Azure

PERIODICALLY RESTART AZURE WEBAPP SERVICE

spursol

AUGUST 05, 2022

Steps:

1. Create an Automation account.
2. Set one of the two methods available for the authentication of automation account.
3. Import necessary modules, create a Runbook and schedule that Runbook.

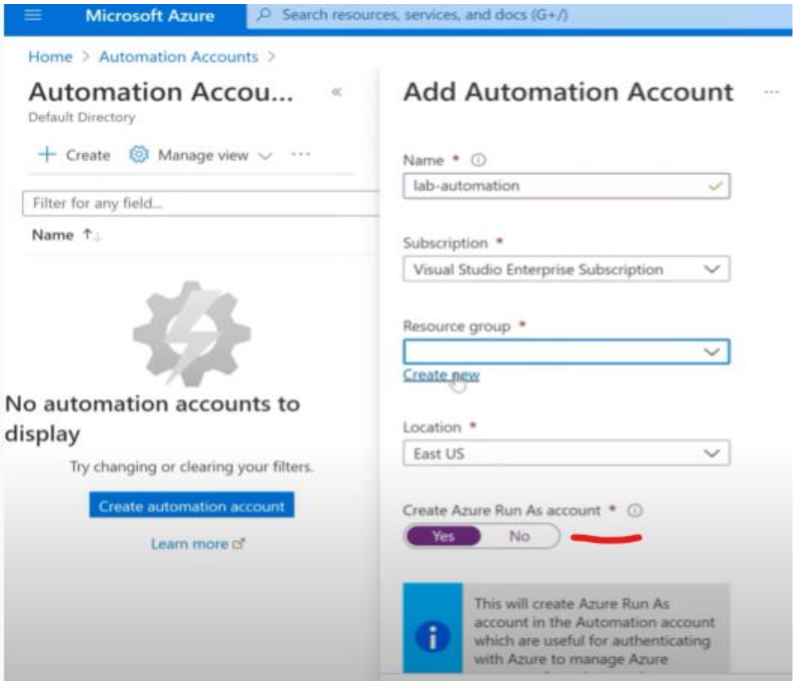
Creation of Automation Account and Methods for Authentication

There are two methods to configure the automation account's authentication.

- Run-As Account
- Managed Identity.

1. **Run-As Account**

Search for Automation Account in Azure resources and create.



No automation accounts to display

Try changing or clearing your filters.

Name * lab-automation

Subscription * Visual Studio Enterprise Subscription

Resource group *

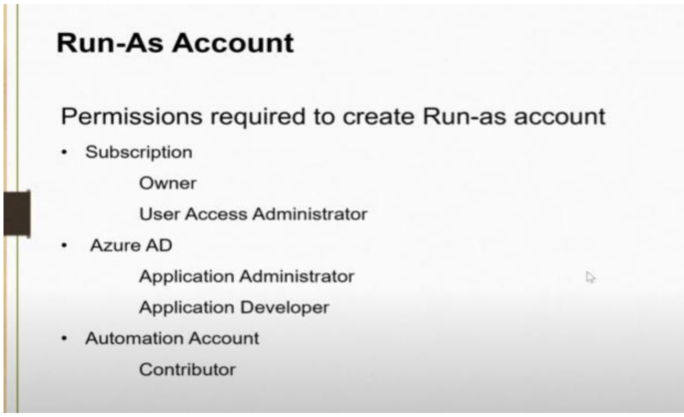
Location * East US

Create Azure Run As account * [✓ Yes] No

This will create Azure Run As account in the Automation account which are useful for authenticating with Azure to manage Azure

Note: - When you create an Automation account, the option to create a Run As account is no longer available. However, you can still create a Run As account in your Automation account from the Azure portal after the creation.

For an automation account to have all permissions to automate services, one must have the following permissions:



