Module 5 - Lab 2: Direct web traffic with Azure Application Gateway

> In this lab, you use the Azure portal to create an application gateway. Then you test it to make sure it works correctly.
>
> The application gateway directs application web traffic to specific resources in a backend pool. You assign listeners to ports, create rules, and add resources to a backend pool. For the sake of simplicity, this article uses a simple setup with a public front-end IP, a basic listener to host a single site on the application gateway, a basic request routing rule, and two virtual machines in the backend pool.

Task 1: Create an application gateway

> You'll create the application gateway using the tabs on the **Create an application gateway** page.

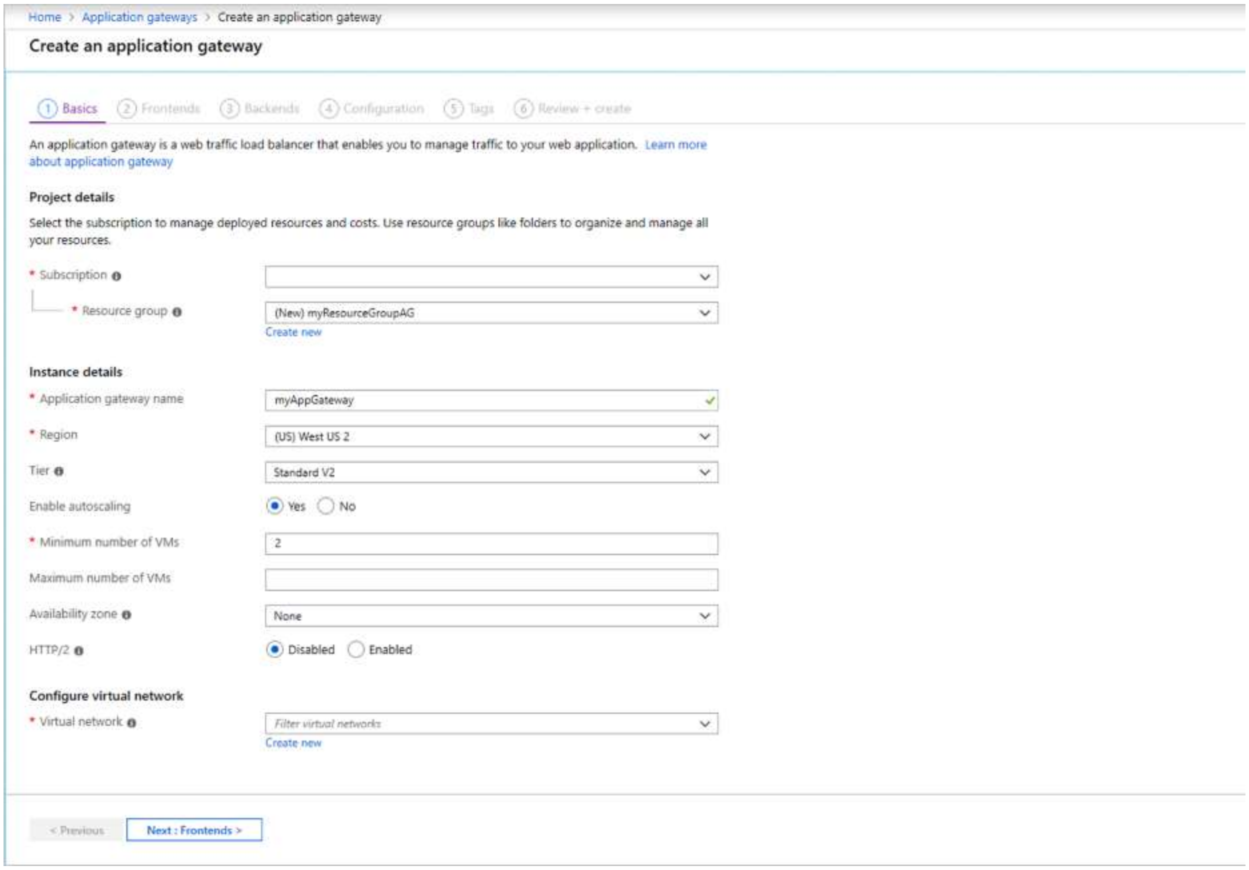
☐ 1. Sign in to the **Azure portal** **https://portal.azure.com** with the username **sheikhnasir2HQY3@gdcs1.com** and password **zqgc5PAulUQ2IStb**

☐ 2. On the **Azure portal** menu or from the **Home** page, select **Create a resource**. The **New** window appears.

☐ 3. Select **Networking** and then select **Application Gateway** in the **Featured** list.

☐ 4. Select + Create and On the **Basics** tab, enter these values for the following application gateway settings:

- **Resource group**: Select **myResourceGroupAG-48CHUWLA7V** for the resource group. it.
- **Application gateway name**: Enter **myAppGateway** for the name of the application gateway.



Home > Application gateways > Create an application gateway

Create an application gateway

An application gateway is a web traffic load balancer that enables you to manage traffic to your web application. Learn more about application gateway

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

* Subscription:

* Resource group: (New) myResourceGroupAG

Create new

Instance details

* Application gateway name: myAppGateway

* Region: (US) West US 2

Tier: Standard V2

Enable autoscaling: ● Yes ○ No

* Minimum number of VMs: 2

Maximum number of VMs:

Availability zone: None

HTTP/2: ● Disabled ○ Enabled

Configure virtual network

* Virtual network: Filter virtual networks

Create new

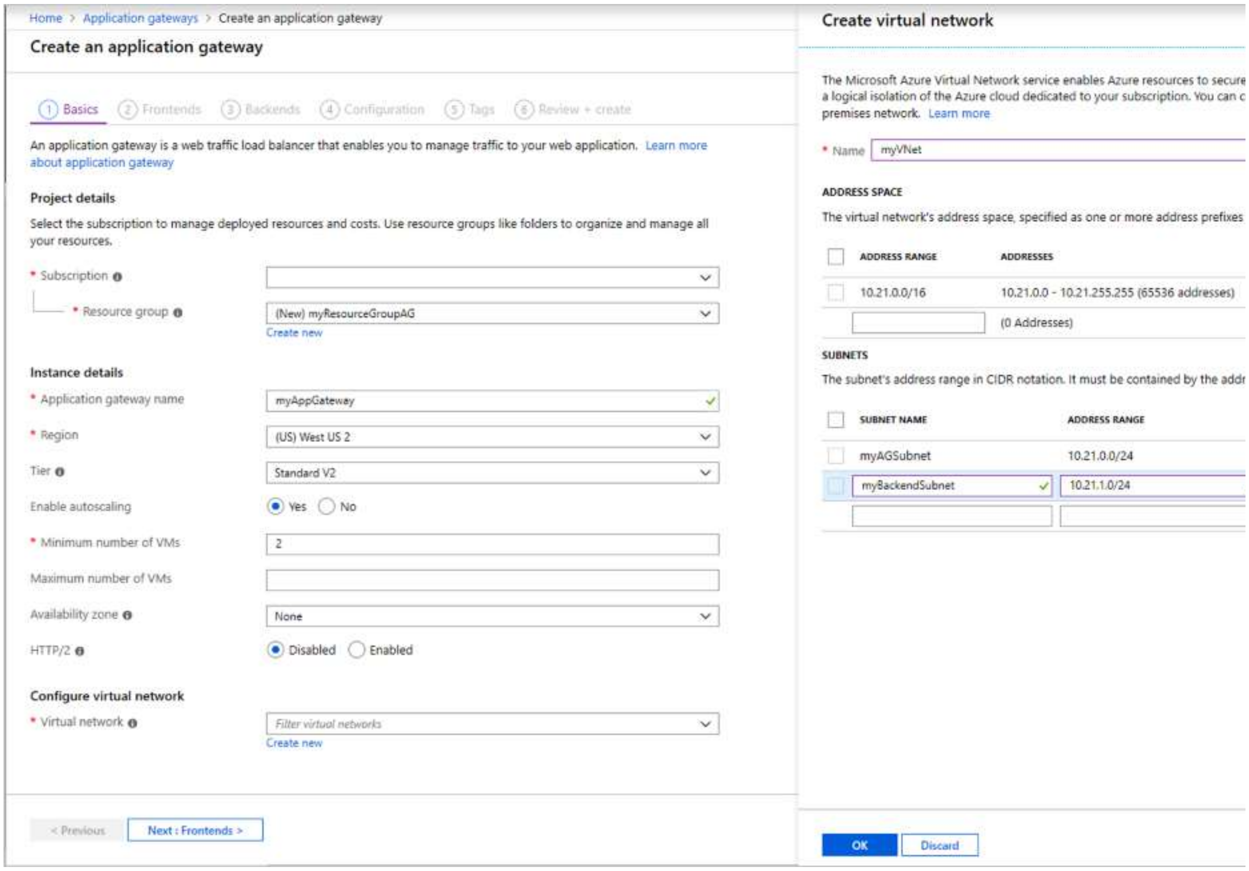
☐ 5. For Azure to communicate between the resources that you create, it needs a virtual network. You can either create a new virtual network or use an existing one. In this example, you'll create a new virtual network at the same time that you create the application gateway. Application Gateway instances are created in separate subnets. You create two subnets in this example: one for the application gateway, and another for the backend servers.

Under **Configure virtual network**, create a new virtual network by selecting **Create new**. In the **Create virtual network** window that opens, enter the following values to create the virtual network and two subnets:

- **Name**: Enter **myVNet** for the name of the virtual network.
- **Subnet name** (Application Gateway subnet): The **Subnets** grid will show a subnet named *Default*. Change the name of this subnet to **myAGSubnet**.
  The application gateway subnet can contain only application gateways. No other resources are allowed.
- **Subnet name** (backend server subnet): In the second row of the **Subnets** grid, enter **myBackendSubnet** in the **Subnet name** column.

- **Address range** (backend server subnet): In the second row of the **Subnets** Grid, enter an address range that doesn't overlap with the address range of *myAGSubnet*. For example, if the address range of *myAGSubnet* is 10.0.0.0/24, enter **10.0.1.0/24** for the address range of *myBackendSubnet*.

  Select **OK** to close the **Create virtual network** window and save the virtual network settings.



Home > Application gateways > Create an application gateway

Create an application gateway

① Basics ② Frontends ③ Backends ④ Configuration ⑤ Tags ⑥ Review + create

An application gateway is a web traffic load balancer that enables you to manage traffic to your web application. Learn more about application gateway

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

* Subscription:
* Resource group: (New) myResourceGroupAG
Create new

Instance details

* Application gateway name: myAppGateway
* Region: (US) West US 2
Tier: Standard V2
Enable autoscaling: ● Yes ○ No
* Minimum number of VMs: 2
Maximum number of VMs:
Availability zone: None
HTTP/2: ● Disabled ○ Enabled

Configure virtual network

* Virtual network: Filter virtual networks
Create new

< Previous | Next : Frontends >

Create virtual network

The Microsoft Azure Virtual Network service enables Azure resources to secure a logical isolation of the Azure cloud dedicated to your subscription. You can c premises network. Learn more

* Name: myVNet

ADDRESS SPACE

The virtual network's address space, specified as one or more address prefixes

□	ADDRESS RANGE	ADDRESSES
□	10.21.0.0/16	10.21.0.0 - 10.21.255.255 (65536 addresses)
		(0 Addresses)

SUBNETS

The subnet's address range in CIDR notation. It must be contained by the addr

□	SUBNET NAME	ADDRESS RANGE
□	myAGSubnet	10.21.0.0/24
□	myBackendSubnet ✓	10.21.1.0/24

