Panduan Pengguna

AWS Toolkit dengan Amazon Q

AWS Toolkit dengan Amazon Q: Panduan Pengguna

Table of Contents

AWS Toolkit dengan Amazon Q

Ini adalah panduan pengguna untuk AWS Toolkit for Visual Studio dengan Amazon Q. Jika Anda mencari AWS Toolkit for VS Code, lihat Panduan Pengguna untuk AWS Toolkit for Visual Studio Code.

Apa itu AWS Toolkit for Visual Studio dengan Amazon Q

AWS Toolkit for Visual Studio with Amazon Q adalah ekstensi untuk Visual Studio IDE yang memudahkan Anda mengembangkan, men-debug, dan menyebarkan aplikasi.NET yang menggunakan Amazon Web Services. AWS Toolkit dengan Amazon Q didukung untuk Visual Studio versi 2022 dan yang lebih baru. Untuk detail tentang cara mengunduh dan menginstal kit, lihat topik Instalasi dan penyiapan di Panduan Pengguna ini.

> **Note**
>
> Toolkit for Visual Studio juga dirilis untuk Visual Studio 2008, 2010, 2012, 2013, 2015, 2017, dan 2019. versi. Namun, versi tersebut tidak lagi didukung. Untuk informasi selengkapnya, lihat topik Instalasi dan penyiapan di Panduan Pengguna ini.

AWS Toolkit dengan Amazon Q berisi fitur-fitur berikut untuk meningkatkan pengalaman pengembangan Anda.

AWS Penjelajah

Jendela alat AWS Explorer dapat diakses di menu Tampilan IDE dan memungkinkan Anda berinteraksi dengan AWS layanan di Visual Studio. Untuk daftar AWS layanan dan fitur yang didukung, lihat topik Bekerja dengan AWS Layanan di Panduan Pengguna ini.

Amazon Q

Mengobrol dengan Pengembang Amazon Q di Visual Studio untuk mengajukan pertanyaan tentang membangun di AWS dan untuk bantuan pengembangan perangkat lunak. Amazon Q dapat menjelaskan konsep pengkodean dan cuplikan kode, menghasilkan pengujian kode dan unit, dan meningkatkan kode melalui debugging atau refactoring.

Untuk menginstal dan menyiapkan Amazon Q untuk Toolkit for Visual Studio, lihat topik Memulai di Panduan Pengguna ini. Untuk mempelajari lebih lanjut tentang bekerja dengan Pengembang Amazon Q, lihat IDEs topik Pengembang Amazon Q di Panduan Pengguna Pengembang Amazon Q. Untuk informasi terperinci tentang paket dan harga Amazon Q, lihat panduan harga Amazon Q.

Informasi Terkait

Untuk membuka masalah atau melihat masalah yang saat ini terbuka, kunjungi https://github.com/aws/aws-toolkit-visual-studio/issues.

Untuk mempelajari lebih lanjut tentang Visual Studio, kunjungi https://visualstudio.microsoft.com/vs/.

Amazon Q

Apa itu Amazon Q

Mulai 30 April 2024, Amazon sekarang menjadi bagian dari CodeWhisperer Pengembang Amazon Q, ini termasuk saran kode sebaris dan pemindaian keamanan.

Untuk mempelajari selengkapnya tentang bekerja dengan Pengembang Amazon Q di bagian AWS Toolkit for Visual Studio, lihat IDEs topik Pengembang Amazon Q di Panduan Pengguna Pengembang Amazon Q. Untuk informasi terperinci tentang paket dan harga Amazon Q, lihat panduan harga Amazon Q.

Mengunduh Toolkit for Visual Studio

Anda dapat mengunduh, menginstal, dan mengatur Toolkit for Visual Studio melalui Visual Studio Marketplace di IDE Anda. Untuk petunjuk terperinci, lihat bagian Memasang AWS Toolkit for Visual Studio di topik Memulai Panduan Pengguna ini.

Mengunduh Toolkit dari Marketplace Visual Studio

Unduh file instalasi Toolkit for Visual Studio dengan menavigasi ke situs dowloads Visual Studio AWS di browser web Anda.

Toolkit IDE tambahan dari AWS

Selain Toolkit for Visual Studio AWS , juga menawarkan IDE Toolkit untuk VS Code dan. JetBrains

AWS Toolkit for Visual Studio Code link

- Ikuti tautan ini untuk Mengunduh AWS Toolkit for Visual Studio Code dari VS Code Marketplace.
- Untuk mempelajari selengkapnya AWS Toolkit for Visual Studio Code, lihat Panduan AWS Toolkit for Visual Studio CodePengguna.

AWS Toolkit for JetBrains link

- Ikuti tautan ini untuk mengunduh AWS Toolkit for JetBrains dari JetBrains Marketplace.
- Untuk mempelajari selengkapnya AWS Toolkit for JetBrains, lihat Panduan AWS Toolkit for JetBrainsPengguna.

Memulai

AWS Toolkit for Visual Studio Ini membuat AWS layanan dan sumber daya Anda tersedia dari lingkungan pengembangan terintegrasi Visual Studio (IDE).

Untuk membantu Anda memulai, topik berikut menjelaskan cara menginstal, mengatur, dan mengkonfigurasi AWS Toolkit for Visual Studio.

Topik

Instalasi dan pengaturan AWS Toolkit for Visual Studio

Topik berikut menjelaskan cara mengunduh, menginstal, mengatur, dan menghapus instalan. AWS Toolkit for Visual Studio

Topik

Prasyarat

Berikut ini adalah prasyarat untuk menyiapkan versi yang didukung dari. AWS Toolkit for Visual Studio

- Visual Studio 19 atau rilis yang lebih baru
- Windows 10 atau rilis Windows yang lebih baru
- Akses administrator ke Windows dan Visual Studio
- Kredensil AWS IAM Aktif

> **Note**
>
> Versi yang tidak didukung AWS Toolkit for Visual Studio tersedia untuk Visual Studio 2008, 2010, 2012, 2013, 2015, dan 2017. Untuk mengunduh versi yang tidak didukung, navigasikan ke halaman AWS Toolkit for Visual Studioarahan dan pilih versi yang Anda inginkan dari daftar tautan unduhan.
> Untuk mempelajari lebih lanjut tentang kredenal IAM atau mendaftar akun, kunjungi gateway Konsol.AWS

Instalasi AWS Toolkit for Visual Studio

Untuk menginstal AWS Toolkit for Visual Studio, temukan versi Visual Studio Anda dari prosedur berikut dan selesaikan langkah-langkah yang diperlukan. Tautan unduhan untuk semua versi AWS Toolkit for Visual Studio dapat ditemukan di halaman AWS Toolkit for Visual Studioarahan.

> **Note**
>
> Jika Anda mengalami masalah saat menginstal AWS Toolkit for Visual Studio, lihat topik Masalah penginstalan pemecahan masalah di panduan ini.

Menginstal AWS Toolkit for Visual Studio untuk Visual Studio 2022

Untuk menginstal AWS Toolkit for Visual Studio 2022 dari Visual Studio, selesaikan langkah-langkah berikut:

1. Dari menu Utama, navigasikan ke Ekstensi dan pilih Kelola Ekstensi.
2. Dari kotak pencarian, cari AWS.
3. Pilih tombol Unduh untuk versi Visual Studio 2022 yang relevan dan ikuti petunjuk penginstalan.

   > **Note**
   >
   > Anda mungkin perlu menutup dan memulai ulang Visual Studio secara manual untuk menyelesaikan proses instalasi.

4. Ketika download dan instalasi selesai, Anda dapat membuka AWS Toolkit for Visual Studio dengan memilih AWS Explorer dari menu View.

Menginstal AWS Toolkit for Visual Studio untuk Visual Studio 2019

Untuk menginstal AWS Toolkit for Visual Studio 2019 dari Visual Studio, selesaikan langkah-langkah berikut:

1. Dari menu Utama, navigasikan ke Ekstensi dan pilih Kelola Ekstensi.
2. Dari kotak pencarian, cari AWS.
3. Pilih tombol Unduh untuk Visual Studio 2017 dan 2019 dan ikuti petunjuknya.

> **Note**
>
> Anda mungkin perlu menutup dan memulai ulang Visual Studio secara manual untuk menyelesaikan proses instalasi.

4. Ketika download dan instalasi selesai, Anda dapat membuka AWS Toolkit for Visual Studio dengan memilih AWS Explorer dari menu View.

Menghapus instalasi AWS Toolkit for Visual Studio

Untuk menghapus instalasi AWS Toolkit for Visual Studio, temukan versi Visual Studio Anda dari prosedur berikut dan selesaikan langkah-langkah yang diperlukan.

Menghapus instalasi AWS Toolkit for Visual Studio untuk Visual Studio 2022

Untuk Menghapus AWS Toolkit for Visual Studio 2022 dari Visual Studio, selesaikan langkah-langkah berikut:

1. Dari menu Utama, navigasikan ke Ekstensi dan pilih Kelola Ekstensi.
2. Dari menu navigasi Kelola Ekstensi, perluas judul Terpasang.
3. Temukan ekstensi AWS Toolkit for Visual Studio 2022 dan pilih tombol Copot pemasangan.