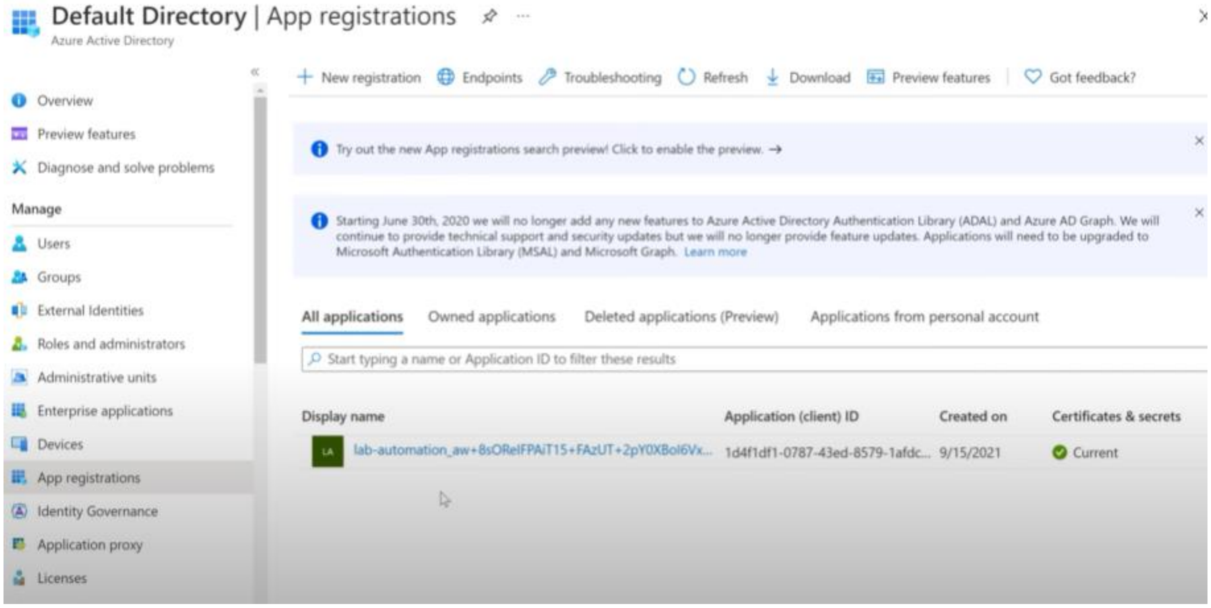
Run-As Account

Permissions required to create Run-as account

- Subscription
  - Owner
  - User Access Administrator
- Azure AD
  - Application Administrator
  - Application Developer
- Automation Account
  - Contributor

To verify:

Go to **Azure Active Directory** > **App Registrations** > There you shall see a new service principal created.



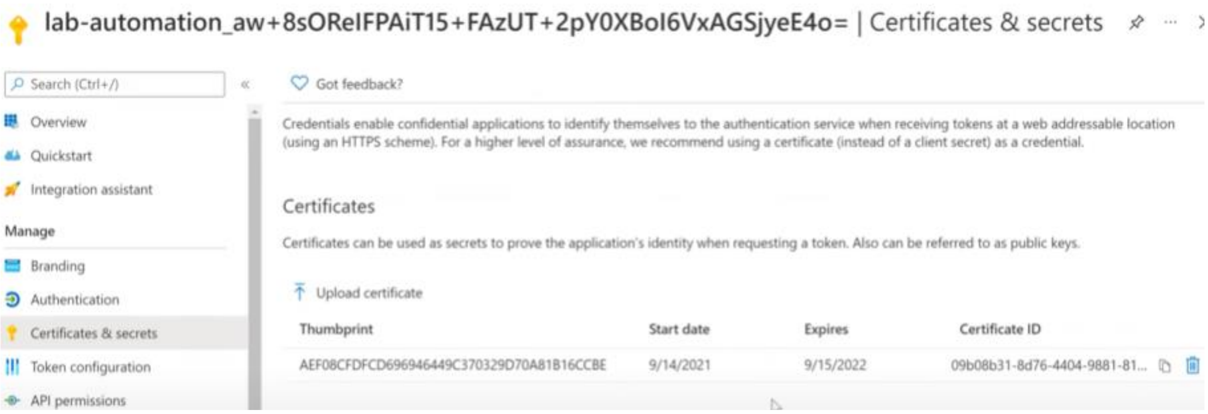
Try out the new App registrations search preview! Click to enable the preview. →

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure AD Graph. We will continue to provide technical support and security updates but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. Learn more

Start typing a name or Application ID to filter these results

Display name	Application (client) ID	Created on	Certificates & secrets
LA lab-automation_aw+8sORelFPAiT15+FAzUT+2pY0XBoI6Vx...	1d4f1df1-0787-43ed-8579-1afdc...	9/15/2021	✓ Current

Select the created principal and go to **Certificates and Secrets**, there a self-signed certificate should also be created.



Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

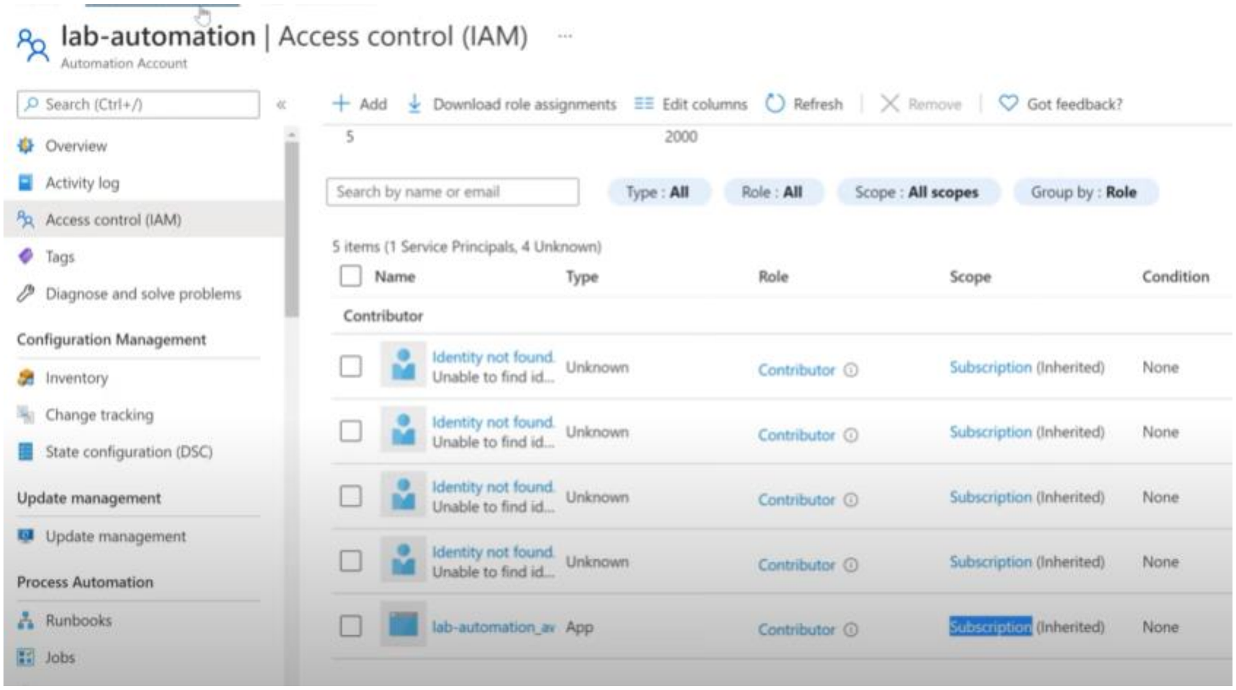
Certificates

Certificates can be used as secrets to prove the application's identity when requesting a token. Also can be referred to as public keys.

Thumbprint	Start date	Expires	Certificate ID
AEF08CFDFCD696946449C370329D70A81B16CCBE	9/14/2021	9/15/2022	09b08b31-8d76-4404-9881-81...

Note: - The automation account should stop working when the certificate expires.

Navigate back to Automation Account and click on **Access Control (IAM)** to check access.



5 items (1 Service Principals, 4 Unknown)

Name	Type	Role	Scope	Condition
Contributor				
Identity not found. Unable to find id...	Unknown	Contributor	Subscription (Inherited)	None
Identity not found. Unable to find id...	Unknown	Contributor	Subscription (Inherited)	None
Identity not found. Unable to find id...	Unknown	Contributor	Subscription (Inherited)	None
Identity not found. Unable to find id...	Unknown	Contributor	Subscription (Inherited)	None
lab-automation_av	App	Contributor	Subscription (Inherited)	None

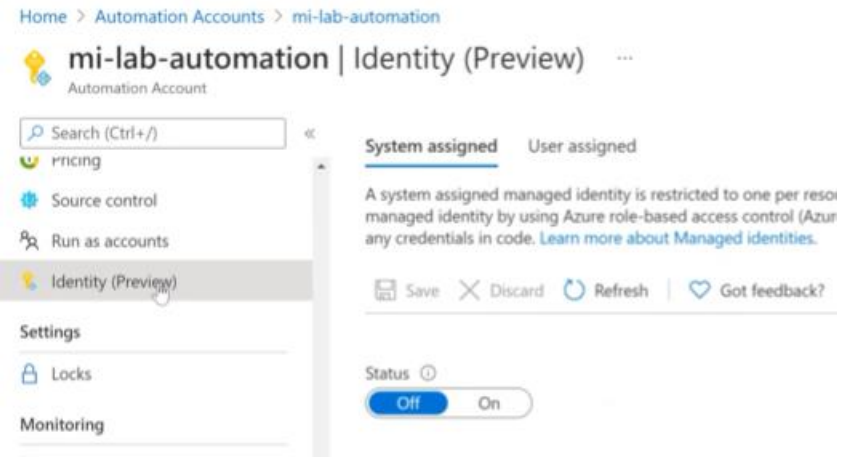
lab-automation is the service principle created in Azure Active Directory.

Note: - All the above-mentioned resources along with Service Principle are automatically generated if you select "Azure Run As account”

2. Managed Identity

In this authentication method, you do not select the “Run as Account” and use the **Identity** feature.

Go to your Automation Account and find Identity.

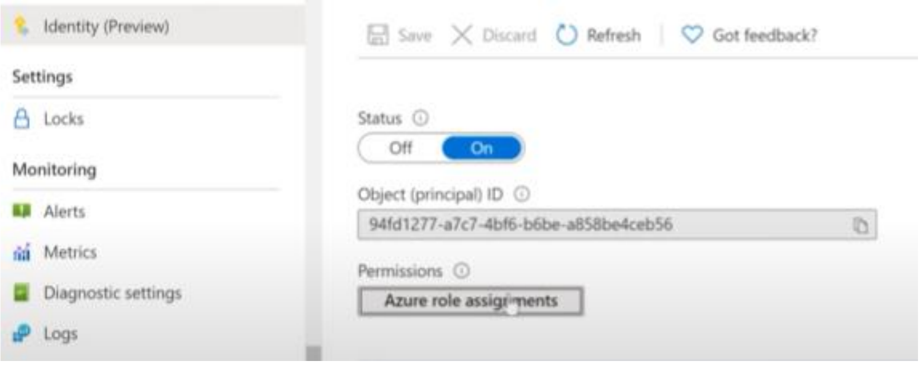


A system assigned managed identity is restricted to one per reso
managed identity by using Azure role-based access control (Azur
any credentials in code. Learn more about Managed identities.

