SANS 2023

Workshop: Beyond Firewalls: Techniques for Protecting Cloud-Based Assets

Prerequisites

- An Azure account which you are the owner / have root access
- A laptop with a web browser

GitHub link: https://github.com/cy63rSANS/workshop1_deploy

Synopsis

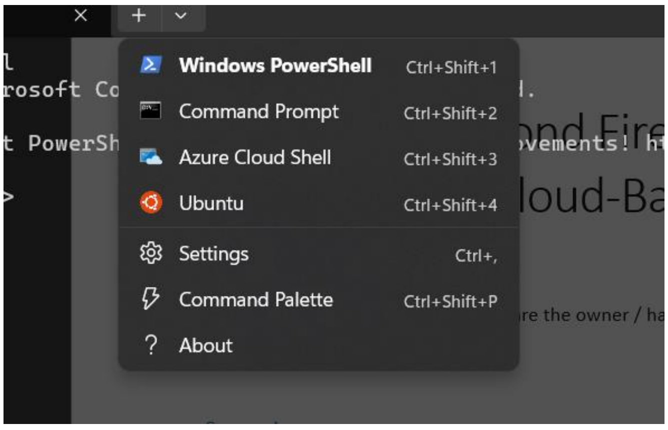
When a system designed for on-premises operation is migrated to a public cloud, it is exposed to additional vulnerabilities and risks of exploitation. This workshop will delve into the realm of cloud native security solutions and techniques, to demonstrate how it is possible to protect such systems that are otherwise considered indefensible particularly in the case of "Lift and Shift".

Stage 1

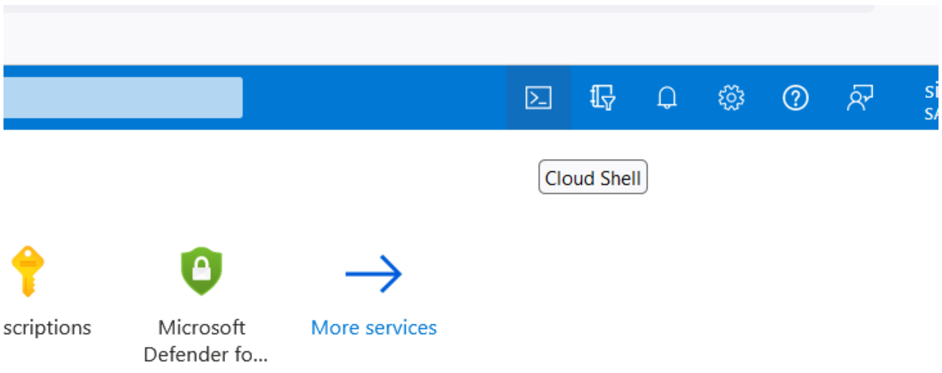
Objective: Deploy lab assets

This entire workshop focuses on using the Azure web portal the Azure shell CLI. There are several ways to access the CLI but we will focus on 2 methods for this lab.

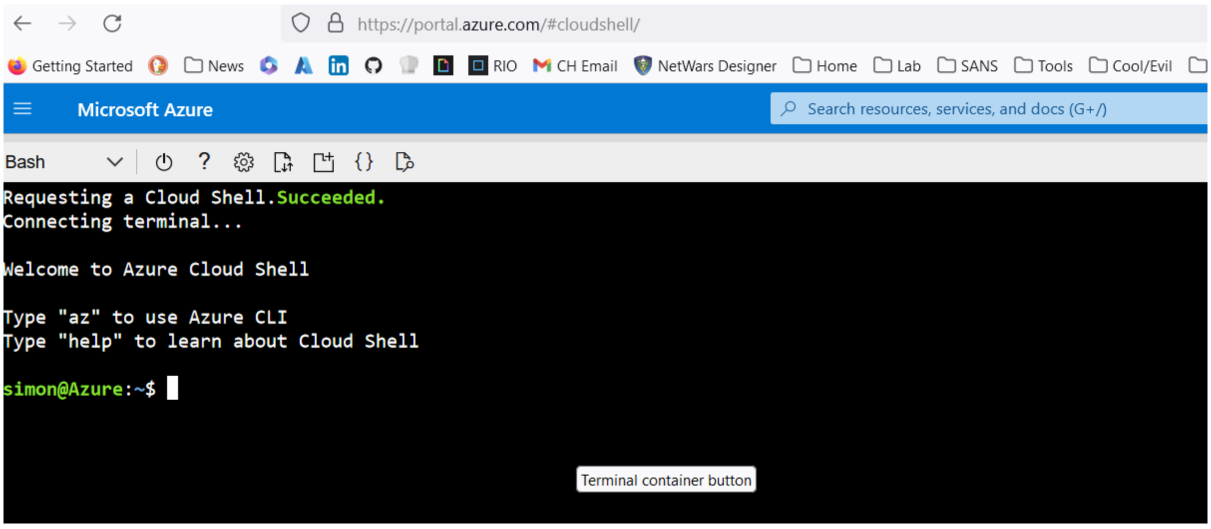
If you are using Windows 11 you can access an inbuilt application called 'Terminal'. Terminal allows you to connect to Azure cloud shell without using the web browser.


Windows PowerShell Ctrl+Shift+1
Command Prompt Ctrl+Shift+2
Azure Cloud Shell Ctrl+Shift+3
Ubuntu Ctrl+Shift+4
Settings Ctrl+,
Command Palette Ctrl+Shift+P
About

If you are using any other operating system other than Windows 11 you can use the browser to directly access the shell using the link on the top right of the Azure portal.



Alternatively, and highly recommended you can also use a full browser-based shell from Azure just by navigating to https://shell.azure.com


https://portal.azure.com/#cloudshell/
Getting Started
News
RIO
CH Email
NetWars Designer
Home
Lab
SANS
Tools
Cool/Evil
Microsoft Azure
Search resources, services, and docs (G+/)
Bash
Terminal container button

```
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

Welcome to Azure Cloud Shell

Type "az" to use Azure CLI
Type "help" to learn about Cloud Shell

simon@Azure:~$
```

Requirement: When you first login to the cloud shell it will ask you for a storage account to log your shell activity and provide you with profile storage space. Please just accept the defaults.

Now you're in the console and the shell we can go ahead and deploy the workshop assets.

Actions: Deployment process

1. In the shell run the following commands:
2. `git clone https://github.com/cy63rSANS/workshop1_deploy`
3. `cd /workshop1_deploy`
4. `terraform init`
5. `terraform apply -auto-approve`

You should now see terraform deploying your infrastructure.

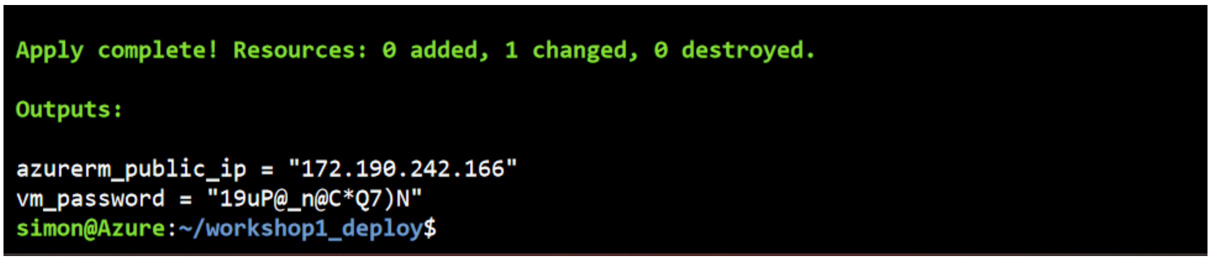
WARNING: You may receive an error from the shell saying ERROR, the was an issue using MSI. If you do receive this error, please run the following commands.

```
az login
```

Follow the instructions to sign in with the device ID.

Once you have completed this, please re-run the deployment instructions from number 3 onwards.

Once the deployment is complete you will see this:



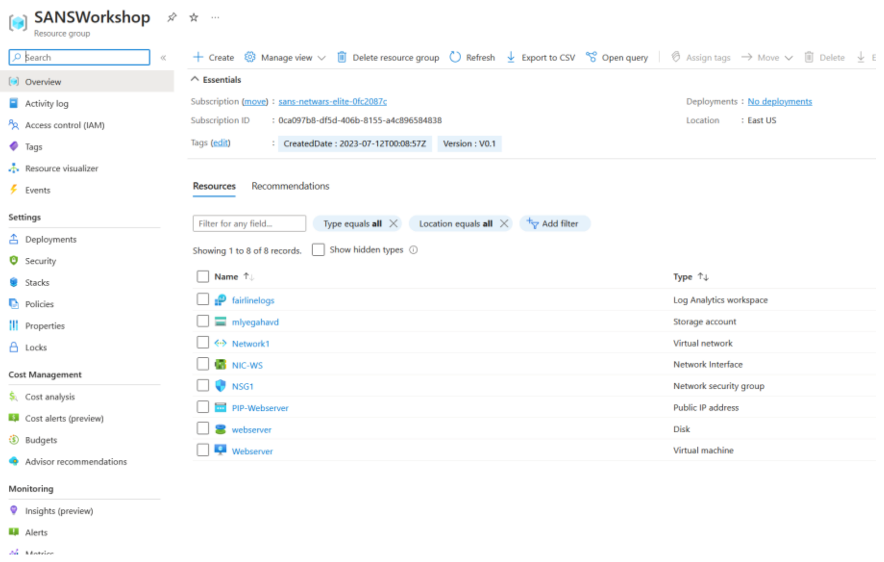
```
Apply complete! Resources: 0 added, 1 changed, 0 destroyed.

Outputs:

azurerm_public_ip = "172.190.242.166"
vm_password = "19uP@_n@C*Q7)N"
simon@Azure:~/workshop1_deploy$
```

Review: What you have just accomplished

The lab environment should now be deployed, you should be able to see the following assets in a resource group called SANSWorkshop. You can find this in the Azure portal under 'Resource Groups'



