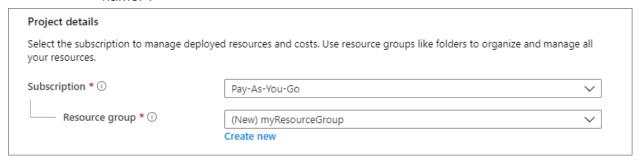
Create a Linux virtual machine in the Azure portal

- 1. Sign in to Azure
 - a. Sign in to the Azure portal.
- 2. Create a virtual machine
 - a. Enter virtual machines in the search.
 - b. Under Services, select Virtual machines.
 - c. On the Virtual Machines page, select "Create" and then "Virtual Machine." The "Create a Virtual Machine" page opens.
 - d. In the Basics tab, under Project details, make sure the correct subscription is selected and then choose to Create new resource group. Enter [Name]-rg for the name.*.



e. Under Instance details, enter myVM for the Virtual machine name, and choose Ubuntu Server 22.04 LTS - Gen2 for your Image. Leave the other defaults. The default size and pricing is only shown as an example. Size availability and pricing are dependent on your region and subscription.

Instance details Virtual machine name * (i) demolinux-vm Region * ① (Europe) West Europe Deploy to an Azure Extended Zone Availability options ① Availability zone Self-selected zone Zone options ① Choose up to 3 availability zones, one VM per zone Azure-selected zone (Preview) Let Azure assign the best zone for your needs Availability zone * ① Zone 1 🕜 You can now select multiple zones. Selecting multiple zones will create one VM per zone. Learn more 🗹 Security type (i) Trusted launch virtual machines Configure security features Image * ① Ubuntu Server 24.04 LTS - x64 Gen2 See all images | Configure VM generation Arm64 VM architecture ① x64 Run with Azure Spot discount ① Size * (i) Standard_DC1s_v2 - 1 vcpu, 4 GiB memory (\$83.95/month) See all sizes Enable Hibernation (1) **Note:** Some users will now see the option to create VMs in multiple zones. Availability zone * (i) Zones 1 🕜 You can now select multiple zones. Selecting multiple zones will create one VM

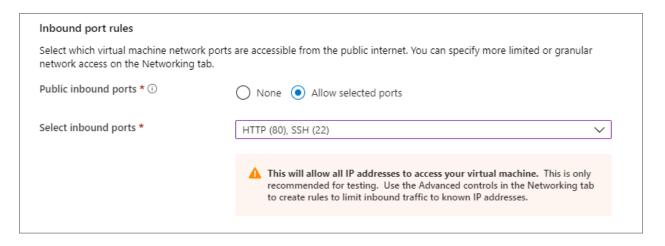
3. Under the Administrator account, select SSH public key.

per zone.

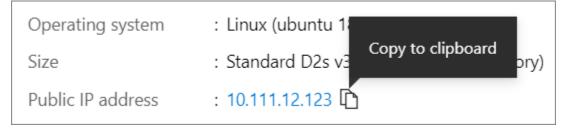
- 4. In Username, enter azureuser.
- 5. For the SSH public key source, leave the default of Generate new key pair, and then enter myKey for the Key pair name.



6. Under Inbound port rules > Public inbound ports, choose Allow selected ports and then select SSH (22) and HTTP (80) from the drop-down.



- 7. Leave the remaining defaults and then select the Review + create button at the bottom of the page.
- 8. On the Create a virtual machine page, you can see the details about the VM you are about to create. When you are ready, select Create.
- 9. When the Generate new key pair window opens, select Download Private Key and create a resource. Your key file will be downloaded as myKey.pem. Make sure you know where the .pem file was downloaded; you will need the path to it in the next step.
- 10. When the deployment is finished, select Go to resource.
- 11. On the page for your new VM, select the public IP address and copy it to your clipboard.



Connect to the virtual machine

- 1. Create an SSH connection with the VM.
 - a. If you are on a Mac or Linux machine, open a Bash prompt and set read-only permission on the .pem file using

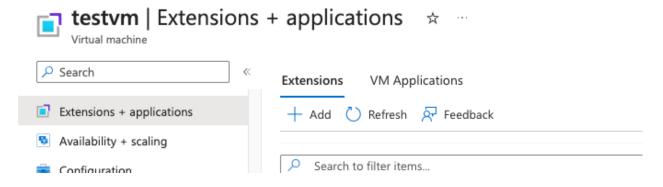
chmod 400 ~/Downloads/myKey.pem.