Status: Off | On (Off selected)

Set the Status to ON and save.

This creates a service principal in the back end and will allow Microsoft to manage the service principal for you.



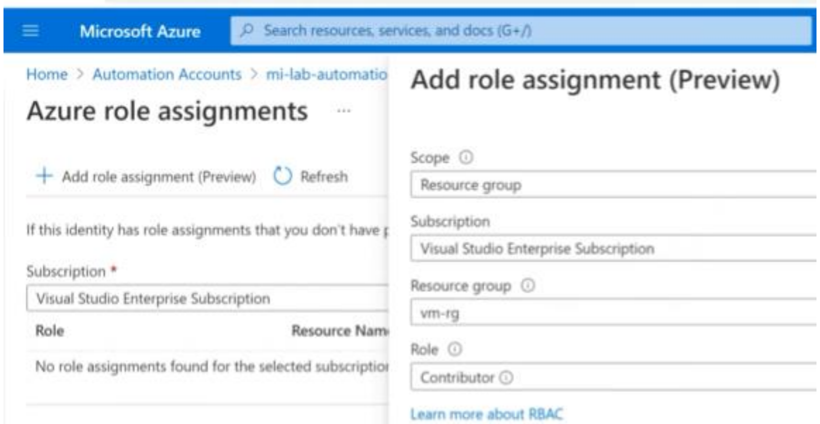
Status: Off | On (On selected)

Object (principal) ID: 94fd1277-a7c7-4bf6-b6be-a858be4ceb56

Permissions

After turning it on, go to the Azure role assignments and assign the role to the service which you want to use for automation.

In above example, Contributor access to a resource group is given. Make sure to give access to the resource group in which the service is present that you want to automate.



Microsoft Azure
Search resources, services, and docs (G+/)
Home > Automation Accounts > mi-lab-automatio
+ Add role assignment (Preview)
Refresh
Learn more about RBAC

Azure role assignments ...

If this identity has role assignments that you don't have p

Subscription * Visual Studio Enterprise Subscription

Role	Resource Nam
No role assignments found for the selected subscriptio	

Add role assignment (Preview)

Scope: Resource group

Subscription: Visual Studio Enterprise Subscription

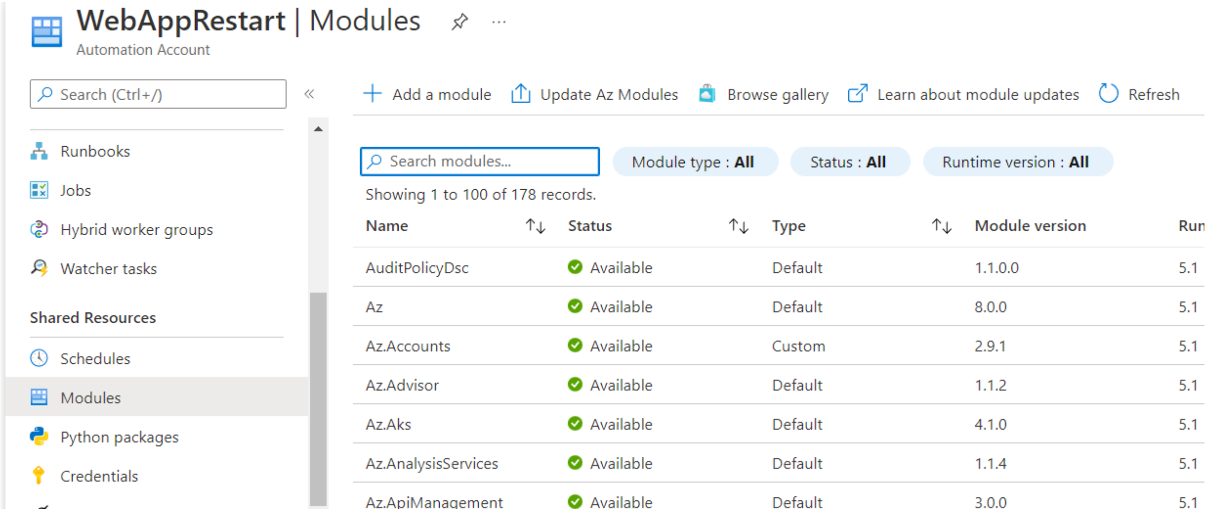
Resource group: vm-rg

Role: Contributor

Note: - If you go back to the Active Directory App Registrations, you won’t see the service principal there because it is managed by Microsoft.

Creation of the Scheduled Runbook

Go to your Automation account and browse the Modules section.