Subscription (move) : sans-netwars-elite-0fc2087c
Subscription ID : 0ca097b8-df5d-406b-8155-a4c896584838
Tags (edit) : CreatedDate : 2023-07-12T00:08:57Z   Version : V0.1
Deployments : No deployments
Location : East US

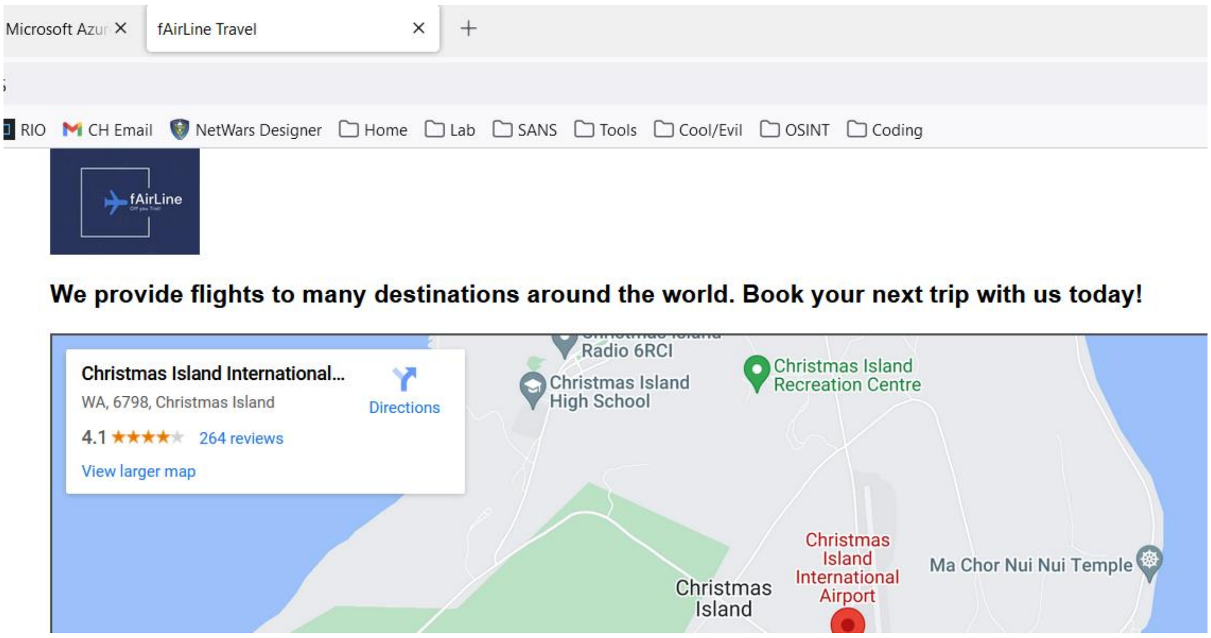
Showing 1 to 8 of 8 records.

Name	Type
fairlinelogs	Log Analytics workspace
mlyegahavd	Storage account
Network1	Virtual network
NIC-WS	Network Interface
NSG1	Network security group
PIP-Webserver	Public IP address
webserver	Disk
Webserver	Virtual machine

You can retrieve your Webservers IP address from the Azure Portal by navigating to the Virtual Machine object called ‘Webserver’ or you can retriceve the IP address by running this command in the shell:

```
az network public-ip list -g SANSWorkshop | jq -r .[].ipAddress
```

Browsing to this IP should present you with this page:



We provide flights to many destinations around the world. Book your next trip with us today!

WARNING: This page may take 5 mins before its available. Make sure you wait until you see this page before proceeding.

Last deployment step:

Run this command from the cloud shell:

```
./final.sh
```

Once this command has completed you can run an attack against your webserver, simply paste your webserver IP address in the page below and it will attack your site.

https://attack.cy63r.ninja

So, Who attacked you?

What was the source IP address?

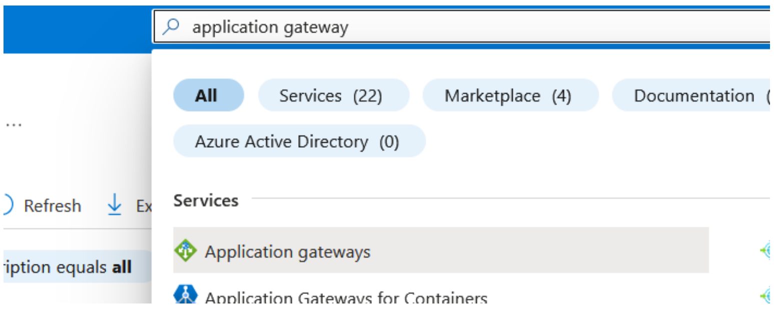
Defence and visibility remediations:

Add Application Gateway Max 15 mins

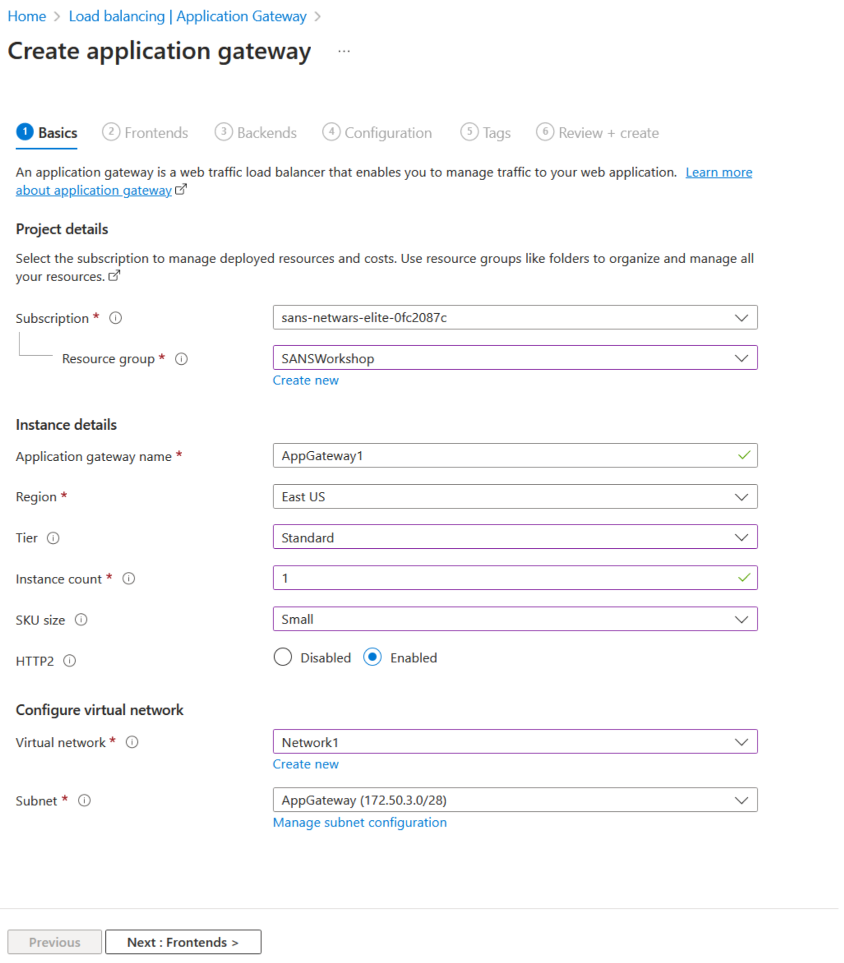
Objective: Add application gateway and WAF to protect webservice

You are going to add an application gateway to the resource group to protect the web application and API

1. Navigate to 'Application Gateways'



2.



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 *	sans-netwars-elite-0fc2087c
Resource group *	SANSWorkshop

Create new

Instance details

Application gateway name *	AppGateway1
Region *	East US
Tier	Standard
Instance count *	1
SKU size	Small
HTTP2	Disabled [✓ Enabled]

Configure virtual network

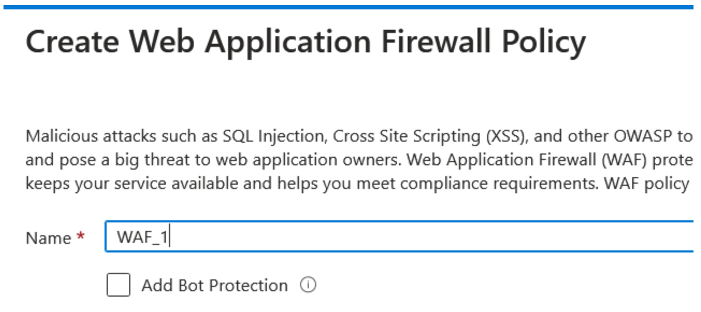
Virtual network *	Network1
Subnet *	AppGateway (172.50.3.0/28)

Create new

Manage subnet configuration

3. Then click 'Next.

4. Click on 'WAF Policy' and Create New, Then give the WAF a name:



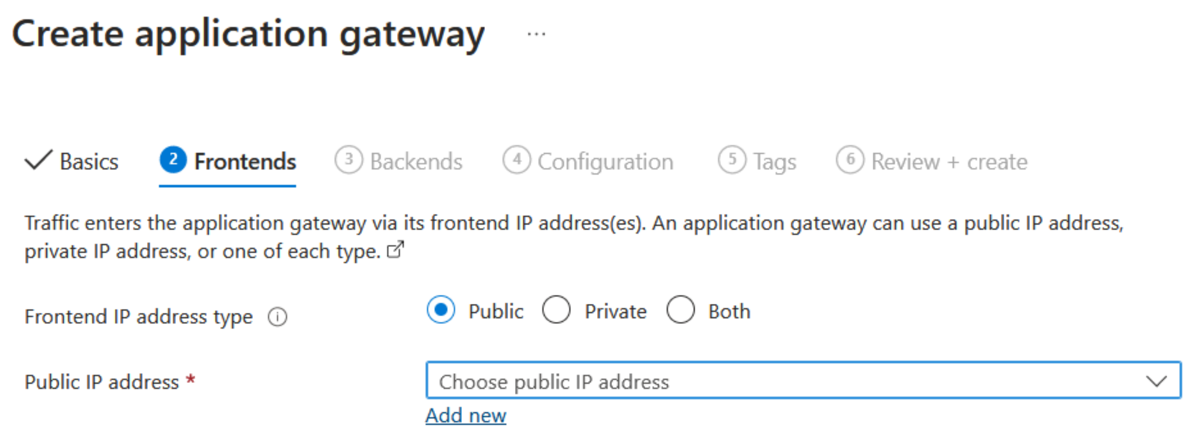
Create Web Application Firewall Policy

Malicious attacks such as SQL Injection, Cross Site Scripting (XSS), and other OWASP to and pose a big threat to web application owners. Web Application Firewall (WAF) prote keeps your service available and helps you meet compliance requirements. WAF policy

Name * WAF_1

☐ Add Bot Protection

5. Click on 'Frontends' then click to **'Add New'** IP address



Create application gateway ...

