



Protecting Your App Using Azure's Cloud Security Features

Cybersecurity
Project Week





Day 2 Recap

In the second day of your project work, you:

01

Created a key vault.

02

Created a self-signed certificate.

03

Imported and bound your self-signed certificate to your web application.*

04

Created and bound an app service managed certificate.*

05

Answered review questions.

* Except for the free domain users.

Today's Class

Today's class time will include:



Introduction to Azure Front Door



Overview of Day 3 tasks



Project work



Azure Front Door

Additional Azure Security

We have created and secured our web applications' traffic using SSL certificates, but our apps are still subject to attack by malicious actors.

Attacks can include:

- Denial of service to make the web application unavailable.
- Attacks against web vulnerabilities, such as cross-site scripting and SQL injection (which we'll cover in more detail during the Web Vulnerabilities unit).
- Attacks against misconfigurations, such as leaving insecure ports open on the web server.

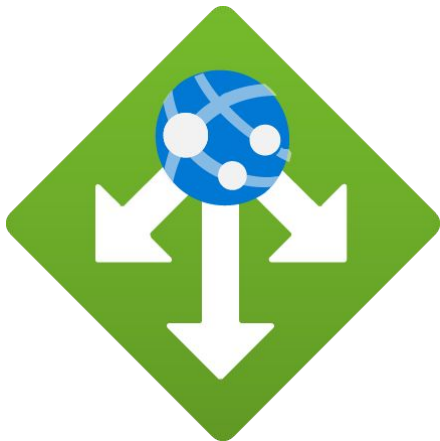


Additional Azure Security

Azure has several technologies that we can use to protect against these attacks:

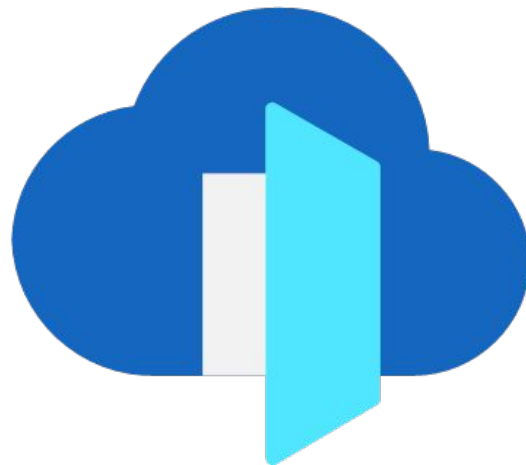
01

Azure Web Application Gateway



02

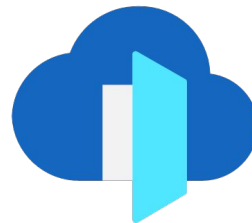
Azure Front Door



Both tools are viable options for securing our web applications.
Let's consider each one's features in order to choose the one that we'll use today.