Module type : **All** Status : **All** Runtime version : **All**

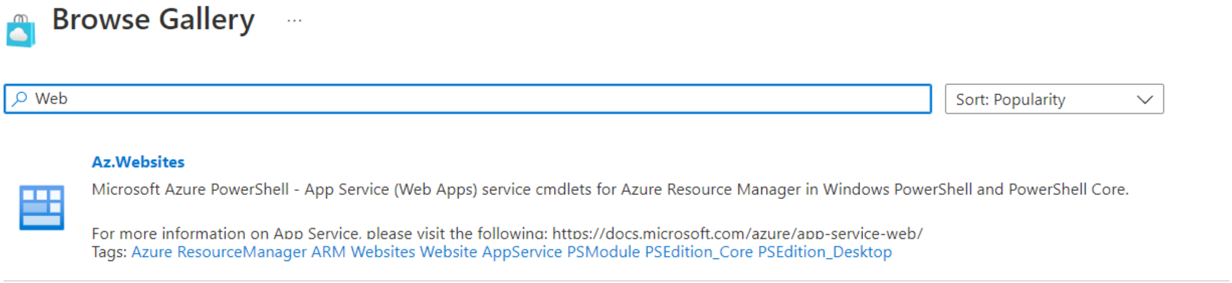
Showing 1 to 100 of 178 records.

Name	Status	Type	Module version	Run
AuditPolicyDsc	Available	Default	1.1.0.0	5.1
Az	Available	Default	8.0.0	5.1
Az.Accounts	Available	Custom	2.9.1	5.1
Az.Advisor	Available	Default	1.1.2	5.1
Az.Aks	Available	Default	4.1.0	5.1
Az.AnalysisServices	Available	Default	1.1.4	5.1
Az.ApiManagement	Available	Default	3.0.0	5.1

From here, add all the necessary modules required in the script for the runbook.

Note: - If the script is written in AzureRM, add those modules, otherwise add the Az modules.

In this case, we have stopped an Azure WebApp so the required module to initiate those commands is Az.Websites

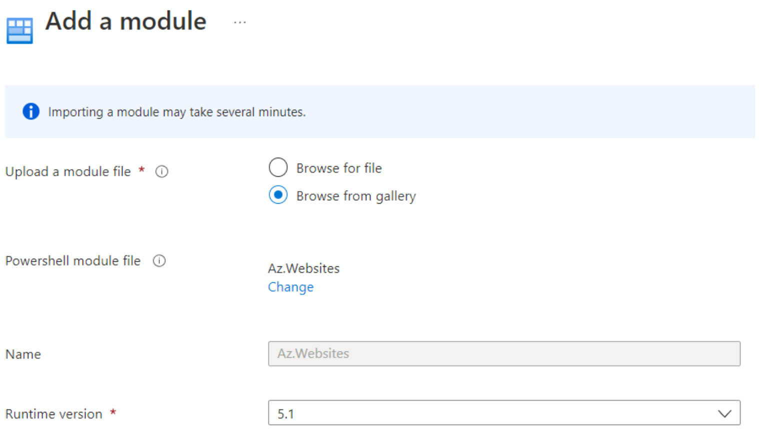


Az.Websites

Microsoft Azure PowerShell - App Service (Web Apps) service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core.

For more information on App Service, please visit the following: https://docs.microsoft.com/azure/app-service-web/
Tags: Azure ResourceManager ARM Websites Website AppService PSModule PSEdition_Core PSEdition_Desktop

Search it in the Gallery, select the module and Import.



Add a module

Importing a module may take several minutes.

Upload a module file * ○ Browse for file ● Browse from gallery

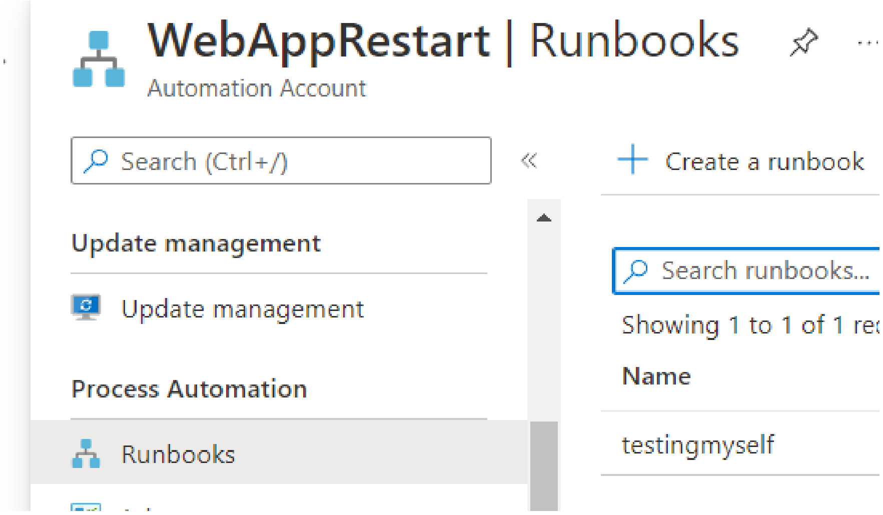
Powershell module file: Az.Websites

Name: Az.Websites

Runtime version *: 5.1

Note: - If you get a dependency error, then you must install the prerequisite module mentioned first.

