Primena Serverless arhitekture uz AWS Lambda, SQS i S3 servise

Veljko Veljović 1937

Napredno Softversko inženjerstvo

Sadržaj

- Zašto AWS
- Rešenje uz pomoć AWS-a
- AWS SQS
- AWS Lambda
- AWS S3
- AWS vs Azure
- Implementacija projekta

AWS

Zašto AWS?

- problemi sa fizičkim serverima
- problemi sa održavanjem
- skalabilnost
- dostupnost
- sigurnost

Rešenje uz pomoć AWS-a

Rešenje uz pomoć AWS servisa
Platforma Amazon Web Services (AWS) omogućava efikasno rešavanje izazova modernih aplikacija kao što su skalabilnost, dostupnost i bezbednost. Umesto manuelnog upravljanja infrastrukturom, AWS pruža skup servisa koji omogućavaju automatizaciju i sigurnost sistema.

- Automatsko skaliranje
- Visoka dostupnost
- Lakše održavanje
- Brz start i razvoj

AWS SQS

Amazon SQS (Simple Queue Service) je u potpunosti upravljani servis za razmenu poruka u redovima, koji omogućava asinhronu komunikaciju između komponenti distribuiranog sistema.

- Asinhrona obrada
- Skalabilnost i otpornost
- Decoupling servisa
- DLQ (Dead Letter Queue)

Zašto koristiti SQS?

Amazon SQS povećava pouzdanost, skalabilnost i fleksibilnost aplikacije. Uklanja uska grla u komunikaciji između mikroservisa i omogućava stabilan rad sistema i pod opterećenjem.

AWS Lambda

AWS Lambda – Serverless izvršavanje koda

- Potpuno serverless
- Event-driven model
- Automatsko skaliranje
- Plaćanje po pozivu
- Jednostavna integracija sa ostalim AWS servisima

Kada koristiti AWS Lambde?

- Obrada događaja
- Automatizacija zadataka
- Mikroservisna arhitektura

AWS S3

Amazon S3 (Simple Storage Service) je servis za čuvanje objekata koji omogućava bezbedno skladištenje i pristup bilo kojoj količini podataka – u bilo kom trenutku i sa bilo kog mesta.

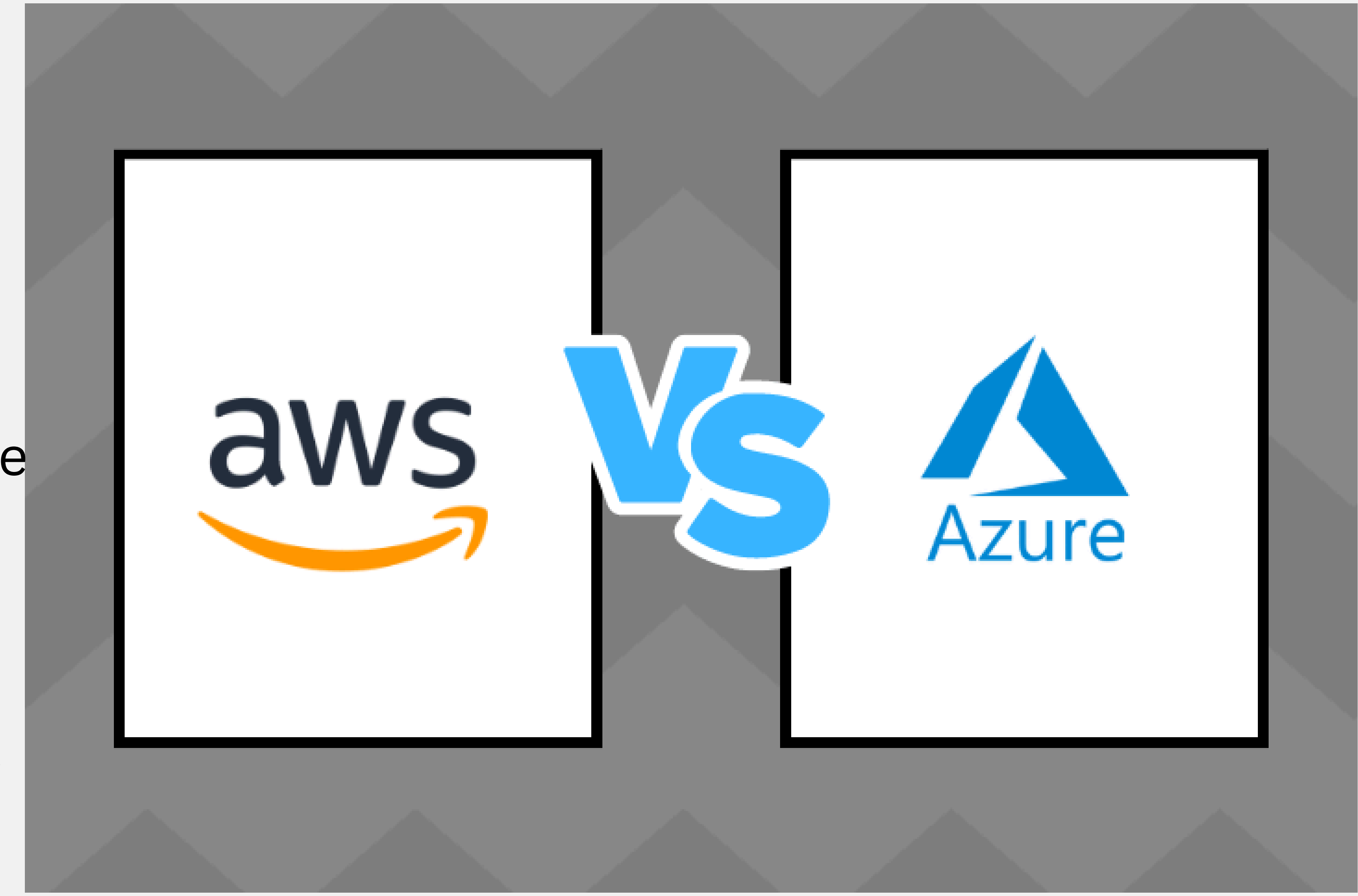
S3 je idealan za:
Staticke fajlove aplikacije (slike, dokumenti, video fajlovi)
Backup i arhiviranje podataka
Logove i analitičke podatke
Hosting statičkih web sajtova

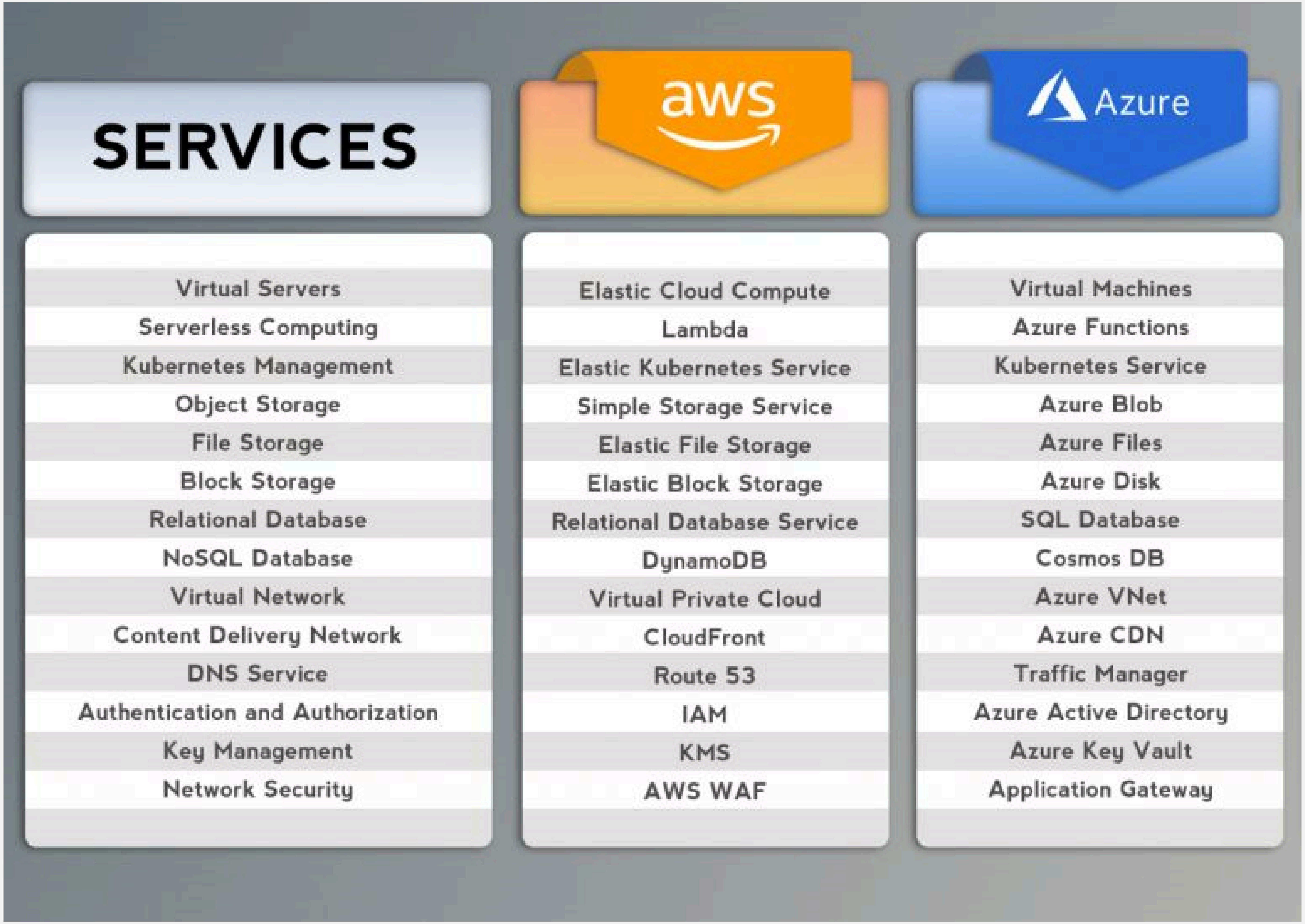
Karakteristike:

- Visoka dostupnost i pouzdanost
- Skalabilnost
- Integracija sa drugim AWS servisima
- Plaćanje po korišćenju

AWS vs Azure

- Azure se bolje integriše sa Microsoft proizvodima, dok je AWS platformno neutralan.
- AWS ima složeniji cenovni model, dok je Azure jednostavniji, naročito za Windows okruženja.
- Free tier i učenje: AWS ima opširniji free tier i lakši je za početnike.
- AWS pokriva najveći broj regiona sa data centrima u svetu.
- AWS ima najveći opseg servisa.