Azure
Web Application
Gateway



Azure
Front Door

Similarities

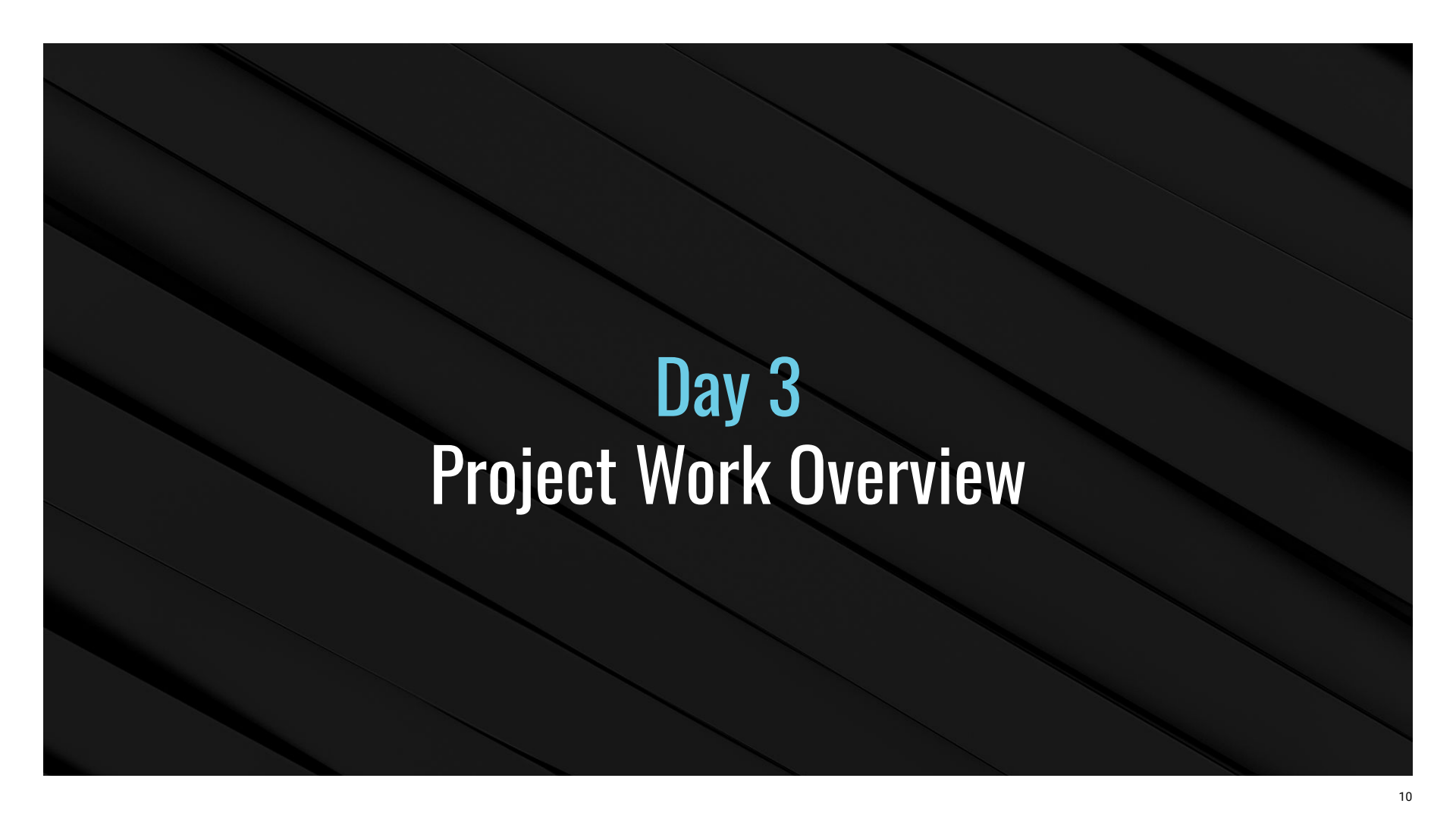
- Both reside in *front* of your web application in order to protect it.
- They work on the Application Layer (7) of the OSI model.
- Their primary solution is a load balancer.
- They can incorporate a web application firewall (WAF) to protect against web vulnerability attacks.
- They have additional features such as URL path-based routing and SSL/TLS termination.

Differences

- The **Web Application Gateway** is more regional and is best suited to protect a web application in a single region in your cloud.
- The **Azure Front Door** is more global and is better suited when you have a variety of regions in a cloud environment.



Azure Front Door is also simpler to implement. So for today's project, you will use its features to protect your web application.

The background of the slide is dark gray with a pattern of diagonal lines that create a sense of depth and movement.

Day 3

Project Work Overview

Day 3 Project Work

Today, we will conclude the project by completing the following steps:

01

Create a Front Door instance.

02

Analyze WAF rule sets.

03

Configure custom WAF rules.

04

Analyze and remediate Security Center recommendations.

05

Answer review questions.

