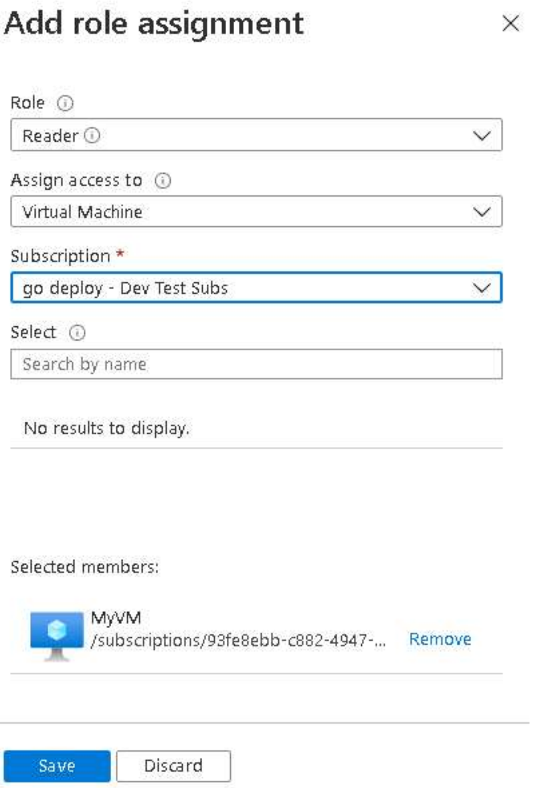
Search by name or email address

□ 9. Then click **+ Add** > **Add role assignment** a new role assignment for your **Windows VM**. Choose **Role** as **Reader**.

□ 10. In the next drop-down, **Assign access to** the resource **Virtual Machine**.

□ 11. Next, ensure your **Subscription** is selected.

□ 12. Finally, in **Select** choose **myVM** and click **Save**.



Add role assignment

Role: Reader

Assign access to: Virtual Machine

Subscription *: go deploy - Dev Test Subs

Select: Search by name

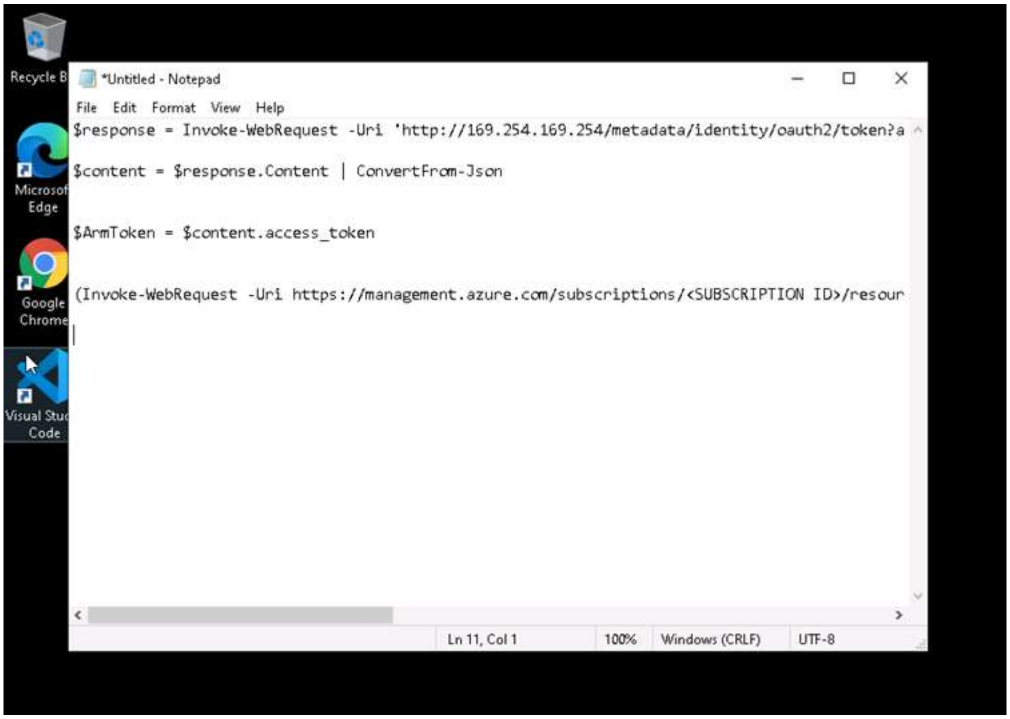
No results to display.

Selected members:

MyVM
/subscriptions/93fe8ebb-c882-4947-...

Task 2: Get an access token using the VM's system-assigned managed identity and use it to call Azure Resource Manager

> **Note**: It is recommended you paste the commands into a Notepad session the Lab VM prior to pasting the commands into the RDP session. On occasions characters are mis-typed.



