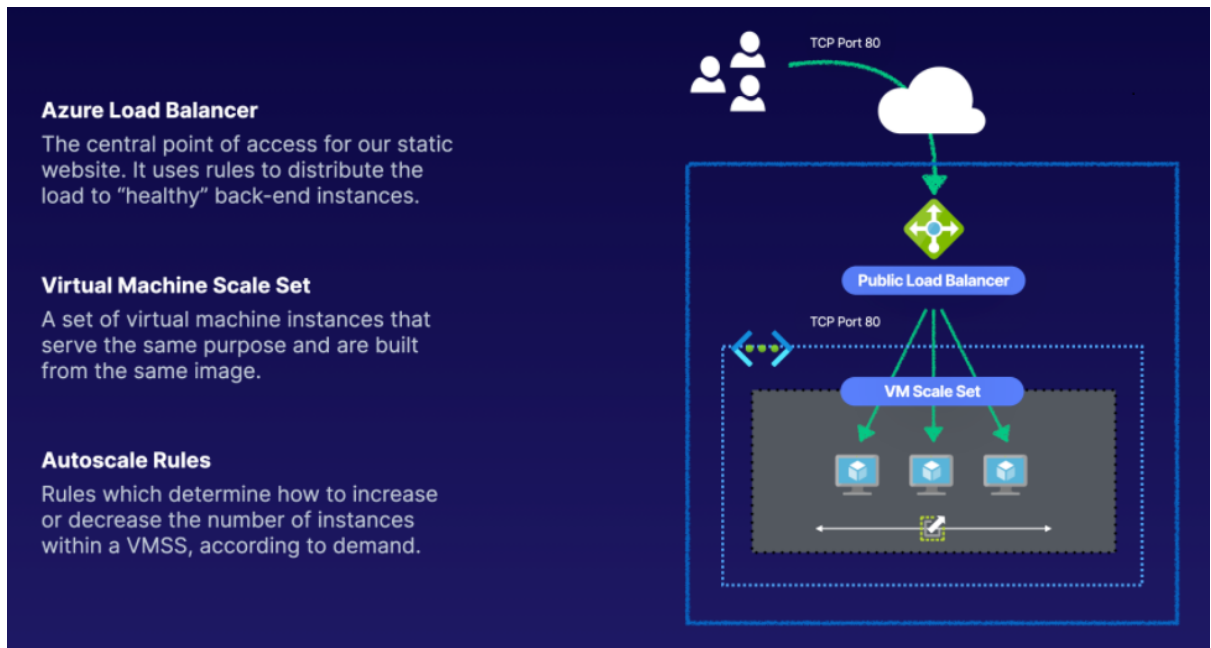


Configure a Load Balanced VM Scale Set in Azure

The goal of this lab is to gain knowledge about:

- Dynamic and elastic compute using a VMSS
- High availability using a load balancer



Create a VM Scale Set with Load Balancing

- 1- Click Home in the upper left.
- 2- Click Create a resource.
- 3- Search for and select Virtual machine scale set.
- 4- Click Create.

Microsoft Azure

Search resources, services, and docs (G+)

[Home](#) >
[1-bd69ebd9-playground-sandbox](#) >
[Create a resource](#) >
[Virtual machine scale set](#) >

Create a virtual machine scale set

Load balancing

You can place this virtual machine scale set in the backend pool of an existing Azure load balancing solution. [Learn more](#)

Use a load balancer ☒

Load balancing settings

- Application Gateway** is an HTTP/HTTPS web traffic load balancer with URL-based routing, SSL termination, session persistence, and web application firewall. [Learn more about Application Gateway](#)
- Azure Load Balancer** supports all TCP/UDP network traffic, port-forwarding, and outbound flows. [Learn more about Azure Load Balancer](#)

Load balancing options * ⓘ

Azure load balancer

Select a load balancer * ⓘ

(new) vmsscaleset01-lb

[Create new](#)

Select a backend pool * ⓘ

(new) bepool

[Create new](#)

Review + create

< Previous

Next : Scaling >

5- Fill out the Basics page:

- Subscription: The default is already selected
- Resource group: Select the existing one in the dropdown
- Virtual machine scale set name: vmsscaleset01
- Region: West US
- Availability zone: None
- Operating system disk image: Ubuntu 18.04 LTS
- Azure Spot instance: No
- Instance size: Standard_AO (Click Select size, search for A0, select it, and click Select.)
- Authentication type: Password
- Username: azureuser
- Password: Use the password you have been provided for the lab

6- Click on Next: Disks >.

7- Fill out the Disks page:

- OS disk type: Standard SSD
- Encryption type: Default

8- Click on Next: Networking.

9- Fill out the Networking page:

- Virtual network: vnet1 (Use the existing vnet available in the resource group provided for you).
- Click on the edit icon for the Network interface section:
 - Name: leave as the default
 - Subnet: subnet1
 - NIC network security group: None (we already have one attached to the subnet)
 - Public IP address: Enabled
 - Click OK.
- Use a load balancer: Yes
- Load balancing options: Azure load balancer
- Leave the defaults as-is.