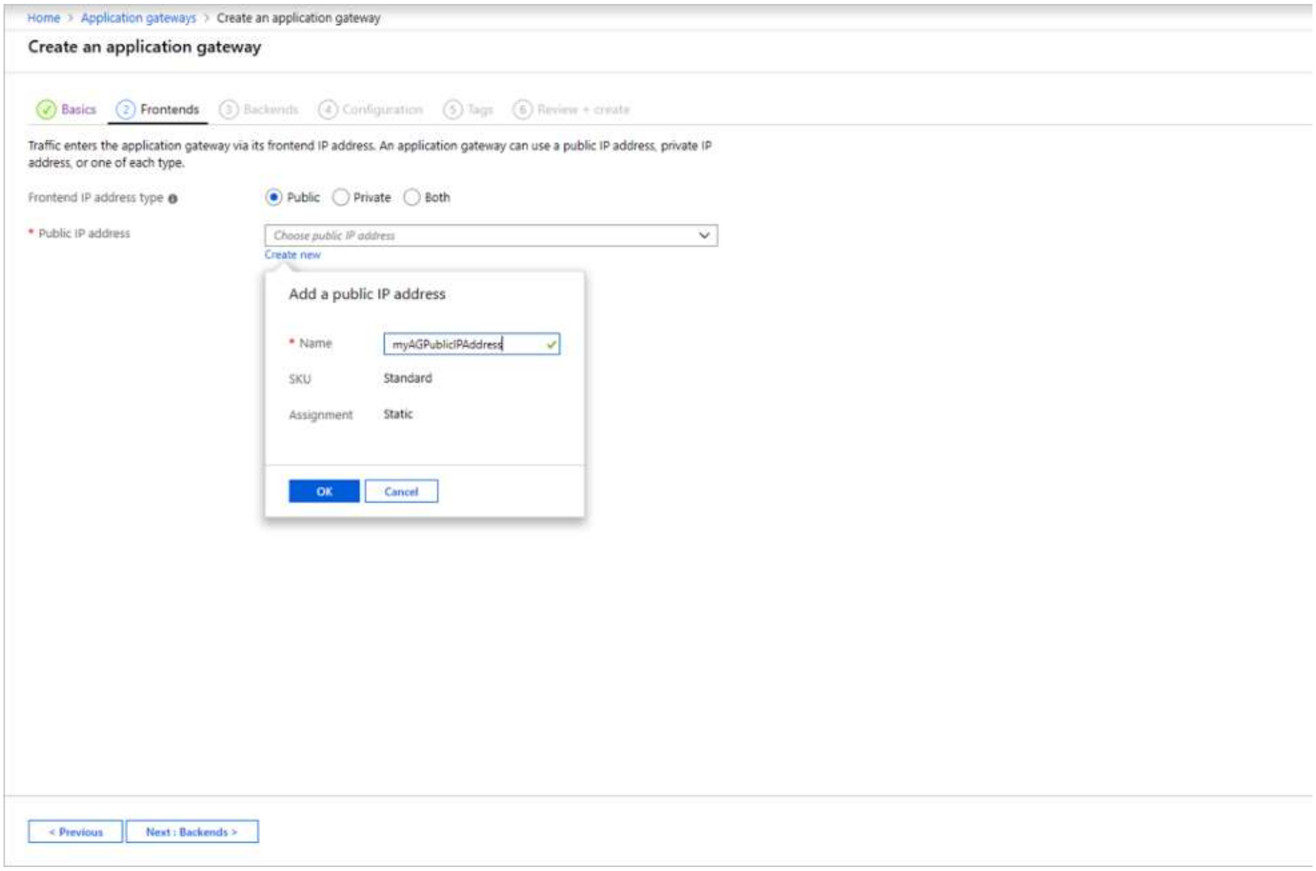
OK | Discard

□ 6. On the **Basics** tab, accept the default values for the other settings and then select **Next: Frontends**.

□ 7. On the **Frontends** tab, verify **Frontend IP address type** is set to **Public**.
You can configure the Frontend IP to be Public or Private as per your use case. In this example, you'll choose a Public Frontend IP.

> ⚠ **Note**: For the Application Gateway v2 SKU, there must be a **Public** frontend IP configuration. You can still have both a Public and a Private frontend IP configuration, but Private only frontend IP configuration (Only ILB mode) is currently not enabled for the v2 SKU.

□ 8. Choose **Add new** for the **Public IP address** and enter **myAGPublicIPAddress** for the public IP address name, and then select **OK**.


Home > Application gateways > Create an application gateway
Create an application gateway
Basics
2 Frontends
3 Backends
4 Configuration
5 Tags
6 Review + create

Traffic enters the application gateway via its frontend IP address. An application gateway can use a public IP address, private IP address, or one of each type.

Frontend IP address type ● Public ○ Private ○ Both

* Public IP address: Choose public IP address

Add a public IP address

* Name: myAGPublicIPAddress

SKU: Standard

Assignment: Static

□ 9. Select **Next: Backends**.

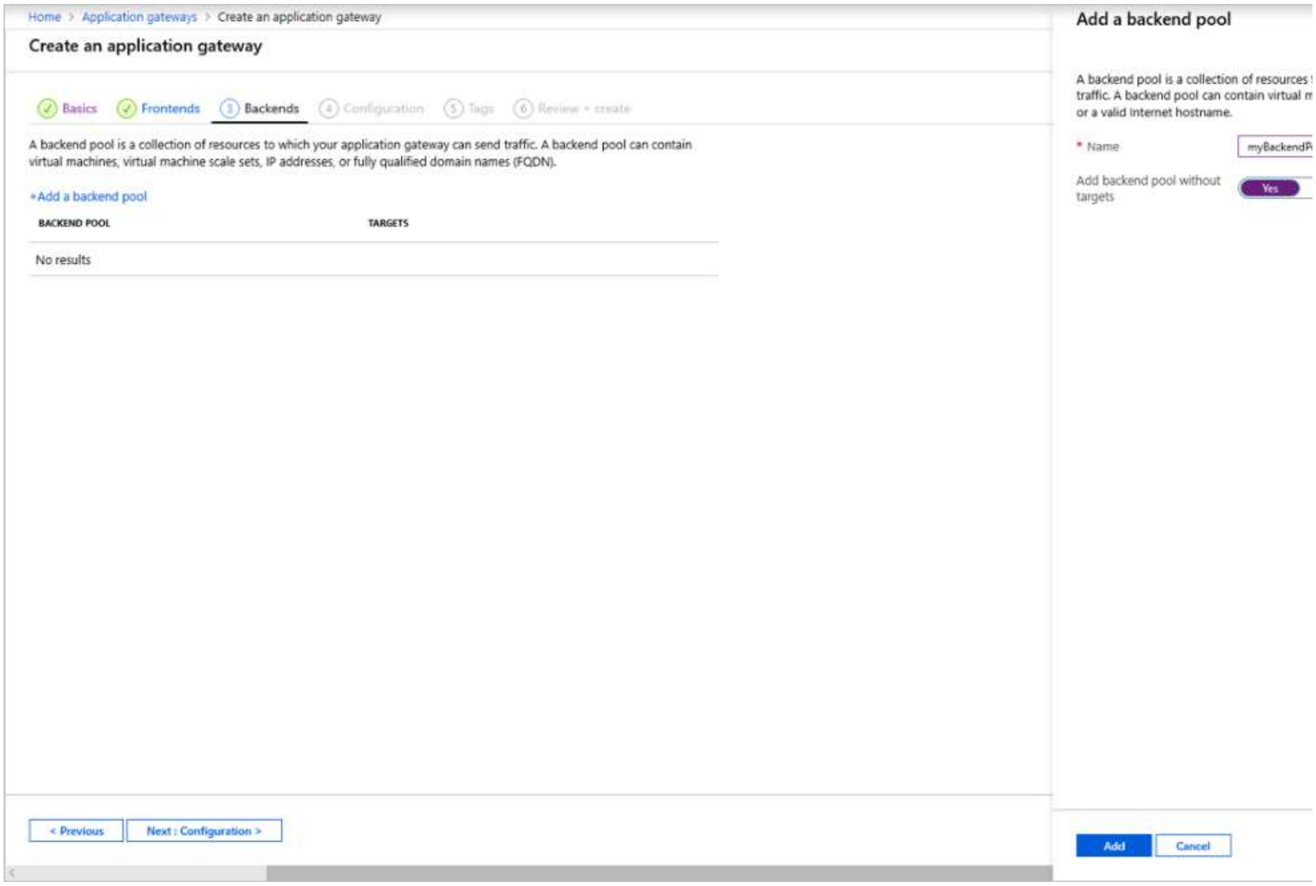
The backend pool is used to route requests to the backend servers that serve the request. Backend pools can be composed of NICs, virtual machine scale sets, public IPs, internal IPs, fully qualified domain names (FQDN), and multi-tenant back-ends like Azure App Service. In this example, you'll create an empty backend pool with your application gateway and then add backend targets to the backend pool.