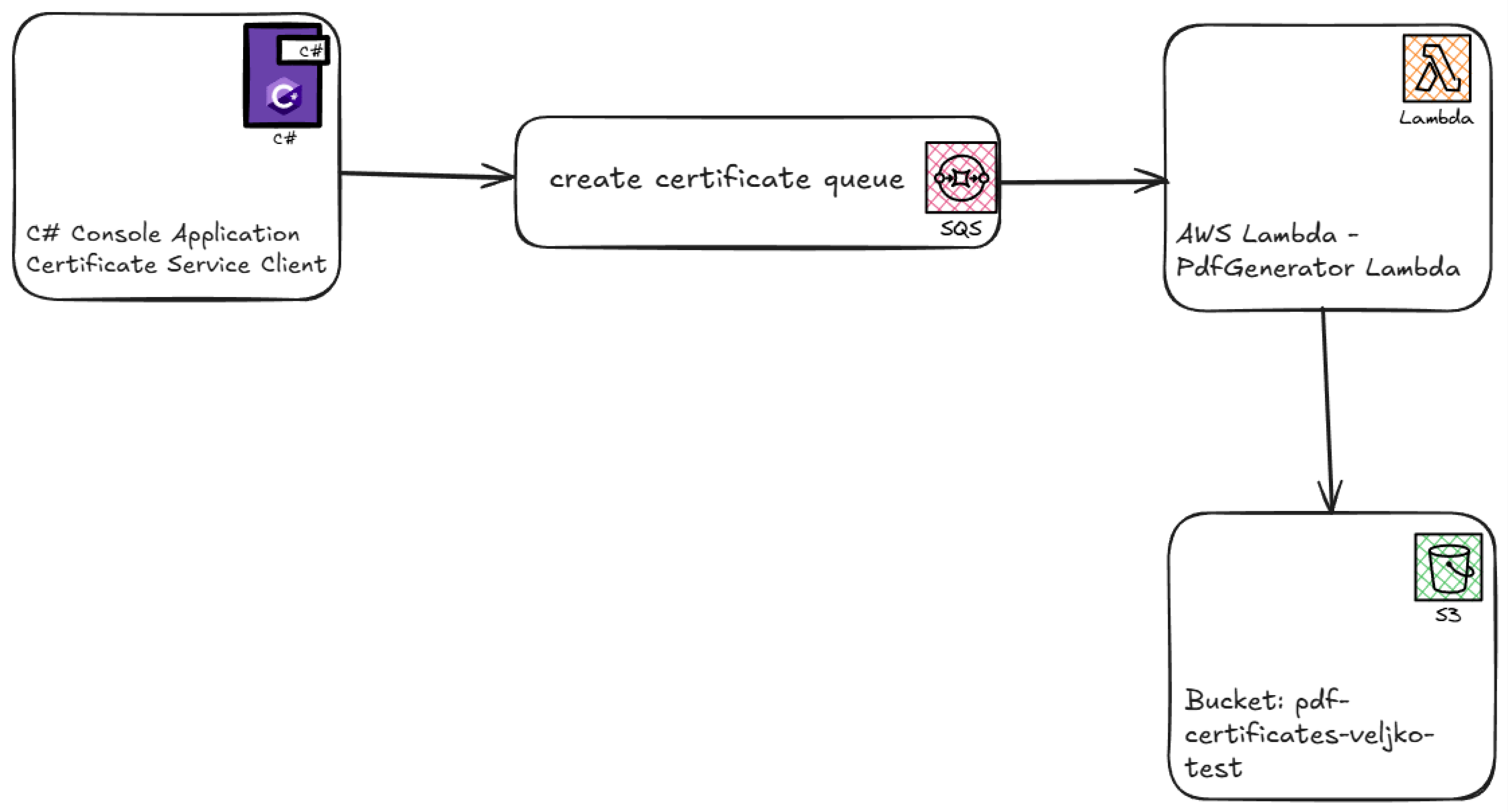




SERVICES	aws	Azure
Virtual Servers	Elastic Cloud Compute	Virtual Machines
Serverless Computing	Lambda	Azure Functions
Kubernetes Management	Elastic Kubernetes Service	Kubernetes Service
Object Storage	Simple Storage Service	Azure Blob
File Storage	Elastic File Storage	Azure Files
Block Storage	Elastic Block Storage	Azure Disk
Relational Database	Relational Database Service	SQL Database
NoSQL Database	DynamoDB	Cosmos DB
Virtual Network	Virtual Private Cloud	Azure VNet
Content Delivery Network	CloudFront	Azure CDN
DNS Service	Route 53	Traffic Manager
Authentication and Authorization	IAM	Azure Active Directory
Key Management	KMS	Azure Key Vault
Network Security	AWS WAF	Application Gateway

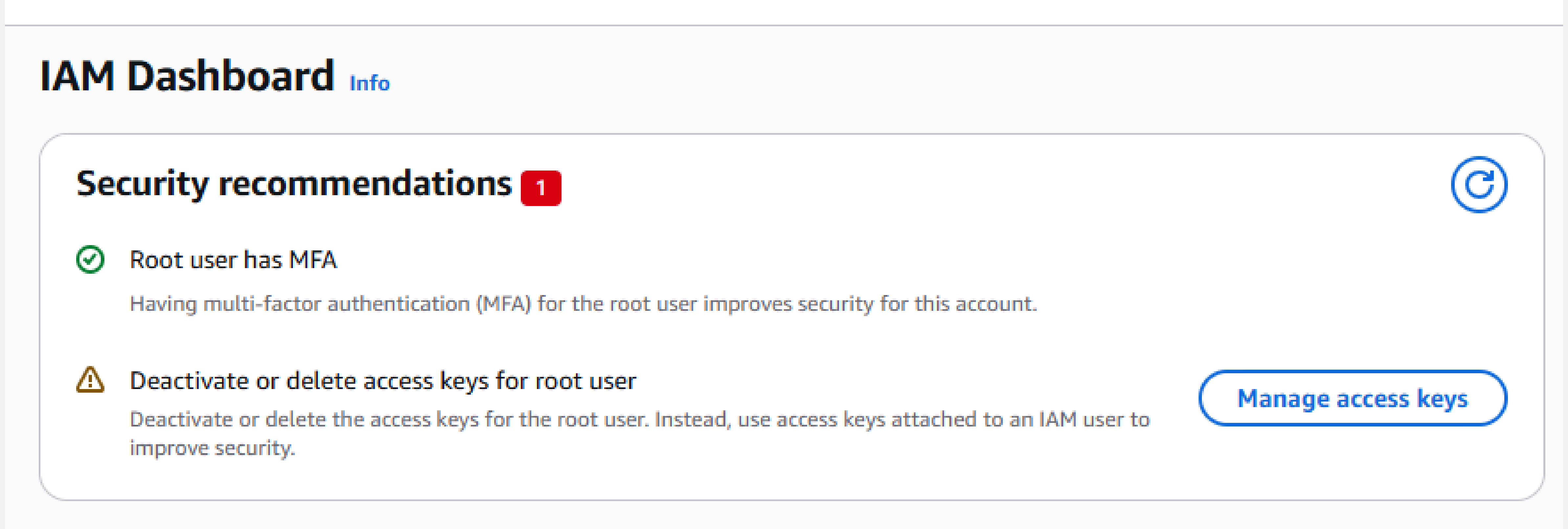
Implementacija Projekta

Dijagram sistema



C#
C# Console Application Certificate Service Client
create certificate queue
SQS
Lambda
AWS Lambda - PdfGenerator Lambda
S3
Bucket: pdf-certificates-veljko-test

Konfiguracija aws-a



IAM Dashboard Info

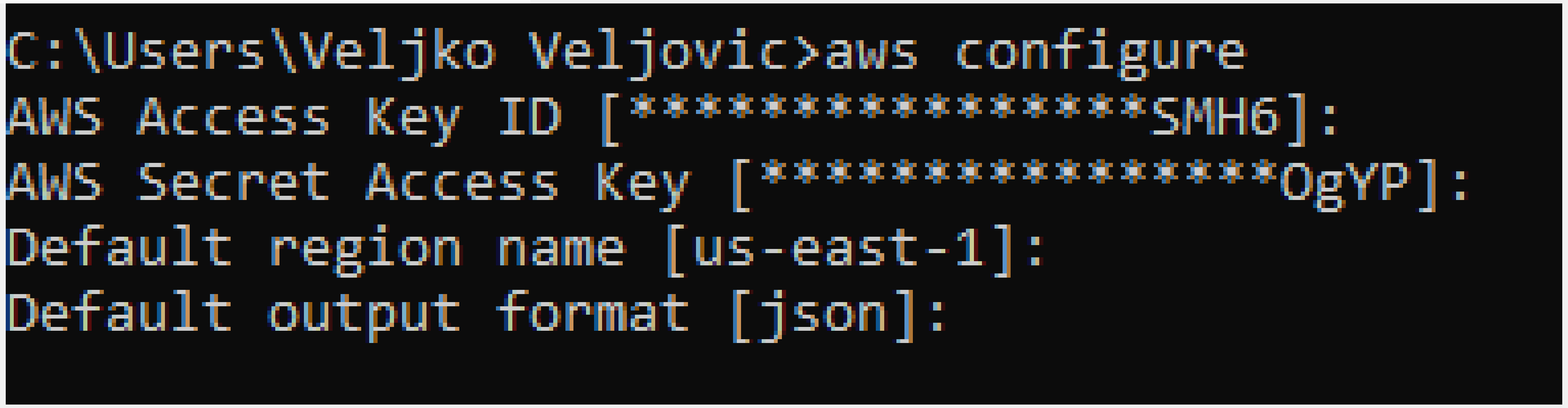
Security recommendations 1

✓ Root user has MFA

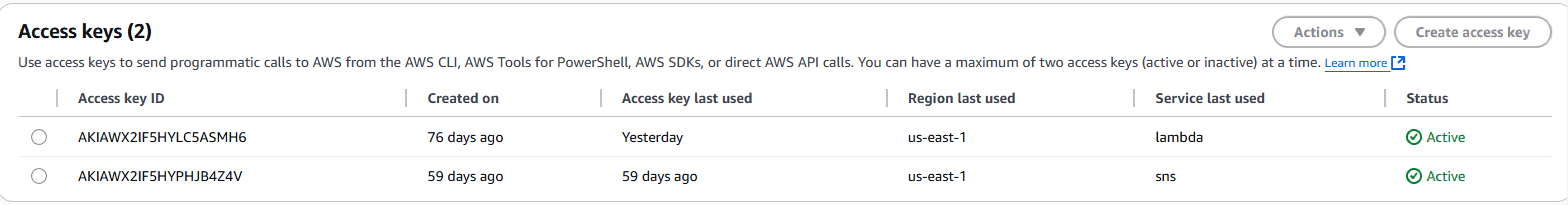
Having multi-factor authentication (MFA) for the root user improves security for this account.

