Create an application gateway (Class Walkthrough)

1. On the Azure portal menu or from the Home page, select Create a resource.
2. Under Categories, select Networking and then select Application Gateway in the Popular Azure services list.
3. On the Basics tab, enter these values for the following application gateway settings:
    - Resource group: Select myResourceGroupAG for the resource group. If it doesn't exist, select Create new to create it.
    - Application gateway name: Enter myAppGateway for the name of the application gateway.

Create 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 * ⓘ

[myResourceGroupAG   ⌄]

Create new

Instance details

Application gateway name * [myAppGateway   ✓]

Region * [Central US   ⌄]

Tier ⓘ [Standard V2   ⌄]

Enable autoscaling   ● Yes ○ No

Minimum instance count * ⓘ [0   ]

Maximum instance count [10   ]

Availability zone ⓘ [None   ⌄]

HTTP2 ⓘ   ● Disabled ○ Enabled

Configure virtual network

Virtual network * ⓘ [⌄]

Create new

[Previous] [Next : Frontends >]

4. For Azure to communicate between the resources that you create, a virtual network is needed. 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.
   - 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. The default IP address range provided is 10.0.0.0/24.

Create virtual network   ✕

The Microsoft Azure Virtual Network service enables Azure resources to securely communicate with each other in a virtual network which is a logical isolation of the Azure cloud dedicated to your subscription. You can connect virtual networks to other virtual networks, or your on-premises network. Learn more ⬀

Name * | myVNet   ✓ |

ADDRESS SPACE

The virtual network's address space, specified as one or more address prefixes in CIDR notation (e.g. 192.168.1.0/24).

Address range	Addresses	Overlap
10.0.0.0/16	10.0.0.0 - 10.0.255.255 (65536 addresses)	None

SUBNETS

The subnet's address range in CIDR notation. It must be contained by the address space of the virtual network.

Subnet name	IPv4 Address range	IPv4 Addresses
myAGSubnet ✓	10.0.0.0/24 ✓	10.0.0.0 - 10.0.0.255 (256 addresses)

5. 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.
6. Select Add new for the Public IP address and enter myAGPublicIPAddress for the public IP address name, and then select OK.

Home > Create a resource >

Create application gateway   ⋯

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

Traffic enters the application gateway via its frontend IP address(es). 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 new

Add a public IP

Name * | myAGPublicIPAddress ✓ |

SKU   ◯ Basic   ⦿ Standard

Assignment   ◯ Dynamic   ⦿ Static

Availability   None
zone

[OK]   [Cancel]

7. Under Backend pools, select the backend pool.
8. Under Target type, select App Services.
9. Under Target, select your App Service.

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 machines scale sets, IP addresses, domain names, or an App Service.

Name

BEPool

Add backend pool without targets

Yes   **No**

Backend targets

1 item

Target type		Target			
App Services	∨	xstof-appgwtest	∨	🗑	···
IP address or FQDN	∨				

10. Under HTTP Settings, select an existing HTTP setting or add a new one.
11. When creating a new HTTP Setting, give it a name
12. Select HTTP as the desired backend protocol using port 443
13. If the certificate is signed by a well known authority, select "Yes" for "User well known CA certificate". Alternatively add authentication/trusted root certificates of backend servers
14. Make sure to set "Override with new host name" to "No"
15. Select the custom HTTPS health probe in the dropdown for "Custom probe".

Add HTTP setting   ✕

HTTP settings name

http-setting-to-app-service

Backend protocol

◯ HTTP   ◉ HTTPS

Backend port *

443

Trusted root certificate

For end-to-end SSL encryption, the backends must be in the allowlist of the application gateway. Upload the public certificate of the backend servers to this HTTP setting.

Use well known CA certificate

◉ Yes   ◯ No

Additional settings

Cookie-based affinity ⓘ

◯ Enable   ◉ Disable

Connection draining ⓘ

◯ Enable   ◉ Disable

Request time-out (seconds) * ⓘ

20

Override backend path ⓘ

Host name

By default, Application Gateway does not change the incoming HTTP host header from the client and sends the header unaltered to the backend. Multi-tenant services like App service or API management rely on a specific host header or SNI extension to resolve to the correct endpoint. Change these settings to overwrite the incoming HTTP host header.

Override with new host name

[Yes   **No**]

Host name override

◯ Pick host name from backend target

◉ Override with specific domain name

e.g. contoso.com

Use custom probe ⓘ

◉ Yes   ◯ No

Custom probe *

probe-for-appvs-backend-https   ⌄

16. Open the "Listeners" section and choose "Add listener" or click an existing one to edit
17. For a new listener: give it a name
18. Under "Frontend IP", select the IP address to listen on
19. Under "Port", select 80
20. Under "Protocol", select "HTTP"
21. Under "Listener Type", select "Basic"
22. Click "Add" to add the listener

Add listener

xstof-appgw

Listener name * ⓘ

> public-https-listener

Frontend IP * ⓘ

> Public

Port * ⓘ

> 443

Protocol ⓘ

○ HTTP ● HTTPS

Choose a certificate

● Create new ○ Select existing

Https Settings

Choose a certificate

○ Upload a certificate ● Choose a certificate from Key Vault

Cert name *

> my-domain-wildcard-cert

Managed identity * ⓘ

> xstof-user-assigned-mi

Key vault * ⓘ

> xstof-domain-kv

Certificate *

> wildcard-xstof-net-with-intermediaries

☐ Enable SSL Profile ⓘ

Additional settings

Listener type ⓘ

● Basic ○ Multi site

[Add] [Cancel]

23. Under "Rules", click to add a new "Request routing rule"
24. Provide the rule with a name
25. Select an HTTP or HTTPS listener that is not bound yet to an existing routing rule
26. Under "Backend targets", choose the Backend Pool in which App Service has been configured
27. Configure the HTTP settings with which Application Gateway should connect to the App Service backend
28. Select "Add" to save this configuration

Add a routing rule
xstof-appgw

Configure a routing rule to send traffic from a given frontend IP address to one or more backend targets. A routing rule must contain a listener and at least one backend target.