After importing the required modules, go to the Runbooks section and create a runbook.



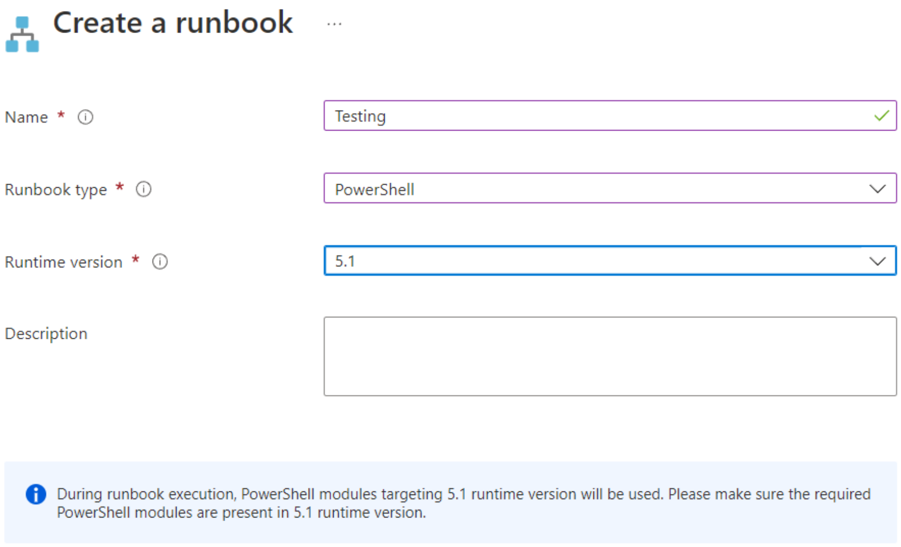
WebAppRestart | Runbooks

Automation Account

Showing 1 to 1 of 1 rec

Name
testingmyself

Note: - Also check the **Import a runbook** option and search if the automation you want to perform already exists as a template, in that case import that.



Name * : Testing

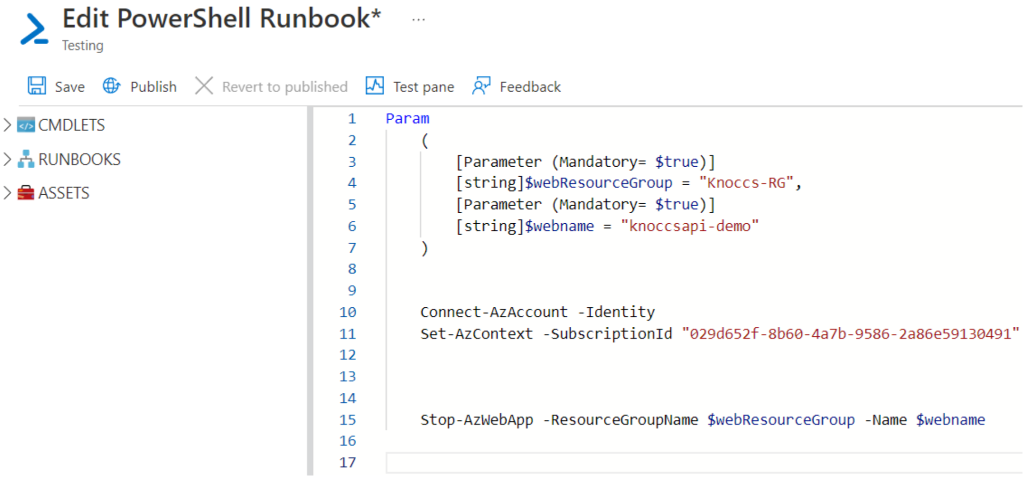
Runbook type * : PowerShell

Runtime version * : 5.1

Description

During runbook execution, PowerShell modules targeting 5.1 runtime version will be used. Please make sure the required PowerShell modules are present in 5.1 runtime version.

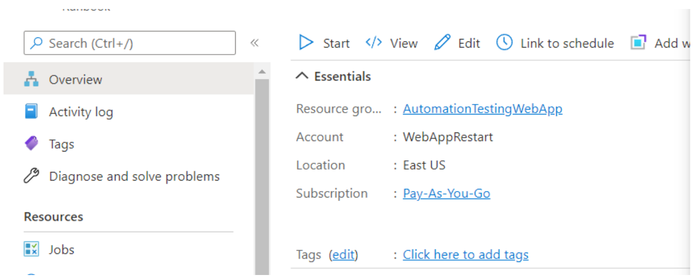
In this example, we are **Stopping** an Azure WebApp



```
Param
    (
        [Parameter (Mandatory= $true)]
        [string]$webResourceGroup = "Knoccs-RG",
        [Parameter (Mandatory= $true)]
        [string]$webname = "knoccsapi-demo"
    )


    Connect-AzAccount -Identity
    Set-AzContext -SubscriptionId "029d652f-8b60-4a7b-9586-2a86e59130491"



    Stop-AzWebApp -ResourceGroupName $webResourceGroup -Name $webname

```

Note: - Click on Test pane if you want to test the script first.