> **Note**
>
> Jika AWS Toolkit for Visual Studio tidak terlihat dari bagian Installed pada menu navigasi, Anda mungkin perlu me-restart Visual Studio.

4. 4. Ikuti petunjuk di layar untuk menyelesaikan proses pencopotan instalasi.

Menghapus instalasi AWS Toolkit for Visual Studio untuk Visual Studio 2019

Untuk menghapus AWS Toolkit for Visual Studio 2019 dari Visual Studio, selesaikan langkah-langkah berikut:

1. Dari menu Utama, navigasikan ke Alat dan pilih Kelola Ekstensi.
2. Dari menu navigasi Kelola Ekstensi, perluas judul Terpasang.
3. Temukan ekstensi AWS Toolkit for Visual Studio 2019 dan pilih tombol Uninstall.
4. 4. Ikuti petunjuk di layar untuk menyelesaikan proses pencopotan instalasi.

Menghapus instalasi AWS Toolkit for Visual Studio untuk Visual Studio 2017

Untuk menghapus AWS Toolkit for Visual Studio 2017 di Visual Studio, selesaikan langkah-langkah berikut:

1. Dari menu Utama, navigasikan ke Alat dan pilih Ekstensi dan Pembaruan.
2. Dari menu navigasi Ekstensi dan Pembaruan, perluas judul Terinstal.
3. Temukan ekstensi AWS Toolkit for Visual Studio 2017 dan pilih tombol Uninstall.
4. 4. Ikuti petunjuk di layar untuk menyelesaikan proses pencopotan instalasi.

Menghapus instalasi AWS Toolkit for Visual Studio untuk Visual Studio 2013 atau 2015

Untuk menghapus instalasi AWS Toolkit for Visual Studio 2013 atau 2015, selesaikan langkah-langkah berikut:

1. Dari Panel Kontrol Windows Anda, buka Program dan Fitur.

   > **Note**
   >
   > Anda dapat membuka Program dan Fitur segera dengan menjalankan `appwiz.cpl` dari prompt perintah Windows atau dialog Windows Run.

2. Dari daftar program yang diinstal, buka menu konteks untuk (klik kanan) AWS Alat untuk Windows.
3. Pilih Uninstall dan ikuti petunjuk untuk menyelesaikan proses uninstall.

> **Note**
>
> Direktori Sampel Anda tidak dihapus selama proses uninstall. Direktori ini dipertahankan jika Anda telah memodifikasi sampel. Direktori ini harus dihapus secara manual.

Menghubungkan ke AWS

Bagian berikut menjelaskan cara memulai dengan AWS Toolkit for Visual Studio dengan Amazon Q. Pertama kali Anda meluncurkan Visual Studio setelah menginstal ekstensi, Memulai akan ditampilkan di jendela editor. Dari tab Memulai Anda dapat menyelesaikan tindakan berikut.

- Aktifkan atau nonaktifkan Amazon Q dan AWS Toolkit.
- Tambahkan dan autentikasi dengan kredensil baru.
- Otentikasi dengan kredensi yang ada.
- Akses dokumentasi dan tutorial untuk membantu Anda mulai bekerja dengan Amazon Q dan AWS Toolkit.

Prasyarat

Untuk mulai bekerja dengan Amazon Q dan AWS Toolkit, Anda perlu mengautentikasi dengan AWS kredensi. Jika sebelumnya Anda telah menyiapkan AWS akun dan otentikasi melalui AWS alat atau layanan lain (seperti AWS Command Line Interface), maka AWS Toolkit secara otomatis mendeteksi kredensil Anda. Jika Anda baru AWS atau belum membuat akun, Anda dapat mendaftar AWS akun dari portal AWS pendaftaran. Untuk informasi terperinci tentang cara menyiapkan AWS akun baru, lihat topik Ringkasan di Panduan Pengguna AWS Pengaturan.

Menghubungkan ke AWS dari Toolkit

Untuk terhubung ke AWS akun Anda dari AWS Toolkit, buka tab Memulai kapan saja dengan menyelesaikan yang berikut ini.

Membuka tab Memulai di Visual Studio

1. Dari Visual Studio, perluas Extensions dari menu utama dan kemudian perluas sub-menu AWS Toolkit.

2. Pilih Memulai.
3. Tab Memulai terbuka di jendela editor Visual Studio.

Dari tab Memulai, ada 2 bagian utama:

- Fitur: Di bagian ini Anda dapat mengaktifkan atau menonaktifkan fitur seperti Amazon Q dan AWS Toolkit.
- Dokumentasi dan Tutorial: Kumpulan referensi ke fitur Anda yang diaktifkan.

> **Note**
>
> Bagian Dokumentasi dan Tutorial hanya terlihat ketika satu atau lebih fitur diaktifkan.

Amazon Q Developer

Dari bagian Amazon Q di tab Memulai, Anda dapat mengaktifkan atau menonaktifkan Amazon Q, menambahkan koneksi baru, atau beralih ke AWS koneksi lain. Sebelum Anda dapat melihat atau mengakses salah satu tindakan ini, Amazon Q harus diaktifkan. Untuk mengaktifkan Amazon Q klik tombol Aktifkan.

Ketika Amazon Q dinonaktifkan, semua fitur dan fungsi Amazon Q sepenuhnya dihapus dari Visual Studio. Mengaktifkan Amazon Q secara otomatis membuka otentikasi Pengaturan untuk Amazon Q di tab Memulai. Untuk melanjutkan, Anda harus mengautentikasi dengan AWS IAM Identity Center kredensi Anda untuk mengakses Tingkat Profesional atau ID AWS Pembangun Anda untuk mengakses Tingkat Gratis. Untuk informasi mendetail tentang masing-masing opsi tingkat, lihat topik Memahami tingkatan layanan untuk Pengembang Amazon Q di Panduan Pengguna Pengembang Amazon Q.

Untuk melanjutkan menyelesaikan salah satu prosedur berikut.

Otentikasi tingkat profesional dengan IAM Identity Center

> **Note**
>
> Bidang Nama Profil, URL Mulai, Wilayah Profil, atau Wilayah SSO yang diperlukan untuk mengautentikasi dengan tingkat Profesional biasanya disediakan oleh administrator di

perusahaan atau organisasi Anda. Untuk informasi rinci tentang kredenal Pusat Identitas IAM, lihat topik Apa itu Pusat Identitas IAM di Panduan Pengguna Pusat Identitas AWS IAM.

1. Dari layar Memulai: AWS Toolkit dengan Amazon Q, pilih tombol Masuk di ubin Amazon Q untuk menavigasi ke otentikasi Pengaturan untuk layar Amazon Q.
2. Dari layar Setup authentication for Amazon Q, navigasikan ke bagian Professional tier, isi kolom yang diperlukan dan pilih tombol Connect.
3. Konfirmasikan bahwa Anda ingin membuka portal permintaan AWS Otorisasi di browser web default Anda.
4. Selesaikan langkah-langkah yang diperlukan oleh portal permintaan AWS Otorisasi, Anda akan diberi tahu saat aman untuk menutup browser Anda dan kembali ke Visual Studio
5. Di tab Memulai, Amazon Q memperbarui untuk menunjukkan bahwa Anda terhubung dengan Pusat Identitas IAM saat proses selesai.

Otentikasi tingkat gratis dengan AWS Builder ID

> **Note**
>
> Untuk detail tambahan tentang AWS Builder ID, lihat topik Masuk dengan AWS Builder ID di Panduan Pengguna AWS Masuk.

1. Dari layar Memulai: AWS Toolkit dengan Amazon Q, pilih tombol Masuk di ubin Amazon Q untuk menavigasi ke otentikasi Pengaturan untuk layar Amazon Q.
2. Dari otentikasi Pengaturan untuk Amazon Q layar, navigasikan ke bagian Tingkat Gratis dan pilih tombol Daftar atau Masuk.
3. Konfirmasikan bahwa Anda ingin membuka portal permintaan AWS Otorisasi di browser web default Anda.
4. Selesaikan langkah-langkah yang diperlukan oleh portal permintaan AWS Otorisasi, Anda akan diberi tahu saat aman untuk menutup browser Anda dan kembali ke Visual Studio.
5. Di tab Memulai, Amazon Q memperbarui untuk menunjukkan bahwa Anda terhubung dengan AWS Builder ID Anda saat proses selesai.

Setelah Anda mengautentikasi dengan Pusat Identitas IAM atau kredenal ID AWS Builder, Anda dapat mengakses Amazon Q di Visual Studio. Selain itu, Anda dapat melakukan tindakan berikut di tab Memulai:

- Keluar: putuskan koneksi kredenal Anda saat ini dari semua fungsi Amazon Q. Amazon Q tetap diaktifkan, tetapi sebagian besar fitur tidak berfungsi.
- Nonaktifkan Amazon Q: Benar-benar menonaktifkan semua fitur Amazon Q di Visual Studio.