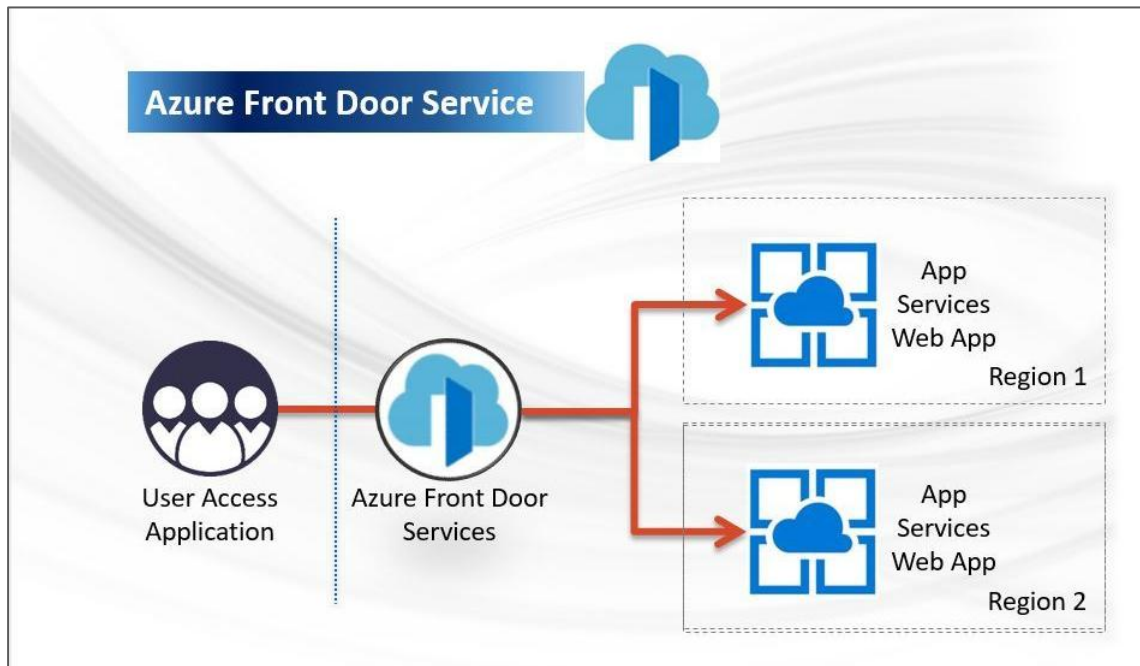
06

Conclude and submit your project.

Step 1

Create a Front Door

Today's first task involves creating an **Azure Front Door** resource to add cloud security protections to your web application.



Steps will be included in the activity guide.

Step 2

Analyze WAF Rule Sets

In this step, you'll analyze Azure Front Door's WAF feature.

Home > PolicyTestGW - Web application firewall > pol1 - Managed rules

pol1 - Managed rules

Application Gateway WAF policy

Search (Ctrl+F)

Save Discard Refresh

A pre-configured rule set is enabled by default. This rule set protects your web application from common threats defined in the top-ten OWASP categories. The default rule set is managed by the Azure WAF service. Rules are updated as needed for new attack signatures. [Learn more](#)

Managed rule set: OWASP_3.0

Expand all Enable Disable

Name	Description	Status
> General		Enabled
> REQUEST-911-METHOD-ENFORCEMENT		Enabled
> REQUEST-913-SCANNER-DETECTION		Enabled
> REQUEST-920-PROTOCOL-ENFORCEMENT		Enabled
> REQUEST-921-PROTOCOL-ATTACK		Enabled
921100	HTTP Request Smuggling Attack	Enabled
<input checked="" type="checkbox"/> 921110	HTTP Request Smuggling Attack	Enabled
921120	HTTP Response Splitting Attack	Enabled
921130	HTTP Response Splitting Attack	Enabled
921140	HTTP Header Injection Attack via headers	Enabled
921150	HTTP Header Injection Attack via payload (CR/LF detected)	Enabled
921151	HTTP Header Injection Attack via payload (CR/LF detected)	Enabled
921160	HTTP Header Injection Attack via payload (CR/LF and header-name detected)	Enabled
921170	HTTP Parameter Pollution	Enabled
921180	HTTP Parameter Pollution (%{TX.1})	Enabled
> REQUEST-930-APPLICATION-ATTACK-LFI		Enabled
> REQUEST-931-APPLICATION-ATTACK-RFI		Enabled
> REQUEST-932-APPLICATION-ATTACK-RFC		Enabled



LOOK FORWARD

We will analyze the web vulnerabilities that it protects against in a later unit!

Step 3

Configure Custom WAF Rules

You will be provided an attack type scenario and learn how to set up a custom rule within the WAF to protect your web application.

Add custom rule

A custom rule is made up of one or more conditions followed by an action. All custom rules for a WAF policy are match rules. [Learn more about custom rules](#)