After creating the runbook, click on save and publish it.



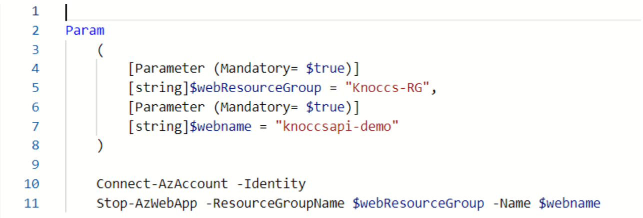
Resource gro... : AutomationTestingWebApp

Account : WebAppRestart

Location : East US

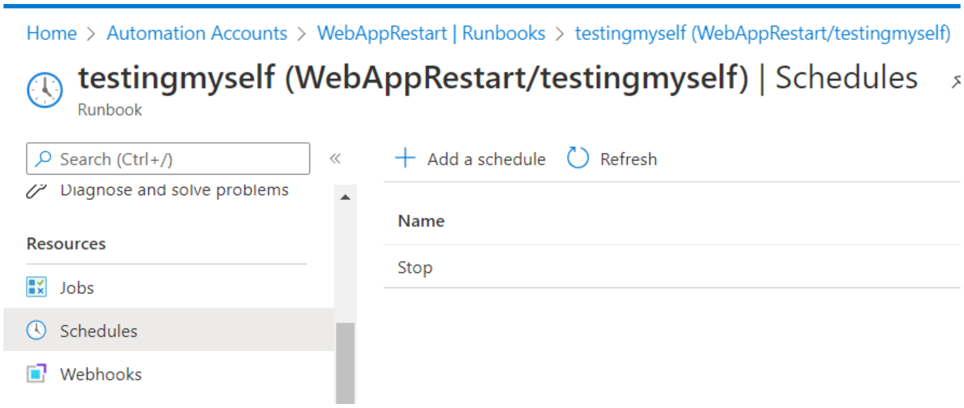
Subscription : Pay-As-You-Go

Tags (edit) : Click here to add tags

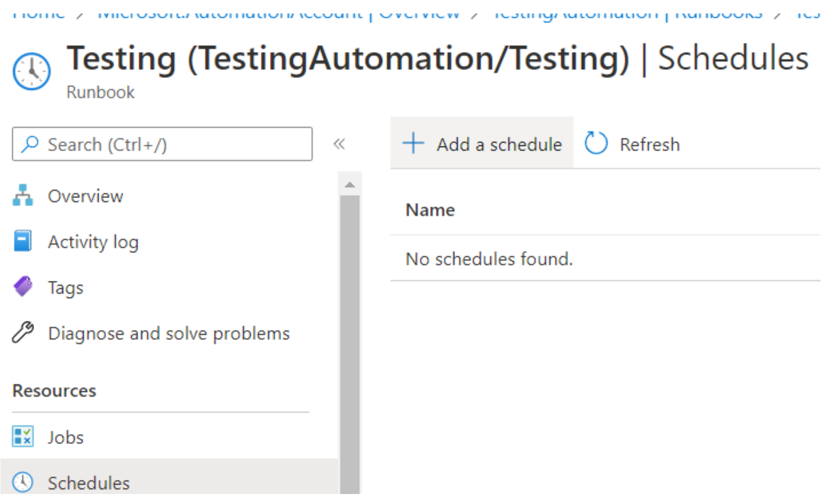


```

Param
    (
        [Parameter (Mandatory= $true)]
        [string]$webResourceGroup = "Knoccs-RG",
        [Parameter (Mandatory= $true)]
        [string]$webname = "knoccsapi-demo"
    )

    Connect-AzAccount -Identity
    Stop-AzWebApp -ResourceGroupName $webResourceGroup -Name $webname
```

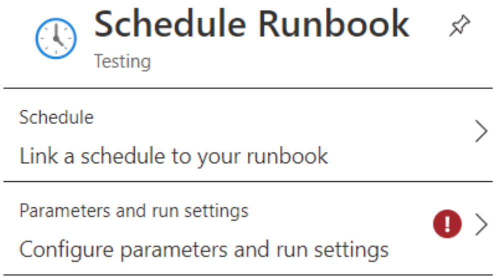
We have successfully, created the Runbook which performs the stop action, now to schedule this Job every day, head over to the Schedule section.


Home > Automation Accounts > WebAppRestart | Runbooks > testingmyself (WebAppRestart/testingmyself)
testingmyself (WebAppRestart/testingmyself) | Schedules
Runbook
Search (Ctrl+/)
«
Diagnose and solve problems
Resources
Jobs
Schedules
Webhooks
Add a schedule
Refresh
Name
Stop

Add the schedule and link your runbook to the schedule to automate the process.