□ 10. On the **Backends** tab, select **+Add a backend pool**.

□ 11. In the **Add a backend pool** window that opens, enter the following values to create an empty backend pool:

- **Name**: Enter myBackendPool for the name of the backend pool.
- **Add backend pool without targets**: Select **Yes** to create a backend pool with no targets. You'll add backend targets after creating the application gateway.

□ 12. In the **Add a backend pool** window, select **Add** to save the backend pool configuration and return to the **Backends** tab.



Home > Application gateways > Create an application gateway

Create an application gateway

A backend pool is a collection of resources to which your application gateway can send traffic. A backend pool can contain virtual machines, virtual machine scale sets, IP addresses, or fully qualified domain names (FQDN).

+Add a backend pool

BACKEND POOL	TARGETS
No results	

Add a backend pool

A backend pool is a collection of resources t traffic. A backend pool can contain virtual m or a valid Internet hostname.

* Name: myBackendP

Add backend pool without targets: Yes

□ 13. On the **Backends** tab, select **Next: Configuration**.

On the **Configuration** tab, you'll connect the frontend and backend pool you created using a routing rule.

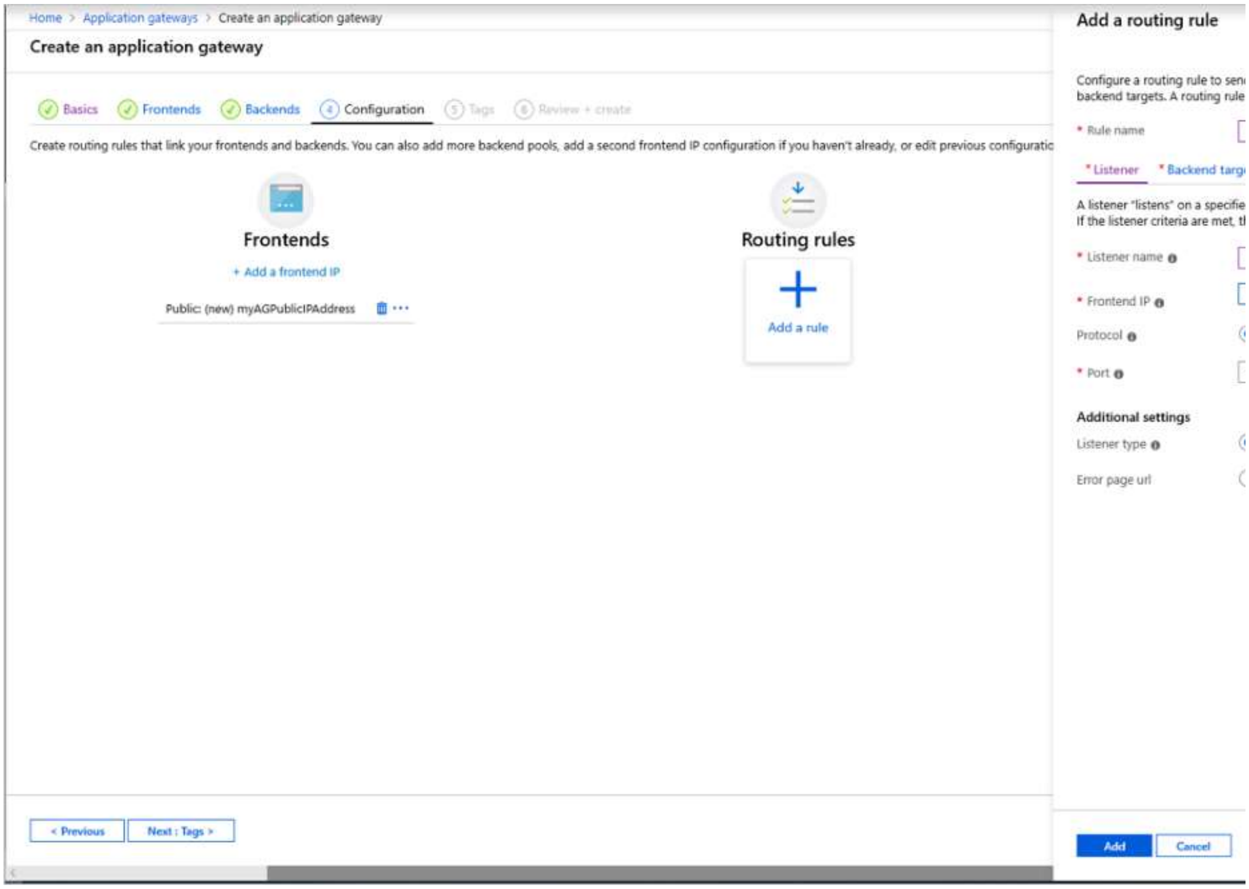
□ 14. Select **Add a rule** in the **Routing rules** column.

□ 15. In the **Add a routing rule** window that opens, enter **myRoutingRule** for the **Rule name**.

□ 16. A routing rule requires a listener. On the **Listener** tab within the **Add a routing rule** window, enter the following values for the listener:

- **Listener name**: Enter **myListener** for the name of the listener.
- **Frontend IP**: Select **Public** to choose the public IP you created for the frontend.

  Accept the default values for the other settings on the **Listener** tab, then select the **Backend targets** tab to configure the rest of the routing rule.