⚠ Deactivate or delete access keys for root user

Deactivate or delete the access keys for the root user. Instead, use access keys attached to an IAM user to improve security.



```
C:\Users\Veljko Veljovic>aws configure
AWS Access Key ID [****************SMH6]:
AWS Secret Access Key [****************OgYP]:
Default region name [us-east-1]:
Default output format [json]:
```

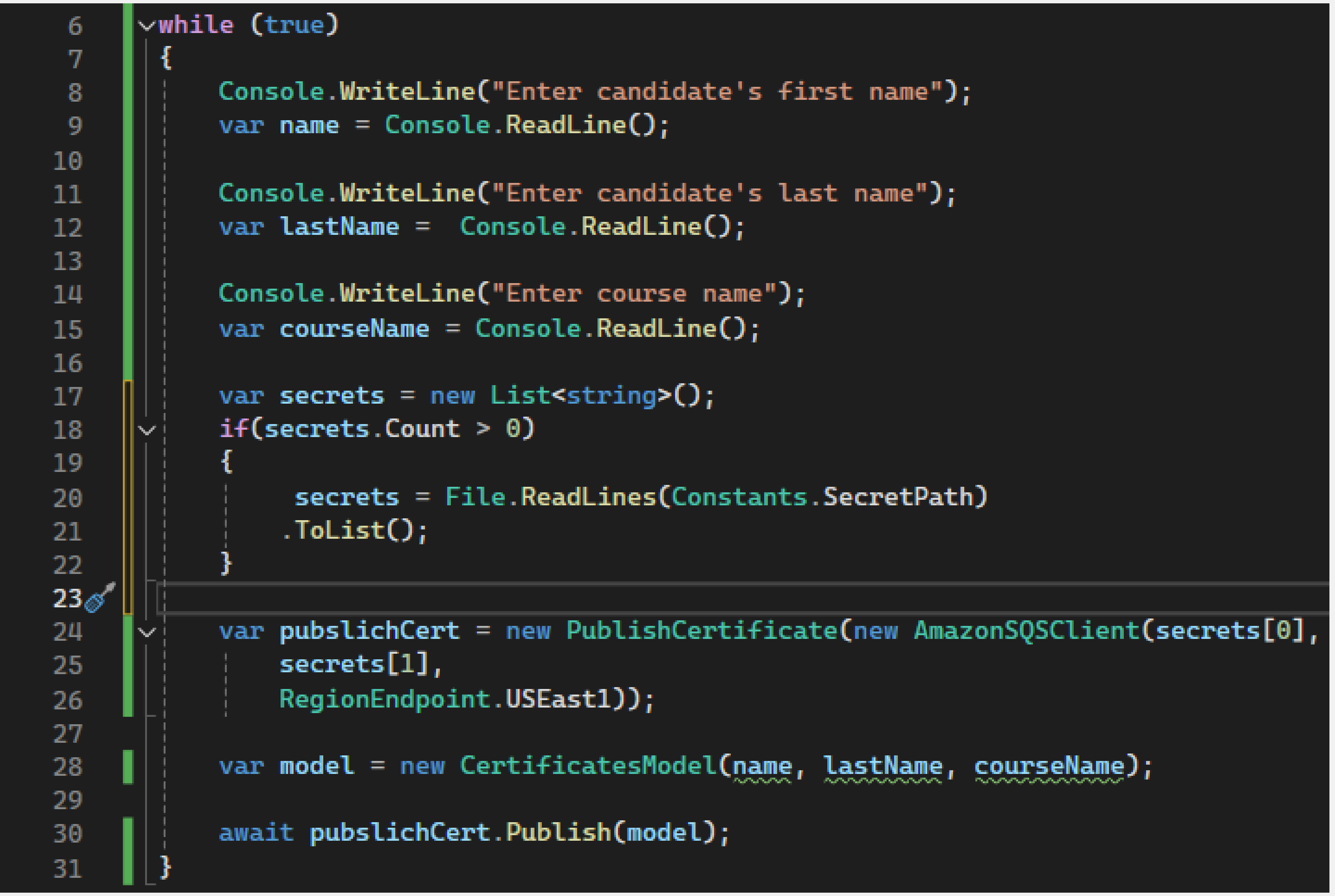


Access keys (2)

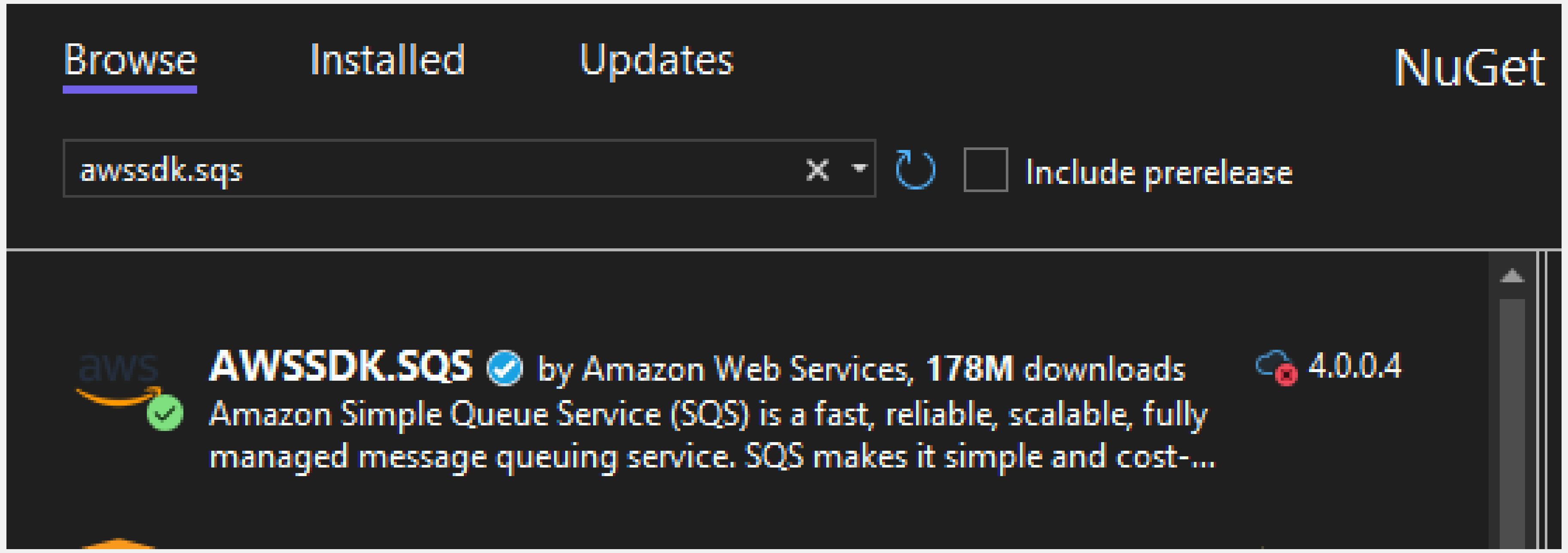
Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. Learn more

	Access key ID	Created on	Access key last used	Region last used	Service last used	Status
○	AKIAWX2IF5HYLC5ASMH6	76 days ago	Yesterday	us-east-1	lambda	✓ Active
○	AKIAWX2IF5HYPHJB4Z4V	59 days ago	59 days ago	us-east-1	sns	✓ Active

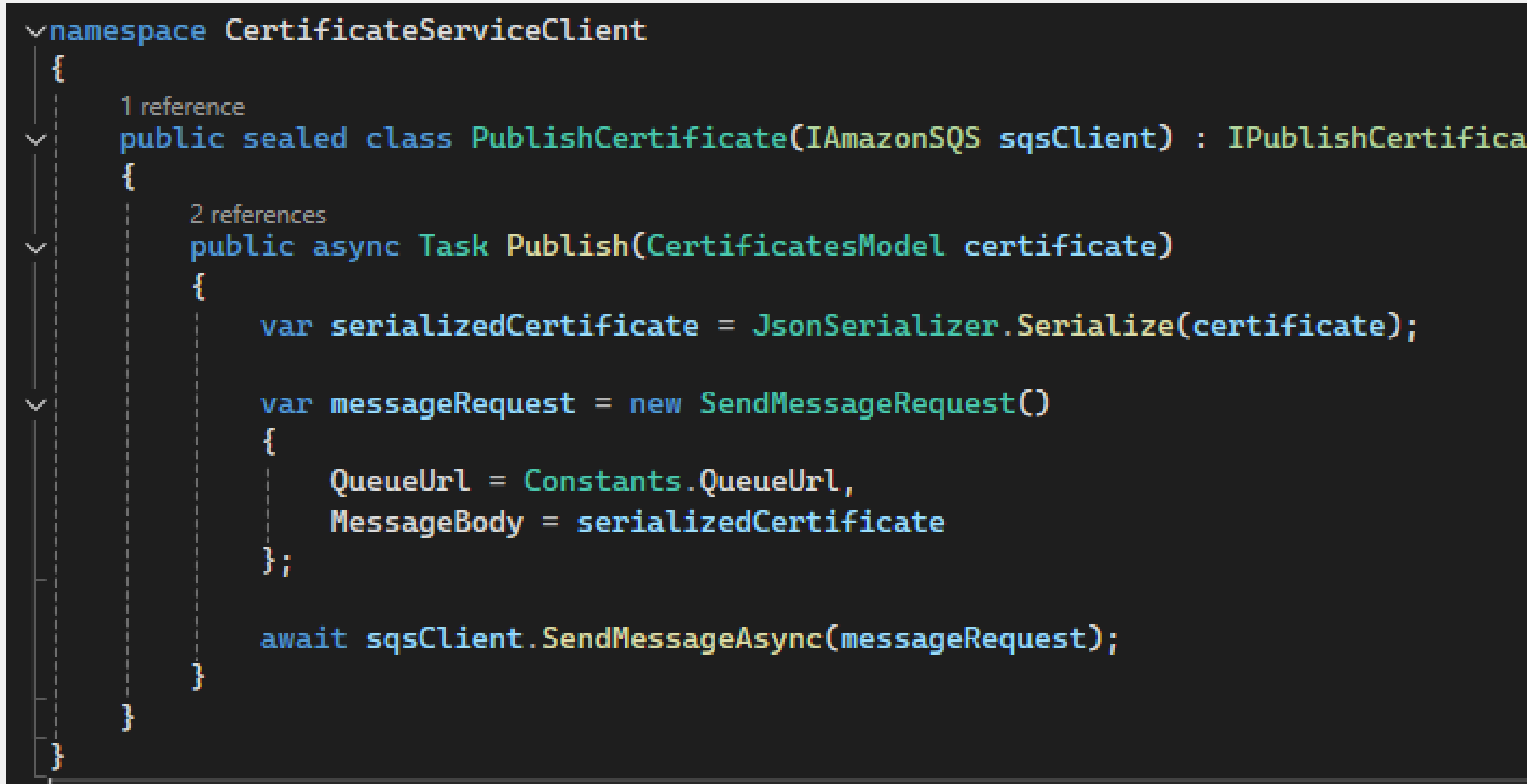
Certificate Service Client



```
while (true)
{
    Console.WriteLine("Enter candidate's first name");
    var name = Console.ReadLine();

    Console.WriteLine("Enter candidate's last name");
    var lastName =  Console.ReadLine();

    Console.WriteLine("Enter course name");
    var courseName = Console.ReadLine();

    var secrets = new List<string>();
    if(secrets.Count > 0)
    {
         secrets = File.ReadLines(Constants.SecretPath)
        .ToList();
    }

    var pubslichCert = new PublishCertificate(new AmazonSQSClient(secrets[0],
        secrets[1],
        RegionEndpoint.USEast1));

    var model = new CertificatesModel(name, lastName, courseName);

    await pubslichCert.Publish(model);
}
```