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

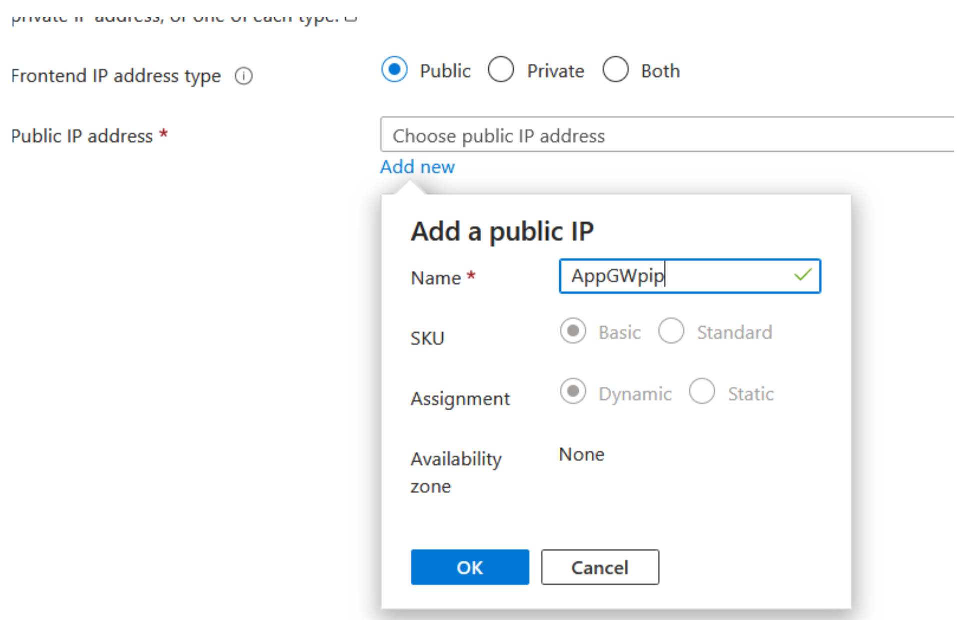
Traffic enters the application gateway via its frontend IP address(es). An application gateway can use a public IP address, private IP address, or one of each type.

Frontend IP address type: [✓ Public] Private Both

Public IP address *: Choose public IP address

Add new

6.



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

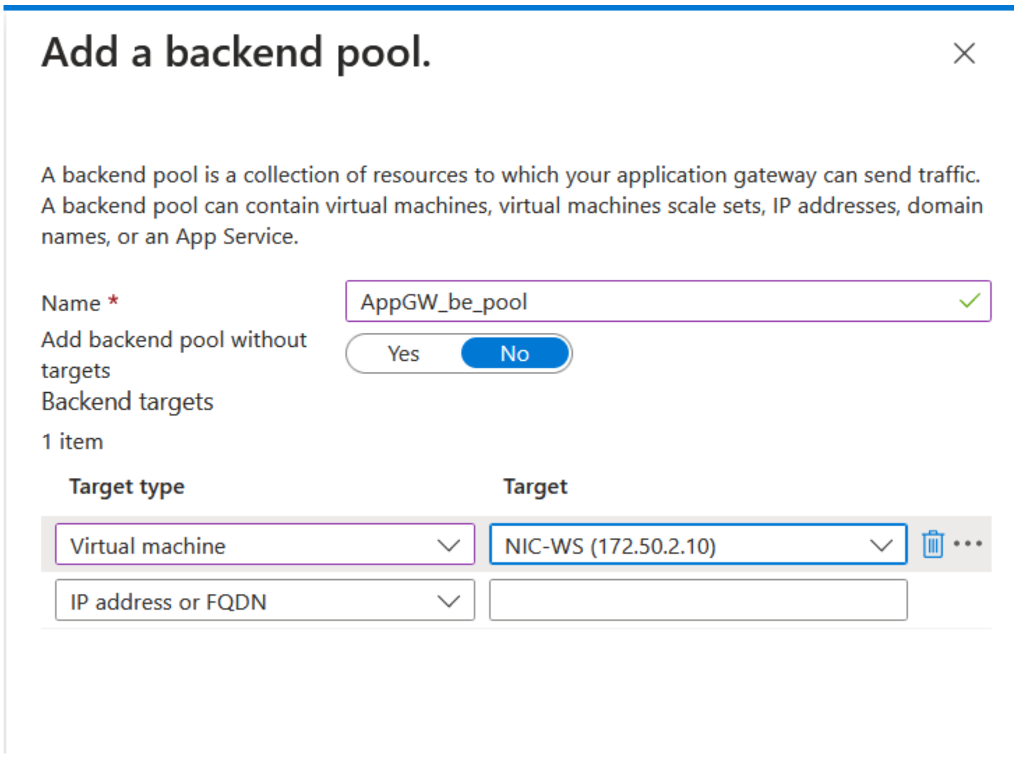
Name * AppGWpip

SKU: [✓ Basic] Standard

Assignment: [✓ Dynamic] Static

Availability zone: None

7. Click on ‘Backends’ and then ‘Add Backend Pool’



Add a 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.

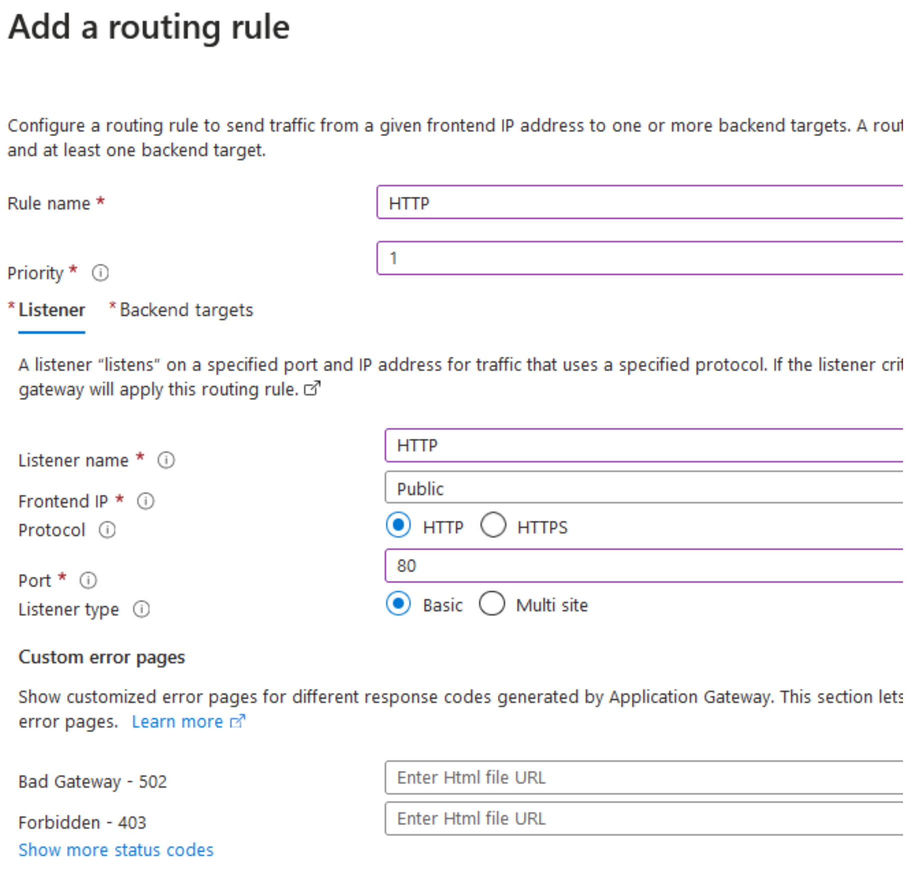
Name *	AppGW_be_pool
Add backend pool without targets	Yes [✓ No]

Backend targets

1 item

Target type	Target
Virtual machine	NIC-WS (172.50.2.10)
IP address or FQDN	

8. Click ‘Next: Configuration’ and select ‘Add a routing Rule’



Add a routing rule

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

Rule name *	HTTP
Priority *	1

*Listener *Backend targets

A listener "listens" on a specified port and IP address for traffic that uses a specified protocol. If the listener crit
gateway will apply this routing rule.