Create routing rules that link your frontends and backends. You can also add more backend pools, add a second frontend IP configuration if you haven't already, or edit previous configuratic

Public: (new) myAGPublicIPAddress

Configure a routing rule to sen backend targets. A routing rule

* Rule name

A listener "listens" on a specifie If the listener criteria are met, tl

* Listener name

* Frontend IP

Protocol

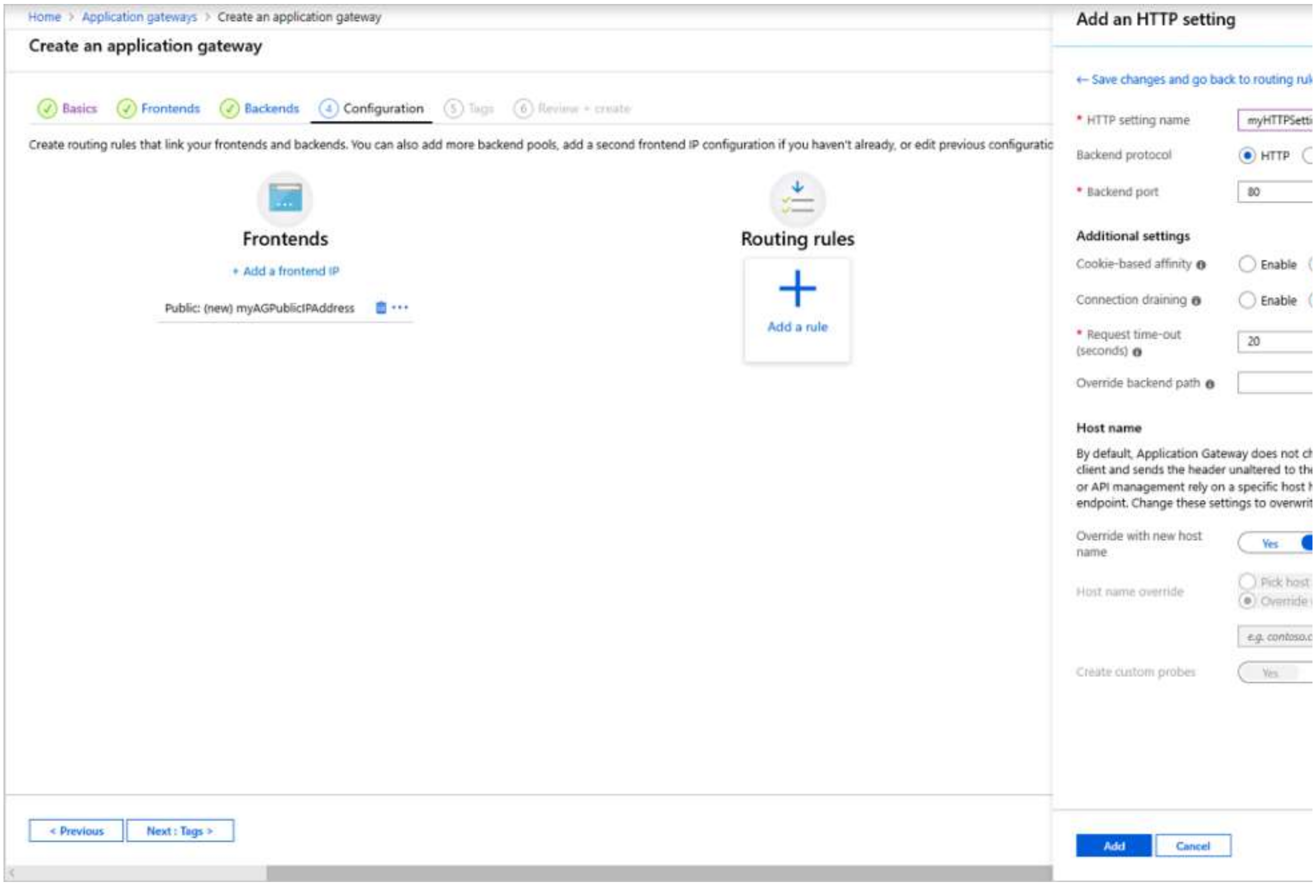
* Port

Listener type

Error page url

□ 17. On the **Backend targets** tab, select **myBackendPool** for the **Backend target**.

□ 18. For the **HTTP setting**, select **Create new** to create a new HTTP setting. The HTTP setting will determine the behavior of the routing rule. In the **Add an HTTP setting** window that opens, enter **myHTTPSetting** for the **HTTP setting name** and *80* for the **Backend port**. Accept the default values for the other settings in the **Add an HTTP setting** window, then select **Add** to return to the **Add a routing rule** window.



Create routing rules that link your frontends and backends. You can also add more backend pools, add a second frontend IP configuration if you haven't already, or edit previous configuratic

Public: (new) myAGPublicIPAddress

* HTTP setting name: myHTTPSetti

Backend protocol: ● HTTP ○

* Backend port: 80

Cookie-based affinity: ○ Enable (

Connection draining: ○ Enable (

* Request time-out (seconds): 20

Override backend path:

By default, Application Gateway does not ch client and sends the header unaltered to the or API management rely on a specific host h endpoint. Change these settings to overwrit