AWS Toolkit

Dari bagian AWS Toolkit di tab Memulai dengan AWS Toolkit, Anda dapat mengaktifkan atau menonaktifkan AWS Toolkit, menambahkan koneksi baru, atau beralih ke koneksi yang berbeda. AWS Sebelum Anda dapat melihat atau mengakses salah satu tindakan ini, AWS Toolkit harus diaktifkan. Untuk mengaktifkan AWS Toolkit, klik tombol Aktifkan.

Ketika AWS Toolkit diaktifkan, otentikasi Setup untuk AWS Toolkit secara otomatis dimuat di tab Memulai dengan Toolkit. AWS Untuk melanjutkan, Anda harus mengautentikasi dengan kredensi Anda atau AWS IAM Identity Centerkredensi Peran Pengguna IAM Anda.

> **Note**
>
> Untuk informasi rinci tentang kredenal Pusat Identitas IAM, lihat topik Apa itu Pusat Identitas IAM di Panduan Pengguna Pusat Identitas AWS IAM. Untuk informasi mendetail tentang kredenal Peran Pengguna IAM, lihat Kunci AWS akses: Topik kredensial jangka panjang di panduan referensi dan Alat.AWS SDKs

Mengautentikasi dan terhubung dengan IAM Identity Center

1. Dari layar Memulai: AWS Toolkit dengan Amazon Q, pilih tombol Masuk di ubin AWS Toolkit untuk menavigasi ke otentikasi Pengaturan untuk layar Toolkit. AWS
2. Dari Setup Authentication for AWS Toolkit layar, pilih IAM Identity Center (Penerus ke Single Sign-on) dari menu drop-down Jenis Profil.
3. Dari menu drop-down Pilih dari Profil yang ada atau tambahkan baru, pilih profil yang ada atau pilih Tambahkan profil baru untuk menambahkan informasi profil baru.

> **Note**
>
> Jika Anda memilih profil yang ada, lanjutkan ke langkah 7.

4. Di bidang Nama Profil, masukkan yang **`profile name`** terkait dengan akun Pusat Identitas IAM yang ingin Anda autentikasi.
5. Di bidang teks URL Mulai, masukkan **`Start URL`** yang dilampirkan ke kredenal Pusat Identitas IAM Anda.
6. Dari menu tarik-turun Wilayah Profil (default ke us-east-1), pilih Wilayah Profil yang ditentukan oleh profil pengguna Pusat Identitas IAM yang Anda autentikasi.
7. Dari menu tarik-turun Wilayah SSO (default ke us-east-1), pilih Wilayah SSO yang ditentukan oleh kredenal Pusat Identitas IAM Anda.
8. Pilih tombol Connect untuk membuka situs permintaan AWS Otorisasi di browser web default Anda.
9. Ikuti petunjuk di browser web default Anda, Anda diberi tahu ketika proses otorisasi selesai, aman untuk menutup browser Anda, dan kembali ke Visual Studio.
10. Di tab Memulai, bagian AWS Toolkit diperbarui untuk menunjukkan bahwa Anda terhubung dengan Pusat Identitas IAM saat proses selesai.

Mengautentikasi dan terhubung dengan kredensi Peran Pengguna IAM

1. Dari layar Memulai: AWS Toolkit dengan Amazon Q, pilih tombol Masuk di ubin AWS Toolkit untuk menavigasi ke otentikasi Pengaturan untuk layar Toolkit. AWS
2. Dari Autentikasi pengaturan untuk AWS Toolkit layar, pilih Peran Pengguna IAM dari menu drop-down Jenis Profil.
3. Di menu drop-down Pilih dari Profil yang ada atau tambahkan baru, pilih**`Add new profile`**.

> **Note**
>
> Jika Anda memilih nama profil yang ada dari daftar, lewati ke Langkah 8.

4. Di bidang teks Nama Profil, masukkan nama untuk profil baru Anda.
5. Di bidang teks ID Kunci Akses, masukkan profil **`Access Key ID`** yang ingin Anda autentikasi.
6. Di bidang teks Kunci Rahasia, masukkan **`Secret Key`** untuk profil yang ingin Anda autentikasi.

7. Dari menu drop-down Storage Location (default ke Shared Credentials File), tentukan apakah Anda ingin menyimpan kredensialnya dengan file Shared Credentials atau dengan.NET Encrypted Store.
8. Dari menu drop-down Profile Region (default ke us-east-1), pilih Partisi dan Wilayah Profil yang dilampirkan ke profil yang ingin Anda autentikasi.
9. Pilih tombol Connect untuk menambahkan profil ini ke lokasi AWS penyimpanan Anda yang and/or diautentikasi AWS.
10. Di tab Memulai, bagian AWS Toolkit diperbarui untuk menunjukkan bahwa Anda terhubung dengan kredenal peran Pengguna IAM Anda saat proses selesai.

Setelah Anda mengautentikasi dengan kredensial IAM Identity Center atau IAM User Role, Anda dapat mengakses AWS Explorer di Toolkit for Visual Studio. Selain itu, Anda dapat Keluar dan Nonaktifkan AWS Toolkit for Visual Studio dengan Amazon Q dari tab Memulai.

Dokumentasi dan Tutorial

Bagian dokumentasi dan Tutorial secara otomatis memperbarui dengan saran dokumentasi dan tutorial berdasarkan preferensi AWS layanan dan fitur Anda. Referensi ini hanya terlihat ketika setidaknya satu fitur telah diaktifkan.

Memecahkan masalah instalasi untuk AWS Toolkit for Visual Studio

Informasi berikut diketahui untuk menyelesaikan masalah instalasi umum saat mengatur AWS Toolkit for Visual Studio.

Jika Anda mengalami kesalahan saat menginstal AWS Toolkit for Visual Studio atau tidak jelas apakah instalasi selesai atau tidak, tinjau informasi di setiap bagian berikut.

Izin administrator untuk Visual Studio

AWS Toolkit for Visual Studio Ekstensi memerlukan izin administrator untuk memastikan bahwa semua AWS layanan dan fitur dapat diakses.

Jika Anda memiliki izin administrator lokal, kemungkinan izin administrator Anda tidak meluas langsung ke instance Visual Studio Anda.

Untuk meluncurkan Visual Studio dengan izin administrator secara lokal:

1. Dari Windows, cari peluncur aplikasi Visual Studio (ikon).
2. Buka menu konteks untuk (klik kanan) ikon Visual Studio untuk membuka menu konteks.
3. Pilih Jalankan sebagai administrator dari menu konteks.

Untuk meluncurkan Visual Studio dengan izin administrator dari jarak jauh:

1. Dari Windows, cari peluncur aplikasi untuk aplikasi yang Anda gunakan untuk terhubung ke instance Visual Studio jarak jauh Anda.
2. Buka menu konteks untuk (klik kanan) aplikasi untuk membuka menu konteks.
3. Pilih Jalankan sebagai administrator dari menu konteks.

> **Note**
>
> Apakah Anda meluncurkan program secara lokal atau menghubungkan dari jarak jauh, Windows mungkin meminta Anda untuk mengonfirmasi kredensi administratif Anda.

Memperoleh log instalasi

Jika Anda telah menyelesaikan langkah-langkah di bagian izin Administrator sebelumnya yang terletak di atas dan dikonfirmasi bahwa Anda menjalankan atau menghubungkan ke Visual Studio dengan izin administrator, maka mendapatkan file log instalasi dapat membantu mendiagnosis masalah lain.

Untuk menginstal file AWS Toolkit for Visual Studio dari `.vsix` file secara manual dan menghasilkan file log instalasi, selesaikan langkah-langkah berikut.

1. Dari halaman AWS Toolkit for Visual Studioarahan, ikuti tautan Unduh dan simpan `.vsix` file AWS Toolkit for Visual Studio versi yang ingin Anda instal.
2. Dari menu utama Visual Studio, perluas header Tools, perluas sub menu Command Line, lalu pilih Visual Studio Developer Command Prompt.
3. Dari Visual Studio Developer Command Prompt masukkan `vsixinstaller` perintah dengan format berikut:

```
vsixinstaller /logFile:[file path to log file] [file path to Toolkit
installation file]
```

4. Ganti `[file path to log file]` dengan nama file dan jalur file lengkap dari direktori tempat Anda ingin log instalasi dibuat. Contoh `vsixinstaller` perintah dengan path file dan nama file yang Anda tentukan menyerupai berikut ini:

   ```
   vsixinstaller /logFile:C:\Users\Documents\install-log.txt [file path to AWSToolkitPackage.vsix]
   ```

5. Ganti `[file path to Toolkit installation file]` dengan jalur file lengkap dari direktori tempat `AWSToolkitPackage.vsix` berada.