- b. If you are on a Windows machine, open a PowerShell prompt.
- 2. At your prompt, open an SSH connection to your virtual machine. Replace the IP address with the one from your VM, and replace the path to the .pem with the path to where the key file was downloaded.

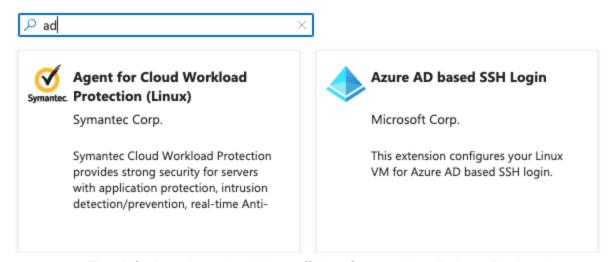
ssh -i ~/Downloads/myKey.pem azureuser@10.111.12.123

Install AADLoginForLinux Extension

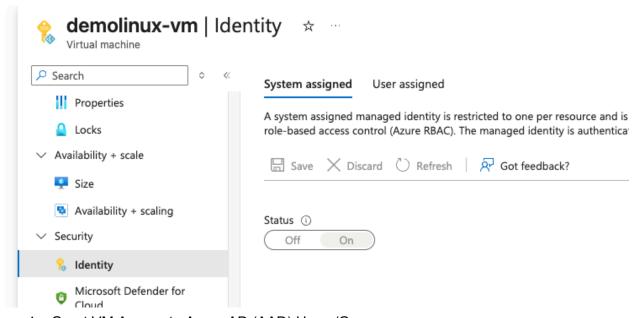
- 1. Navigate to Your VM:
 - a. In the left sidebar, click on "Virtual Machines".
 - b. Find and click on the name of the virtual machine where you want to install the extension.
- Add the AADLoginForLinux Extension:
 - a. In the VM's pane, under the "Settings" section, click on "Extensions".



- b. Click on "+ Add" at the top of the page to add a new extension.
- c. In the search box, type **Azure AD based SSH Login** to filter the list.
- d. From the results, select **Azure AD based SSH Login** by Microsoft and click "Create".



- e. The default settings should be sufficient for most installations. Review the settings, and if everything looks good, click "OK".
- 3. The virtual machine needs a managed identity to communicate with Entra ID
 - a. On the virtual machine menu
 - b. Go to Security->Identity
 - c. On the System Assigned set status to On
 - d. Save



- 4. Grant VM Access to Azure AD (AAD) Users/Groups:
 - a. This step is important as installing the extension alone doesn't grant any AAD user access to the VM. You need to explicitly provide access:
 - b. While still in the VM's settings, click on "Access Control(IAM)" from the sidebar.
 - c. Click on Role assignments
 - i. For individual users:

- Click "Add user", and then search for and select the Azure AD user. Assign a role (like "Virtual Machine Administrator Login") and set the type as "Azure AD user". Click "Save".
- ii. For groups:
 - Click "Add group", and then search for and select the Azure AD group. Assign a role (like "Virtual Machine Administrator Login") and set the type as "Azure AD group". Click "Save".

Add role assignment —

Role	Members	Conditions	Review + assign		
Selected role		Virtual Machine Administrator Login			
Assign access to		User, group, or service principal Managed identity			
Members		+ Select members			
		Name		Object ID	Туре
		Dorin Hus	eras	b86bb32c-1e9a-4aa5-a	af4e-b985061fa066 User

- 5. Restart the VM
- 6. SSH Using AAD Credentials:
 - a. With the extension installed and access granted:
 - b. AAD users or members of the AAD groups you've granted access to can SSH into the VM using:

```
az login -t TENANT_ID
sudo az ssh vm --ip IP.ADR.FROM.PORTAL
```

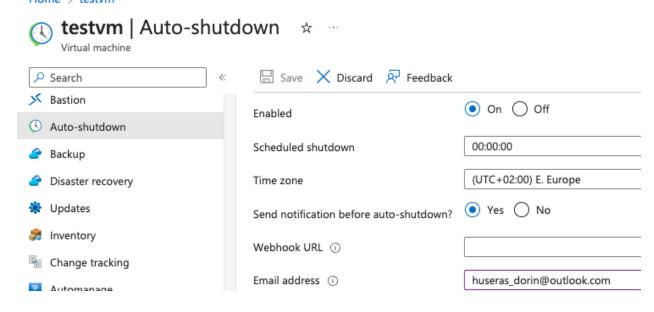
Auto-shutdown

If the VM is still needed, Azure provides an Auto-shutdown feature for virtual machines to help manage costs and ensure you are not billed for unused resources.