Rule name *   `public-https-to-app-svc-https` ✓

* Listener   *** Backend targets**

Choose a backend pool to which this routing rule will send traffic. You will also need to specify a set of HTTP settings that define the behavior of the routing rule.

Target type   ⦿ Backend pool   ◯ Redirection

Backend target * ⓘ   `be-pool-for-appsvc` ⌄

HTTP settings * ⓘ   `http-setting-to-app-service` ⌄

Open the "Backend health" section and ensure the "Status" column indicates the combination for HTTP Setting and Backend Pool shows as "Healthy".

Set up Azure Firewall (Class Walkthrough)

1. Navigate to the Azure Portal
   - Go to the Azure Portal.
   - Sign in with your Azure credentials.
2. Find your WAF Policy
   - In the left-hand menu, click on "All services" and then search for "Web Application Firewall policies."
   - Select your WAF policy from the list. If you haven't already created a policy, you would need to create one.
3. Set the Policy in Prevention Mode
   - In the WAF policy dashboard, under "Settings," click on "Policy settings."
   - For the option "Mode," select "Prevention."

- ○ Click the "Save" button to apply the changes.
4. Enable or Disable Specific Firewall Rules
    - ○ Still in the WAF policy dashboard, under "Settings," click on "Custom rules" if you're looking to enable/disable custom rules. For managed rules, click on "Managed rules."
    - ○ Here you'll see a list of the rules.
        - i. For Custom Rules:
            1. Click on the rule you want to enable or disable.
            2. Toggle the "State" option to "Enabled" or "Disabled."
            3. Click "Save."
        - ii. For Managed Rules:
            1. Find the rule set you are interested in.
            2. Click on the specific rule set to expand and view individual rules.
            3. For each rule, you can toggle it "On" or "Off."
            4. After making changes, go back to the "Managed rules" page and click "Save" at the top.
5. Verify the Changes
    - ○ Check the operational logs or perform tests to ensure that the WAF is operating in prevention mode.
    - ○ For the rules you enabled or disabled, test to ensure they are acting as expected.

Deploy a bastion Host

1. Sign in to the Azure portal.
2. Go to your virtual network.
3. On the page for your virtual network, in the left pane, select Bastion to open the Bastion page.
4. On the Bastion page, expand Dedicated Deployment Options.
5. Select Configure manually. This lets you configure specific additional settings (such as the SKU) when deploying Bastion to your virtual network.

6. On the Create a Bastion page, configure the settings for your bastion host. Project details are populated from your virtual network values. Configure the Instance details values.
   - Name: Type the name that you want to use for your bastion resource.
   - Region: The Azure public region in which the resource will be created. Choose the region in which your virtual network resides.
   - Tier: The tier is also known as the SKU. For this tutorial, select Standard. For information about the features available for each SKU, see Configuration settings - SKU
   - Instance count: This is the setting for host scaling and is available for the Standard SKU. Host scaling is configured in scale unit increments. Use the slider or type a number to configure the instance count that you want. For this tutorial, you can select the instance count you'd prefer. For more information, see Host scaling and Pricing.

7. Configure the virtual network settings. Select your virtual network from the dropdown. If you don't see your virtual network in the dropdown list, make sure you selected the correct Region in the previous settings on this page.
8. To configure the AzureBastionSubnet, select Manage subnet configuration.

Configure virtual networks

Virtual network * ⓘ | VNet1 ⌄ |

Create new

❌ To associate a virtual network with a Bastion, it must contain a subnet with name AzureBastionSubnet and a prefix of at least /26

Subnet * | ⌄ |

Manage subnet configuration

9. On the Subnets page, select +Subnet to open the Add subnet page.
10. On the Add subnet page, create the 'AzureBastionSubnet' subnet using the following values. Leave the other values as default.
    ○ The subnet name must be AzureBastionSubnet.
    ○ The subnet must be at least /26 or larger (/26, /25, /24 etc.) to accommodate features available with the Standard SKU.
    ○ Select Save at the bottom of the page to save your values.
11. At the top of the Subnets page, select Create a Bastion to return to the Bastion configuration page

Home > TestRG1 > VNet1 | Bastion > Create a Bastion > VNet1

‹·› VNet1 | Subnets ☆ ⋯
Virtual network

| 🔍 Search (Ctrl+/) | « | + Subnet + Gateway subnet ↻ Refresh | ⧣ Manage users 🗑 Delete |

‹·› Overview
▫ Activity log
⧣ Access control (IAM)
◆ Tags
𝒫 Diagnose and solve problems

Settings
‹·› Address space

🔍 Search subnets

Name ↑↓	IPv4 ↑↓	IPv6 ↑↓	Available IPs ↑↓	Delegated to ↑↓
FrontEnd	10.1.0.0/24	-	249	-
AzureBastionSubnet	10.1.1.0/26	-	59	-

12. The Public IP address section is where you configure the public IP address of the Bastion host resource on which RDP/SSH will be accessed (over port 443). The public IP address must be in the same region as the Bastion resource you're creating. Create a new IP address. You can leave the default naming suggestion.
13. When you finish specifying the settings, select Review + Create. This validates the values.
14. Once validation passes, you can deploy Bastion. Select Create. You'll see a message letting you know that your deployment is in process. Status displays on this page as the resources are created. It takes about 10 minutes for the Bastion resource to be created and deployed.
15. Connect to the vm using Bastion in your browser