Listener name *	HTTP
Frontend IP *	Public
Protocol	[✓ HTTP] HTTPS
Port *	80
Listener type	[✓ Basic] Multi site

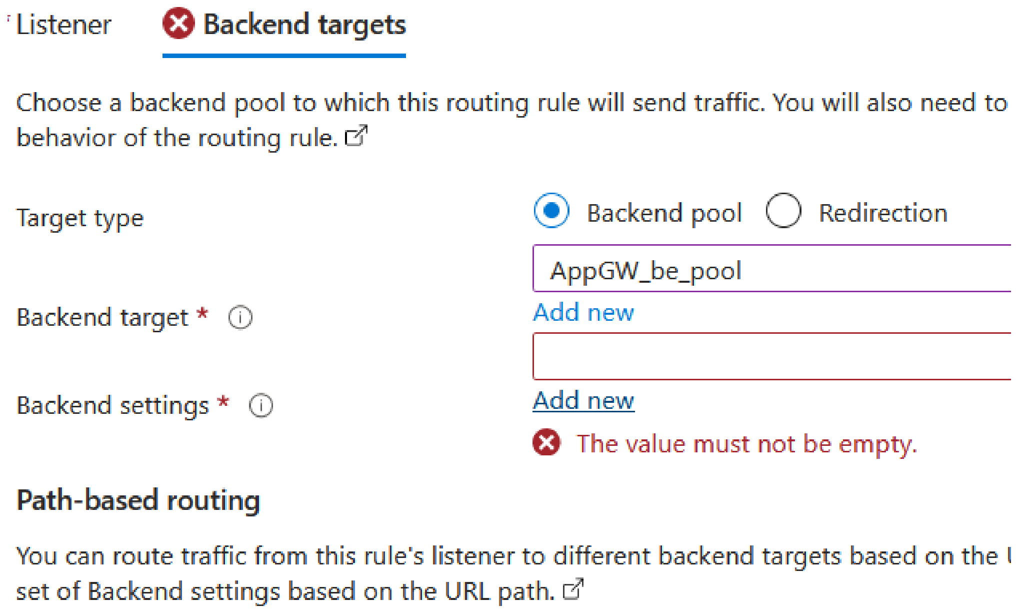
Custom error pages

Show customized error pages for different response codes generated by Application Gateway. This section lets
error pages. Learn more

Bad Gateway - 502	Enter Html file URL
Forbidden - 403	Enter Html file URL

Show more status codes

9. Complete as above then click 'Backend Targets' and click on Backend settings 'Add New'



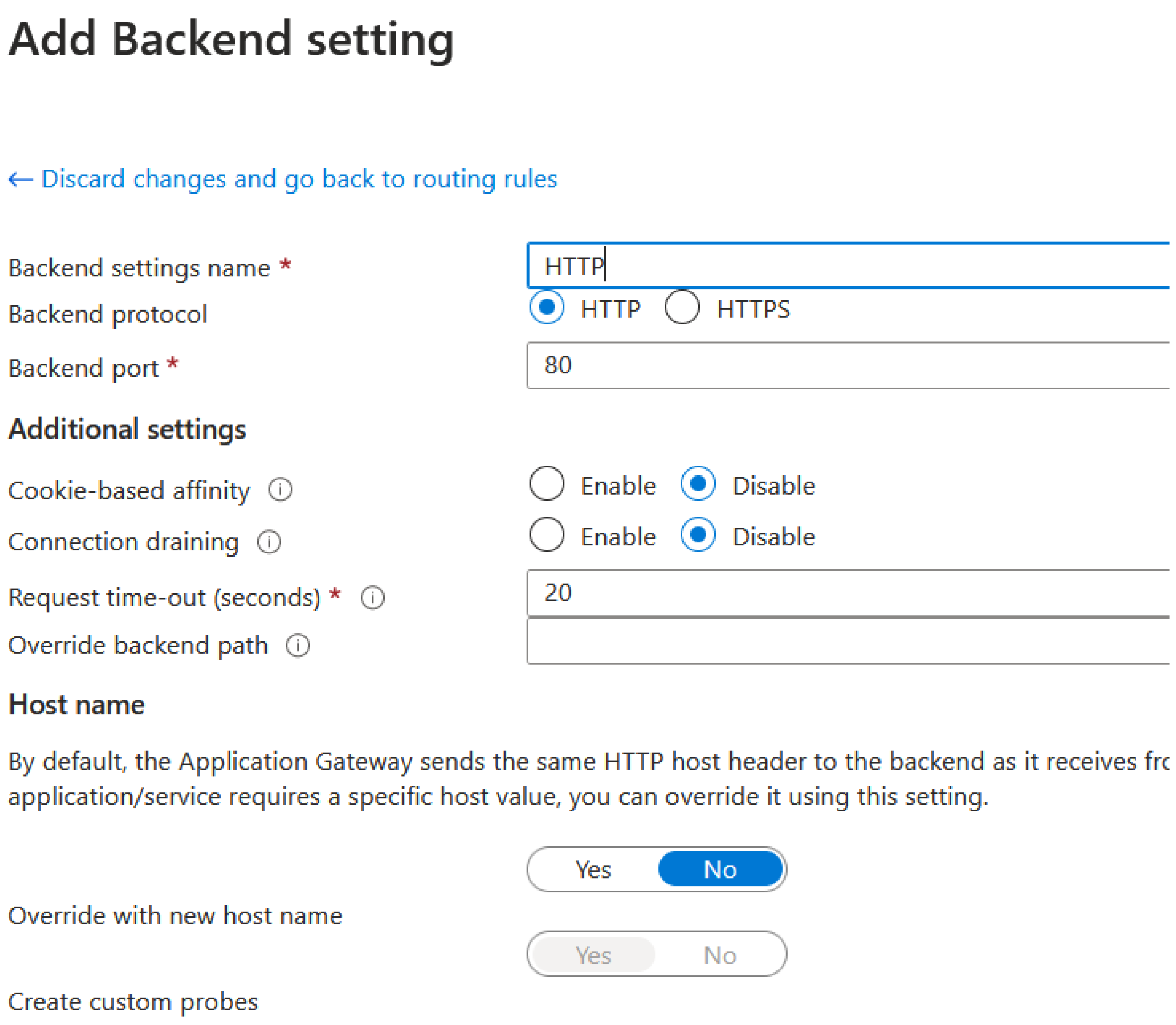
Listener | Backend targets

Choose a backend pool to which this routing rule will send traffic. You will also need to
behavior of the routing rule.

Target type	[✓ Backend pool] Redirection
Backend target *	AppGW_be_pool Add new
Backend settings *	 Add new The value must not be empty.

Path-based routing

You can route traffic from this rule's listener to different backend targets based on the
set of Backend settings based on the URL path.



Add Backend setting

← Discard changes and go back to routing rules

Backend settings name *	HTTP
Backend protocol	[✓ HTTP] HTTPS
Backend port *	80

Additional settings

Cookie-based affinity	Enable [✓ Disable]
Connection draining	Enable [✓ Disable]
Request time-out (seconds) *	20
Override backend path	

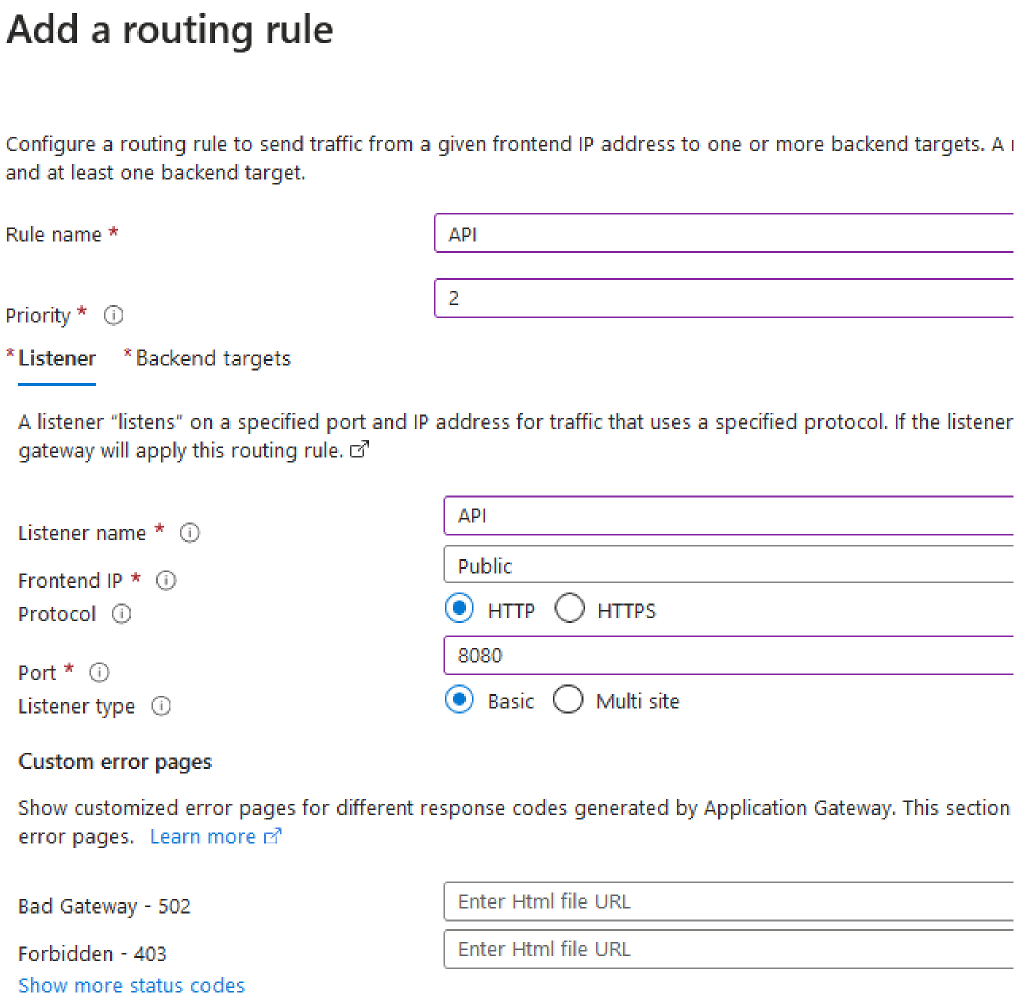
Host name

By default, the Application Gateway sends the same HTTP host header to the backend as it receives fr
application/service requires a specific host value, you can override it using this setting.

Override with new host name	Yes [✓ No]
Create custom probes	Yes No

10. Complete as above then click 'Add', then 'Add' again

11. Again, click on ‘Add routing rule’



Add a routing rule

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

Rule name *	API
Priority * ⓘ	2

*Listener   *Backend targets

A listener "listens" on a specified port and IP address for traffic that uses a specified protocol. If the listener
gateway will apply this routing rule.