10- Click Next: Scaling.

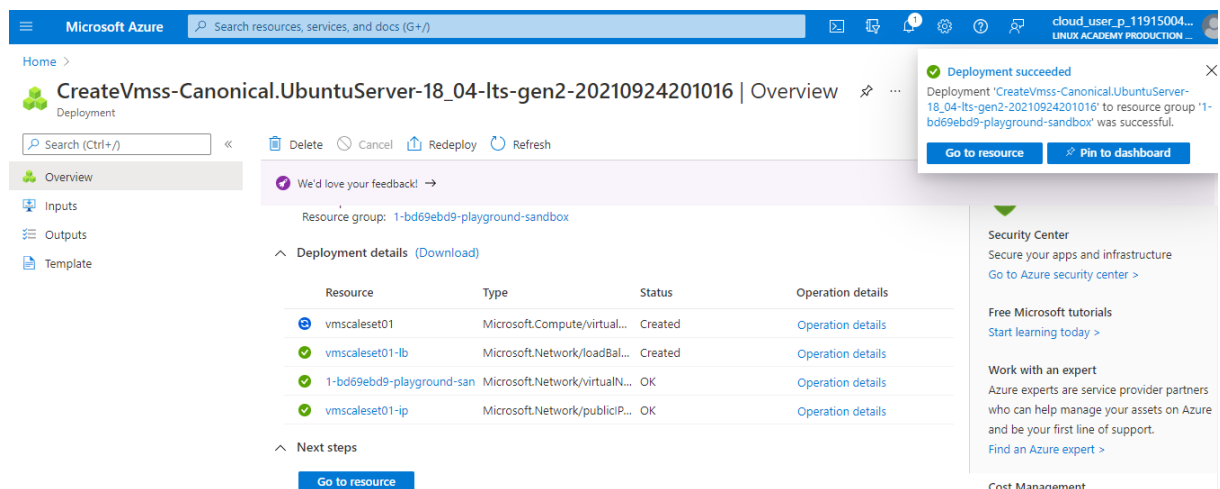
11- Fill out Scaling:

- Initial instance count: 0 (we will change this later)
- Scaling policy: Manual (we will change this later) . Click Review + create.

12- Click Create once validation is passed.

13- Navigate to the newly created VM Scale Set.

Note: There can be an error with the Azure Portal that results in the Manual Scale being set to 2 instead of 0. Make sure you change this to 0 before proceeding.



The screenshot shows the Azure portal interface for a deployment. The deployment is titled 'CreateVmss-Canonical.UbuntuServer-18_04-Its-gen2-20210924201016' and is in the 'Overview' tab. A notification at the top right states 'Deployment succeeded'. Below the notification, a table lists the resources created:

Resource	Type	Status	Operation details
vmsscaleset01	Microsoft.Compute/virtual...	Created	Operation details
vmsscaleset01-lb	Microsoft.Network/loadBal...	Created	Operation details
1-bd69ebd9-playground-san	Microsoft.Network/virtualN...	OK	Operation details
vmsscaleset01-ip	Microsoft.Network/publicIP...	OK	Operation details

Below the table, there is a 'Next steps' section with a 'Go to resource' button. On the right side of the portal, there are links to 'Security Center', 'Free Microsoft tutorials', and 'Work with an expert'.

Configure a Custom Script Extension

1- Download the file to use for your VM Extension here:

<https://raw.githubusercontent.com/linuxacademy/content-az300-lbvmyscaleset/master/script.sh>

2- Navigate to Storage Accounts.

3- Select the storage account that was already created for you.

4- Click Containers.

5- Click +Create:

- Name: scripts
- Public access level: Private
- Click Create

6- Browse to the new container (note you may need to click refresh if it doesn't appear).

7- Click Upload.

8- Select the script.sh file you downloaded earlier.

9- Click Upload.

The screenshot shows the 'Install extension' page in the Microsoft Azure portal. The breadcrumb navigation is: Home > vmyscale01 > New resource > Custom Script For Linux > Install extension. The page has a close button (X) in the top right. There are two input fields: 'Script files' with a dropdown menu showing 'Select a file' and a 'Browse' button, and 'Command' with a dropdown menu showing 'sh script.sh'.