   Contoh `vsixinstaller` perintah dengan path file lengkap ke file instalasi Toolkit harus menyerupai yang berikut:

   ```
   vsixinstaller /logFile:[file path to log file] C:\Users\Downloads\AWSToolkitPackage.vsix
   ```

6. Periksa untuk memastikan nama file dan jalur Anda benar, lalu jalankan `vsixinstaller` perintah.

   Contoh `vsixinstaller` perintah lengkap menyerupai yang berikut:

   ```
   vsixinstaller /logFile:C:\Users\Documents\install-log.txt C:\Users\Downloads\AWSToolkitPackage.vsix
   ```

Menginstal ekstensi Visual Studio yang berbeda

Jika Anda telah memperoleh file log instalasi dan Anda masih tidak dapat menentukan mengapa proses instalasi gagal, periksa untuk melihat apakah Anda dapat menginstal ekstensi Visual Studio lainnya. Menginstal ekstensi Visual Studio yang berbeda dapat memberikan wawasan tambahan untuk masalah instalasi Anda. Jika Anda tidak dapat menginstal ekstensi Visual Studio apa pun, mungkin perlu untuk memecahkan masalah dengan Visual Studio, bukan. AWS Toolkit for Visual Studio

Menghubungi dukungan

Jika Anda telah meninjau semua bagian yang terkandung dalam panduan ini dan memerlukan sumber daya atau dukungan tambahan, Anda dapat melihat masalah sebelumnya atau membuka masalah baru dari situs Masalah AWS Toolkit for Visual Studio Github.

Untuk membantu mempercepat solusi untuk masalah Anda:

- Periksa masalah masa lalu dan saat ini untuk melihat apakah orang lain mengalami situasi serupa.
- Simpan catatan terperinci dari setiap langkah yang telah Anda ambil untuk mengatasi masalah ini.
- Simpan file log apa pun yang Anda peroleh dari menginstal AWS Toolkit for Visual Studio atau ekstensi lainnya.
- Lampirkan file log AWS Toolkit for Visual Studio instalasi Anda ke masalah baru.

Profil dan Window Binding

Profil dan Window Binding untuk Toolkit for Visual Studio

Saat bekerja dengan alat penerbitan, penyihir, dan fitur lain dari Toolkit for Visual Studio, perhatikan hal-hal berikut:

- Jendela AWS Explorer terikat pada satu profil dan wilayah pada satu waktu. Windows dibuka dari default AWS Explorer ke profil dan wilayah terikat itu.
- Setelah jendela baru dibuka, Anda dapat menggunakan instance AWS Explorer tersebut untuk beralih ke profil atau wilayah yang berbeda.
- Alat dan fitur penerbitan Toolkit for Visual Studio secara otomatis default ke profil dan wilayah yang ditetapkan AWS di Explorer.
- Jika profil atau wilayah baru ditentukan dalam alat penerbitan, panduan, atau fitur: semua sumber daya yang dibuat setelahnya akan terus menggunakan pengaturan profil dan wilayah baru.
- Jika Anda memiliki beberapa instance Visual Studio yang terbuka, setiap instance dapat diikat ke profil dan wilayah yang berbeda.
- AWS Explorer menyimpan profil dan wilayah terakhir yang ditentukan dan instance Visual Studio terakhir yang ditutup akan memiliki nilainya tetap ada.

Otentikasi dan akses

Anda tidak perlu melakukan autentikasi AWS untuk mulai bekerja dengan AWS Toolkit for Visual Studio dengan Amazon Q. Namun, AWS sebagian besar sumber daya dikelola melalui akun. AWS Untuk mengakses semua AWS Toolkit for Visual Studio dengan layanan dan fitur Amazon Q, Anda memerlukan setidaknya 2 jenis otentikasi akun:

1. Baik AWS Identity and Access Management (IAM) atau AWS IAM Identity Centerotentikasi untuk akun Anda AWS . Sebagian besar AWS layanan dan sumber daya dikelola melalui IAM dan IAM Identity Center.
2. AWS Builder ID adalah opsional untuk AWS layanan tertentu lainnya.

Topik berikut berisi rincian tambahan dan mengatur instruksi untuk setiap jenis kredensi dan metode otentikasi.

Topik

AWS Identitas Pusat Identitas IAM di AWS Toolkit for Visual Studio

AWS IAM Identity Center adalah praktik terbaik yang disarankan untuk mengelola otentikasi AWS akun Anda.

Untuk petunjuk terperinci tentang cara mengatur Pusat Identitas IAM untuk Kit Pengembangan Perangkat Lunak (SDKs) dan AWS Toolkit for Visual Studio, lihat bagian autentikasi Pusat Identitas IAM dari Panduan Referensi Alat AWS SDKs dan Alat.

Mengautentikasi dengan IAM Identity Center dari AWS Toolkit for Visual Studio

Untuk mengautentikasi dengan IAM Identity Center dari AWS Toolkit for Visual Studio dengan menambahkan profil IAM Identity Center ke `config` file `credentials` atau Anda, selesaikan langkah-langkah berikut.

1. Dari editor teks pilihan Anda, buka informasi AWS kredensional yang disimpan dalam file. `<hone-directory>\.aws\credentials`
2. Dari bagian `credentials file` bawah`[default]`, tambahkan template untuk profil Pusat Identitas IAM bernama. Berikut ini adalah contoh template:

   > **Important**
   >
   > Jangan gunakan profil kata saat membuat entri dalam `credential` file karena membuat konflik dengan konvensi penamaan `credential` file.
   > Sertakan kata awalan `profile_` hanya saat mengonfigurasi profil bernama dalam file. `config`

   ```
   [sso-user-1]
   sso_start_url = https://example.com/start
   sso_region = us-east-2
   sso_account_id = 123456789011
   sso_role_name = readOnly
   region = us-west-2
   ```

- **`sso_start_url`**: URL yang mengarah ke portal pengguna Pusat Identitas IAM organisasi Anda.
- **`sso_region`**: AWS Wilayah yang berisi host portal Pusat Identitas IAM Anda. Ini bisa berbeda dari AWS Wilayah yang ditentukan nanti dalam `region` parameter default.
- **`sso_account_id`**: ID AWS akun yang berisi peran IAM dengan izin yang ingin Anda berikan kepada pengguna Pusat Identitas IAM ini.
- **`sso_role_name`**: Nama peran IAM yang menentukan izin pengguna saat menggunakan profil ini untuk mendapatkan kredensil melalui IAM Identity Center.
- **`region`**: AWS Wilayah default tempat pengguna IAM Identity Center ini masuk.

> **Note**
>
> Anda juga dapat menambahkan profil yang diaktifkan Pusat Identitas IAM ke profil Anda AWS CLI dengan menjalankan `aws configure sso` perintah. Setelah menjalankan perintah ini, Anda memberikan nilai untuk URL awal Pusat Identitas IAM (`sso_start_url`) dan AWS Region (`region`) yang menghosting direktori IAM Identity Center.
> Untuk informasi selengkapnya, lihat Mengonfigurasi AWS CLI untuk AWS menggunakan Single Sign-On di Panduan Pengguna.AWS Command Line Interface

Masuk dengan IAM Identity Center

Saat masuk dengan profil Pusat Identitas IAM, browser default diluncurkan ke yang `sso_start_url` ditentukan di profil Anda`credential file`. Anda harus memverifikasi login IAM Identity Center Anda sebelum Anda dapat mengakses AWS sumber daya Anda di AWS Toolkit for Visual Studio. Jika kredensional Anda kedaluwarsa, Anda harus mengulangi proses koneksi untuk mendapatkan kredensi sementara yang baru.

AWS Kredensi IAM

AWS Kredensi IAM mengautentikasi dengan AWS akun Anda melalui kunci akses yang disimpan secara lokal.

Bagian berikut menjelaskan cara mengatur kredensial IAM untuk mengautentikasi dengan akun Anda AWS dari. AWS Toolkit for Visual Studio

> **Important**
>
> Sebelum menyiapkan kredensi IAM untuk mengautentikasi dengan AWS akun Anda, perhatikan bahwa:
>
> - Jika Anda telah menyetel kredensil IAM melalui AWS layanan lain (seperti AWS CLI), maka AWS Toolkit for Visual Studio secara otomatis mendeteksi kredensil tersebut.
> - AWS merekomendasikan menggunakan AWS IAM Identity Center otentikasi. Untuk informasi tambahan tentang praktik terbaik AWS IAM, lihat praktik terbaik Keamanan di bagian IAM dari Panduan Pengguna AWS Identity and Access Management.
> - Untuk menghindari risiko keamanan, jangan gunakan pengguna IAM untuk otentikasi saat mengembangkan perangkat lunak yang dibuat khusus atau bekerja dengan data nyata.

Sebaliknya, gunakan federasi dengan penyedia identitas seperti AWS IAM Identity Center. Untuk informasi lebih lanjut lihat Apa itu Pusat Identitas IAM? dalam AWS IAM Identity Center User Guide.