Listener name * ⓘ	API
Frontend IP * ⓘ	Public
Protocol ⓘ	[✓ HTTP] HTTPS
Port * ⓘ	8080
Listener type ⓘ	[✓ Basic] Multi site

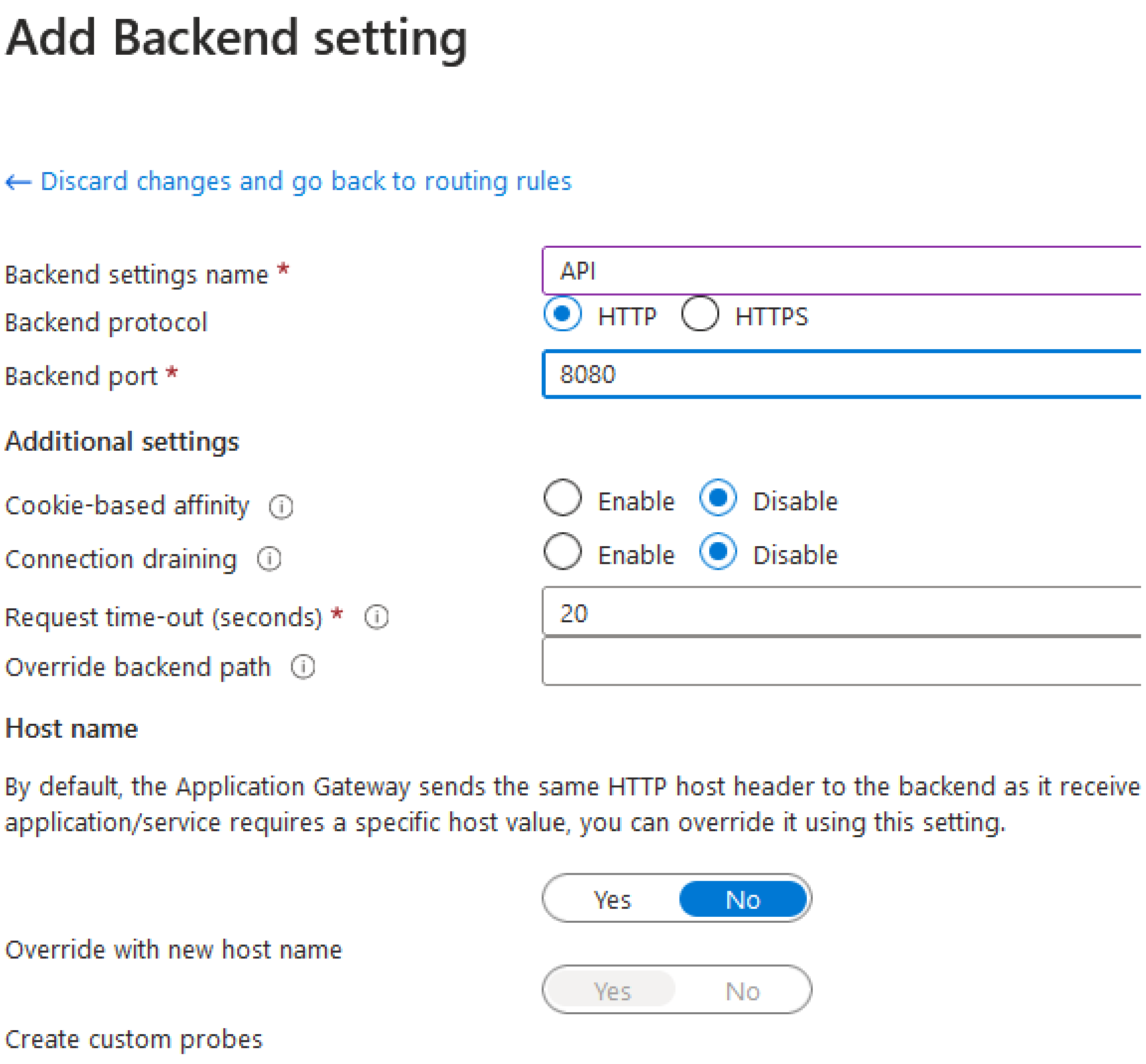
Custom error pages

Show customized error pages for different response codes generated by Application Gateway. This section
error pages. Learn more

Bad Gateway - 502	Enter Html file URL
Forbidden - 403	Enter Html file URL

Show more status codes

12. Complete as above then click on ‘Backend Targets’ and on Backend Settings select ‘Add New’



Add Backend setting

← Discard changes and go back to routing rules

Backend settings name *	API
Backend protocol	[✓ HTTP] HTTPS
Backend port *	8080

Additional settings

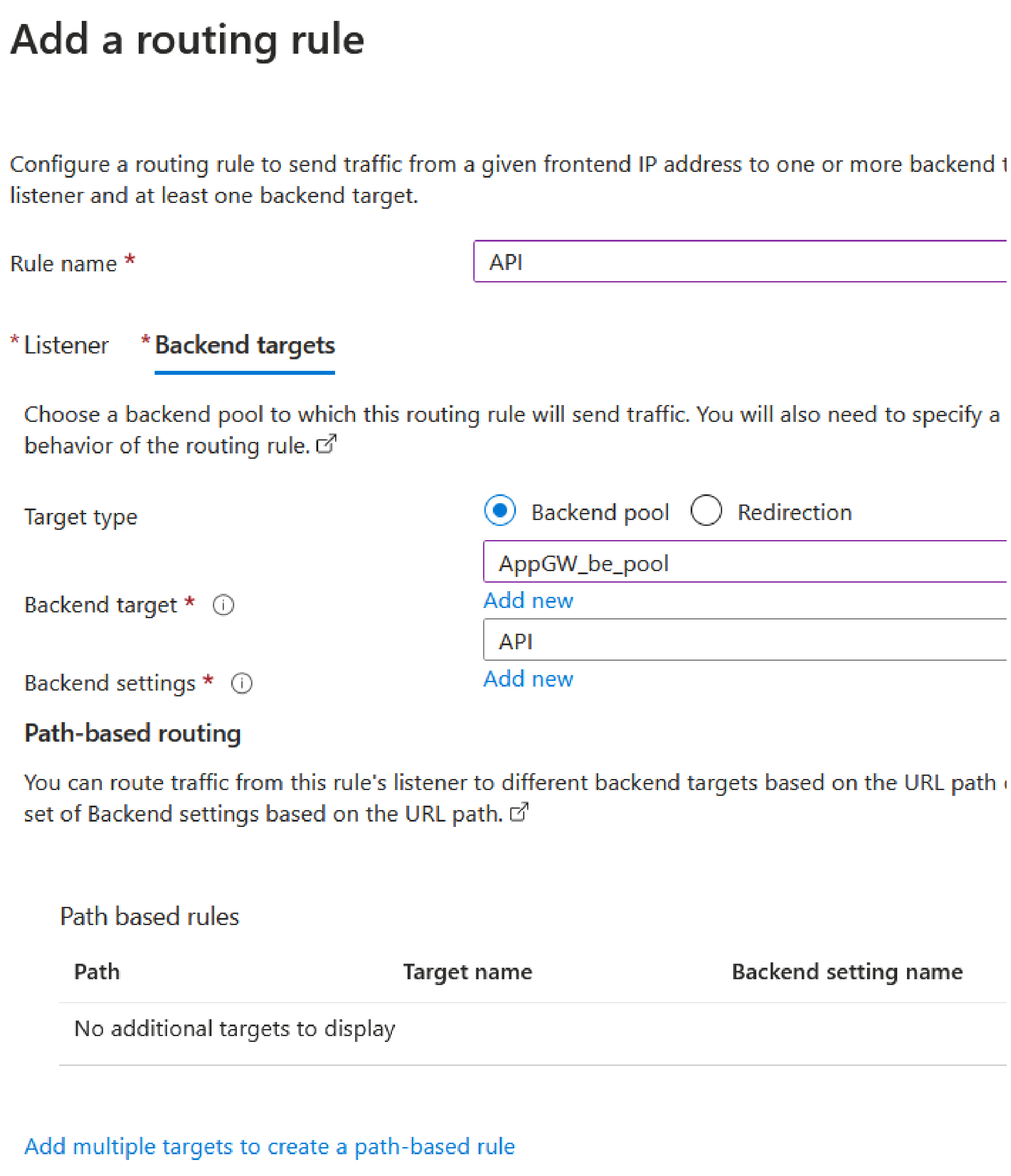
Cookie-based affinity ⓘ	Enable [✓ Disable]
Connection draining ⓘ	Enable [✓ Disable]
Request time-out (seconds) * ⓘ	20
Override backend path ⓘ	

Host name

By default, the Application Gateway sends the same HTTP host header to the backend as it receive
application/service requires a specific host value, you can override it using this setting.

Override with new host name	Yes [✓ No]
Create custom probes	Yes No

13.



Add a routing rule

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

Rule name * API

*Listener ***Backend targets**

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

Target type [✓ Backend pool] Redirection

Backend target * AppGW_be_pool
Add new

Backend settings * API
Add new

Path-based routing

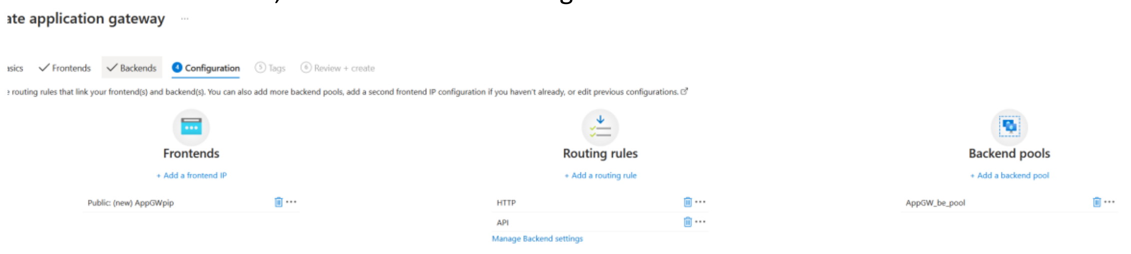
You can route traffic from this rule's listener to different backend targets based on the URL path
set of Backend settings based on the URL path.