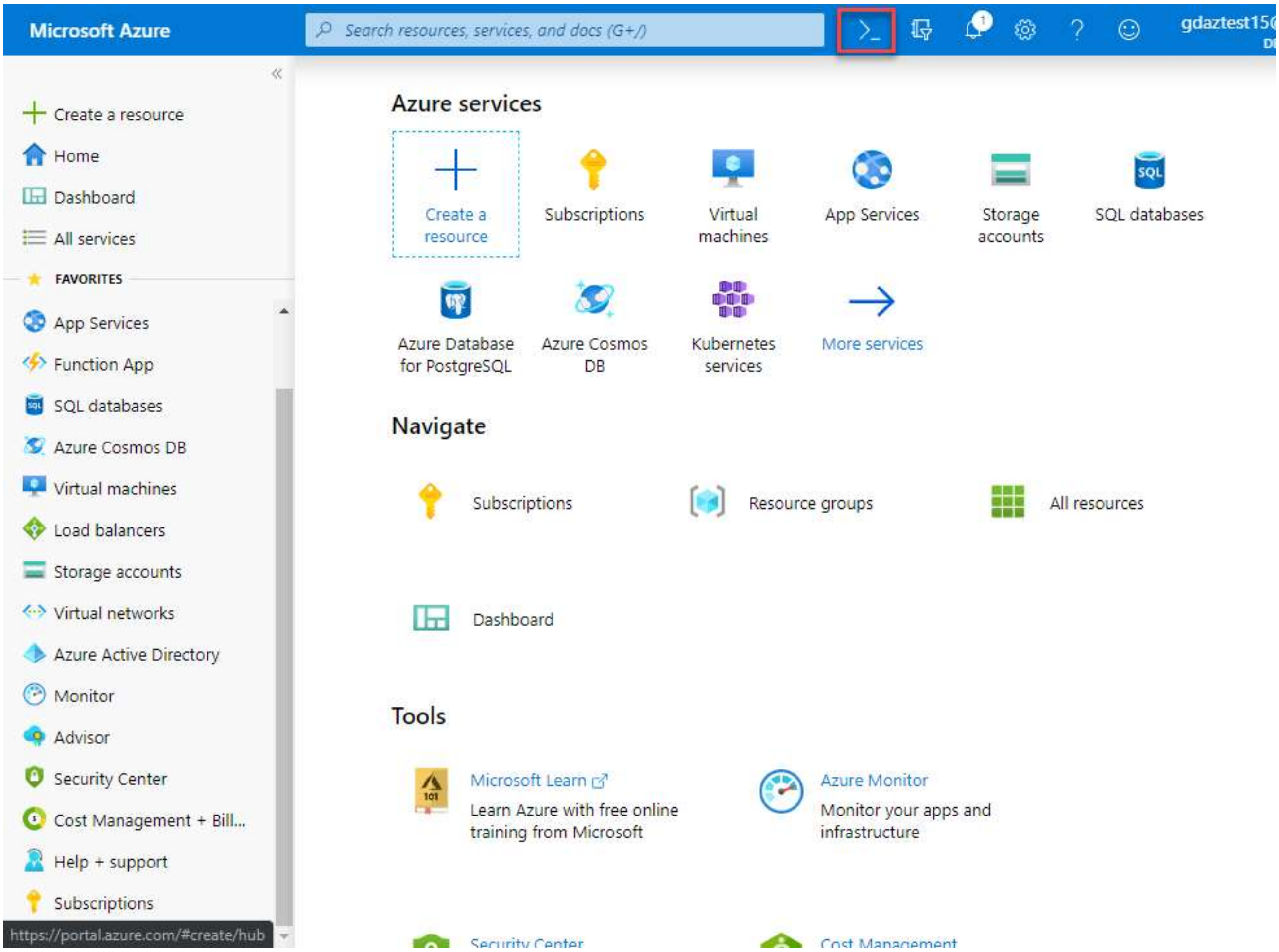
```
$response = Invoke-WebRequest -Uri 'http://169.254.169.254/metadata/identity/oauth2/token?a

$content = $response.Content | ConvertFrom-Json


$ArmToken = $content.access_token


(Invoke-WebRequest -Uri https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resour
```

□ 1. Select **Cloud Shell** from the Azure Portal tool bar.



□ 2. Select **PowerShell** on the Welcome screen.

□ 3. In the **You have no storage mounted** pane, click **Show advanced settings**, perform the following tasks:

- Leave the **Subscription** drop-down list entry set to its default value.
- In the **Cloud Shell region** East US.

