.NET 8 Support on AWS Guide

Overview

This guide describes .NET 8 support provided by AWS services and tools. The guide will be updated as new support is added.

Related Content:

- GitHub home for .NET on AWS
- AWS Developer Center - .NET on AWS

Introduction

On November 14th, Microsoft released .NET 8, the latest Long-Term Support (LTS) version of the .NET platform. .NET 8 includes extensive performance improvements, container enhancements, C# language simplified syntax, Blazor support for full-stack web applications, and ASP.NET Core partial support for Native Ahead of Time compilation (NativeAOT). .NET 8 can be used with a number of AWS services such as Amazon EC2 where the runtime is installed manually.

This guide is focused on AWS services and tools that have either been updated, or are being updated, to provide .NET 8 support. While we work to test and validate support for .NET 8 across our services and tools, there are some steps you can take to use .NET 8 today. These are called out in the sections below.

Please open an Issue on this repository with questions about .NET 8 support on AWS. We use this feedback to help us prioritize updates.

Compute Services

Amazon EC2

Customers can install .NET 8 on over 400 Amazon EC2 instances types. The following EC2 User Data example installs .NET 8 on an Amazon Linux 2023 instance:

```
#!/bin/bash
sudo rpm --import https://packages.microsoft.com/keys/microsoft.asc
sudo wget -O /etc/yum.repos.d/microsoft-prod.repo
https://packages.microsoft.com/config/fedora/37/prod.repo
```

```
sudo dnf install -y dotnet-sdk-8.0
dotnet --version > /tmp/dotnet-version
```

Instructions for installing .NET on Linux can be found at https://learn.microsoft.com/en-us/dotnet/core/install/linux#packages. Install scripts for .NET can be found at https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#scripted-install.

Customers can use automation facilities in the AWS Systems Manager service to automatically install .NET runtimes using automation documents, and use the EC2 Image Builder service to pre-create EC2 Images with the .NET Runtime pre-installed.

AWS Elastic Beanstalk

As of the 12/05/2023 platform update, Elastic Beanstalk Windows supports .NET 8.

Elastic Beanstalk Linux currently supports the .NET 6 runtime. To use a version of the .NET Core runtime that Elastic Beanstalk doesn't include in its platforms, provide a self-contained application as described under Bundling applications for the .NET Core on Linux platform (AWS) and .NET application publishing overview (Microsoft).

AWS Lambda

You can currently run .NET 8 applications on AWS Lambda in several ways. You can create your own custom runtime, deploy a container, or publish native code to Lambda using .NET 8's NativeAOT compilation.

Video: The Simplest Way To Build .NET 8 Native AOT Lambda Functions

Blog post: Building serverless .NET applications on AWS Lambda using .NET 7

Lambda runtime support for .NET 8 is planned. Visit the AWS Lambda Runtimes documentation to check status. Future support for .NET runtimes is based on Amazon Linux 2023 (see Introducing the Amazon Linux 2023 runtime for AWS Lambda).

The AWS .NET Lambda packages have been updated to target .NET 8 and address Native AOT warnings. This makes it easier and safer to use them for Native AOT Lambda functions.

The .NET Lambda Annotations Framework simplifies the programming model and lets you write .NET Lambda functions more naturally in C#. When using custom runtimes or native ahead of time (AOT) compilation, the framework removes the need to manually bootstrap the Lambda runtime and can auto-generate the Main method. See .NET Lambda Annotations Design - Auto Generate Main.

You can track AWS Lambda .NET 8 support at https://github.com/aws/aws-lambda-dotnet/issues/1611.

Containers

AWS customers can deploy .NET applications running on either Windows or Linux containers to Amazon Elastic Container Software (ECS) or Amazon Elastic Kubernetes Service (EKS). AWS Fargate is a service that you can use to run and manage the lifecycle of ECS and EKS containers without the need to manage the container infrastructure yourself.

AWS App Runner is a fully managed service that makes it easy to quickly deploy containerized web applications and APIs, scaling up or down automatically to meet application traffic needs. To use with .NET 8 applications, upload an image with the .NET 8 application to Amazon Elastic Container Registry (ECR) and use the source image support to configure AWS App Runner to start, run, scale, and load balance the application.