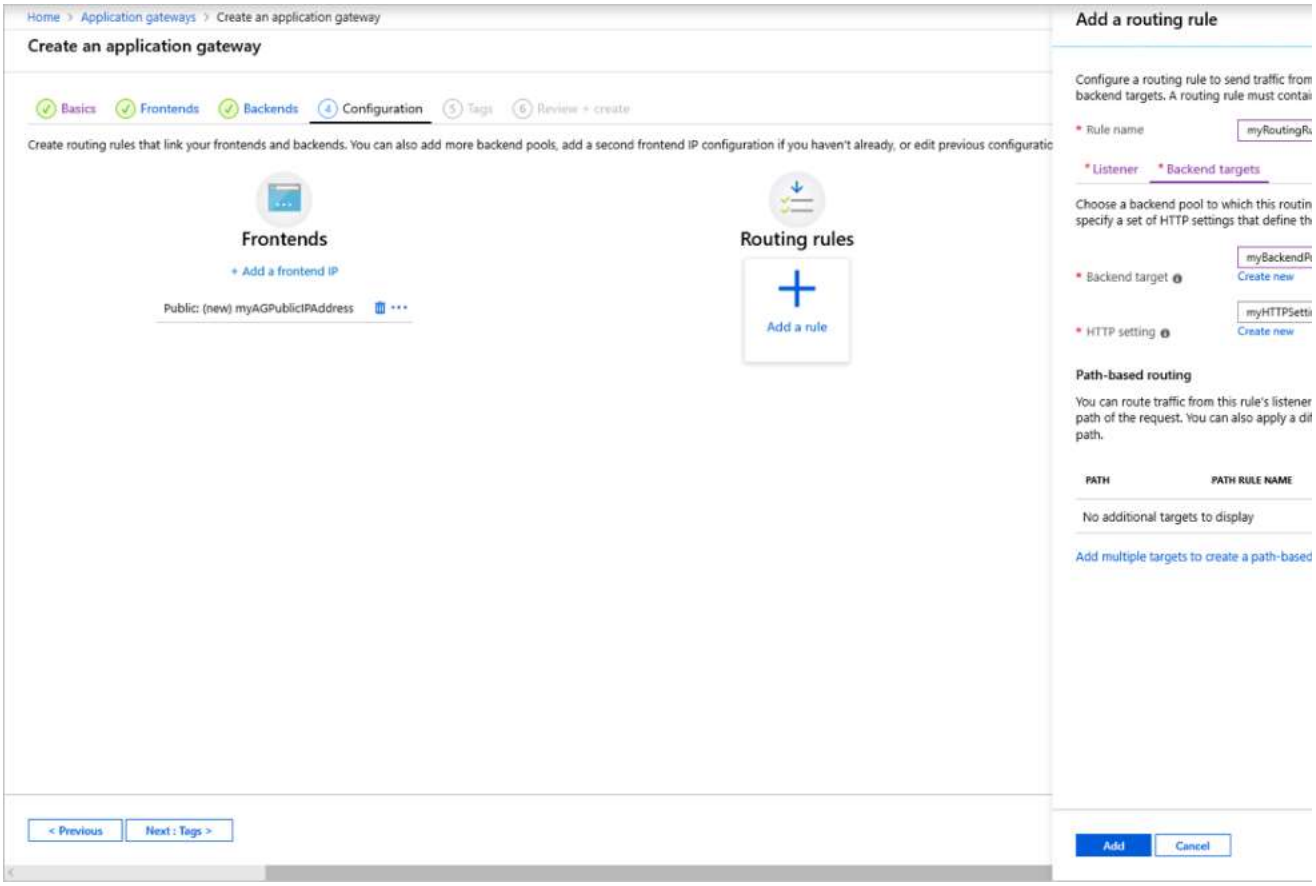
Override with new host name: Yes

Host name override: ○ Pick host ● Override

e.g. contoso.c

Create custom probes: Yes

□ 19. On the **Add a routing rule** window, select **Add** to save the routing rule and return to the **Configuration** tab.



Home > Application gateways > Create an application gateway

Create an application gateway

Create routing rules that link your frontends and backends. You can also add more backend pools, add a second frontend IP configuration if you haven't already, or edit previous configuratic

Public: (new) myAGPublicIPAddress

Add a routing rule

Configure a routing rule to send traffic from backend targets. A routing rule must contai

* Rule name: myRoutingRu

Choose a backend pool to which this routin specify a set of HTTP settings that define th

* Backend target: myBackendP

* HTTP setting: myHTTPSetti

Path-based routing

You can route traffic from this rule's listener path of the request. You can also apply a dif path.

PATH	PATH RULE NAME
No additional targets to display	

□ 20. Select **Next: Tags** and then **Next: Review + create**.

Review the settings on the **Review + create** tab, and then select **Create** to create the virtual network, the public IP address, and the application gateway. It may take several minutes for Azure to create the application gateway. Wait until the deployment finishes successfully before moving on to the next section.

Task 2: Add backend targets

> In this task, you'll use virtual machines as the target backend. You can either use existing virtual machines or create new ones. You'll create two virtual machines as backend servers for the application gateway.
>
> To do this, you'll:
>
> 1. Create two new VMs, *myVM* and *myVM2*, to be used as backend servers.
> 2. Install IIS on the virtual machines to verify that the application gateway was created successfully.
> 3. Add the backend servers to the backend pool.

□ 1. On the Azure portal menu or from the **Home** page, select **Create a resource**. The **New** window appears.

□ 2. Select **Windows Server 2019 Datacenter** in the **Popular** list. The **Create a virtual machine** page appears.

Application Gateway can route traffic to any type of virtual machine used in its backend pool. In this example, you use a Windows Server 2019 Datacenter.

□ 3. Enter these values in the **Basics** tab for the following virtual machine settings:

- **Resource group**: Select **myResourceGroupAG-48CHUWLA7V** for the resource group name.
- **Virtual machine name**: Enter **myVM** for the name of the virtual machine.
- **Region**: Select the same region where you created the application gateway.
- **Username**: Type **localadmin** for the administrator user name.
- **Password**: **zqgc5PAulUQ2lStb**

□ 4. Accept the other defaults and then select **Next: Disks**.

□ 5. Accept the **Disks** tab defaults and then select **Next: Networking**.

□ 6. On the **Networking** tab, verify that **myVNet** is selected for the **Virtual network** and the **Subnet** is set to **myBackendSubnet**. Accept the other defaults and then select **Next: Management**.

Application Gateway can communicate with instances outside of the virtual network that it is in, but you need to ensure there's IP connectivity.

□ 7. On the **Management** tab, set **Boot diagnostics** to **Disabled**. Accept the other defaults and then select **Review + create**.

□ 8. On the **Review + create** tab, review the settings, correct any validation errors, and then select **Create**.