Membuat pengguna IAM

Sebelum Anda dapat mengatur AWS Toolkit for Visual Studio untuk mengautentikasi dengan AWS akun Anda, Anda harus menyelesaikan Langkah 1: Buat pengguna IAM Anda dan Langkah 2: Dapatkan kunci akses Anda di Authenticate using long-term credentials topic in the and Tools Reference Guide.AWS SDKs

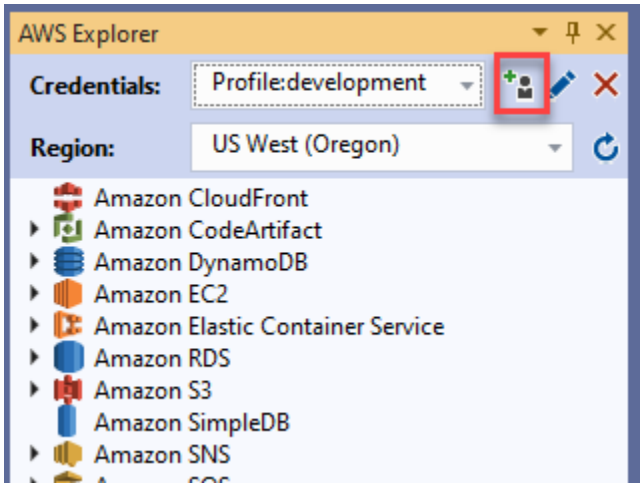
> **Note**
>
> Langkah 3: Perbarui kredenal bersama adalah opsional.
> Jika Anda menyelesaikan Langkah 3, secara AWS Toolkit for Visual Studio otomatis mendeteksi kredensil Anda dari. `credentials file`
> Jika Anda belum menyelesaikan Langkah 3, AWS Toolkit for Visual Studio memandu Anda melalui proses membuat `credentials file` seperti yang dijelaskan dalam Membuat file kredensial dari AWS Toolkit for Visual Studio bagian, yang terletak di bawah ini.

Membuat file kredensial

Untuk menambahkan pengguna ke atau membuat `credentials file` dari AWS Toolkit for Visual Studio:

> **Note**
>
> Ketika profil pengguna baru ditambahkan dari toolkit:
>
> - Jika `credentials file` sudah ada, informasi pengguna baru ditambahkan ke file yang ada.
> - Jika `credentials file` tidak ada file baru dibuat.

1. Dari AWS Explorer pilih ikon Profil Akun Baru untuk membuka dialog Profil Akun Baru.



2. Lengkapi kolom wajib di dialog Profil Akun Baru dan pilih tombol OK untuk membuat pengguna IAM.

Mengedit kredensi pengguna IAM dari toolkit

Untuk mengedit kredensi pengguna IAM dari toolkit, selesaikan langkah-langkah berikut:

1. Dari drop-down Credentials di AWS Explorer, pilih kredensi pengguna IAM yang ingin Anda edit.
2. Pilih ikon Edit Profil untuk membuka dialog Edit Profil.
3. Dari dialog Edit Profil, selesaikan pembaruan Anda dan pilih OK tombol untuk menyimpan perubahan Anda.

Untuk menghapus kredensi pengguna IAM dari toolkit, selesaikan langkah-langkah berikut:

1. Dari tarik-turun Kredensial di AWS Explorer, pilih kredensi pengguna IAM yang ingin Anda hapus.
2. Pilih ikon Hapus Profil untuk membuka prompt Hapus Profil.
3. Konfirmasikan bahwa Anda ingin menghapus profil untuk menghapusnya dari profil Anda`Credentials file`.

> **Important**
>
> Profil yang mendukung fitur akses lanjutan, seperti Pusat Identitas IAM atau otentikasi Multi-faktor (MFA) dalam dialog Edit Profil, tidak dapat diedit dari file. AWS Toolkit for Visual Studio

Untuk membuat perubahan pada jenis profil ini, Anda harus mengedit `credentials file` menggunakan editor teks.

Mengedit kredensi pengguna IAM dari editor teks

Selain mengelola pengguna IAM dengan AWS Toolkit for Visual Studio, Anda dapat mengedit `credential files` dari editor teks pilihan Anda. Lokasi default `credential file` di Windows adalah`C:\Users\`*`USERNAME`*`\.aws\credentials`.

Untuk detail selengkapnya tentang lokasi dan struktur`credential files`, lihat bagian File konfigurasi bersama dan panduan Referensi AWS SDKs Alat.

Membuat pengguna IAM dari AWS Command Line Interface ()AWS CLI

AWS CLI Ini adalah alat lain yang dapat Anda gunakan untuk membuat pengguna IAM di`credentials file`, menggunakan perintah`aws configure`.

Untuk informasi rinci tentang membuat pengguna IAM dari AWS CLI lihat Mengkonfigurasi AWS CLI topik dalam AWS CLI Panduan Pengguna.

Toolkit for Visual Studio mendukung properti konfigurasi berikut:

```
aws_access_key_id
aws_secret_access_key
aws_session_token
credential_process
credential_source
external_id
mfa_serial
role_arn
role_session_name
source_profile
sso_account_id
sso_region
sso_role_name
sso_start_url
```

AWS ID Pembangun

AWS Builder ID adalah metode AWS otentikasi tambahan yang mungkin diperlukan untuk menggunakan layanan atau fitur tertentu, seperti mengkloning repositori pihak ketiga dengan Amazon. CodeCatalyst

Untuk informasi lebih lanjut tentang metode autentikasi AWS Builder ID, lihat topik Masuk dengan AWS Builder ID di Panduan Pengguna AWS Masuk.

Untuk informasi tambahan tentang mengkloning repositori CodeCatalyst dari AWS Toolkit for Visual Studio, lihat CodeCatalyst topik Bekerja dengan Amazon di Panduan Pengguna ini.

Otentikasi multi-faktor (MFA) di Toolkit for Visual Studio

Otentikasi multi-faktor (MFA) adalah keamanan tambahan untuk akun Anda. AWS MFA mengharuskan pengguna untuk memberikan kredensi masuk dan otentikasi unik dari mekanisme MFA yang AWS didukung saat mengakses situs web atau layanan. AWS

AWS mendukung berbagai perangkat virtual dan perangkat keras untuk otentikasi MFA. Berikut ini adalah contoh perangkat MFA virtual yang diaktifkan melalui aplikasi smartphone. Untuk informasi selengkapnya tentang opsi perangkat MFA, lihat Menggunakan otentikasi multi-faktor (MFA) AWS di Panduan Pengguna IAM.

Langkah 1: Membuat peran IAM untuk mendelegasikan akses ke pengguna IAM

Prosedur berikut menjelaskan cara mengatur deligasi peran untuk menetapkan izin ke pengguna IAM. Untuk informasi rinci tentang deligasi peran, lihat Membuat peran untuk mendelegasikan izin ke topik pengguna IAM di Panduan Pengguna.AWS Identity and Access Management

1. Pergi ke konsol IAM di https://console.aws.amazon.com/iam.
2. Pilih Peran di bilah navigasi, lalu pilih Buat Peran.
3. Di halaman Buat peran, pilih AWS Akun lain.
4. Masukkan ID Akun yang Anda butuhkan dan tandai kotak centang Memerlukan MFA.

> **Note**
>
> Untuk menemukan 12 digit nomor akun (ID) Anda, buka bilah navigasi di konsol, lalu pilih Support, Support Center.

5. Pilih Berikutnya: Izin.
6. Lampirkan kebijakan yang ada ke peran Anda atau buat kebijakan baru untuknya. Kebijakan yang Anda pilih di halaman ini menentukan AWS layanan mana yang dapat diakses pengguna IAM dengan Toolkit.
7. Setelah melampirkan kebijakan, pilih Berikutnya: Tag untuk opsi menambahkan tag IAM ke peran Anda. Kemudian pilih Berikutnya: Tinjau untuk melanjutkan.
8. Di halaman Tinjauan, masukkan nama Peran yang diperlukan (toolkit-role, misalnya). Anda juga dapat menambahkan deskripsi Peran opsional.
9. Pilih Buat peran.
10. Ketika pesan konfirmasi ditampilkan (“Peran toolkit-peran telah dibuat”, misalnya), pilih nama peran dalam pesan.
11. Di halaman Ringkasan, pilih ikon salin untuk menyalin ARN Peran dan tempelkan ke dalam file. (Anda memerlukan ARN ini saat mengonfigurasi pengguna IAM untuk mengambil peran.).

Langkah 2: Membuat pengguna IAM yang mengasumsikan izin peran

Langkah ini membuat pengguna IAM tanpa izin sehingga kebijakan in-line dapat ditambahkan.

1. Pergi ke konsol IAM di https://console.aws.amazon.com/iam.
2. Pilih Pengguna di bilah navigasi dan kemudian pilih Tambah pengguna.
3. Di halaman Tambah pengguna, masukkan nama pengguna yang diperlukan (toolkit-user, misalnya) dan tandai kotak centang Akses program.
4. Pilih Berikutnya: Izin, Berikutnya: Tag, dan Berikutnya: Tinjau untuk bergerak melalui halaman berikutnya. Anda tidak menambahkan izin pada tahap ini karena pengguna akan mengambil izin peran.
5. Di halaman Tinjauan, Anda diberi tahu bahwa Pengguna ini tidak memiliki izin. Pilih Create user (Buat pengguna).
6. Di halaman Sukses, pilih Unduh.csv untuk mengunduh file yang berisi ID kunci akses dan kunci akses rahasia. (Anda memerlukan keduanya saat menentukan profil pengguna di file kredensial.)