- 1. On the Operations section for the VM, select the Auto-shutdown option.
 - a. A page will open where you can configure the auto-shutdown time. Select the On option to enable and then set a time that works for you.

2. Once you have set the time, select Save at the top to enable your Auto-shutdown configuration.

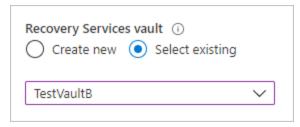
Home > testvm



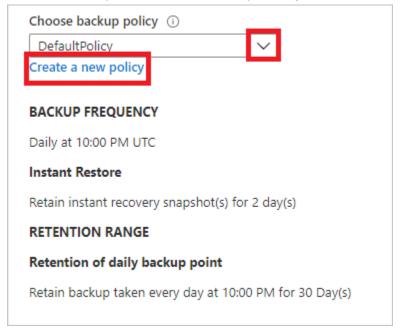
Backup an Azure VM from the VM settings

Azure Backup provides independent and isolated backups to guard against unintended destruction of the data on your VMs. Backups are stored in a Recovery Services vault with built-in management of recovery points. Configuration and scaling are simple, backups are optimized, and you can easily restore as needed. You can back up Azure VMs using a couple of methods:

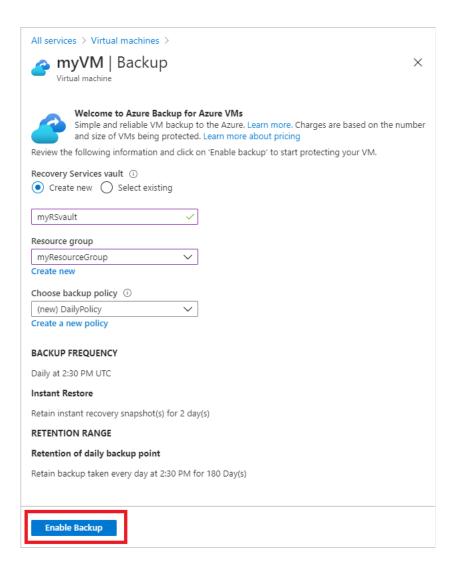
- 1. Sign in to the Azure portal.
 - a. Select all services and in the Filter, type Virtual machines, and then select Virtual machines.
 - b. From the list of VMs, select the VM you want to back up.
- 2. On the VM menu, select Backup.
- 3. In Recovery Services vault, do the following:
 - a. If you already have a vault, select Select existing, and select a vault.
 - b. If you don't have a vault, select Create new. Specify a name for the vault. It's created in the same region and resource group as the VM. You can't modify these settings when you enable backup directly from the VM settings.



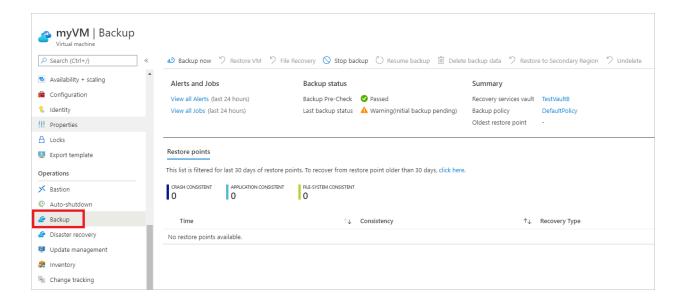
- 4. In Choose backup policy, do one of the following:
 - a. Leave the default policy. This backs up the VM once a day at the time specified, and retains backups in the vault for 30 days.
 - b. Select an existing backup policy if you have one.
 - c. Create a new policy, and define the policy settings.



5. Select Enable Backup. This associates the backup policy with the VM.



- 6. You can track the configuration progress in the portal notifications.
- 7. After the job completes, in the VM menu, select Backup. The page shows backup status for the VM, information about recovery points, jobs running, and alerts issued.



- 8. After enabling backup, an initial backup will run. You can start the initial backup immediately or wait until it starts in accordance with the backup schedule.
 - a. Until the initial backup completes, the Last backup status shows as Warning (Initial backup pending).
 - b. To see when the next scheduled backup will run, select the backup policy name.

