

## Module 08 - Hybrid Runbook Worker

### Exercise 1: Prepare the Environment

#### Introduction

This lab requires multiple servers to be built. We will use Bicep - an simpler version of ARM template to deploy the infrastructure.

#### Estimated Time to Complete This Exercise

30 minutes

#### Task Description

You should stop any running machines before executing this.

We will use the Azure Cloud Shell to deploy these machines as it already has the Bicep executable installed.

1. [ ] In a browser - navigate to <https://shell.azure.com> and sign in with your credentials. Ensure that the **PowerShell** shell is selected.
2. [ ] Run the following command in the cloud shell to prep the environment for deployment incase some previous labs were not completed:

```
Register-AzResourceProvider -ProviderNamespace Microsoft.OperationsManagement
```

3. [ ] Upload the files *deploy.ps1* and *azuredeploy.bicep* in C:\Labs\Module8\ to the shell by using the **Upload/Download Files** option.
4. [ ] We have to add a location field in the **deploy.ps1** file. You can choose 1 of 3 different ways to complete this.
  - Use the built in shell editors - either vi or nano (Advanced)
  - Use the code below to replace the value in the file (Medium)

```
sed -i "s/update_location/southeastasia/g" deploy.ps1
```

- Use the build in graphical code editor in Azure Cloud Shell by using the code below (Easy)

```
code .
```

Whichever way you choose - the line in the file should look similar to the below file when complete. If you desire a different location simply update it accordingly.

```
New-AzResourceGroup -Name HybridRunbookWorker -Location "australiaeast" -Force -Verbose
```

## Module 8: Hybrid Runbook Worker

[!note]: For the location of resource group don't use 'eastus'. Make sure VMs from previous lab are deleted or you could hit a limit on the cores per region. Canadacentral is an example of a region that should work. Some other regions will have VM size limitations.

5. [] The Ubuntu machine we build will require an SSH certificate to connect to it. We can generate this in the Cloud Shell by using the code below.

```
ssh-keygen
```

Simply press Enter through all the prompts.

6. [] To deploy the template simply run the command below to create the resource group and complete the deployment. It will take a few minutes to run and you can monitor the process.

```
./deploy.ps1
```

[!note]: If you get the error **Code:PrincipalNotFound** when running the deploy.ps1 script, just run it again.

```
az resource list --resource-group HybridRunbookWorker --output table
```

The output should be similar to the below.

## Module 8: Hybrid Runbook Worker

Name Type	Status	ResourceGroup	Location
-----	-----	-----	-----
hrw-aa Microsoft.Automation/automationAccounts		HybridRunbookWorker	australiaeast
linux-server_OsDisk_1_6b396131e24c4ba5b494b2e247b118d8 Microsoft.Compute/disks		HYBRIDRUNBOOKWORKER	australiaeast
win-server_OsDisk_1_9fb5a1454b444cb082ce9af54260690f Microsoft.Compute/disks		HYBRIDRUNBOOKWORKER	australiaeast
linux-server Microsoft.Compute/virtualMachines		HybridRunbookWorker	australiaeast
linux-server/linuxmma Microsoft.Compute/virtualMachines/extensions		HybridRunbookWorker	australiaeast
win-server Microsoft.Compute/virtualMachines		HybridRunbookWorker	australiaeast
win-server/winmma Microsoft.Compute/virtualMachines/extensions		HybridRunbookWorker	australiaeast
uami-yknzz3t2ulkre Microsoft.ManagedIdentity/userAssignedIdentities		HybridRunbookWorker	australiaeast
linux-nic Microsoft.Network/networkInterfaces		HybridRunbookWorker	australiaeast
win-nic Microsoft.Network/networkInterfaces		HybridRunbookWorker	australiaeast
hrwnsg1 Microsoft.Network/networkSecurityGroups		HybridRunbookWorker	australiaeast
linux-pip Microsoft.Network/publicIPAddresses		HybridRunbookWorker	australiaeast
win-pip Microsoft.Network/publicIPAddresses		HybridRunbookWorker	australiaeast
hrw_vnet Microsoft.Network/virtualNetworks		HybridRunbookWorker	australiaeast
yknzz3 Microsoft.OperationalInsights/workspaces		HybridRunbookWorker	australiaeast
addIntelligencePack Microsoft.Resources/deploymentScripts		HybridRunbookWorker	australiaeast
delayforUAMI Microsoft.Resources/deploymentScripts		HybridRunbookWorker	australiaeast

That is a quick overview of Bicep. Ask your trainer for more information.