7. Pilih Tutup.

Langkah 3: Menambahkan kebijakan untuk memungkinkan pengguna IAM mengambil peran

Prosedur berikut membuat kebijakan in-line yang memungkinkan pengguna untuk mengambil peran (dan izin peran tersebut).

1. Di halaman Pengguna konsol IAM, pilih pengguna IAM yang baru saja Anda buat (toolkit-user, misalnya).
2. Di tab Izin pada halaman Ringkasan, pilih Tambahkan kebijakan sebaris.
3. Di halaman Buat kebijakan, pilih Pilih layanan, masukkan STS di Temukan layanan, lalu pilih STS dari hasilnya.
4. Untuk Tindakan, mulailah memasukkan istilah AssumeRole. Tandai kotak AssumeRolecentang saat muncul.
5. Di bagian Sumber Daya, pastikan Spesifik dipilih, dan klik Tambahkan ARN untuk membatasi akses.
6. Dalam Tambahkan ARN kotak dialog, untuk Tentukan ARN untuk peran tambahkan ARN dari peran yang Anda buat di Langkah 1.

   Setelah Anda menambahkan ARN peran, akun tepercaya dan nama peran yang terkait dengan peran tersebut akan ditampilkan di Akun dan nama Peran dengan jalur.
7. Pilih Tambahkan.
8. Kembali ke halaman Buat kebijakan, pilih Tentukan kondisi permintaan (opsional), tandai kotak centang MFA wajib, lalu pilih dekat untuk mengonfirmasi..
9. Pilih Tinjau kebijakan
10. Di halaman Kebijakan tinjauan, masukkan Nama untuk kebijakan, lalu pilih Buat kebijakan.

    Tab Izin menampilkan kebijakan inline baru yang dilampirkan langsung ke pengguna IAM.

Langkah 4: Mengelola perangkat MFA virtual untuk pengguna IAM

1. Unduh dan instal aplikasi MFA virtual ke ponsel cerdas Anda.

   Untuk daftar aplikasi yang didukung, lihat halaman sumber daya Otentikasi Multi-faktor.

2. Di konsol IAM, pilih Pengguna dari bilah navigasi dan kemudian pilih pengguna yang mengambil peran (toolkit-user, dalam hal ini).
3. Di halaman Ringkasan, pilih tab Security credentials, dan untuk perangkat MFA yang Ditugaskan pilih Kelola.
4. Di panel Kelola perangkat MFA, pilih Perangkat MFA virtual, lalu pilih Lanjutkan.
5. Di panel Siapkan perangkat MFA virtual, pilih Tampilkan kode QR dan kemudian pindai kode menggunakan aplikasi MFA virtual yang Anda instal di ponsel cerdas Anda.
6. Setelah Anda memindai kode QR, aplikasi MFA virtual menghasilkan kode MFA satu kali. Masukkan dua kode MFA berturut-turut dalam kode MFA 1 dan kode MFA 2.
7. Pilih Tugaskan MFA.
8. Kembali ke tab Security credentials untuk pengguna, salin ARN dari perangkat MFA yang Ditugaskan baru.

   ARN menyertakan ID akun 12 digit Anda dan formatnya mirip dengan yang berikut:. `arn:aws:iam::123456789012:mfa/toolkit-user` Anda memerlukan ARN ini saat mendefinisikan profil MFA di langkah berikutnya.

Langkah 5: Membuat profil untuk memungkinkan MFA

Prosedur berikut membuat profil yang memungkinkan MFA saat mengakses AWS layanan dari Toolkit for Visual Studio.

Profil yang Anda buat mencakup tiga bagian informasi yang telah Anda salin dan simpan selama langkah-langkah sebelumnya:

- Kunci akses (ID kunci akses dan kunci akses rahasia) untuk pengguna IAM
- ARN dari peran yang mendelegasikan izin ke pengguna IAM
- ARN dari perangkat MFA virtual yang ditetapkan untuk pengguna IAM

Di file kredenal AWS bersama atau SDK Store yang berisi AWS kredensi Anda, tambahkan entri berikut:

```
[toolkit-user]
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
```

```
[mfa]
source_profile = toolkit-user
role_arn = arn:aws:iam::111111111111:role/toolkit-role
mfa_serial = arn:aws:iam::111111111111:mfa/toolkit-user
```

Ada dua profil yang didefinisikan dalam contoh yang diberikan:

- `[toolkit-user]`profil termasuk kunci akses dan kunci akses rahasia yang dihasilkan dan disimpan saat Anda membuat pengguna IAM di Langkah 2.
- `[mfa]`profil mendefinisikan bagaimana otentikasi multi-faktor didukung. Ada tiga entri:

  ◦`source_profile`: Menentukan profil yang kredensialnya digunakan untuk mengambil peran yang ditentukan oleh `role_arn` pengaturan ini dalam profil ini. Dalam hal ini, itu adalah `toolkit-user` profilnya.

  ◦`role_arn`: Menentukan Nama Sumber Daya Amazon (ARN) dari peran IAM yang ingin Anda gunakan untuk melakukan operasi yang diminta menggunakan profil ini. Dalam hal ini, ini adalah ARN untuk peran yang Anda buat di Langkah 1.

  ◦`mfa_serial`: Menentukan identifikasi atau nomor seri perangkat MFA yang harus digunakan pengguna saat mengambil peran. Dalam hal ini, itu adalah ARN dari perangkat virtual yang Anda atur di Langkah 3.

Menyiapkan kredensi eksternal

Jika Anda memiliki metode untuk menghasilkan atau mencari kredensional yang tidak didukung secara langsung AWS, Anda dapat menambahkan ke file kredensi bersama profil yang berisi setelan. `credential_process` Pengaturan ini menentukan perintah eksternal yang dijalankan untuk menghasilkan atau mengambil kredensi otentikasi untuk digunakan. Misalnya, Anda mungkin menyertakan entri yang mirip dengan yang berikut ini dalam `config` file:

```
[profile developer]
credential_process = /opt/bin/awscreds-custom --username helen
```

Untuk informasi selengkapnya tentang penggunaan kredensi eksternal dan risiko keamanan terkait, lihat Sumber kredensi dengan proses eksternal di Panduan Pengguna.AWS Command Line Interface

Memperbarui firewall dan gateway untuk memungkinkan akses

Jika Anda memfilter akses ke AWS domain atau titik akhir URL tertentu dengan menggunakan solusi pemfilteran konten web, titik akhir berikut harus diizinkan terdaftar untuk mengakses semua layanan dan fitur yang tersedia melalui AWS Toolkit for Visual Studio Amazon Q. Untuk langkah-langkah terperinci tentang cara memecahkan masalah firewall dan pengaturan proxy untuk Toolkit AWS dengan Amazon Q, lihat bagian Firewall dan pengaturan proxy di topik Pemecahan Masalah dalam Panduan Pengguna ini. Untuk informasi terperinci tentang mengonfigurasi proxy perusahaan untuk Amazon Q, lihat topik Mengonfigurasi proxy perusahaan di Amazon Q di Panduan Pengguna Pengembang Amazon Q.

AWS Toolkit for Visual Studio Titik akhir

Berikut ini adalah daftar titik akhir AWS Toolkit for Visual Studio tertentu dan referensi yang perlu diizinkan terdaftar.

Titik akhir

```
https://idetoolkits-hostedfiles.amazonaws.com/*
https://idetoolkits.amazonwebservices.com/*
http://vstoolkit.amazonwebservices.com/*
https://aws-vs-toolkit.s3.amazonaws.com/*
https://raw.githubusercontent.com/aws/aws-toolkit-visual-studio/main/version.json
https://aws-toolkit-language-servers.amazonaws.com/*
```

Titik akhir plugin Amazon Q

Berikut ini adalah daftar titik akhir dan referensi khusus plugin Amazon Q yang perlu diizinkan terdaftar.

```
https://idetoolkits-hostedfiles.amazonaws.com/*    (Plugin for configs)
https://idetoolkits.amazonwebservices.com/*   (Plugin for endpoints)
https://aws-toolkit-language-servers.amazonaws.com/*  (Language Server Process)
https://client-telemetry.us-east-1.amazonaws.com/ (Telemetry)
https://cognito-identity.us-east-1.amazonaws.com    (Telemetry)
https://aws-language-servers.us-east-1.amazonaws.com (Language Server Process)
```

Titik akhir Pengembang Amazon Q

Berikut ini adalah daftar titik akhir dan referensi khusus Pengembang Amazon Q yang perlu diizinkan terdaftar.

```
https://codewhisperer.us-east-1.amazonaws.com (Inline,Chat, QSDA,...)
https://q.us-east-1.amazonaws.com (Inline,Chat, QSDA....)
https://desktop-release.codewhisperer.us-east-1.amazonaws.com/ (Download URL for CLI.)
https://specs.q.us-east-1.amazonaws.com (URL for auto-complete specs used by CLI)
* aws-language-servers.us-east-1.amazonaws.com (Local Workspace context)
```

Titik Akhir Transformasi Kode Q Amazon

Berikut ini adalah daftar titik akhir dan referensi spesifik Amazon Q Code Transform yang perlu diizinkan terdaftar.

```
https://docs.aws.amazon.com/amazonq/latest/qdeveloper-ug/security_iam_manage-access-
with-policies.html
```

Titik akhir otentikasi

Berikut ini adalah daftar endpoint otentikasi dan referensi yang perlu diizinkan terdaftar.

```
[Directory ID or alias].awsapps.com
* oidc.[Region].amazonaws.com
*.sso.[Region].amazonaws.com
*.sso-portal.[Region].amazonaws.com
*.aws.dev
*.awsstatic.com
*.console.aws.a2z.com
*.sso.amazonaws.com
```

Titik Akhir Identitas

Daftar berikut berisi titik akhir yang spesifik untuk identitas, seperti AWS IAM Identity Center dan AWS Builder ID.