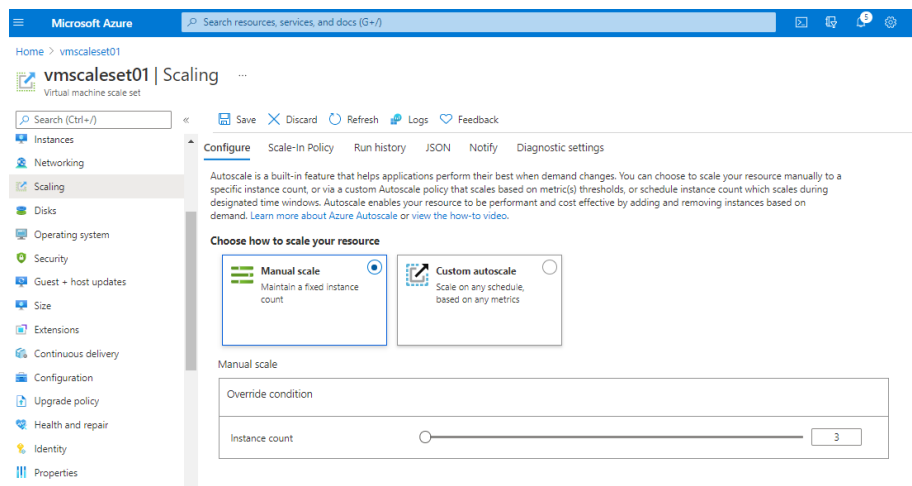
The screenshot shows the 'script' container page in the Microsoft Azure portal. The breadcrumb navigation is: Home > vmyscale01 > New resource > Custom Script For Linux > Install extension > Storage accounts > Containers > script. The page has an 'Upload' button and a 'Refresh' button. Below these, there is an 'Authentication method' section with a link to 'Access key (Switch to Azure AD User Account)' and a 'Location' section with a search bar. A table with columns 'Name', 'Modified', 'Access tier', and 'Blob type' is shown, with 'No results' displayed. A 'Select' button is at the bottom left. On the right, an 'Upload blob' sidebar is open, showing a 'Files' section with a text input containing 'script.sh' and a 'Browse' button. There is an 'Overwrite if files already exist' checkbox and an 'Advanced' section. An 'Upload' button is at the bottom of the sidebar.

Follow the steps below to configure the Custom Script Extension for the VM Scale Set. Ensure there are currently no instances running, or they will not have the appropriate configuration (without being upgraded).

1. Open the recently created VM Scale Set.
2. Click on Extensions within the Settings menu.
3. Click on the + Add icon.
4. Choose Custom Script For Linux.
5. Click Create.
6. Click on Browse to select a script file.
7. Navigate to the scripts container you created earlier.
8. Tick the checkbox beside the script.sh file.
9. Click on Select.
10. Click OK.

Configure Autoscale

- 1- Navigate to Settings then Scaling.
- 2- Click on the Manual autoscale option.
- 3- Set the instance count to 3 and check the number of instances created



Configure Custom Autoscale

1. Navigate to Settings then Scaling.
2. Click on the Custom autoscale option.
3. Enter a name for the autoscale: autoscaletesting01.
4. Leave the resource group as-is (select your resource group if it is not pre-populated).
5. Configure the Default autoscale profile:
 - Leave Scale based on a metric selected.

- Click Add a rule, to add a rule for scaling out:
 - ✓ Leave everything but Duration as their default.
 - ✓ Change Duration to 10.
- Click Add a rule again to create a rule for scaling in:
 - ✓ Leave all but the designated fields as their defaults
 - ✓ Change Operator to Less than.
 - ✓ Change Metric threshold to trigger scale action to 10.
 - ✓ Change Duration to 5.
 - ✓ Change section Operation section to Decrease count by.

Scale rule

Percentage CPU (Average)

Operator * Less than Metric threshold to trigger scale action * 10

Duration (minutes) * 5

Time grain (minutes) 1 Time grain statistic * Average

Action

Operation * Decrease count by Cool down (minutes) * 5

Instance count * 1

Add

- Configure instance limits: minimum of 1, maximum of 3, and default of 1.
- Click Save.

vmsscaleset01 | Scaling

Scale mode: ☒ Scale based on a metric ☐ Scale to a specific instance count

Rules

Scale out

When vmsscaleset01 (Average) Percentage CPU > 70 Increase count by 1

Scale in

When vmsscaleset01 (Average) Percentage CPU > 10 Decrease count by 1

+ Add a rule

Instance limits

Minimum 2 Maximum 4 Default 3

Schedule

This scale condition is executed when none of the other scale condition(s) match

Test the Scaling

1. Go to the Instances section.
2. Check that our instance status is Running.
3. Click on **Overview**.
4. Copy the Public IP address and paste it into a new browser window. We'll see our page is working correctly.
5. Go back to Instances and select the one we created.
6. Copy the public IP.
7. Open a terminal login via SSH using the public IP you copied and the **azureuser**:

ssh azureuser@<VM_PUBLIC_IP>
8. At the prompt, enter the password provided on the lab page.
9. Run: **stress --cpu 1**
10. Go back to the Azure Portal and go to the Overview page.
11. Watch the monitoring of the VMSS and confirm that your instance count increases to confirm that we set up the scaling correctly.
12. Select Scaling from the sidebar and then the Run history tab. Here, choose ScaleupResult to show that our scaling succeeded.