## Exercise 2: Register the Hybrid Runbook Workers

### Introduction

The hybrid runbook worker can now be registered to the automation account. Please be aware that images may differ from what is shown in this manual.

### Prerequisites (if applicable)

Exercise 1

### Estimated Time to Complete This Lab

15 minutes

### Task 1: Register the Windows Hybrid Runbook Worker

1. First we will generate a command to register the Hybrid Runbook Worker on the Windows machine. In the Azure Cloud Shell upload the file at C:\Labs\Module8\Scripts\GenerateWinCommand.ps1 . Run this script using the code below - the output will be the command we need to run on the virtual machine.

```
cd ~
pwsh GenerateWinCommand.ps1
```

2. Get the public IP address of the Windows Server - we need to RDP to the machine. Run the code below in the Cloud Shell to get the public IP.

```
az network public-ip show --name win-pip --resource-group hybridrunbookworker -o table
```

3. Use RDP to connect the IP address and log in to the server using the username **aa-admin** and the password **RedDwarf2017** .
4. When you are in the command console type `powershell` to launch PowerShell
5. Copy the output from the command in the Cloud Shell (from the GenerateWinCommand.ps1 file) and paste it in the PowerShell windows in the Windows virtual machine.

[!note]: If you receive an error of unable to find the Powershell command to register the hybrid worker, you just need to wait a few minutes for the Hybrid worker solution to finish registering with the log analytics workspace, which will trigger that Powershell module to be downloaded. To verify that the solution is installed, go to the Log analytics workspace in the Hybridrunbookworker resource group and under general, go to solutions and you should see AzureAutomation listed. Note that this lab creates a new Automation account **Hybridrunbookworker**

6. When the script block completes you can check the Azure Automation Account in the portal → click on Hybrid worker groups and verify that the **Windows** group is present.

User hybrid worker groups [System hybrid worker groups](#)

These are hybrid worker groups that you create and maintain.

GROUP NAME	NUMBER OF WORKERS	LAST REGISTRATION TIME	LAST SEEN TIME
Windows	1	30/08/2019, 10:14 am	4 minutes ago

{400}

### Task 2: Register the Linux Hybrid Runbook Worker

1. First we will generate a command to register the Hybrid Runbook Worker on the Linux machine. In the Azure Cloud Shell upload the file at C:\Labs\Module8\Scripts\GenerateLinuxCommand.ps1 . Run this script using the code below - the output will be the command we need to run on the virtual machine.

```
cd ~
pwsh GenerateLinuxCommand.ps1
```

Make a note of the output of this command because we will use it to install the Hybrid worker role.

## Module 8: Hybrid Runbook Worker

2. [ ] Get the public IP address of the Linux Server - we need to SSH to the machine. Run the code below in the Cloud Shell to get the public IP.

```
az network public-ip show --name linux-pip --resource-group hybridrunbookworker -o table
```

3. [ ] Enter the command below to connect to the Linux machine - replacing the IP address with the public IP obtained from the command above. SSH will use the private key you generated to connect to the machine without the need for a password.

```
ssh aa-admin@13.76.177.80
```

When the connection is successful you should see the command line prompt change to something like below.

```
aa-admin@linux-server:~$
```

4. [ ] Copy the output from the command in the Cloud Shell (from the GenerateLinuxCommand.ps1 file) and paste it in the Cloud Shell at the prompt. You may have to press Enter to run the command.
5. [ ] When the script block completes you can check the Azure Automation Account in the portal → click on Hybrid worker groups and verify that the Linux group is present.