Install a Web server

1. To see your VM in action, install the NGINX web server. From your SSH session, update your package sources and then install the latest NGINX package.

```
sudo apt-get -y update
sudo apt-get -y install nginx
```

- 2. When done, type exit to leave the SSH session.
- 3. View the web server in action
 - a. Use a web browser of your choice to view the default NGINX welcome page. Type the public IP address of the VM as the web address. The public IP address can be found on the VM overview page or as part of the SSH connection string you used earlier.

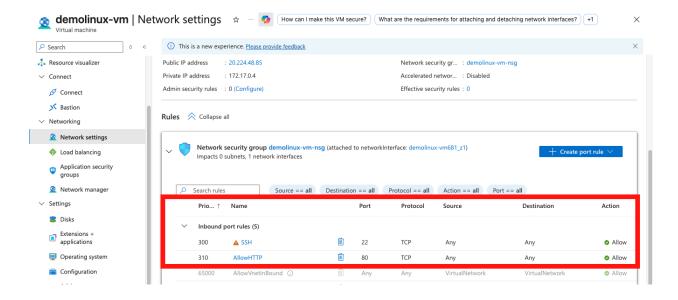
Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to <u>nginx.org</u>. Commercial support is available at <u>nginx.com</u>.

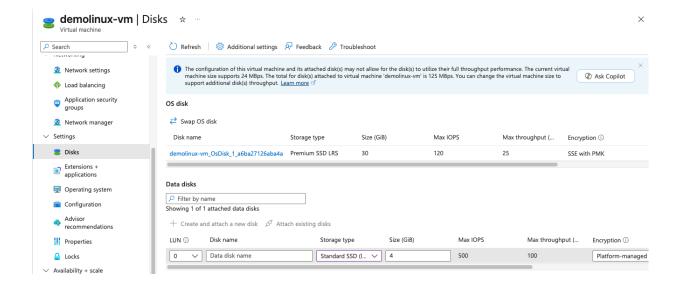
Thank you for using nginx.

9. Make sure you allow access to port 80 from the internet.



Create and attach a data disk

- 1. Go to your VM
 - a. In the Azure Portal, open Virtual Machines.
 - b. Select the Linux VM where you want to attach the disk.
- 2. Open the Disks blade
 - a. In the VM's menu (left-hand side), click Disks.
- 3. Add a data disk
 - a. Click + Add data disk.
 - b. You now have two options:
 - Create disk → Azure will let you create a new managed disk (you choose size, performance tier, etc.).
 - d. Attach existing disk \rightarrow pick one of your already-created managed disks.
- 4. Save changes
 - a. After choosing/creating the disk, click Save at the top.
 - b. Azure will hot-attach the disk to your VM.



Mount the disk

- 1. SSH into your VM.
- 2. Run:

- a. \rightarrow You'll see the new disk (likely /dev/sdc).
- 3. Partition & format:

```
sudo fdisk /dev/sdc
  - When inside fdisk /dev/sdc:
  - Press n → new partition
  - Choose p (primary).
  - Partition number: 1.
  - First sector: press Enter (default).
  - Last sector: press Enter (use entire disk).
  - Press w → write changes and exit.
sudo mkfs -t ext4 /dev/sdc1
sudo mkdir /datadisk
```

```
sudo mount /dev/sdc1 /datadisk

#check it out
df -h | grep datadisk
```

```
$ sudo mkdir /datadisk

$ sudo mount /dev/sdc1 /datadisk

$ sudo su

|root@demolinux-vm:/home/dorin.huseras# cd /

|root@demolinux-vm:/# ls

bin boot dev home lib.usr-is-merged lost+found mnt proc run sbin.usr-is-merged srv tmp var

bin.usr-is-merged datadisk etc lib lib64 media opt root sbin snap sys usr

root@demolinux-vm:/#
```

Restore from backup

- 1. Go to Recovery Services vaults → select the vault where your VM is backed up.
- 2. In the left menu, click Backup items → Azure Virtual Machine.
- 3. Find and click your VM.
- 4. Click Restore VM.
- 5. Choose a restore point (date/time snapshot).
- 6. Choose a restore type:
 - a. Create new VM → Azure deploys a brand-new VM from the backup.
 - b. Replace existing → overwrites the current VM (careful, this will stop the existing VM).
- 7. Pick the resource group, network, and storage settings for the restore.
- 8. Click Restore.





Create Virtual Machine Scale Set

- 1. Go to Azure Portal \rightarrow Virtual Machine Scale Sets \rightarrow + Create.
- Basics tab:
 - a. Subscription: select your subscription.
 - b. Resource group: create new → rg-vmss-demo.
 - c. Scale set name: vmss-demo.
 - d. Region: West Europe.