AWS Elastic Beanstalk allows customers to package up their applications and deploy to Elastic Beanstalk as a container image. By using containers, customer can deploy .NET 8 applications on Elastic Beanstalk.

Tools, Libraries, and SDK

AWS SDK for .NET

The AWS SDK for .NET allows .NET developers to integrate AWS services into their application code in a familiar and consistent manner. Starting with version 3.7.300 the SDK added a new .NET 8 target framework and has made the SDK compatible with Native AOT by addressing all trimming warnings. The library is available from NuGet. Learn how to get started with the AWS SDK for .NET in the Developer Guide.

AWS Code Build

AWS Code Build is a fully managed service to help developers automatically build applications from source code. The code build service allows the user to customize the build environment, to suit the needs of the application being built. This includes the ability to install additional .NET runtimes. Users can add support for building .NET 8 applications by adding the following snippet to their applications buildspec.yml file.

```
install:
  commands:
    - curl -sSL https://dot.net/v1/dotnet-install.sh | bash /dev/stdin --channel 8.0
```

This will automatically download and install the .NET 8 SDK as part of the Install phase of CodeBuild.

AWS Deploy Tool for .NET

The AWS Deploy Tool for .NET command line interface (CLI) is an interactive assistant that provides compute recommendations for .NET applications and deploys them to AWS in a few easy steps. The Deploy Tool supports .NET 8 applications by creating container images for container based services like Amazon ECS and AWS AppRunner or using .NET self contained publishing for Elastic Beanstalk.

AWS Toolkit for Visual Studio

The AWS Toolkit for Visual Studio is an extension for Microsoft Visual Studio on Windows that makes it easier for developers to develop, debug, and deploy .NET applications using Amazon Web Services. Visual Studio 2022 supports .NET 8 development, and customers can download the AWS Toolkit for Visual Studio 2022 from the Visual Studio Marketplace.

The Toolkit's *Publish to AWS* feature integrates with the AWS Deploy Tool for .NET, and can deploy .NET 8 projects to various AWS services from Visual Studio.

.NET Modernization Tools

AWS provides assistive tools that help architects, developers, and IT professionals modernize .NET workloads. The following AWS modernization tools now support .NET 8:

AWS App2Container (A2C) is a command line tool that containerizes your applications. It automatically generates a container image configured with the correct dependencies, network configurations, and deployment instructions for Amazon ECS or Amazon EKS. A2C can now detect a .NET 8 runtime version and containerize the application using the corresponding runtime base images.

AWS Microservice Extractor for .NET is an assistive tool that serves as an advisor to assess and visualize monolithic code, and recommend microservice candidates using artificial intelligence and heuristics. It also serves as a robotic builder to simplify microservices extraction. Microservice Extractor now supports analyzing .NET 8 applications for visualization, grouping, and extraction. With its integrated strangler-fig porting capability, you can also use Microservice Extractor to break down a large .NET Framework-based application with hundreds of projects and thousands of classes into manageable groups and port those directly to .NET 8.

Migration Hub Strategy Recommendations (MHSR) helps you plan migration and modernization initiatives by offering strategy recommendations for viable transformation paths for your applications. MHSR can now detect .NET 8 applications and provide recommendations for them.

AWS Toolkit for .NET Refactoring is a Visual Studio extension that helps you refactor legacy .NET applications to cloud-based alternatives on AWS. It provides a compatibility assessment report and helps port your code. The Toolkit for .NET Refactoring can now target .NET 8 for assessment, porting, and test deployment.

You can take full advantage of .NET 8 as you plan, migrate, and modernize .NET workloads on AWS using these assistive tools. For more information on .NET modernization use cases and tools, see Modernize .NET Workloads on AWS at the .NET on AWS developer center.

Security and Diagnostics

AWS X-Ray

AWS X-Ray helps developers analyze and debug distributed applications, such as those built using a microservices architecture. .NET 8 applications can integrate AWS X-Ray with AWS X-Ray SDK for .NET and the AWS Distro for OpenTelemetry .NET. We don’t recommend using AWS X-Ray with NativeAOT .NET applications at this time.

Document Revision History

2023-11-22 v1.0 Created.
2023-12-06 v1.1 Added Elastic Beanstalk Windows support for .NET 8.