User hybrid worker groups [System hybrid worker groups](#)

These are hybrid worker groups that you create and maintain.

Search to filter items...

GROUP NAME	NUMBER OF WORKERS	LAST REGISTRATION TIME
Linux	1	30/08/2019, 11:01 am
Windows	1	30/08/2019, 10:14 am

{400}

## Exercise 3: Testing the Hybrid Runbook Workers

### Introduction

The hybrid runbook workers can now access and run Runbooks in the automation account.

### Prerequisites (if applicable)

Exercise 1 and 2

### Estimated Time to Complete This Lab

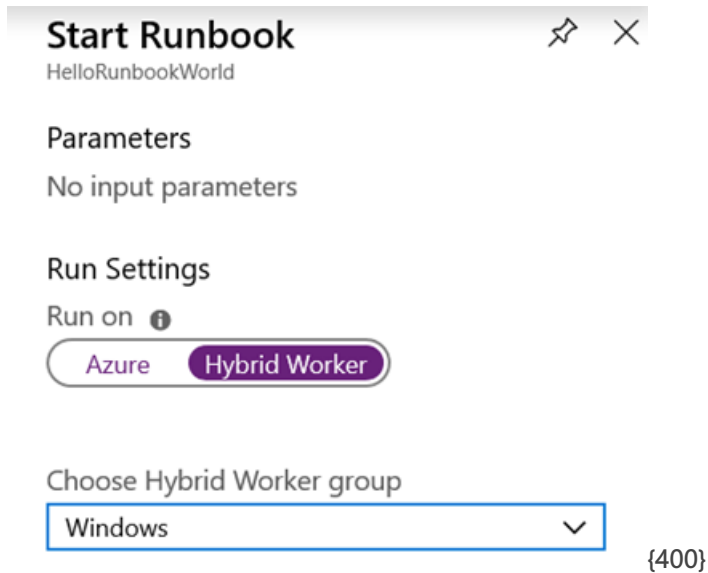
10 minutes

### Task 1: Test the Windows Hybrid Runbook Worker

1. [ ] Add both runbooks in C:\Labs\Module8\RunbookExamples to the Automation Account. If you need help to do this please refer to the lab exercise in Module 3.

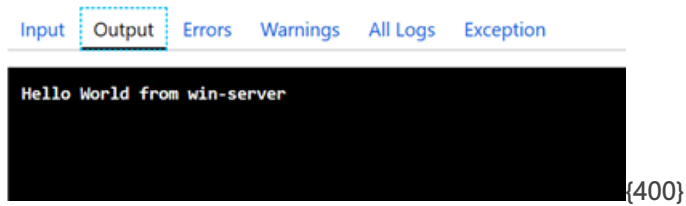
[!note]: If you import the runbooks via the portal you will have to publish them before you can click Start. You can also run both of these tests from the runbook Test Pane.

2. ☐ Select the **HelloRunbookWorld** Runbook and click **Start**. (If the start button is greyed out - ensure the runbook is Published)
3. ☐ Change the **Run Settings** to **Hybrid Worker** > **Windows** and click **OK**.