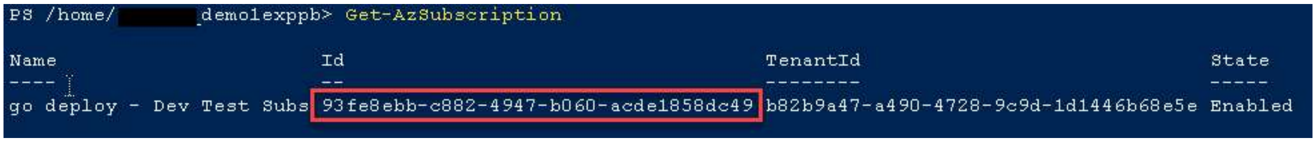
- In the **Resource group** section, select the Resource Group that has been created for you.
- In the **Storage account** section, ensure that the **Create new** option is selected and then, in the text box below, type a unique name consisting of a combination of between 3 and 24 characters and digits.
- In the **File share** section, ensure that the **Create new** option is selected and then, in the text box below, type **cloudshell**.
- Click the **Create storage** button.

□ 4. Wait for the **Cloud Shell** to finish its first-time setup procedures before you proceed to the next task.

□ 5. Run the following command to get your Azure Subscription ID:

```
Get-AzSubscription
```

□ 6. Copy the Subscription ID to your Notepad file.



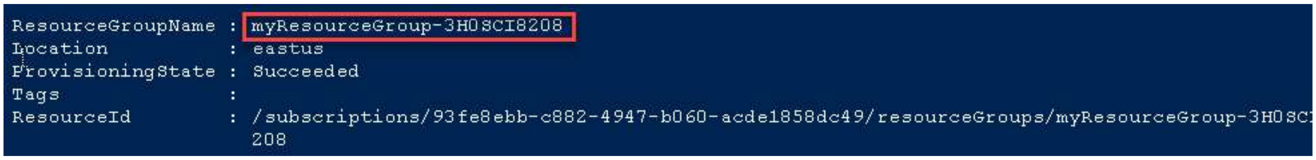
```
PS /home/[redacted]_demolexppb> Get-AzSubscription

Name                     Id                                   TenantId                             State
----                     --                                   --------                             -----
go deploy - Dev Test Subs 93fe8ebb-c882-4947-b060-acde1858dc49 b82b9a47-a490-4728-9c9d-1d1446b68e5e Enabled
```

□ 7. Run the following command to get your Resource Group ID:

```
Get-AzResourceGroup
```

□ 8. Copy the Resource Group name to your Notepad file.



```
ResourceGroupName : myResourceGroup-3HOSCI8208
Location          : eastus
ProvisioningState : Succeeded
Tags              :
ResourceId        : /subscriptions/93fe8ebb-c882-4947-b060-acde1858dc49/resourceGroups/myResourceGroup-3HOSCI
                    208
```

□ 9. In the portal, navigate to **Virtual Machines** and go to your Windows virtual machine and in the **Overview**, click **Connect > RDP**. Download and run the RDP file.

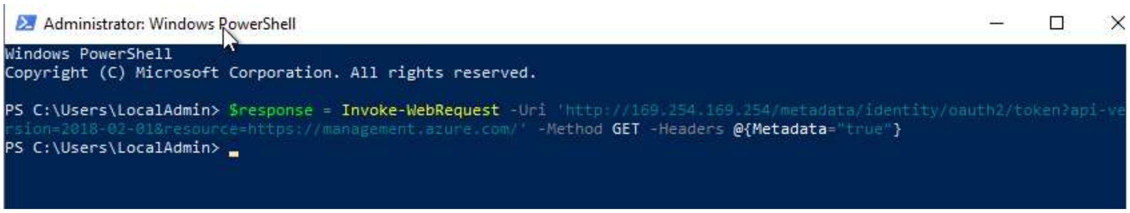
□ 10. Enter the username **localadmin** and password **hHdZF87FqAsjNMqF**.

□ 11. Now that you have created a **Remote Desktop Connection** with the virtual machine, open **PowerShell** in the remote session.

□ 12. Using the Invoke-WebRequest cmdlet, make a request to the local managed identity for Azure resources endpoint to get an access token for Azure Resource Manager.

```
$response = Invoke-WebRequest -Uri 'http://169.254.169.254/metadata/identity/oauth2/token?api-version=2018-02-01&resource=https://manage
```

> ⚠ **Note**: The value of the "resource" parameter must be an exact match for what is expected by Azure AD. When using the Azure Resource Manager resource ID, you must include the trailing slash on the URI.