- e. Availability zone: Zones 1, 2.f. Orchestration mode: Flexible.
- g. Security type: Trusted launch virtual machines.
- h. Scaling: choose Autoscaling.

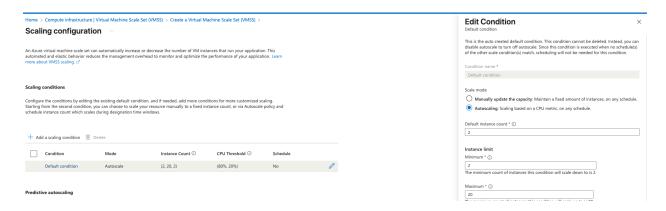
Create a Virtual Machine Scale Set (VMSS)

⚠ Changing Basic options may reset selecti	ons you have made. Review all options prior to creating the virtual machine.				
Virtual machine scale set name *	dorin-vmss 🗸				
Region *	(Europe) West Europe Deploy to an Azure Extended Zone				
Availability zone ①	Zones 1, 2				
Orchestration					
A scale set has a "scale set model" that defines the attributes of virtual machine instances (size, number of data disks, etc). As the number of instances in the scale set changes, new instances are added based on the scale set model. Learn more about the scale set model 🗗					
Orchestration mode * ③	 Flexible: achieve high availability at scale with identical or multiple virtual machine types 				
	Uniform: optimized for large scale stateless workloads				
Security type ①	Trusted launch virtual machines				
	Configure security features				
Scaling					
Scaling mode ①	Manually update the capacity: Maintain a fixed amount of instances.				
	Autoscaling: Scaling based on a CPU metric, on any schedule.				
	No scaling profile: manual attach virtual machines after deployment				
Scaling configuration	Scaling configuration Scaling condition count: 1 Predictive autoscaling: Disabled Diagnostic logs: Disabled Scale-in policy: Default Force delete: Disabled Configure Select configure to review all scaling options prior to creating the virtual machine scale set.				

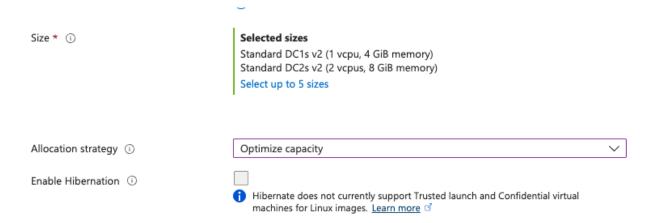
3. Set scaling configuration

- a. Default instance count: 2. (Start with 2 VMs running.)
- b. Minimum instances: 2. (Never go below 2 for availability.)

- c. Maximum instances: 20. (Scale out to handle heavy traffic.)
- d. Scale out rule: Add 1 VM if CPU > 80%. (More servers when traffic is high.)
- e. Scale in rule: Remove 1 VM if CPU < 20%. (Save costs when traffic is low.)
- f. Scale-in policy: Default (balances across zones). (Removes the "safest" VM first.)



- 4. Image: Ubuntu Server 24.04 LTS (x64). (OS template used for all VMs.)
- 5. VM size: Standard B2s (or DC1s). (Defines CPU + RAM per VM.)
- 6. Authentication type: SSH public key. (Secure login method for Linux VMs.)
 - a. Username: azureuser. (Default admin account on the VM.)
- 7. Disks: Only OS disk. (No extra data disks, keeps it simple.)
- 8. Spot instances: Disabled. (Prevents random eviction ensures stability.)



9. A network load balancer is required

- a. Without a load balancer, every VM in your scale set would need its own public IP. With autoscaling, new VMs get created and deleted meaning their IPs keep changing. That makes it almost impossible for clients to know *where* to connect.
- b. The load balancer makes sure requests are spread across all available VMs (round robin or by rules). This prevents one VM from being overloaded while others sit idle.
- c. Load balancing options: Azure Load Balancer. (Distributes traffic across VM instances.)

- d. Select load balancer: Create new → lb-vmss-demo. (Load balancer resource to manage traffic.)
- e. This ensures all VM instances are reachable through one public IP.