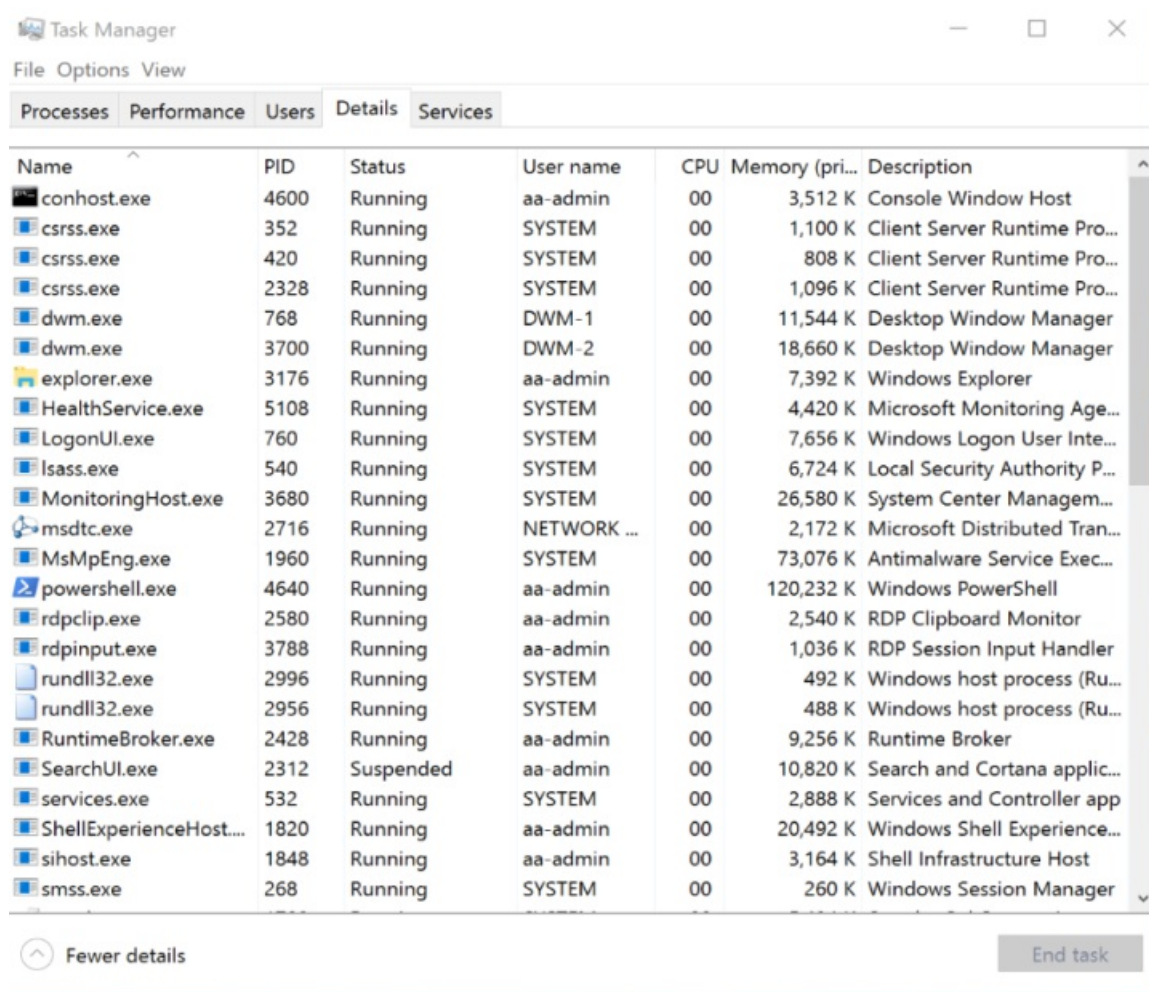
The screenshot shows the 'Start Runbook' dialog for 'HelloRunbookWorld'. It has a title bar with a pin icon and a close button. Under 'Parameters', it says 'No input parameters'. Under 'Run Settings', there is a 'Run on' section with an information icon, showing two buttons: 'Azure' and 'Hybrid Worker', with 'Hybrid Worker' selected. Below this is a 'Choose Hybrid Worker group' section with a dropdown menu showing 'Windows' and a checkmark icon. To the right of the dropdown is the text '{400}'.

4. ☐ Click **Output** and follow the output. It should be similar to below which shows that the Runbook has executed on the Hybrid Runbook Worker:



The screenshot shows the 'Output' tab of the runbook execution pane. The tabs are 'Input', 'Output', 'Errors', 'Warnings', 'All Logs', and 'Exception'. The 'Output' tab is selected and highlighted with a dashed blue border. Below the tabs is a black terminal window with the text 'Hello World from win-server' in white. To the right of the terminal window is the text '{400}'.