Path based rules

Path	Target name	Backend setting name
No additional targets to display		

Add multiple targets to create a path-based rule

14.
15. When it looks like this , Click ‘Add’ then ‘Add’ again



ate application gateway

asics ✓ Frontends ✓ Backends ❹ Configuration ⑤ Tags ⑥ Review + create

routing rules that link your frontend(s) and backend(s). You can also add more backend pools, add a second frontend IP configuration if you haven't already, or edit previous configurations.

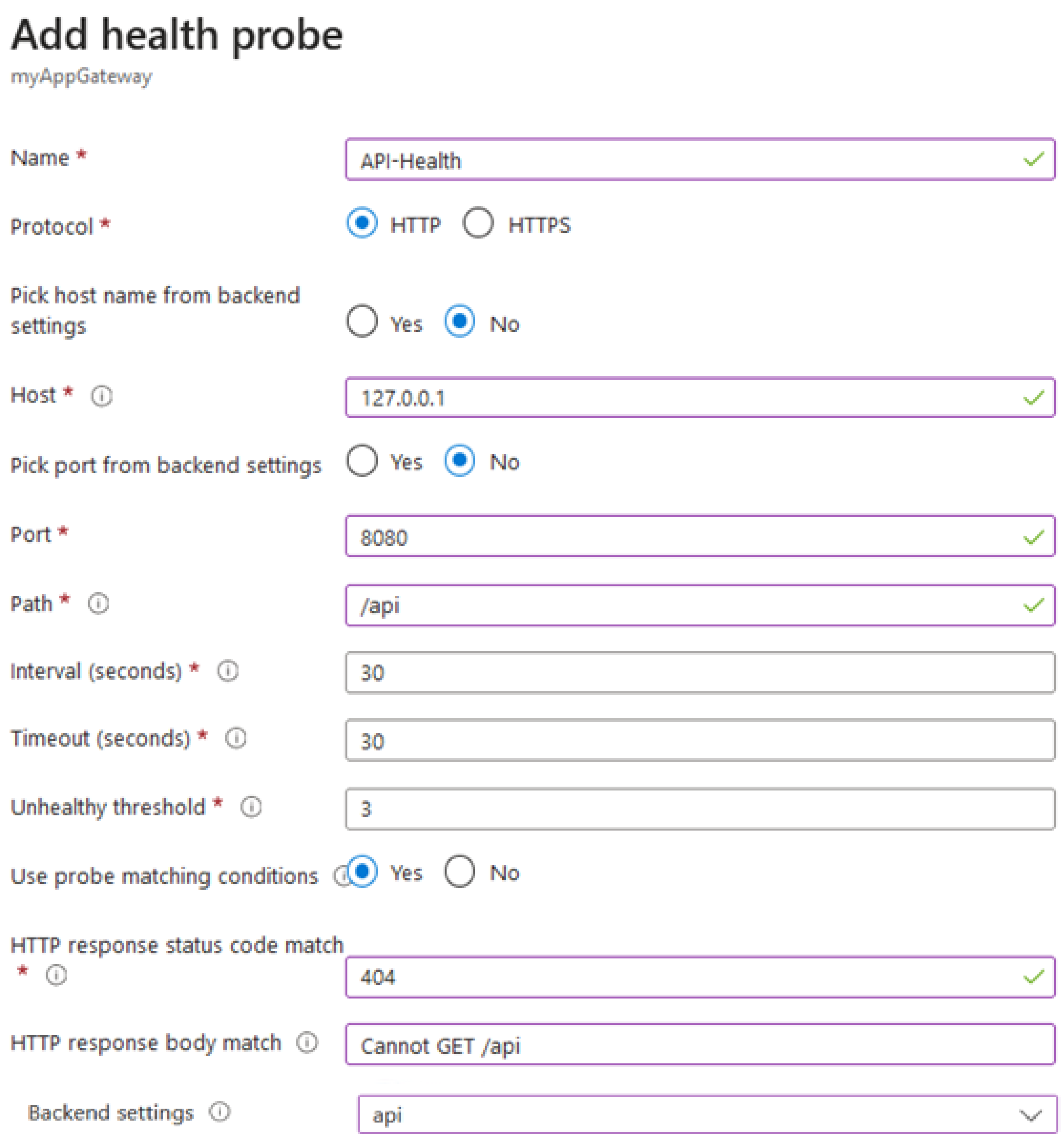
Frontends
+ Add a frontend IP
Public: (new) AppGWpip

Routing rules
+ Add a routing rule
HTTP
API
Manage Backend settings

Backend pools
+ Add a backend pool
AppGW_be_pool

16.
17. It should now look like this, if so go to ‘Next: Tags’ then ‘Review and Create’, Then ‘Create’
18. This will take up to 15 minutes to provision.

19. Once provisioning is complete, return to the Application Gateway configuration and navigate to 'Health Rules'
20. Create new health rule that matches the following configuration:
21.



Add health probe

myAppGateway

Name *	API-Health
Protocol *	[✓ HTTP] HTTPS
Pick host name from backend settings	Yes [✓ No]
Host *	127.0.0.1
Pick port from backend settings	Yes [✓ No]
Port *	8080
Path *	/api
Interval (seconds) *	30
Timeout (seconds) *	30
Unhealthy threshold *	3
Use probe matching conditions	[✓ Yes] No
HTTP response status code match *	404
HTTP response body match	Cannot GET /api
Backend settings	api

22.

23. Click on test, then 'Add'
24. Application Gateway configuration is now complete.
25. You now have to update some webserver configurations to complete this upgrade, I have provided a script to do this. Back in the Azure shell run the following script:

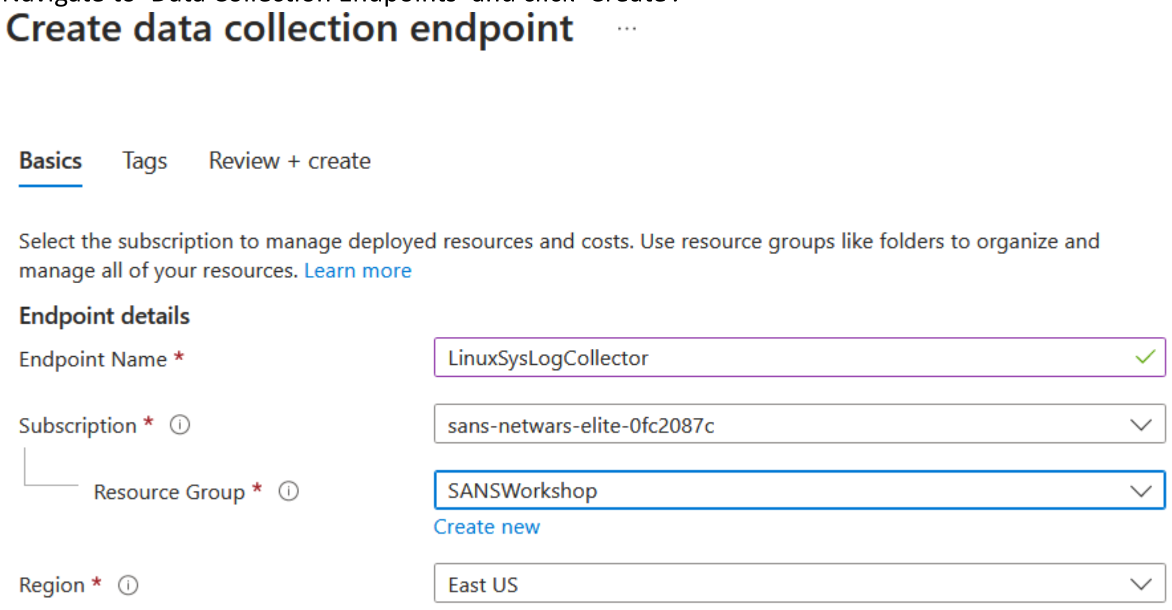
```
26.      ./appGwFix.sh
```

Your new webserver IP will be shown, you can now return to the attack website to generate some traffic for your logs.

Objective: Configure log visibility

To get logs from this webserver we need to use the AMA logging agent, Since Ubuntu 22.04 is not supported for the Log Analytics agent.

1. Navigate to 'Data Collection Endpoints' and click 'Create'.



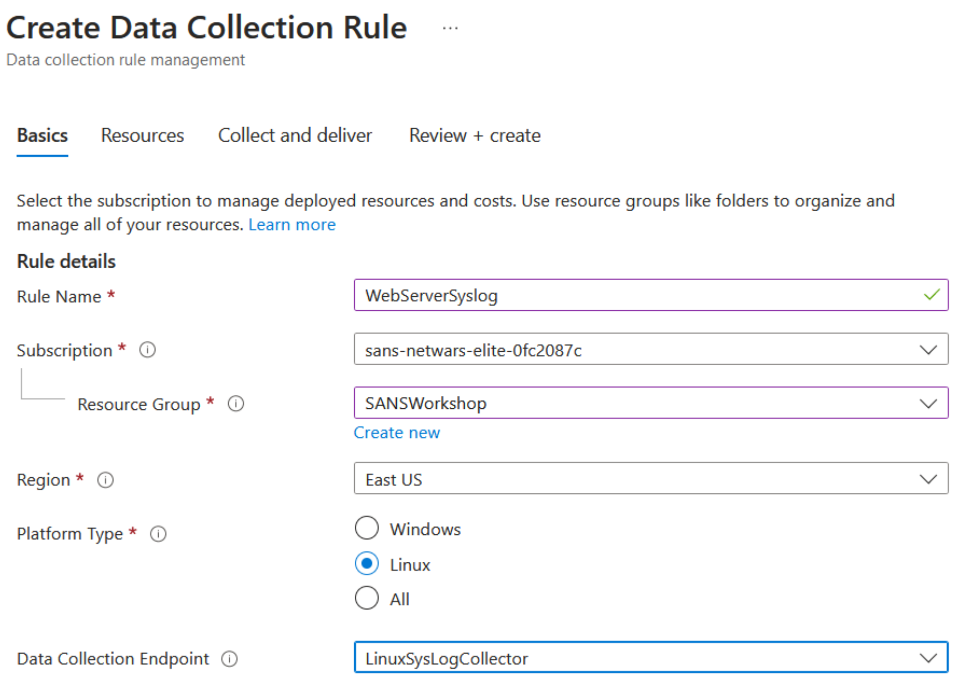
Create data collection endpoint

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

Endpoint details

Endpoint Name *	LinuxSysLogCollector
Subscription *	sans-netwars-elite-0fc2087c
Resource Group *	SANSWorkshop
Region *	East US