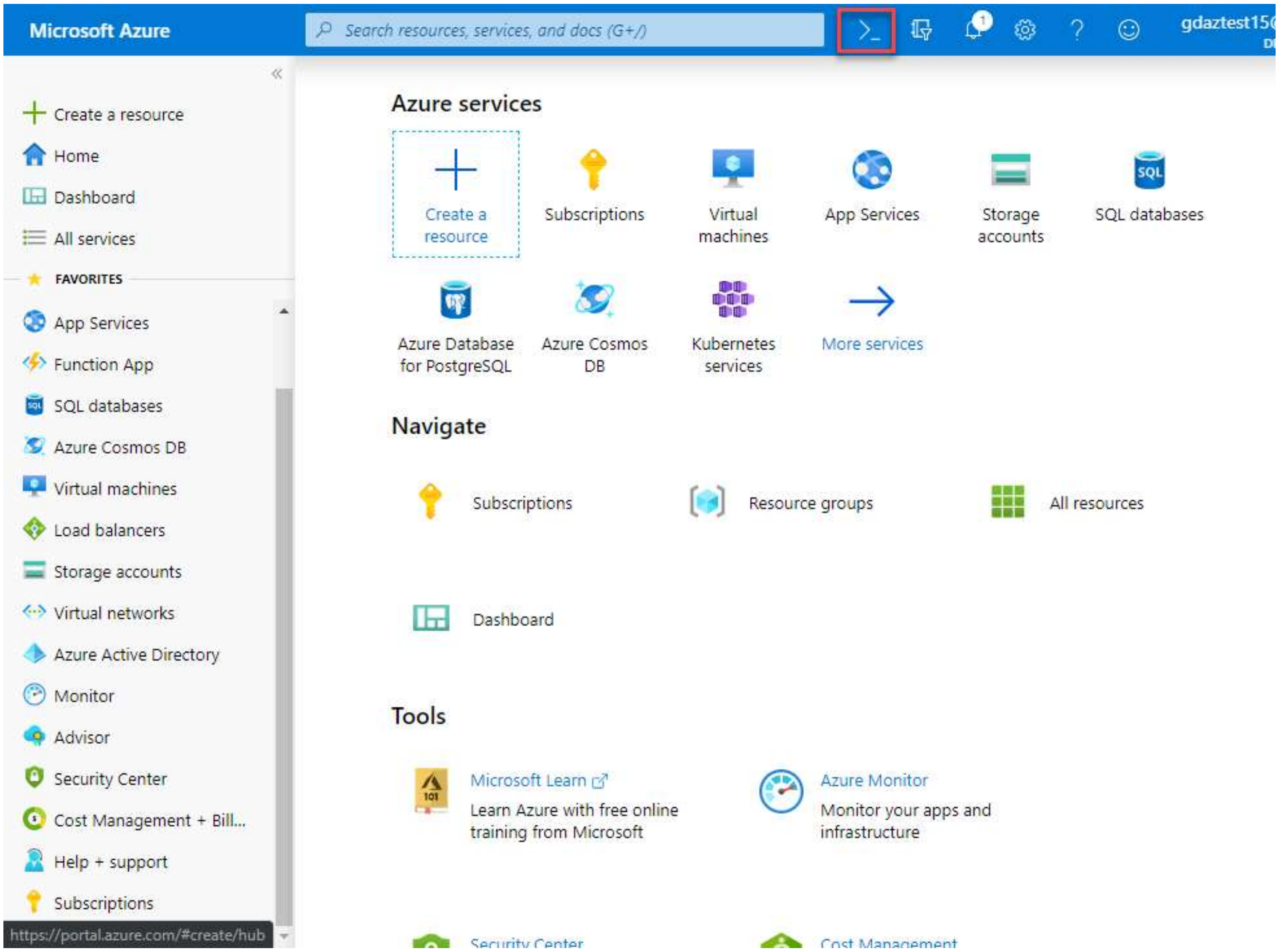
□ 9. Wait for the virtual machine creation to complete before continuing.

Task 3: Install IIS for testing

> In this task, you install IIS on the virtual machines only to verify Azure created the application gateway successfully.

□ 1. Login to the Azure Portal **https://portal.azure.com** with the username **sheikhnasir2HQY3@gdcs1.com** and password **zqgc5PAulUQ2IStb**

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



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

□ 4. 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** drop-down list, select the Azure region matching or near the location where you intend to deploy resources in this exercise.
- In the **Resource group:** East US.
- 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.

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

□ 6. Run the following commands to install IIS on the virtual machine. Change the *Location* parameter if necessary:

```
$ResourceGroup = (Get-AzResourceGroup).ResourceGroupName
```

```
Set-AzVMExtension -ResourceGroupName $ResourceGroup -ExtensionName IIS -VMName myVM -Publisher Microsoft.Compute -ExtensionType CustomSc
```

□ 7. Create a second virtual machine and install IIS by using the steps that you previously completed. Use *myVM2* for the virtual machine name and for the **VMName** setting of the **Set-AzVMExtension** cmdlet.

Task 4: Add backend servers to backend pool

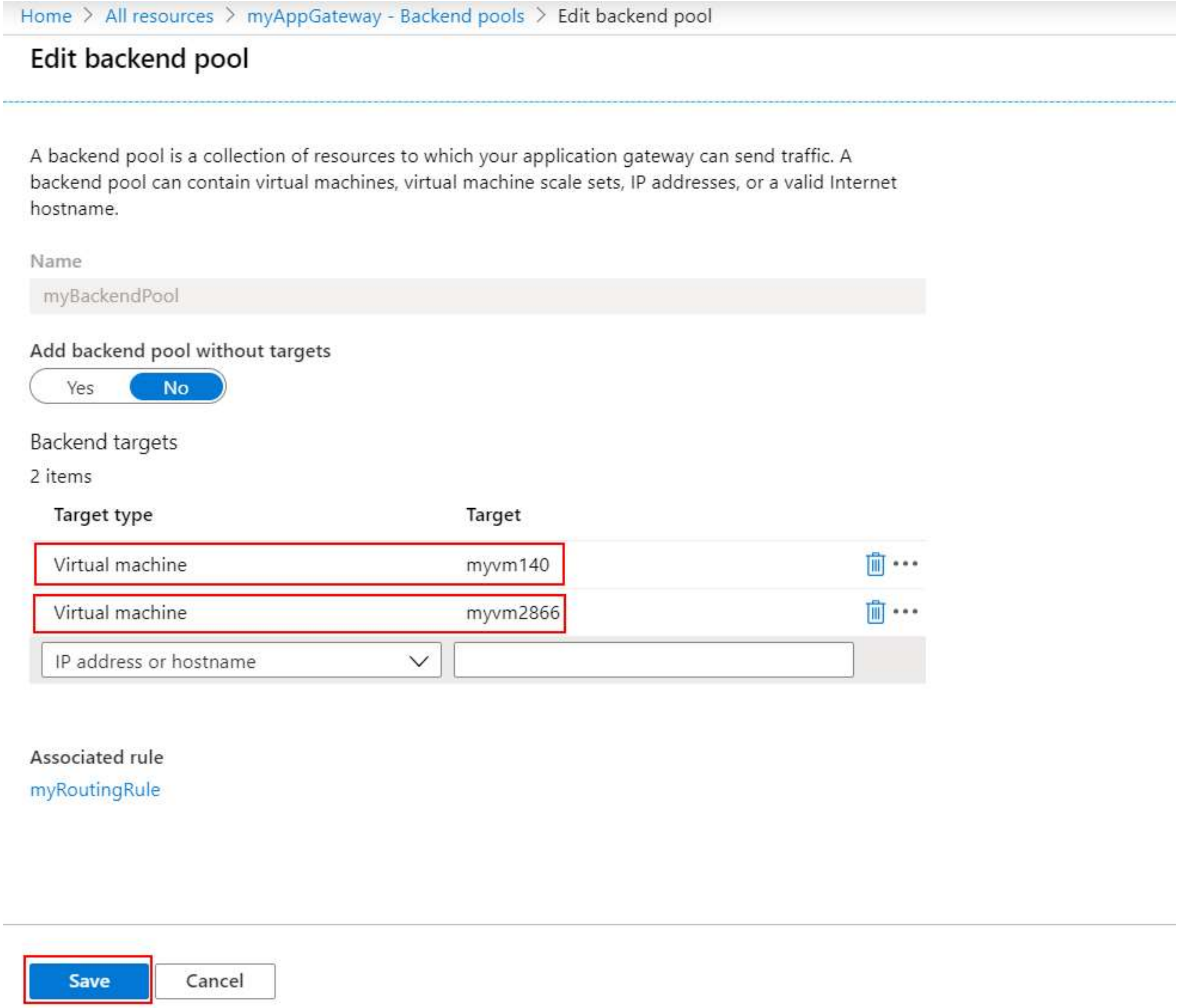
□ 1. On the Azure portal menu, select **All resources** or search for and select *All resources*. Then select **myAppGateway**.