You can place this virtual machine in the backend pool of an existing Azure load balancing solution. Learn more None Azure load balancer Supports all TCP/UDP network traffic, port-forwarding, and outbound flows. Application gateway Web traffic load balancer for HTTP/HTTPS with URL-based routing, SSL termination, session persistence, and web application firewall. To allow traffic from your load balancing product, please update the appropriate port configuration on your network security group associated with your network interface. Select a load balancer ★ ① (new) public-nlb

10. Advanced - Add Cloud-init

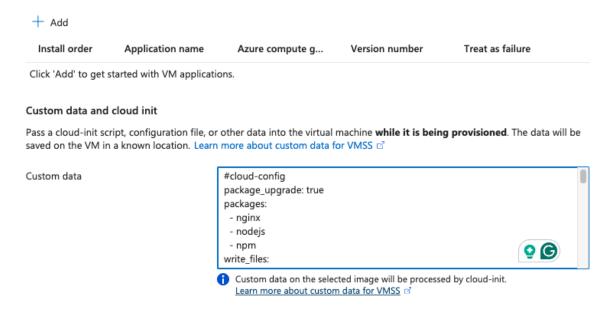
a. In Advanced tab → Custom data, paste the provided #cloud-config.

Create a load balancer

- b. What it does:
 - i. Installs nginx, nodejs, npm.
 - ii. Configures Nginx as a reverse proxy to forward traffic to Node.js.
 - iii. Creates a simple Node.js Express app.
 - iv. Starts the app automatically.
- c. This means every new VM in the scale set configures itself at boot.

Create a Virtual Machine Scale Set (VMSS)

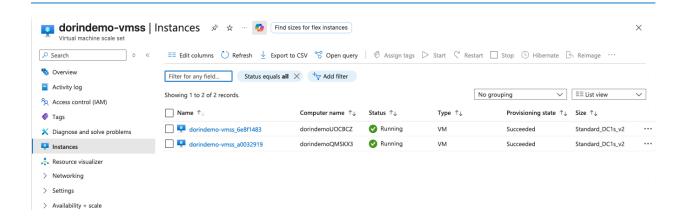
VM applications contain application files that are securely and reliably downloaded on your VM after deployment. In addition to the application files, an install and uninstall script are included in the application. You can easily add or remove applications on your VM after create. Learn more 🗹



Hear data

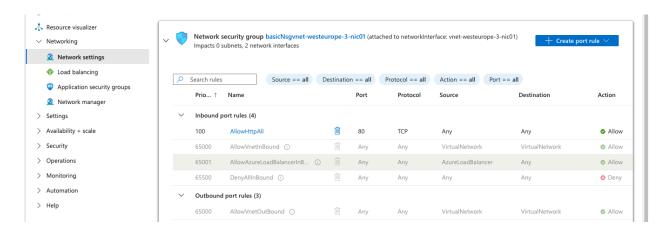
11. Review + Create

- a. Review settings. (Azure validates config, estimates cost.)
- b. Click Create. (Deploys VMSS, load balancer, scaling rules, and VMs.)



Test VMSS scaling capability

- 1. Allow HTTP traffic
 - a. In the Azure Portal, go to your VMSS → Networking → Inbound port rules.
 - b. Add a rule to allow TCP/80 (HTTP).
 - c. This makes your Nginx + Node.js app accessible via the Load Balancer's public IP.



- 2. Access the application
 - a. Open a web browser and go to the public IP of your Load Balancer (you can find it in the Load balancer → Overview).
 - b. You should see the response from your Node.js app:

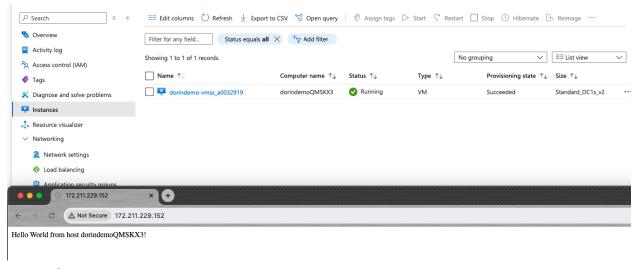


- c. Each VM will return its own hostname (that's how you know which VM responded).
- 3. Simulate removing a VM
 - a. From the portal, note the VM name shown in the browser.

- b. Go to VMSS \rightarrow Instances.
- Select the instance with that name and delete it.



d. Now, refresh your browser → the Load Balancer will route traffic to the other VM in the scale set, and you'll see a different hostname.



- 4. Observe autoscaling recovery
 - Since your minimum instance count is 2, VMSS will detect that one VM is missing.
 - b. Within a few minutes, it will automatically provision a new VM instance.
 - c. Refresh your browser again after a while \rightarrow you'll see the new VM's hostname responding alongside the existing one.