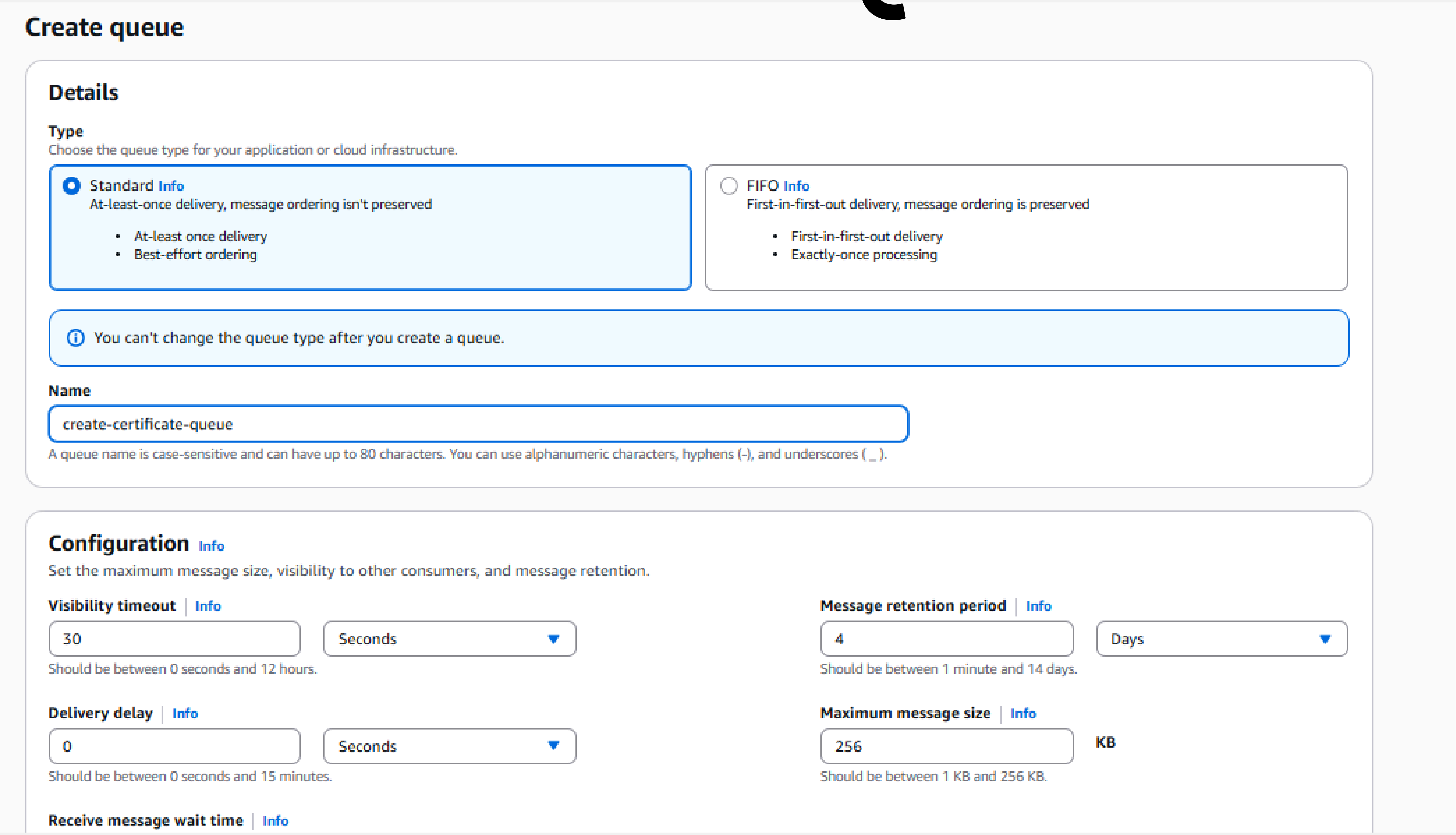


AWSSDK.SQS by Amazon Web Services, **178M** downloads 4.0.0.4
Amazon Simple Queue Service (SQS) is a fast, reliable, scalable, fully managed message queuing service. SQS makes it simple and cost-...



```
namespace CertificateServiceClient
{
    1 reference
    public sealed class PublishCertificate(IAmazonSQS sqsClient) : IPublishCertifica
    {
        2 references
        public async Task Publish(CertificatesModel certificate)
        {
            var serializedCertificate = JsonSerializer.Serialize(certificate);

            var messageRequest = new SendMessageRequest()
            {
                QueueUrl = Constants.QueueUrl,
                MessageBody = serializedCertificate
            };

            await sqsClient.SendMessageAsync(messageRequest);
        }
    }
}
```

SQS



Create queue

Details

Type

Choose the queue type for your application or cloud infrastructure.

● Standard Info
At-least-once delivery, message ordering isn't preserved

- At-least once delivery
- Best-effort ordering

○ FIFO Info
First-in-first-out delivery, message ordering is preserved

- First-in-first-out delivery
- Exactly-once processing

ⓘ You can't change the queue type after you create a queue.

Name

create-certificate-queue

A queue name is case-sensitive and can have up to 80 characters. You can use alphanumeric characters, hyphens (-), and underscores (_).

Configuration Info

Set the maximum message size, visibility to other consumers, and message retention.

Visibility timeout | Info
30 Seconds ▼
Should be between 0 seconds and 12 hours.

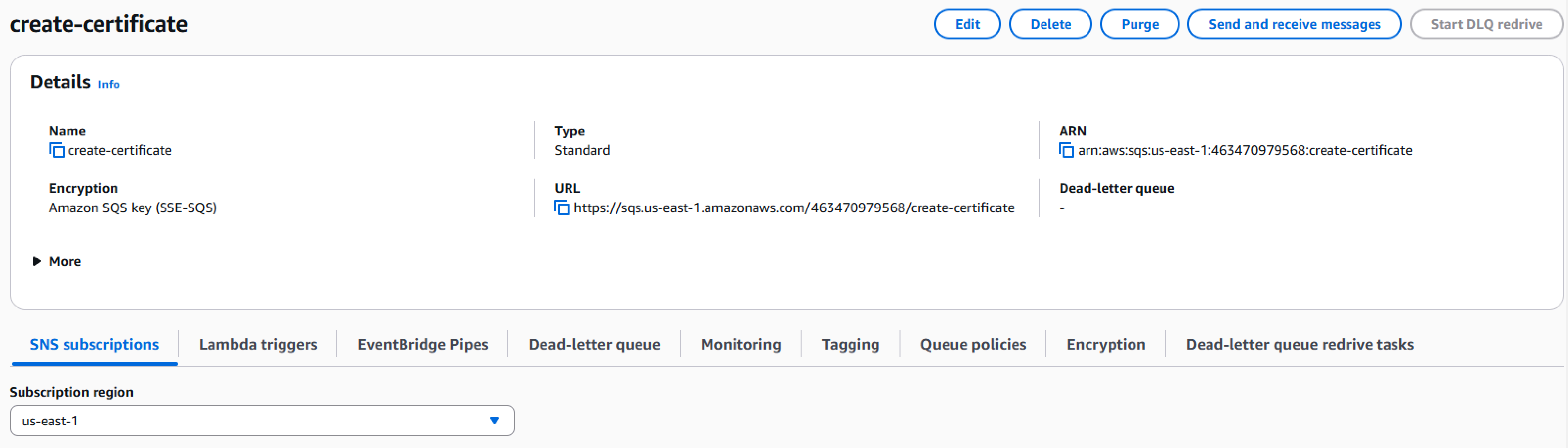
Message retention period | Info
4 Days ▼
Should be between 1 minute and 14 days.

Delivery delay | Info
0 Seconds ▼
Should be between 0 seconds and 15 minutes.

Maximum message size | Info
256 KB
Should be between 1 KB and 256 KB.

Receive message wait time | Info

SQS



create-certificate

Details Info

Name
create-certificate

Type
Standard

ARN
arn:aws:sqs:us-east-1:463470979568:create-certificate

Encryption
Amazon SQS key (SSE-SQS)

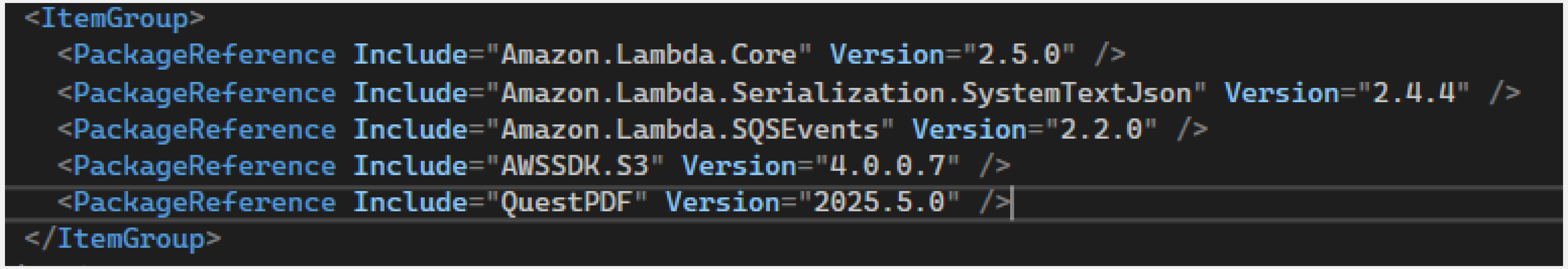
URL
https://sqs.us-east-1.amazonaws.com/463470979568/create-certificate