□ 2. Select **Backend pools** from the left menu.

□ 3. Select **myBackendPool**.

□ 4. Under **Backend targets**, **Target type**, select **Virtual machine** from the drop-down list.

□ 5. Under **Target**, select the **myVM** and **myVM2** virtual machines and their associated network interfaces from the drop-down lists.



Home > All resources > myAppGateway - Backend pools > Edit backend pool

Edit backend pool

A backend pool is a collection of resources to which your application gateway can send traffic. A backend pool can contain virtual machines, virtual machine scale sets, IP addresses, or a valid Internet hostname.

Name: myBackendPool

Add backend pool without targets: Yes [✓ No]

Backend targets

2 items

Target type	Target
Virtual machine	myvm140
Virtual machine	myvm2866
IP address or hostname	

Associated rule: myRoutingRule

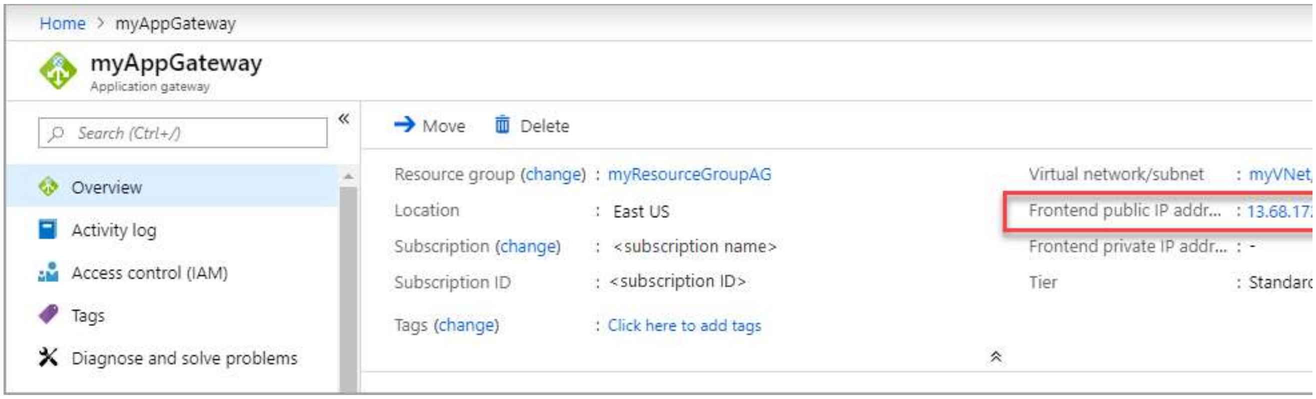
□ 6. Select **Save**.

□ 7. Wait for the deployment to complete before proceeding to the next step.

Task 5: Test the application gateway

Although IIS isn't required to create the application gateway, you installed it in this quickstart to verify if Azure successfully created the application gateway. Use IIS to test the application gateway:

□ 1. Find the public IP address for the application gateway on its **Overview** page.

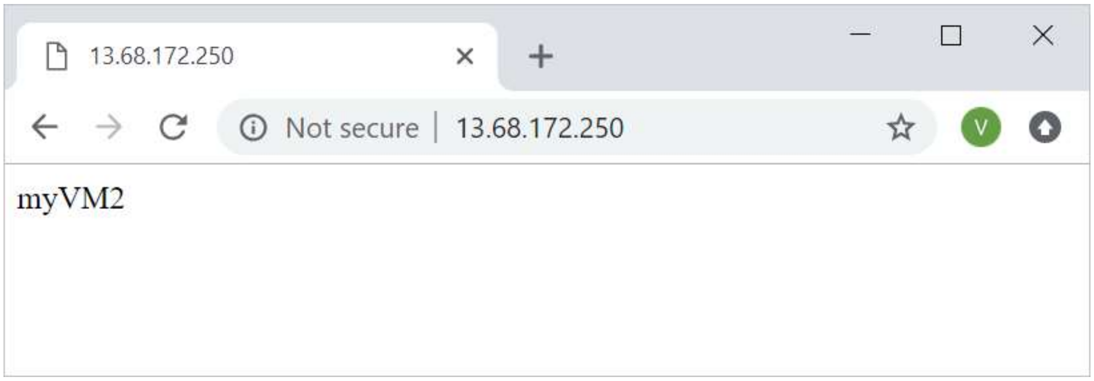


Resource group (change) : myResourceGroupAG
Location : East US
Subscription (change) : <subscription name>
Subscription ID : <subscription ID>
Tags (change) : Click here to add tags
Virtual network/subnet : myVNet,
Frontend public IP addr... : 13.68.17
Frontend private IP addr... : -
Tier : Standard

Or, you can select **All resources**, enter *myAGPublicIPAddress* in the search box, and then select it in the search results. Azure displays the public IP address on the **Overview** page.

☐ 2. Copy the public IP address, and then paste it into the address bar of your browser to browse that IP address.

☐ 3. Check the response. A valid response verifies that the application gateway was successfully created and can successfully connect with the backend.



myVM2

Refresh the browser multiple times and you should see connections to both myVM and myVM2.