Custom rule name *

Status ⓘ Enabled Disabled

Rule type ⓘ Match Rate limit

Priority * ⓘ

Conditions

If

Match type ⓘ Geo location

Match variable
RemoteAddr

Operation
☒ Is ☐ Is not

Country/Region * 0 selected

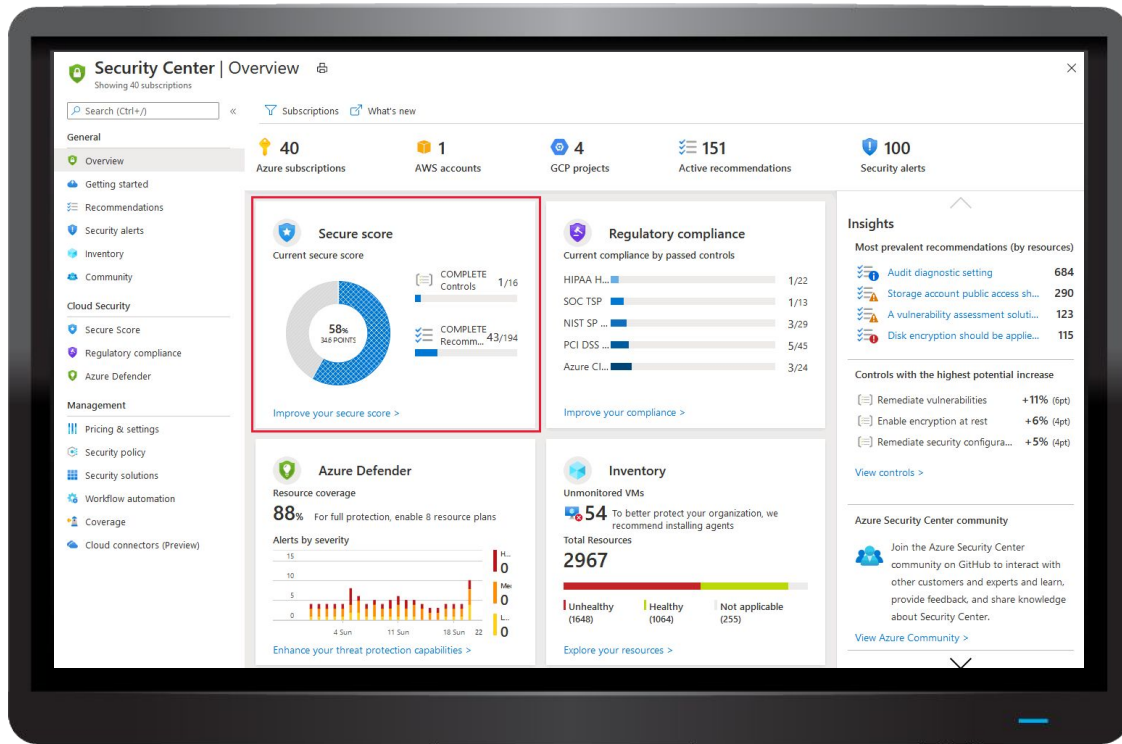
Step 4

Analyze and Remediate Security Center Recommendations

You will view security recommendations from Azure's Security Center tool and fix one of the recommendations from the Security Center dashboard.



Azure's Security Center has many features, such as custom alerting, which will not be covered in this project.





Important



The time it takes for the security recommendations to display can vary.

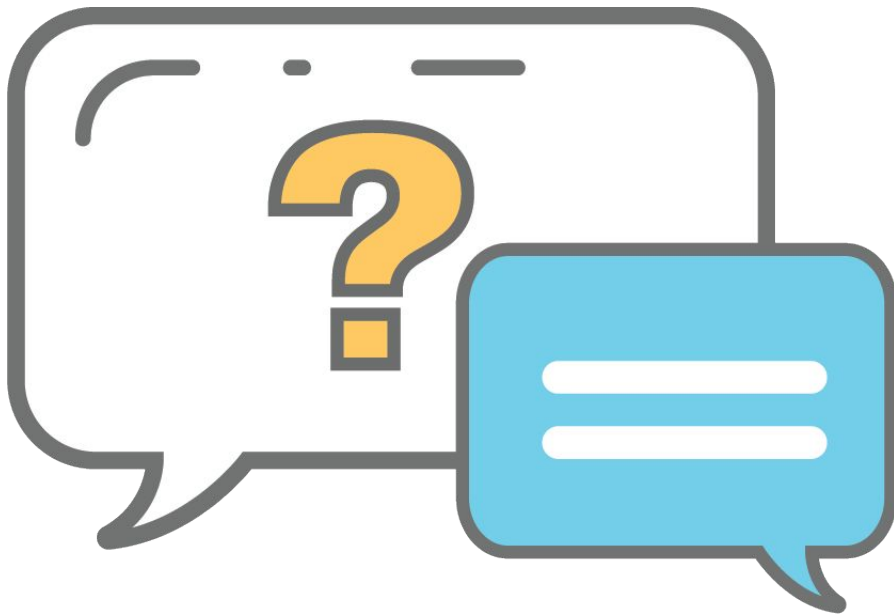
This step is not required to complete the project.

Step 5

Answer Review Questions

Once you complete today's activities, you will answer several questions about the project and how it relates to concepts that we've covered in class.

Feel free to use any resources available (e.g., class notes, slides, online resources) to answer these questions.



Step 6

Conclude and Submit Your Project

At the end of today's class, you will receive a guide with instructions for:



Submitting
project
deliverables.

Disabling any
paid features.

Adding your
project to your
resume.



Important



You are responsible for any costs that you incur by maintaining your website and/or not deleting your resources.

Refer to the provided guide to assist with deleting resources and monitoring future expenses.



Activity: Protect Your Web Application with Azure's Security Features

In this activity, you'll complete the Day 3 tasks of your project.

Suggested Time:

To end of class



Let's Get Started



For the remainder of today's class, you will work on the daily tasks.

While each student is responsible for completing their own project, you can use your classmates, TAs, or the instructor to assist if you have any questions.

You'll add advanced security features to your web application.

Today's activity guide also provides resources explaining how to add your project to your resume and how to discuss your project to networkers and in interview scenarios.

Questions?



Project Work Time

*The
End*