Dead-letter queue
-

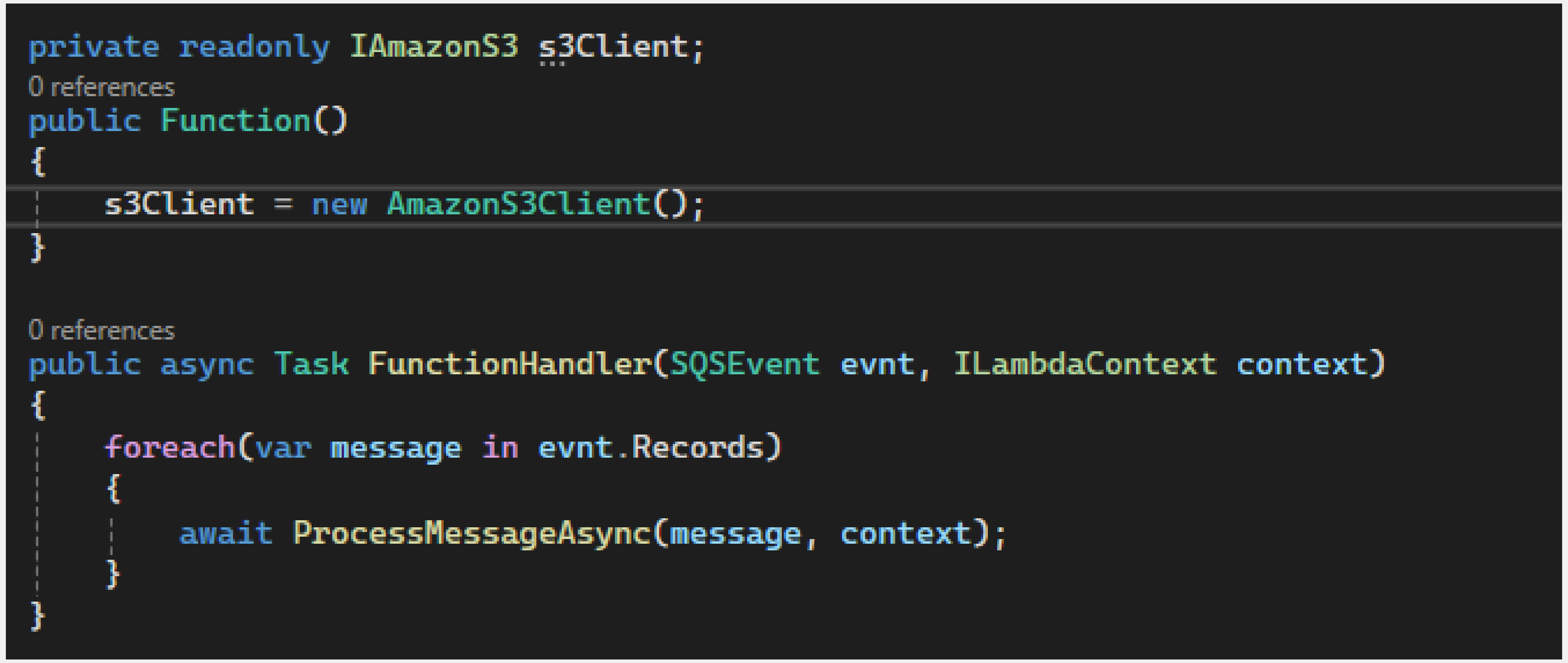
▶ More

Subscription region
us-east-1

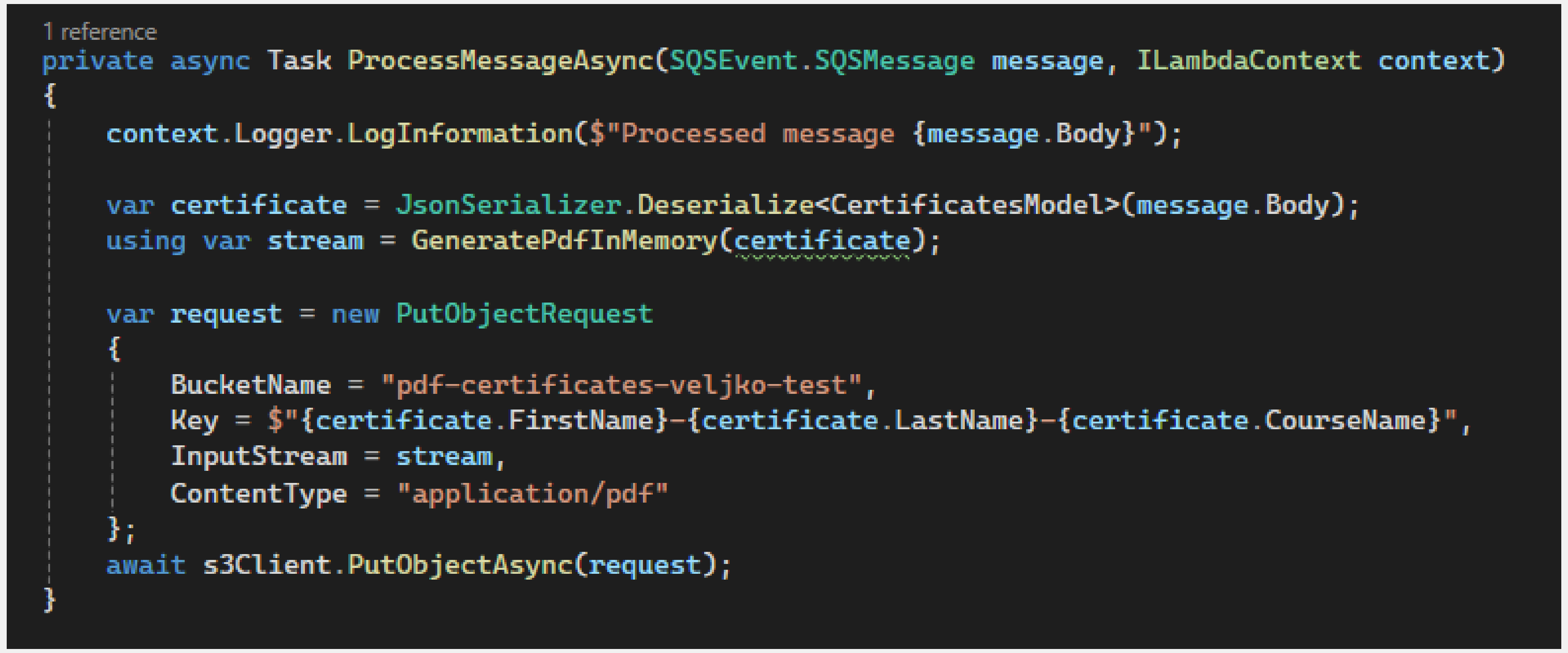
PdfGeneratorLambda



```xml
<ItemGroup>
  <PackageReference Include="Amazon.Lambda.Core" Version="2.5.0" />
  <PackageReference Include="Amazon.Lambda.Serialization.SystemTextJson" Version="2.4.4" />
  <PackageReference Include="Amazon.Lambda.SQSEvents" Version="2.2.0" />
  <PackageReference Include="AWSSDK.S3" Version="4.0.0.7" />
  <PackageReference Include="QuestPDF" Version="2025.5.0" />
</ItemGroup>
```

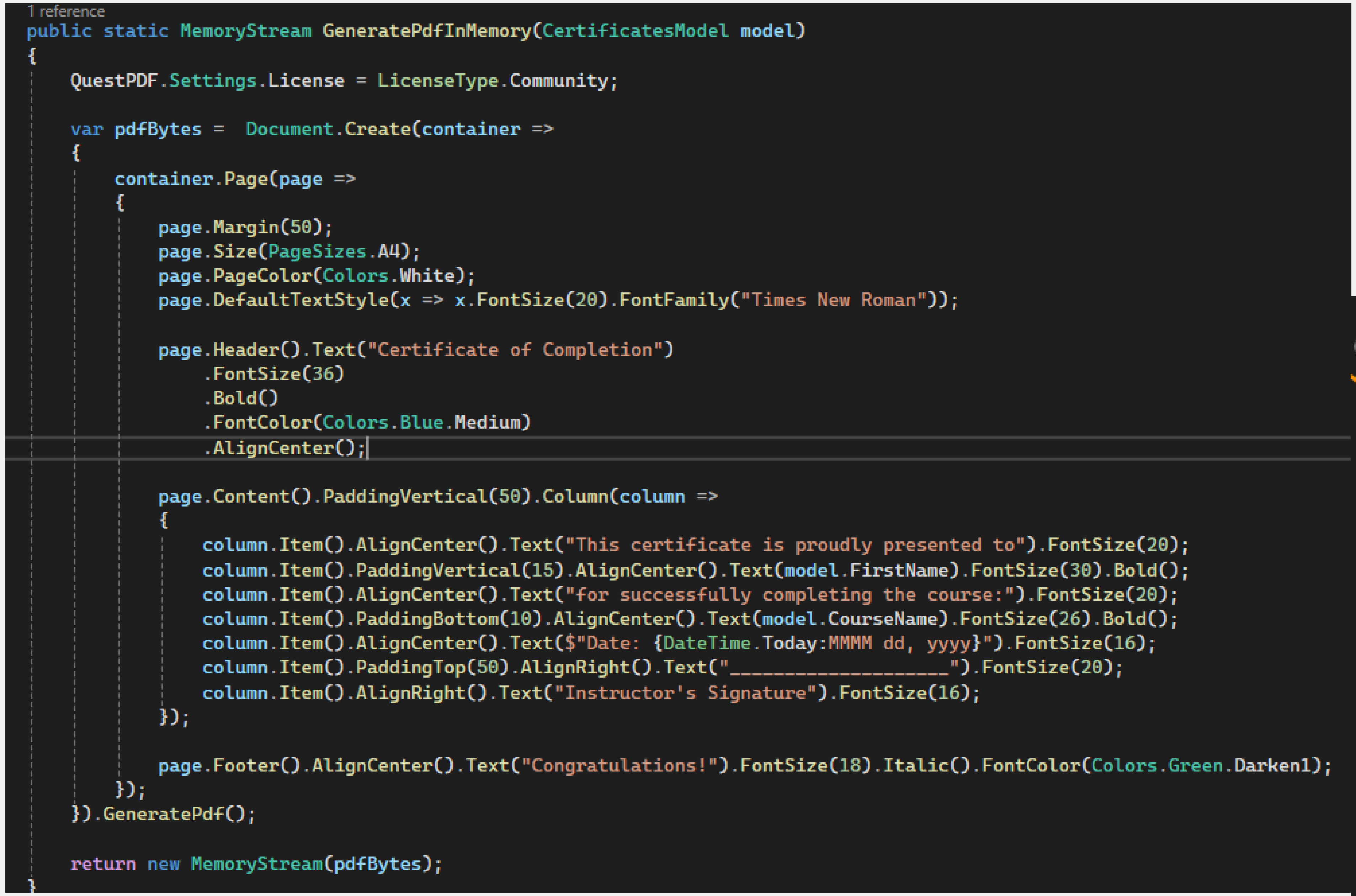


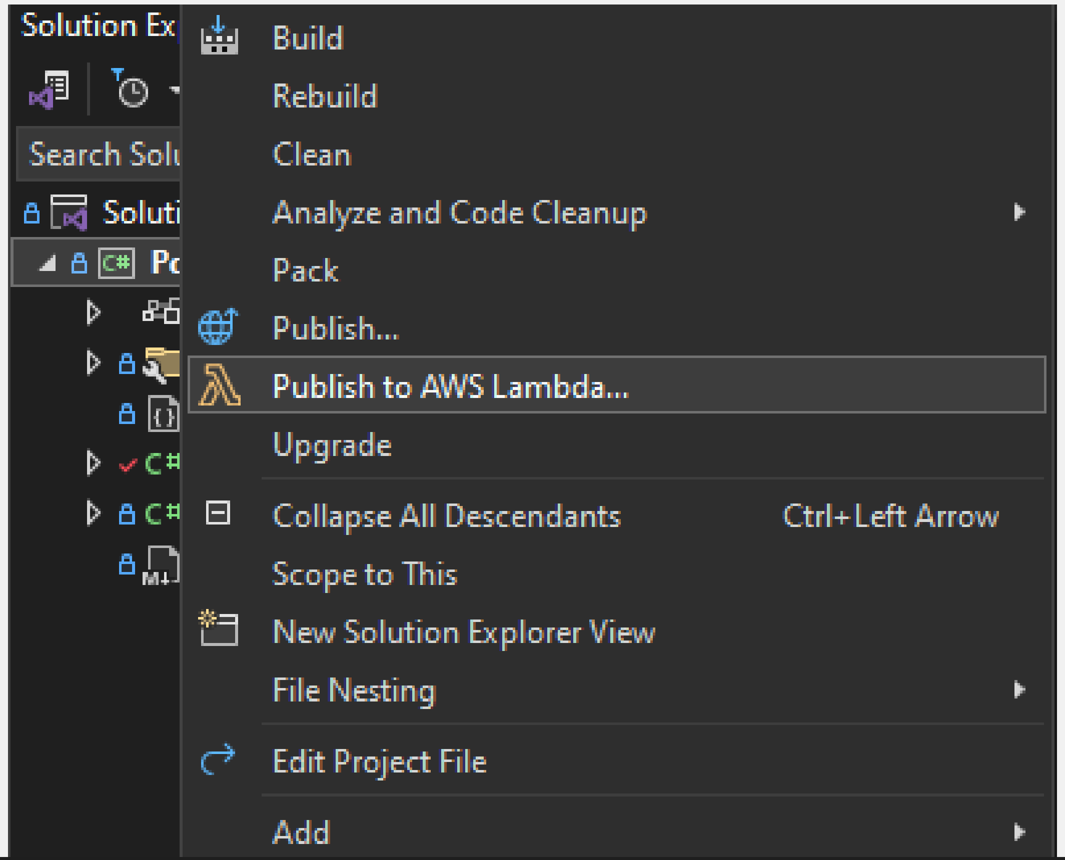
```csharp
private readonly IAmazonS3 s3Client;
0 references
public Function()
{
    s3Client = new AmazonS3Client();
}

0 references
public async Task FunctionHandler(SQSEvent evnt, ILambdaContext context)
{
    foreach(var message in evnt.Records)
    {
        await ProcessMessageAsync(message, context);
    }
}
```