2.
3. Click 'Review and Create
4. Now Navigate to 'Monitor' and under 'Settings' select 'Data Collection Rules, and click 'Create'



Create Data Collection Rule

Data collection rule management

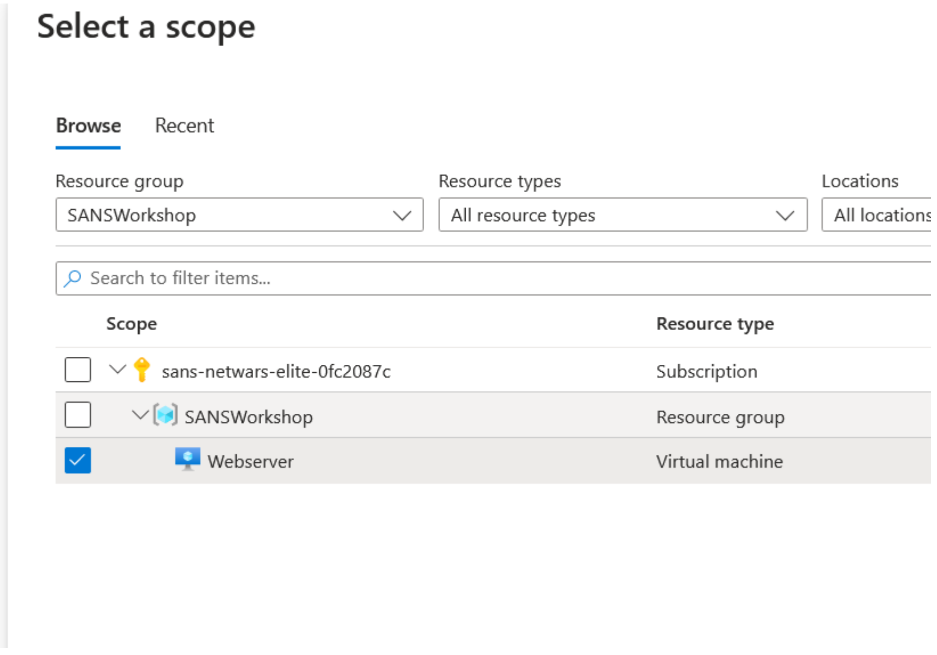
Basics | Resources | Collect and deliver | Review + create

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

Rule details

Rule Name *	WebServerSyslog
Subscription *	sans-netwars-elite-0fc2087c
Resource Group *	SANSWorkshop
Region *	East US
Platform Type *	Windows [✓ Linux] All
Data Collection Endpoint	LinuxSysLogCollector

5.
6. Click on 'Resources' and the 'Add Resources' the select the Webserver:



Select a scope

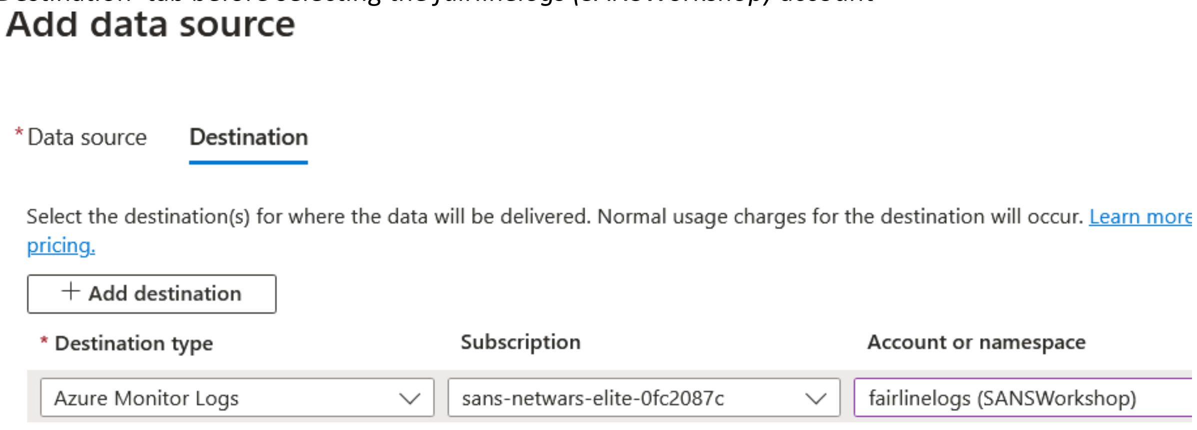
Browse | Recent

Resource group	Resource types	Locations
SANSWorkshop	All resource types	All locations

Search to filter items...

	Scope	Resource type
☐	sans-netwars-elite-0fc2087c	Subscription
☐	SANSWorkshop	Resource group
✓	Webserver	Virtual machine

7. Click on ‘Collect and Deliver’ then ‘+Add Data Source’ and select ‘Linux Syslog’, then ‘Destination’ tab before selecting the *fairlinelogs (SANSWorkshop)* account



Add data source

*Data source Destination

Select the destination(s) for where the data will be delivered. Normal usage charges for the destination will occur. Learn more pricing.

+ Add destination

* Destination type	Subscription	Account or namespace
Azure Monitor Logs	sans-netwars-elite-0fc2087c	fairlinelogs (SANSWorkshop)

8. Then click ‘review and Create’

If you now navigate to the Virtual Machine ‘Webserver’ and navigate to ‘Extensions and applications’ you should see 2 agents added, these are you data collector agents and within a few minutes you should start receiving logs.

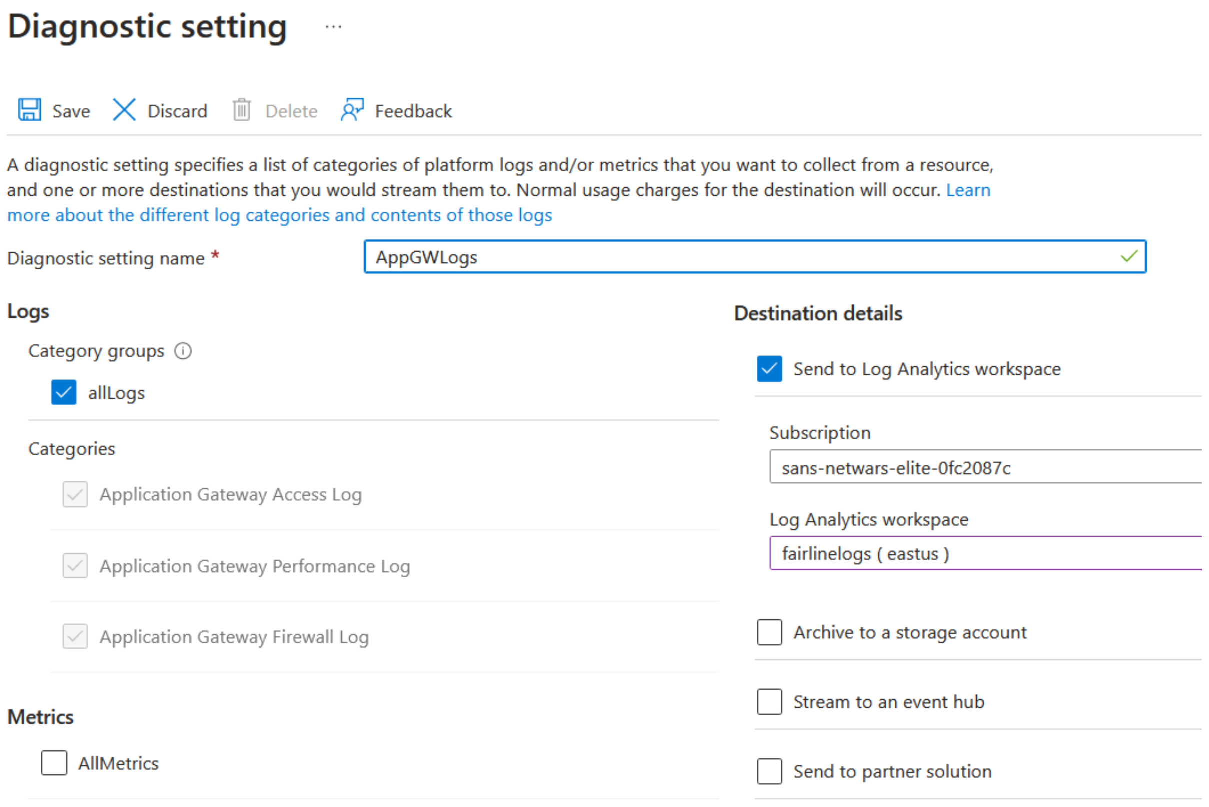
Try navigating to ‘Logs’ under the Monitoring section and running this query:

```
Syslog
| top 100 by TimeGenerated desc
```

Objective: Application Gateway adjustments and logging

Now you should check your application gateway is working properly and configure some additional logging.

1. Navigate to 'Diagnostic settings' and click on '+Add diagnostic Setting'



Diagnostic setting ...