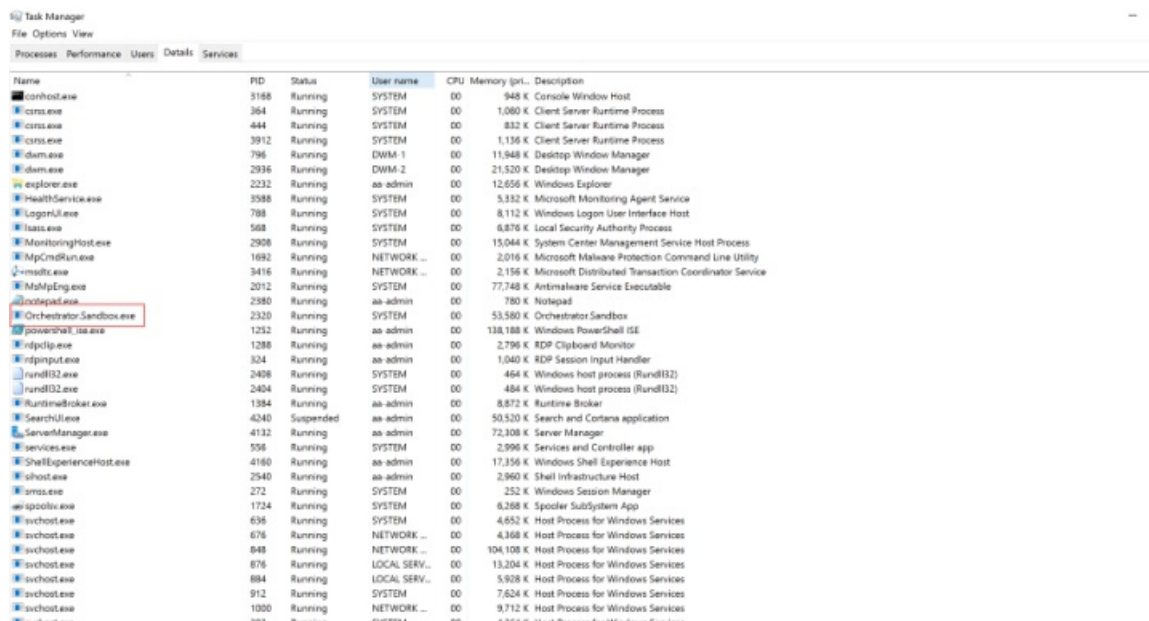
5. ☐ In the Portal, navigate to **Virtual Machines**. Click on **win-server**. Click **Connect** to logon to the desktop with the below details:
  - **User Name:** aa-admin
  - **Password:** RedDwarf2017
6. ☐ At the command prompt enter **taskmgr** and press Enter. Select **More Details** and choose the **Details** tab.



{400}

7. In the Azure portal, start the runbook again and ensure it is set to run on the hybrid worker.

8. Quickly switch back to task manager on win-server and observe the OrchestratorSandbox.exe process. This is the hybrid runbook worker process picking up the job and executing it.