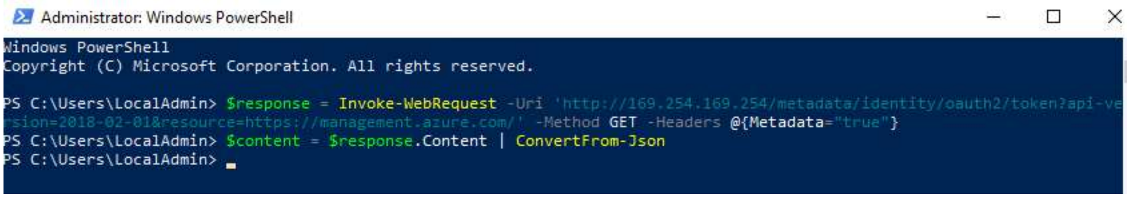


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\LocalAdmin> $response = Invoke-WebRequest -Uri 'http://169.254.169.254/metadata/identity/oauth2/token?api-ve
rsion=2018-02-01&resource=https://management.azure.com/' -Method GET -Headers @{Metadata="true"}
PS C:\Users\LocalAdmin> _
```

Next, extract the full response, which is stored as a JavaScript Object Notation (JSON) formatted string in the $response object.

```
$content = $response.Content | ConvertFrom-Json
```



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\LocalAdmin> $response = Invoke-WebRequest -Uri 'http://169.254.169.254/metadata/identity/oauth2/token?api-ve
rsion=2018-02-01&resource=https://management.azure.com/' -Method GET -Headers @{Metadata="true"}
PS C:\Users\LocalAdmin> $content = $response.Content | ConvertFrom-Json
PS C:\Users\LocalAdmin> _
```

Next, extract the access token from the response.

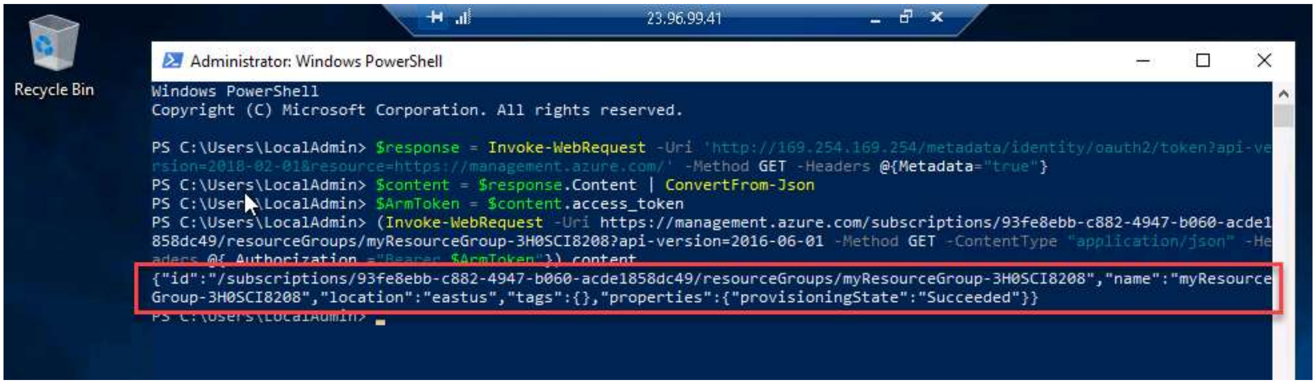
```
$ArmToken = $content.access_token
```

Finally, call Azure Resource Manager using the access token. In this example, we're also using the Invoke-WebRequest cmdlet to make the call to Azure Resource Manager, and include the access token in the Authorization header. Ensure you replace **<SUBSCRIPTION ID>** and **<RESOURCE GROUP>** with the attributes you copied at the start of this task.

```
(Invoke-WebRequest -Uri https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup-WTRZR9T68S?api-versi
```

⚠ **Note**: The URL is case-sensitive, so ensure if you are using the exact same case as you used earlier when you named the Resource Group, and the uppercase "G" in "resourceGroups."

The command returns the details of the Resource Group:



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\LocalAdmin> $response = Invoke-WebRequest -Uri 'http://169.254.169.254/metadata/identity/oauth2/token?api-ve
rsion=2018-02-01&resource=https://management.azure.com/' -Method GET -Headers @{Metadata="true"}
PS C:\Users\LocalAdmin> $content = $response.Content | ConvertFrom-Json
PS C:\Users\LocalAdmin> $ArmToken = $content.access_token
PS C:\Users\LocalAdmin> (Invoke-WebRequest -Uri https://management.azure.com/subscriptions/93fe8ebb-c882-4947-b060-acde1
858dc49/resourceGroups/myResourceGroup-3H0SCI8208?api-version=2016-06-01 -Method GET -ContentType "application/json" -He
aders @{ Authorization ="Bearer $ArmToken"}).content
{"id":"/subscriptions/93fe8ebb-c882-4947-b060-acde1858dc49/resourceGroups/myResourceGroup-3H0SCI8208","name":"myResource
Group-3H0SCI8208","location":"eastus","tags":{},"properties":{"provisioningState":"Succeeded"}}
PS C:\Users\LocalAdmin>
```

✓ In this quickstart, you learned how to use a system-assigned managed identity to access the Azure Resource Manager API.