A diagnostic setting specifies a list of categories of platform logs and/or metrics that you want to collect from a resource, and one or more destinations that you would stream them to. Normal usage charges for the destination will occur. Learn more about the different log categories and contents of those logs

Diagnostic setting name * AppGWLogs

Logs

Category groups

- [✓] allLogs

Categories

- [✓] Application Gateway Access Log
- [✓] Application Gateway Performance Log
- [✓] Application Gateway Firewall Log

Metrics

- [ ] AllMetrics

Destination details

- [✓] Send to Log Analytics workspace

Subscription: sans-netwars-elite-0fc2087c

Log Analytics workspace: fairlinelogs (eastus)

- [ ] Archive to a storage account
- [ ] Stream to an event hub
- [ ] Send to partner solution

2. Click on Save, Test the website and if everything is working:

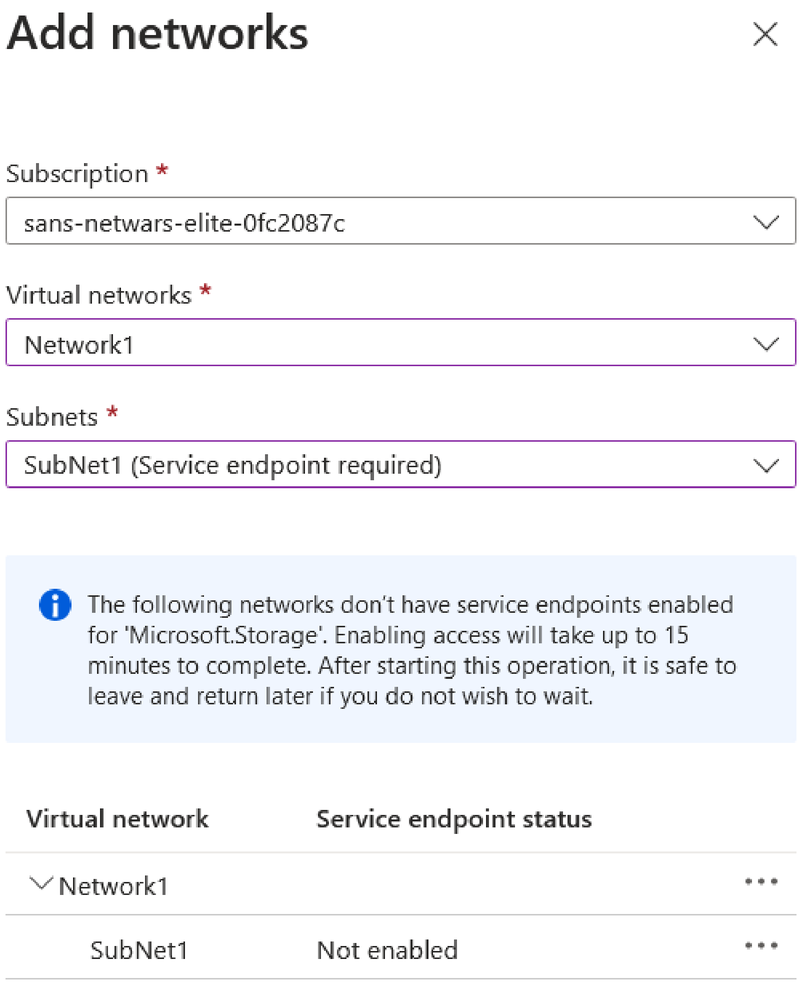
Now we have to update the NSG to stop direct traffic from hitting the webserver so all traffic now has to traverse the Application Gateway.

1. Navigate to the resource group and click on the NSG1
2. Click on the WebApp_Inbound rule and change the source to 'Service Tag' then select the 'Source Service Tag' *'GatewayManager'*
3. Click 'Save' and wait a few minutes before testing the AppGW IP and Old IP for access to the webserver.

Bonus Objective: Add security to Storage Account

You can also add security to the Storage account which is currently public, you can do this configuring the Network rules to only allow access from the webserver vNet and Subnet.

1. Navigate to the Storage account from the resource group
2. Click on ‘Networking’
3. Select ‘Enabled from selected virtual networks and IP addresses’



Add networks

Subscription *
sans-netwars-elite-0fc2087c

Virtual networks *
Network1

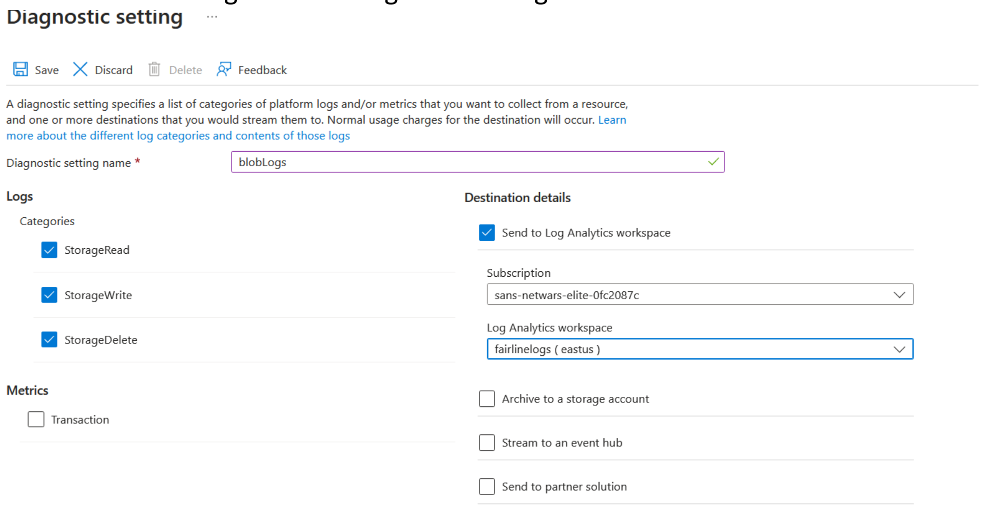
Subnets *
SubNet1 (Service endpoint required)

The following networks don't have service endpoints enabled for 'Microsoft.Storage'. Enabling access will take up to 15 minutes to complete. After starting this operation, it is safe to leave and return later if you do not wish to wait.

Virtual network	Service endpoint status
Network1	
SubNet1	Not enabled

4.
5. Configure as above and click ‘Enable’ then wait a few minutes until you are notified the endpoint has been created.
6. Now click ‘Add’ and then ‘Save’ at the top of the window.
7. On the left hand side, scroll down to ‘Diagnostics settings’ and click on ‘Blob’

8. ‘Click on ‘+Add diagnostic Settings’ and configure as below:


Diagnostic setting

A diagnostic setting specifies a list of categories of platform logs and/or metrics that you want to collect from a resource, and one or more destinations that you would stream them to. Normal usage charges for the destination will occur. Learn more about the different log categories and contents of those logs

Diagnostic setting name * blobLogs

Logs

Categories

- [✓] StorageRead
- [✓] StorageWrite
- [✓] StorageDelete

Metrics

- [ ] Transaction

Destination details

- [✓] Send to Log Analytics workspace

Subscription: sans-netwars-elite-0fc2087c

Log Analytics workspace: fairlinelogs (eastus)

- [ ] Archive to a storage account
- [ ] Stream to an event hub
- [ ] Send to partner solution

You have now configured all defensive and logging agents and services.

Request some further attacks against your web application from here:

https://attack.cy63r.ninja

You can now view in the logs the source of the attacks, the method used and the files acquired.

Below are some hints and tips the types of queries you can use:

Monitor Queries:

Heartbeat : check to make sure logs are arriving

VM:

```
Heartbeat
| where TimeGenerated > ago(1h)
| summarize NoHeartbeatPeriod = now() - max(TimeGenerated) by
Computer
| top 10 by NoHeartbeatPeriod desc
```

Syslog – SSH attack

```
Syslog
| top 100 by TimeGenerated desc
| where Facility == "authpriv"
```

Syslog – User ID of SSH attacks

```
Syslog
| top 100 by TimeGenerated desc
| where Facility != "user"
```

Log Analytics Queries:

All Application Data logging

App gateway:

```
AzureDiagnostics
| where Category == "ApplicationGatewayAccessLog"
```

Application Data Logging successful downloads / connections

```
AzureDiagnostics
| where ResourceType == "APPLICATIONGATEWAYS"

        and OperationName == "ApplicationGatewayAccess"

        and httpStatus_d > 200
```

```
AzureDiagnostics
| where Category == "ApplicationGatewayAccessLog"
| where host_s == "xxx.xxx.xxx.xxx"
| summarize count() by host_s, bin(TimeGenerated, 30m)
| render timechart
```

Total requests by URL

```
AzureDiagnostics
| where ResourceProvider == "MICROSOFT.NETWORK" and Category ==
"ApplicationGatewayAccessLog"
| summarize count() by requestUri_s
```

Requests per minute by URL in timechart

```
AzureDiagnostics
| where ResourceProvider == "MICROSOFT.NETWORK" and Category ==
"ApplicationGatewayAccessLog"
| summarize count() by requestUri_s, bin(TimeGenerated, 1m)
| render timechart
```

Requests of URL by IP address

```
AzureDiagnostics
| where ResourceProvider == "MICROSOFT.NETWORK" and Category ==
"ApplicationGatewayAccessLog"
| summarize count() by requestUri_s,clientIP_s,requestQuery_s
```

Requests resulting in >500+ responses by URL in the last hour

```
AzureDiagnostics
| where ResourceProvider == "MICROSOFT.NETWORK" and Category ==
"ApplicationGatewayAccessLog"
| where httpStatus_d >= 500
| summarize count(httpStatus_d) by httpStatus_d,requestUri_s,
bin(TimeGenerated, 1h)
| order by count_httpStatus_d desc
| project httpStatus_d, requestUri_s, TimeGenerated,
count_httpStatus_d
```

Failed requests over time

```
AzureDiagnostics
| where ResourceProvider == "MICROSOFT.NETWORK" and Category ==
"ApplicationGatewayAccessLog"
| where httpStatus_d >= 400
| parse requestQuery_s with "SERVER-ROUTED=" serverRouted "&"
| extend httpStatus = tostring(httpStatus_d)
| summarize count() by serverRouted, bin(TimeGenerated, 5m)
| render timechart
```