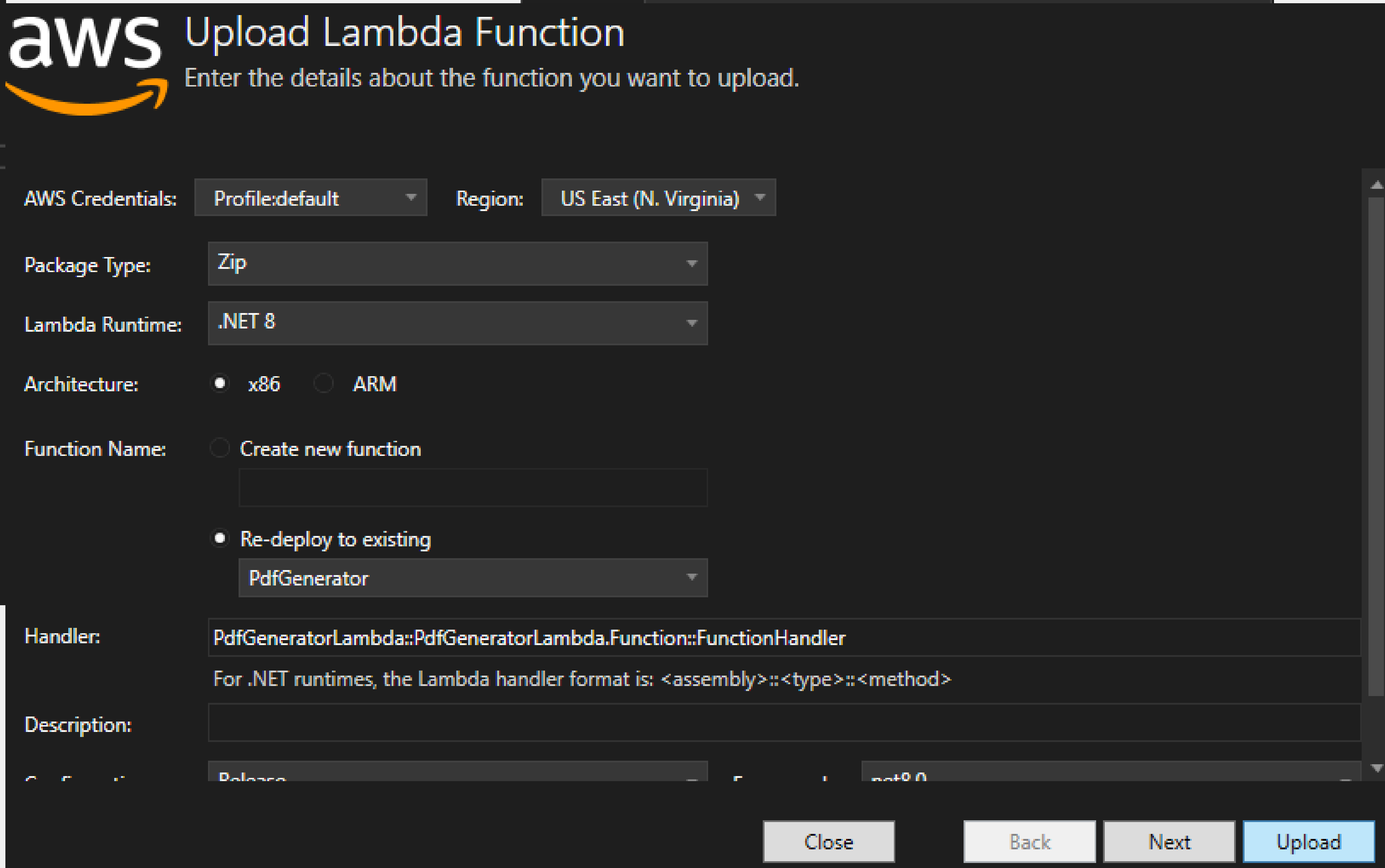


```csharp
1 reference
private async Task ProcessMessageAsync(SQSEvent.SQSMessage message, ILambdaContext context)
{
    context.Logger.LogInformation($"Processed message {message.Body}");

    var certificate = JsonSerializer.Deserialize<CertificatesModel>(message.Body);
    using var stream = GeneratePdfInMemory(certificate);

    var request = new PutObjectRequest
    {
        BucketName = "pdf-certificates-veljko-test",
        Key = $"{certificate.FirstName}-{certificate.LastName}-{certificate.CourseName}",
        InputStream = stream,
        ContentType = "application/pdf"
    };
    await s3Client.PutObjectAsync(request);
}
```

PdfGeneratorLambda


```
1 reference
public static MemoryStream GeneratePdfInMemory(CertificatesModel model)
{
    QuestPDF.Settings.License = LicenseType.Community;

    var pdfBytes =  Document.Create(container =>
    {
        container.Page(page =>
        {
            page.Margin(50);
            page.Size(PageSizes.A4);
            page.PageColor(Colors.White);
            page.DefaultTextStyle(x => x.FontSize(20).FontFamily("Times New Roman"));

            page.Header().Text("Certificate of Completion")
                .FontSize(36)
                .Bold()
                .FontColor(Colors.Blue.Medium)
                .AlignCenter();

            page.Content().PaddingVertical(50).Column(column =>
            {
                column.Item().AlignCenter().Text("This certificate is proudly presented to").FontSize(20);
                column.Item().PaddingVertical(15).AlignCenter().Text(model.FirstName).FontSize(30).Bold();
                column.Item().AlignCenter().Text("for successfully completing the course:").FontSize(20);
                column.Item().PaddingBottom(10).AlignCenter().Text(model.CourseName).FontSize(26).Bold();
                column.Item().AlignCenter().Text($"Date: {DateTime.Today:MMMM dd, yyyy}").FontSize(16);
                column.Item().PaddingTop(50).AlignRight().Text("____________________").FontSize(20);
                column.Item().AlignRight().Text("Instructor's Signature").FontSize(16);
            });

            page.Footer().AlignCenter().Text("Congratulations!").FontSize(18).Italic().FontColor(Colors.Green.Darken1);
        });
    }).GeneratePdf();

    return new MemoryStream(pdfBytes);
}
```




Upload Lambda Function

Enter the details about the function you want to upload.

AWS Credentials: Profile:default

Region: US East (N. Virginia)

Package Type: Zip

Lambda Runtime: .NET 8

Architecture: ● x86 ○ ARM

Function Name: ○ Create new function

● Re-deploy to existing

PdfGenerator

Handler: PdfGeneratorLambda::PdfGeneratorLambda.Function::FunctionHandler

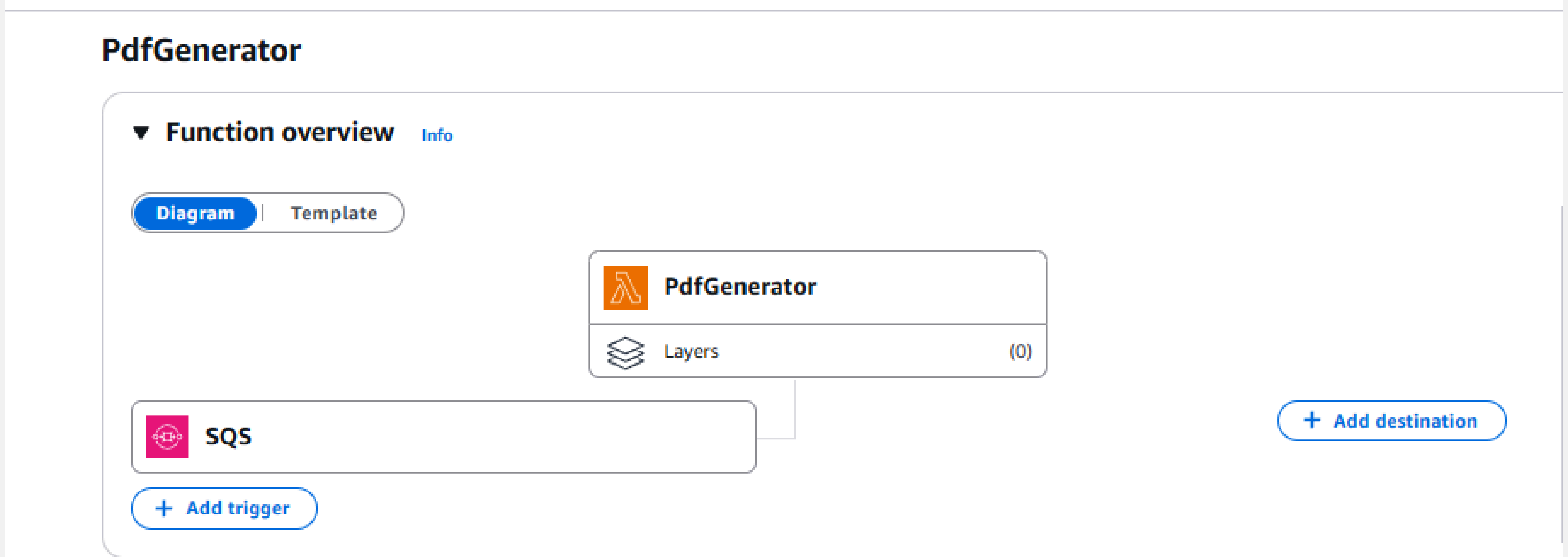
For .NET runtimes, the Lambda handler format is: <assembly>::<type>::<method>

Description:

Release

net8.0

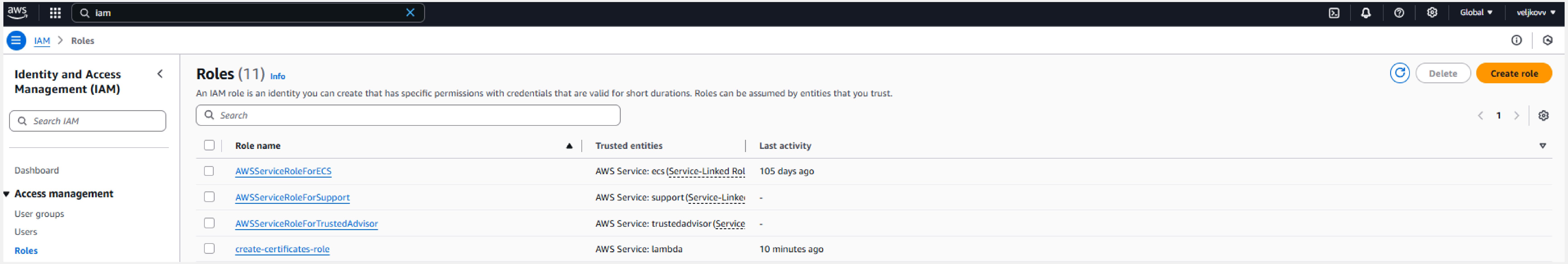
Lambda



PdfGenerator

▼ Function overview Info

IAM

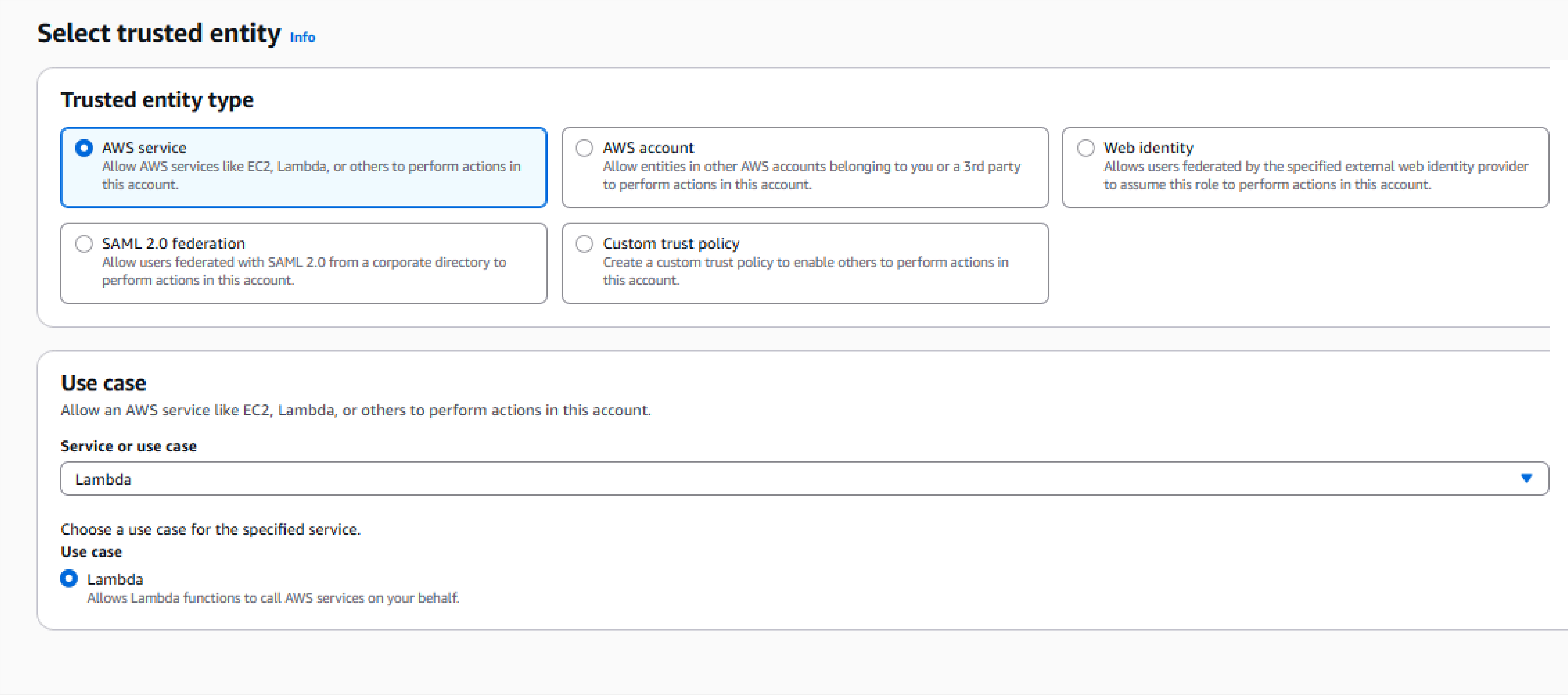


Identity and Access Management (IAM)

Roles (11) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

☐	Role name	Trusted entities	Last activity
☐	AWSServiceRoleForECS	AWS Service: ecs (Service-Linked Rol	105 days ago
☐	AWSServiceRoleForSupport	AWS Service: support (Service-Linke	-
☐	AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service	-
☐	create-certificates-role	AWS Service: lambda	10 minutes ago



Select trusted entity Info

Trusted entity type

- ● AWS service
  Allow AWS services like EC2, Lambda, or others to perform actions in this account.
- ○ AWS account
  Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
- ○ Web identity
  Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
- ○ SAML 2.0 federation
  Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- ○ Custom trust policy
  Create a custom trust policy to enable others to perform actions in this account.

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

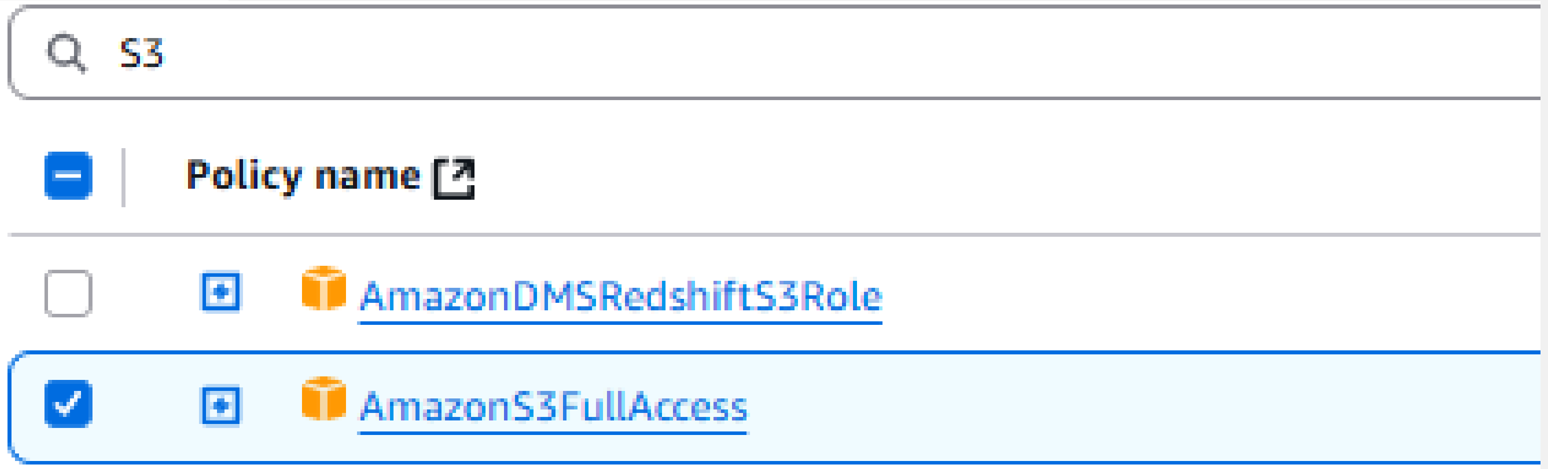
Service or use case

Lambda

Choose a use case for the specified service.

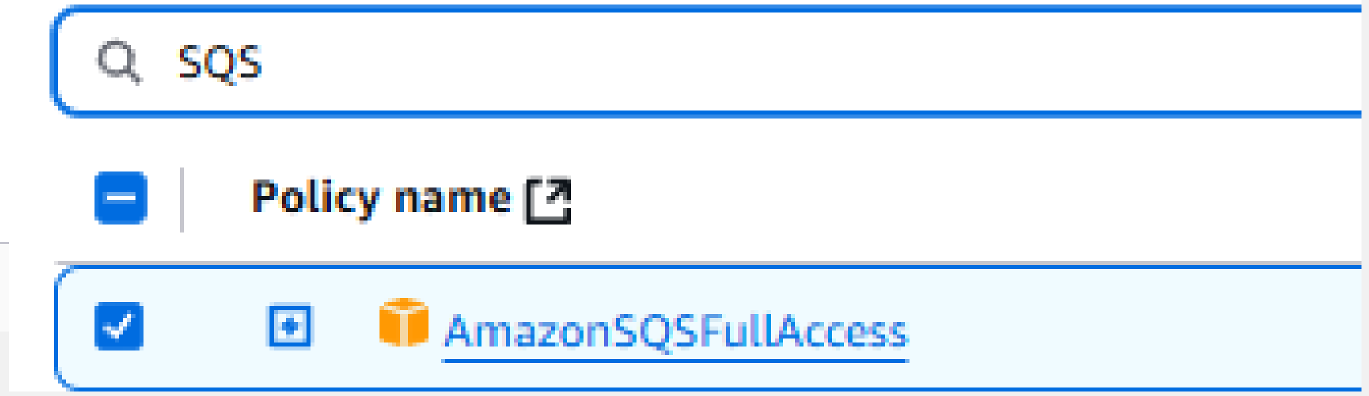
Use case

- ● Lambda
  Allows Lambda functions to call AWS services on your behalf.