Home > Microsoft.Automation/Account | Overview > TestingAutomation | Runbooks > Tes
Testing (TestingAutomation/Testing) | Schedules
Runbook
Search (Ctrl+/)
«
Overview
Activity log
Tags
Diagnose and solve problems
Resources
Jobs
Schedules
Add a schedule
Refresh
Name
No schedules found.

First click on Link a schedule to link it, then configure the parameters.


Schedule Runbook

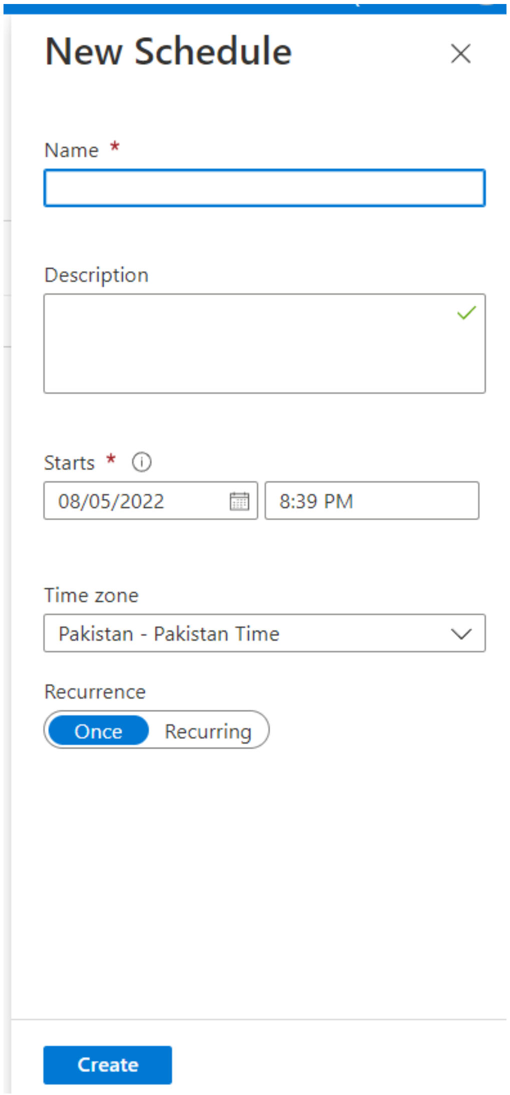
Testing

Schedule

Link a schedule to your runbook

Parameters and run settings

Configure parameters and run settings


New Schedule

Name *

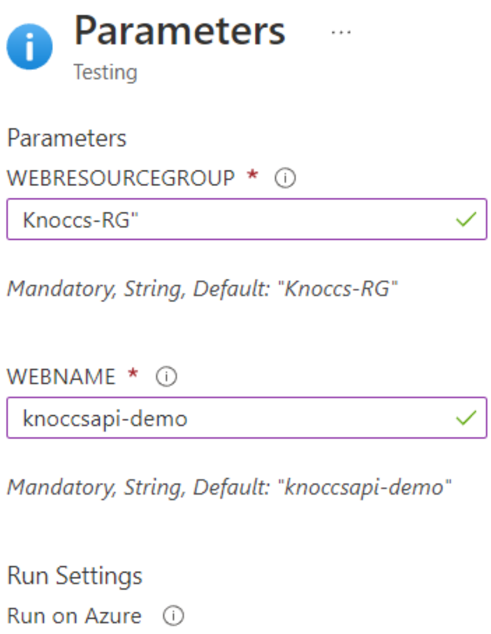
Description

Starts * 08/05/2022 8:39 PM

Time zone: Pakistan - Pakistan Time

Recurrence: Once Recurring

Fill the required details and hit on ok.


Parameters

Testing

Parameters

WEBRESOURCEGROUP *

Knoccs-RG"

Mandatory, String, Default: "Knoccs-RG"

WEBNAME *

knoccsapi-demo

Mandatory, String, Default: "knoccsapi-demo"

Run Settings

Run on Azure

After that, the runbook will run in the mentioned schedule.

IMPORTANT!

To start working on the runbook, Use the following line of code in the beginning of the script in case of **Identity Management Authentication** to set a connection.

```
 Param
         (
             [Parameter (Mandatory= $true)]
             [string]$VmResourceGroup,
             [Parameter (Mandatory= $true)]
             [string]$VmName
         )

         Connect-AzAccount -Identity
         Set-AzContext -SubscriptionId "xxxx-xxxxxx-xxxxxx-xxxxx"
```

In case **of Run-As Account**, use the following line of code in the beginning to set the connection.

```
Param
(
    [Parameter (Mandatory= $true)]
    [string]$VmResourceGroup,
    [Parameter (Mandatory= $true)]
    [string]$VmName
)

$connectionName = "AzureRunAsConnection"

    $servicePrincipalConnection = Get-AutomationConnection -Name $connectionName
    Connect-AzAccount `
        -ServicePrincipal `
        -TenantId $servicePrincipalConnection.TenantId `
        -ApplicationId $servicePrincipalConnection.ApplicationId `
        -CertificateThumbprint $servicePrincipalConnection.CertificateThumbprint
```