AWS IAM Identity Center

Untuk detail tentang titik akhir yang diperlukan untuk Pusat Identitas IAM, lihat topik Aktifkan Pusat Identitas IAM di Panduan Pengguna. AWS IAM Identity Center

Pusat Identitas IAM Perusahaan

```
https://[Center director id].awsapps.com/start (should be permitted to initiate auth)
https://us-east-1.signin.aws (for facilitating authentication, assuming IAM Identity
 Center is in IAD)
https://oidc.(us-east-1).amazonaws.com
https://log.sso-portal.eu-west-1.amazonaws.com
https://portal.sso.eu-west-1.amazonaws.com
```

AWS ID Pembangun

```
https://view.awsapps.com/start (must be blocked to disable individual tier)
https://codewhisperer.us-east-1.amazonaws.com and q.us-east-1.amazonaws.com (should be
 permitted)
```

Telemetri

Berikut ini adalah titik akhir khusus Telemetri yang perlu diizinkan terdaftar.

```
https://telemetry.aws-language-servers.us-east-1.amazonaws.com/
https://client-telemetry.us-east-1.amazonaws.com
```

Referensi

Berikut ini adalah daftar referensi endpoint.

```
idetoolkits-hostedfiles.amazonaws.com
cognito-identity.us-east-1.amazonaws.com
```

```
amazonwebservices.gallery.vsassets.io
eu-west-1.prod.pr.analytics.console.aws.a2z.com
prod.pa.cdn.uis.awsstatic.com
portal.sso.eu-west-1.amazonaws.com
log.sso-portal.eu-west-1.amazonaws.com
prod.assets.shortbread.aws.dev
prod.tools.shortbread.aws.dev
prod.log.shortbread.aws.dev
a.b.cdn.console.awsstatic.com
assets.sso-portal.eu-west-1.amazonaws.com
oidc.eu-west-1.amazonaws.com
aws-toolkit-language-servers.amazonaws.com
aws-language-servers.us-east-1.amazonaws.com
idetoolkits.amazonwebservices.com
```

Bekerja dengan AWS Layanan

Topik berikut menjelaskan cara memulai bekerja dengan AWS layanan dari AWS Toolkit for Visual Studio dengan Amazon Q.

Topik

Amazon CodeCatalyst untuk AWS Toolkit for Visual Studio dengan Amazon Q

Apa itu Amazon CodeCatalyst?

Amazon CodeCatalyst adalah ruang kolaborasi berbasis cloud untuk tim pengembangan perangkat lunak. Menggunakan AWS Toolkit for Visual Studio dengan Amazon Q, Anda dapat melihat dan

CodeCatalyst mengelola sumber daya langsung AWS dari Toolkit for Visual Studio dengan Amazon Q. Untuk CodeCatalyst informasi selengkapnya, lihat Panduan Pengguna Amazon. CodeCatalyst

Topik berikut menjelaskan cara menghubungkan AWS Toolkit for Visual Studio dengan Amazon Q dan cara bekerja CodeCatalyst dengan melalui AWS Toolkit for Visual Studio CodeCatalyst dengan Amazon Q.

Topik

Memulai Amazon CodeCatalyst dan AWS Toolkit for Visual Studio dengan Amazon Q

Untuk mulai bekerja dengan Amazon CodeCatalyst dari AWS Toolkit for Visual Studio dengan Amazon Q, lengkapi yang berikut ini.

Topik

Menginstal AWS Toolkit for Visual Studio dengan Amazon Q

Sebelum Anda mengintegrasikan AWS Toolkit for Visual Studio dengan Amazon Q dengan akun CodeCatalyst Anda, pastikan Anda menggunakan Toolkit for Visual Studio versi AWS terbaru dengan Amazon Q. Untuk detail tentang cara menginstal dan menyiapkan Toolkit for Visual Studio AWS versi terbaru dengan Amazon Q, lihat Menyiapkan Toolkit for Visual Studio dengan Amazon Q AWS bagian dari Panduan Pengguna ini.

Membuat CodeCatalyst akun dan AWS Builder ID

Selain menginstal AWS Toolkit for Visual Studio versi terbaru dengan Amazon Q, Anda harus memiliki Builder ID AWS aktif CodeCatalyst dan akun untuk terhubung AWS dengan Toolkit for Visual

Studio dengan Amazon Q. Jika Anda tidak memiliki Builder CodeCatalyst ID atau AWS akun aktif, lihat bagian CodeCatalyst CodeCatalystMenyiapkan dengan di Panduan Pengguna.

> **Note**
>
> AWS Builder ID berbeda dari AWS Kredensial Anda. Untuk petunjuk tentang cara mendaftar dan mengautentikasi dengan AWS Builder ID, lihat topik Autentikasi dan akses: AWS Builder ID di Panduan Pengguna ini.
> Untuk informasi lebih lanjut tentang AWS Builder IDs, lihat topik AWS Builder ID di Panduan Pengguna Referensi AWS Umum.

Menghubungkan AWS Toolkit for Visual Studio dengan Amazon Q CodeCatalyst

Untuk menghubungkan AWS Toolkit for Visual Studio dengan Amazon Q dengan akun CodeCatalyst Anda, selesaikan langkah-langkah berikut.

1. Dari item menu Git di Visual Studio, pilih Clone Repository... .
2. Dari bagian Browse a Repository, pilih Amazon CodeCatalyst sebagai penyedia.
3. Dari bagian Koneksi, pilih Connect with AWS Builder ID untuk membuka CodeCatalyst konsol di browser web pilihan Anda.
4. Dari browser Anda, masukkan AWS Builder ID Anda ke dalam kolom yang disediakan dan ikuti petunjuk untuk melanjutkan.
5. Saat diminta, pilih Izinkan untuk mengonfirmasi koneksi antara AWS Toolkit for Visual Studio dengan Amazon Q dan CodeCatalyst akun Anda. Ketika proses koneksi selesai, CodeCatalyst menampilkan konfirmasi yang menunjukkan bahwa aman untuk menutup browser Anda.

Bekerja dengan CodeCatalyst sumber daya Amazon dari AWS Toolkit for Visual Studio dengan Amazon Q

Bagian berikut memberikan ikhtisar fitur manajemen CodeCatalyst sumber daya Amazon Amazon yang tersedia untuk AWS Toolkit for Visual Studio dengan Amazon Q.

Topik

- Mengkloning repositori

Mengkloning repositori

CodeCatalyst adalah layanan berbasis cloud yang mengharuskan Anda terhubung ke cloud untuk mengerjakan CodeCatalyst proyek. Untuk mengerjakan proyek secara lokal, Anda dapat mengkloning CodeCatalyst repositori ke mesin lokal Anda dan menyinkronkan dengan CodeCatalyst proyek Anda saat berikutnya Anda terhubung ke cloud.

Untuk mengkloning repositori ke mesin lokal Anda, selesaikan langkah-langkah berikut.

1. Dari item menu Git di Visual Studio, pilih Clone Repository... .
2. Dari bagian Browse a Repository, pilih Amazon CodeCatalyst sebagai penyedia.

> **Note**
>
> Jika bagian Sambungan menampilkan `Not Connected` pesan, selesaikan langkah-langkah di bagian Autentikasi dan akses: AWS Builder ID dari Panduan Pengguna ini sebelum melanjutkan.

3. Pilih Space dan Project tempat Anda ingin mengkloning repositori.
4. Dari bagian Repositori, pilih repositori yang ingin Anda kloning.
5. Dari bagian Path, pilih folder yang ingin Anda kloning repositori Anda.

> **Note**
>
> Folder ini awalnya harus kosong untuk mengkloning dengan sukses.

6. Pilih Clone untuk mulai mengkloning repositori.
7. Setelah repositori dikloning, Visual Studio akan memuat solusi kloning Anda

> **Note**
>
> Jika Visual Studio tidak membuka solusi di repositori kloning, opsi Visual Studio Anda dapat disesuaikan dari otomatis memuat solusi saat membuka pengaturan repositori Git, yang terletak di Pengaturan Global Git, dari menu Kontrol Sumber.

Pemecahan masalah

Berikut ini adalah topik pemecahan masalah untuk mengatasi masalah yang diketahui saat bekerja dengan Amazon CodeCatalyst dari AWS Toolkit for Visual Studio dengan Amazon Q.