S3

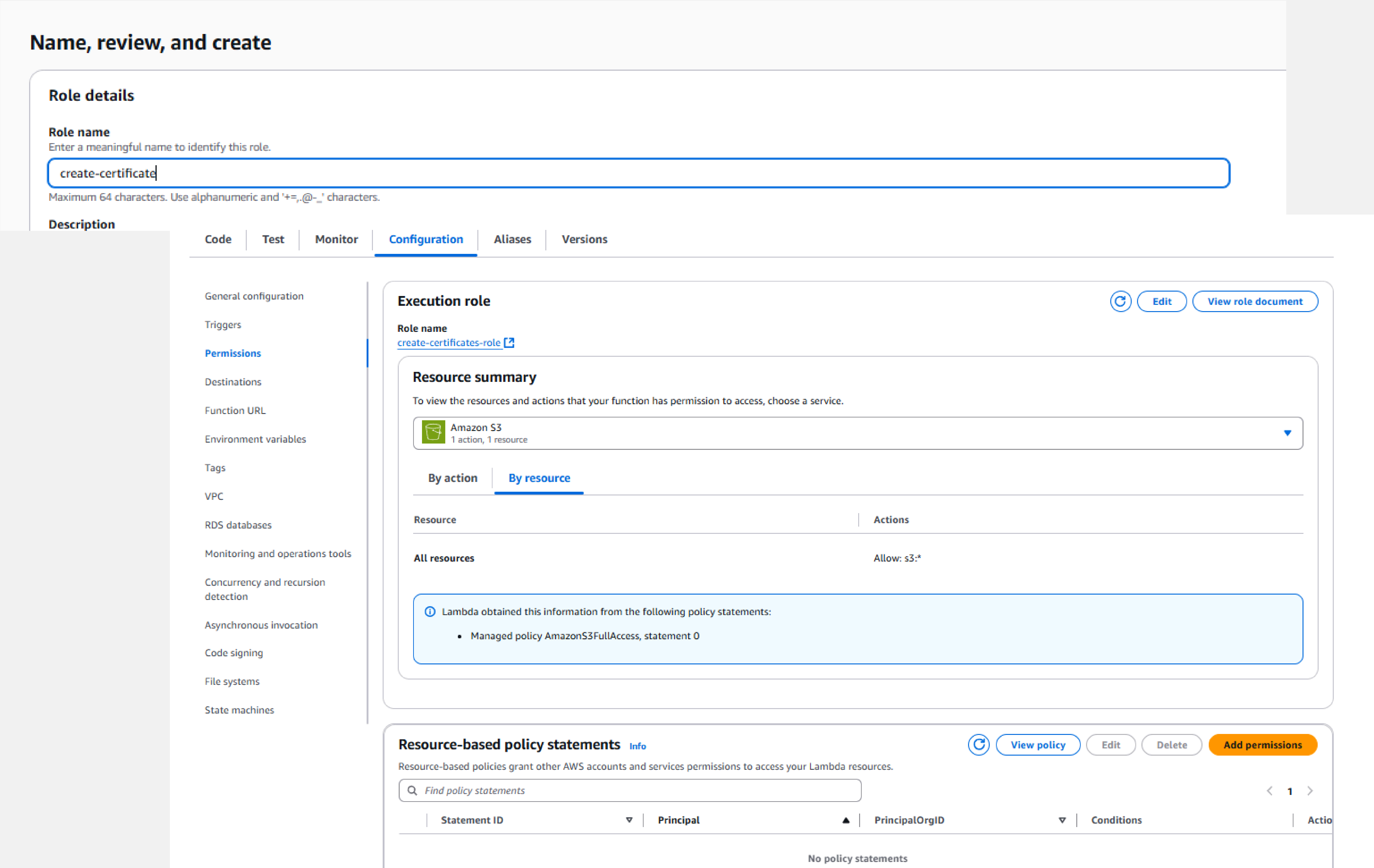
☐	Policy name
☐	AmazonDMSRedshiftS3Role
☑	AmazonS3FullAccess



SQS

☐	Policy name
☑	AmazonSQSFullAccess

IAM



Name, review, and create

Role details

Role name

Enter a meaningful name to identify this role.

create-certificate

Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.

Description

Execution role

Role name

create-certificates-role

Resource summary

To view the resources and actions that your function has permission to access, choose a service.

Amazon S3
1 action, 1 resource

Resource	Actions
All resources	Allow: s3:*

Lambda obtained this information from the following policy statements:

- Managed policy AmazonS3FullAccess, statement 0

Resource-based policy statements Info

Resource-based policies grant other AWS accounts and services permissions to access your Lambda resources.

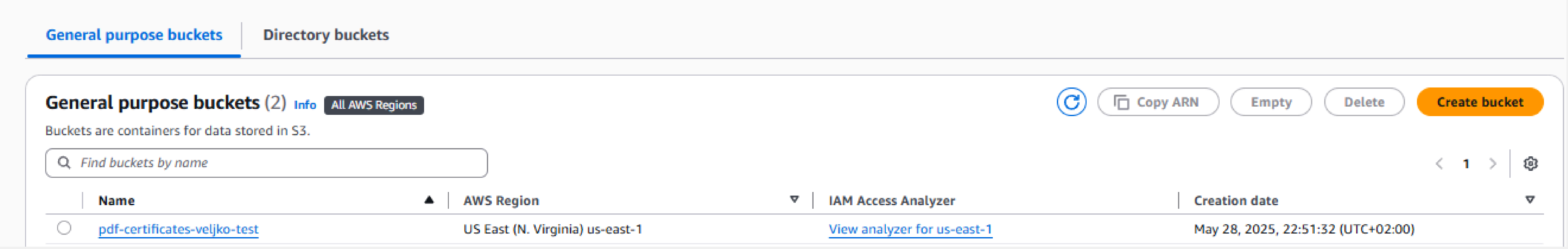
Find policy statements

1

Statement ID	Principal	PrincipalOrgID	Conditions	Actio

No policy statements

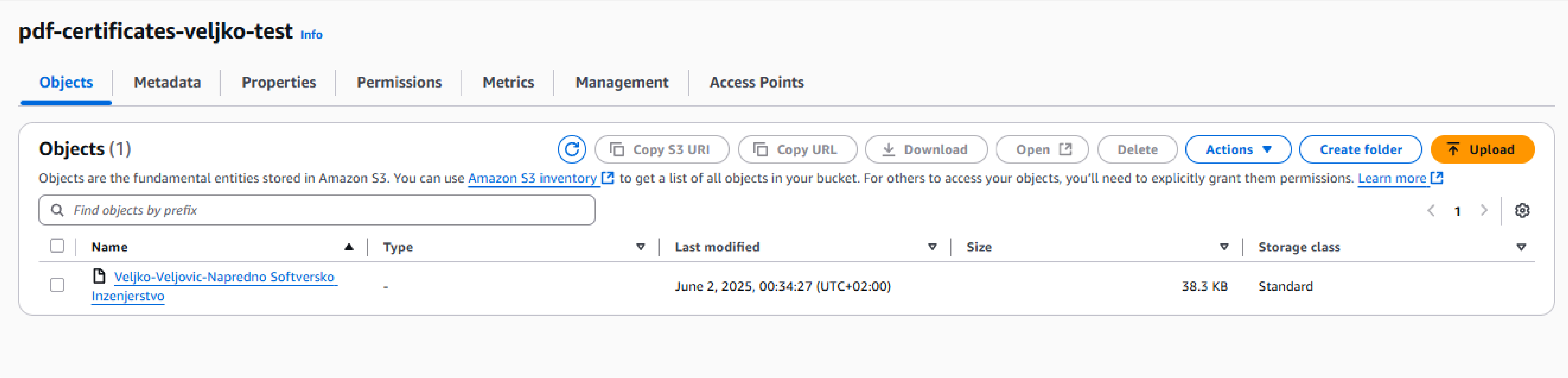
S3



General purpose buckets (2)

Buckets are containers for data stored in S3.

	Name ▲	AWS Region	IAM Access Analyzer	Creation date
○	pdf-certificates-veljko-test	US East (N. Virginia) us-east-1	View analyzer for us-east-1	May 28, 2025, 22:51:32 (UTC+02:00)



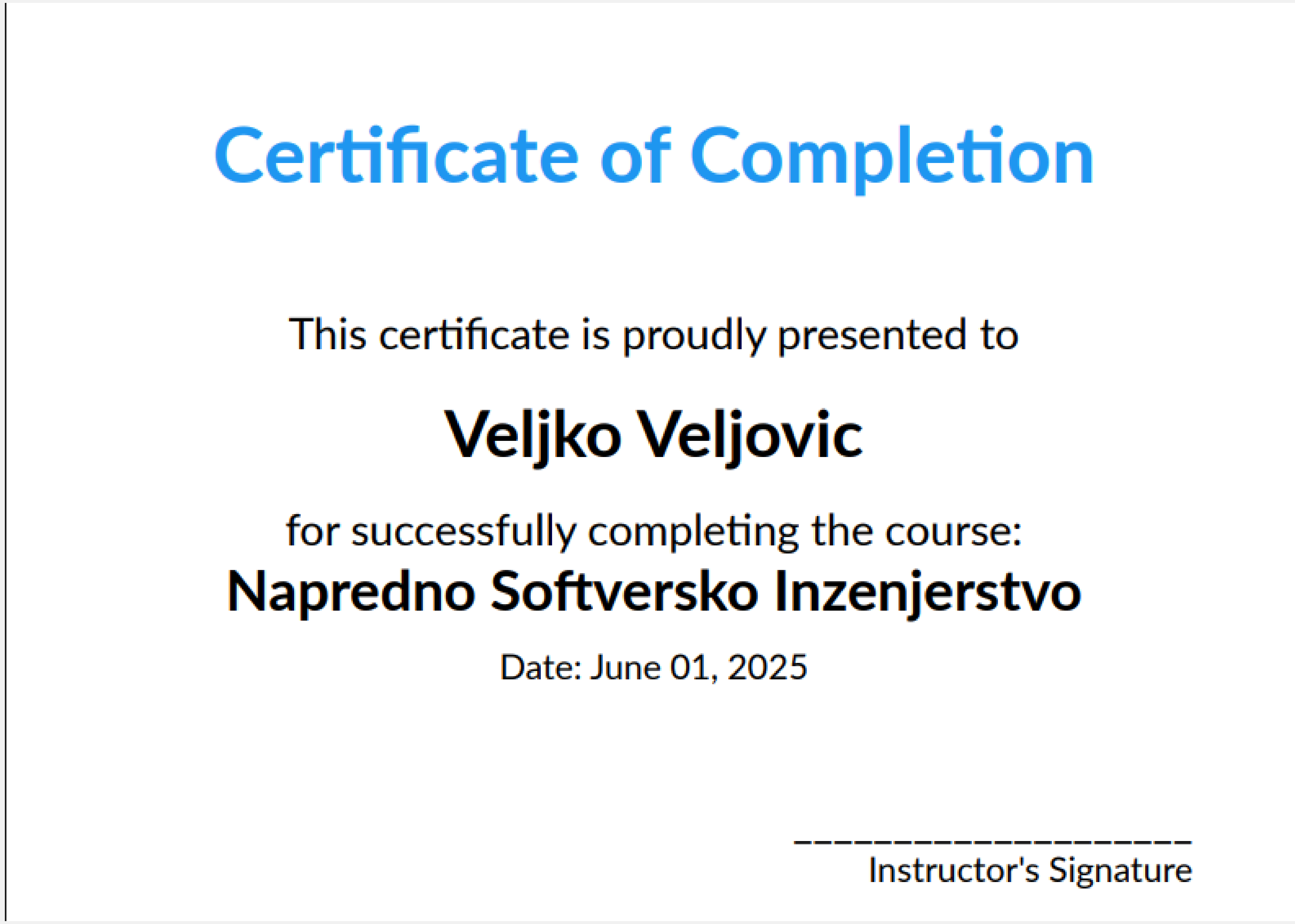
pdf-certificates-veljko-test

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name ▲	Type	Last modified	Size	Storage class
□	Veljko-Veljovic-Napredno Softversko Inzenjerstvo	-	June 2, 2025, 00:34:27 (UTC+02:00)	38.3 KB	Standard

S3



Certificate of Completion

This certificate is proudly presented to

Veljko Veljovic

for successfully completing the course:

Napredno Softversko Inzenjerstvo

Date: June 01, 2025

____

Instructor's Signature

Hvala na pažnji!