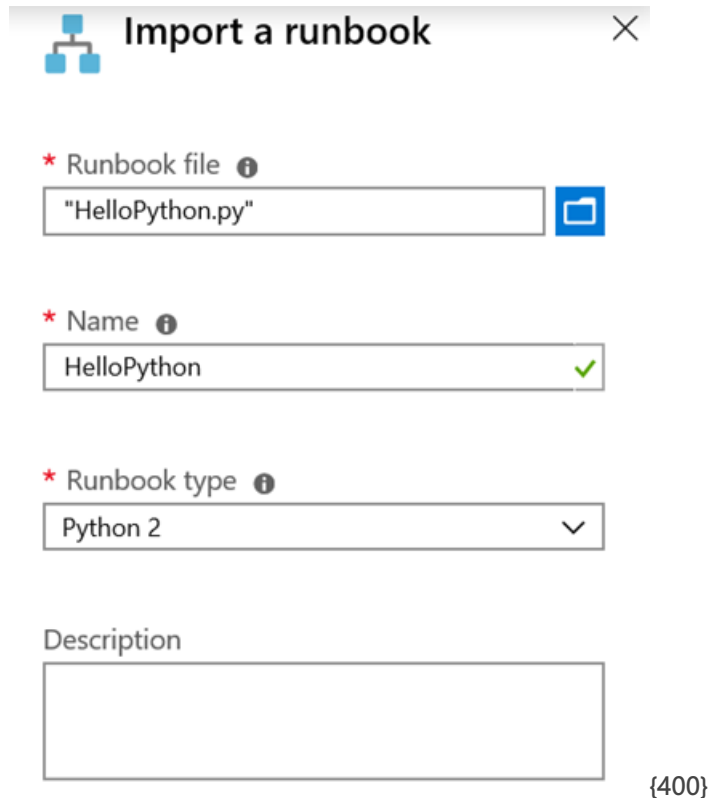


{400}

## Task 2: Test the Linux Hybrid Runbook Worker

## Module 8: Hybrid Runbook Worker

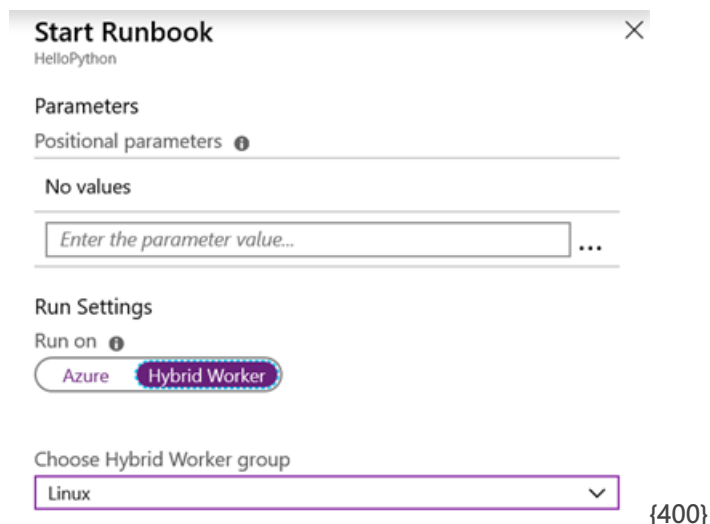
1. In the Microsoft Azure portal, select the hybrid automation account and click on Runbooks.
2. Click on **Import a Runbook**.
3. Click on the Runbook File icon and navigate to C:\Labs\Module8\RunbookExamples. Select the HelloPython.py file and click open.
4. Ensure the Runbook Type is "Python 2" and click **Create**.



The screenshot shows the 'Import a runbook' dialog box. It has a title bar with a tree icon and a close button. The form contains the following fields:

- Runbook file**: A text input field containing 'HelloPython.py' and a file explorer icon.
- Name**: A text input field containing 'HelloPython' with a green checkmark icon.
- Runbook type**: A dropdown menu showing 'Python 2'.
- Description**: A large text area with a placeholder '{400}'.

5. When the editor pane opens click on **Publish**.
6. Select the **HelloPython** Runbook and click **Start**.
7. Change the **Run Settings** to **Hybrid Worker > Linux** and click **OK**.



The screenshot shows the 'Start Runbook' dialog box for the 'HelloPython' runbook. It has a title bar with a close button. The form contains the following sections:

- Parameters**: A section with a sub-header 'Positional parameters' and a text input field containing 'Enter the parameter value...' with a three-dot menu icon.
- Run Settings**: A section with a sub-header 'Run on' and two buttons: 'Azure' and 'Hybrid Worker' (which is selected and highlighted with a blue border).
- Choose Hybrid Worker group**: A dropdown menu showing 'Linux'.

8. Click **Output** and follow the output. It should be similar to below which shows that the Runbook has



## Module 8: Hybrid Runbook Worker

executed on the Hybrid Runbook Worker:

[Input](#) [Output](#) [Errors](#) [Warnings](#) [All Logs](#) [Exception](#)

```
Hello from linux-server
```

{400}