Topik

Kredensial

Jika Anda menemukan dialog yang meminta kredensial saat mencoba mengkloning repositori berbasis git dari, pembantu AWS CodeCommit Credential Anda dapat dikonfigurasi secara CodeCatalyst global, menyebabkan interferensi. CodeCatalyst Untuk informasi tambahan tentang pembantu AWS CodeCommit kredenal, lihat Mengatur langkah untuk koneksi HTTPS ke AWS CodeCommit repositori di Windows dengan bagian pembantu kredenal AWS CLI pada Panduan Pengguna. AWS CodeCommit

Untuk membatasi penanganan AWS CodeCommit Credential helper saja CodeCommit URLs, selesaikan langkah-langkah berikut.

1. buka file konfigurasi git global di: `%userprofile%\.gitconfig`
2. Temukan bagian berikut di file Anda:

```
[credential]
    helper = !aws codecommit credential-helper $@
    UseHttpPath = true
```

3. Ubah bagian itu menjadi yang berikut:

```
[credential "https://git-codecommit.*.amazonaws.com"]
    helper = !aws codecommit credential-helper $@
    UseHttpPath = true
```

4. Simpan perubahan Anda, lalu selesaikan langkah-langkah untuk mengkloning repositori Anda.

Integrasi Amazon CloudWatch Logs untuk Visual Studio

Integrasi Amazon CloudWatch Logs dari AWS Toolkit for Visual Studio dengan Amazon Q memberi Anda kemampuan untuk memantau, menyimpan, dan CloudWatch mengakses sumber daya Log, tanpa harus meninggalkan IDE Anda. Untuk mempelajari lebih lanjut tentang menyiapkan CloudWatch layanan dan cara bekerja dengan fitur CloudWatch Log, pilih dari topik berikut.

Topik

Menyiapkan integrasi CloudWatch Log untuk Visual Studio

Sebelum Anda dapat menggunakan integrasi Amazon CloudWatch Logs dengan AWS Toolkit dengan Amazon Q, Anda memerlukan AWS akun. Anda dapat membuat AWS akun baru dari situs AWS masuk. Sebagian besar fitur CloudWatch Log yang tersedia dari AWS Toolkit dengan Amazon Q dapat diakses dengan AWS kredensi aktif. Jika fitur tertentu memerlukan konfigurasi tambahan, persyaratan disertakan dalam bagian yang relevan dari panduan Bekerja dengan CloudWatch Log.

Untuk informasi dan opsi tambahan tentang pengaturan CloudWatch Log, lihat bagian Menyiapkan pada panduan Amazon CloudWatch Logs.

Bekerja dengan CloudWatch Log di Visual Studio

Integrasi Amazon CloudWatch Logs memungkinkan Anda memantau, menyimpan, dan mengakses CloudWatch Log dari AWS Toolkit for Visual Studio dengan Amazon Q. Memiliki akses CloudWatch ke fitur Log — tanpa perlu meninggalkan IDE Anda — meningkatkan efisiensi dengan menyederhanakan proses pengembangan Log dan mengurangi gangguan CloudWatch pada alur kerja Anda. Topik berikut menjelaskan cara bekerja dengan fitur dasar dan fungsi integrasi CloudWatch Log.

Topik

CloudWatch Grup Log

A `log group` adalah grup `log streams` yang berbagi pengaturan retensi, pemantauan, dan kontrol akses yang sama. Tidak ada batas jumlah log stream yang dapat tergabung dalam satu grup log.

Melihat Grup Log

`View Log Groups`Fitur ini menampilkan daftar grup log di CloudWatch Log Groups Explorer.

Untuk mengakses fitur View Log Groups dan membuka CloudWatch Log Groups Explorer, selesaikan langkah-langkah berikut.

1. Dari AWS Explorer, perluas Amazon CloudWatch.
2. Klik dua kali Grup Log atau buka menu konteks (klik kanan) dan pilih Lihat, untuk membuka Penjelajah Grup CloudWatch Log.

> **Note**
>
> Penjelajah Grup CloudWatch Log akan terbuka di lokasi jendela yang sama dengan Solutions Explorer.

Memfilter Grup Log

Akun individual Anda dapat berisi ribuan grup log yang berbeda. Untuk menyederhanakan pencarian Anda untuk grup tertentu, gunakan `filtering` fitur yang dijelaskan di bawah ini.

1. Dari Penjelajah Grup CloudWatch Log, atur kursor Anda di bilah pencarian yang terletak di bagian atas jendela.
2. Mulai mengetik awalan yang terkait dengan grup log yang Anda cari.
3. CloudWatch Penjelajah Grup Log diperbarui secara otomatis untuk menampilkan hasil yang cocok dengan istilah pencarian yang Anda tentukan pada langkah sebelumnya.

Hapus Grup Log

Untuk menghapus grup log tertentu, lihat prosedur berikut.

1. Dari Penjelajah Grup CloudWatch Log, klik kanan Grup Log yang ingin Anda hapus.

2. Saat diminta, konfirmasikan bahwa Anda ingin menghapus Grup Log yang saat ini dipilih.
3. Memilih tombol Ya menghapus grup log yang dipilih, lalu menyegarkan Penjelajah Grup CloudWatch Log.

Segarkan Grup Log

Untuk menyegarkan daftar grup log saat ini yang ditampilkan di Penjelajah Grup CloudWatch Log, pilih tombol ikon Segarkan yang terletak di bilah alat.

Salin Grup Log ARN

Untuk menyalin ARN dari grup log tertentu, selesaikan langkah-langkah yang dijelaskan di bawah ini.

1. Dari Penjelajah Grup CloudWatch Log, klik kanan Grup Log tempat Anda ingin menyalin ARN.
2. Pilih opsi Salin ARN dari menu.
3. ARN sekarang disalin ke clipboard lokal Anda dan siap ditempel.

CloudWatch Aliran Log

Pengaliran log adalah urutan log acara yang berbagi sumber yang sama.

> **Note**
>
> Saat melihat aliran log, perhatikan properti berikut:
>
> - Secara default, aliran log diurutkan berdasarkan stempel waktu peristiwa terbaru.
> - Kolom yang terkait dengan aliran log dapat diurutkan dalam urutan naik atau turun, dengan mengaktifkan tanda sisipan yang terletak di header kolom.
> - Entri yang difilter hanya dapat diurutkan berdasarkan Nama Aliran Log.

Melihat Aliran Log

1. Dari Penjelajah Grup CloudWatch Log klik dua kali Grup Log, atau klik kanan grup log dan pilih Lihat Aliran Log dari menu konteks.
2. Tab baru akan terbuka di jendela dokumen, yang berisi daftar aliran log yang terkait dengan grup log Anda.

Memfilter Aliran Log

1. Dari tab Aliran Log, di jendela dokumen, atur kursor Anda di bilah pencarian.
2. Mulai ketikkan awalan yang terkait dengan aliran log yang Anda cari.
3. Saat Anda mengetik, tampilan saat ini secara otomatis diperbarui untuk memfilter Aliran Log Anda berdasarkan input Anda.

Segarkan Aliran Log

Untuk menyegarkan daftar aliran log saat ini yang ditampilkan di jendela dokumen, pilih tombol ikon Refresh, yang terletak di bilah alat, di sebelah bilah pencarian.

Salin Aliran Log ARN

Untuk menyalin ARN dari aliran log tertentu, selesaikan langkah-langkah yang dijelaskan di bawah ini.

1. Dari tab Log Streams, di jendela dokumen, klik kanan aliran log yang ingin Anda salin ARN.
2. Pilih opsi Salin ARN dari menu.
3. ARN sekarang disalin ke clipboard lokal Anda dan siap ditempel.

Unduh Aliran Log

Fitur Export Log Stream mengunduh dan menyimpan aliran log yang dipilih secara lokal, di mana ia dapat diakses oleh alat dan perangkat lunak khusus untuk pemrosesan tambahan.

1. Dari tab Log Streams, di jendela dokumen, klik kanan aliran log yang ingin Anda unduh.
2. Pilih Ekspor Log Stream untuk membuka dialog Ekspor ke file teks.
3. Pilih lokasi tempat Anda ingin menyimpan file secara lokal dan tentukan nama di bidang teks yang disediakan.
4. Konfirmasikan unduhan dengan memilih OK. Status unduhan ditampilkan di Pusat Status Tugas Visual Studio

CloudWatch Peristiwa Log

Peristiwa log adalah catatan aktivitas yang direkam oleh aplikasi atau sumber daya yang dipantau oleh CloudWatch.

Tindakan Peristiwa Log

Peristiwa log ditampilkan sebagai tabel. Secara default, acara diurutkan dari acara tertua ke yang terbaru.

Tindakan berikut dikaitkan dengan peristiwa log di Visual Studio:

- Mode teks terbungkus: Anda dapat mengaktifkan teks yang dibungkus dengan mengklik acara.
- Tombol pembungkus teks: terletak di`document window` **`toolbar`**, tombol ini mengaktifkan dan menonaktifkan pembungkus teks, untuk semua entri.
- Salin pesan ke clipboard Anda: pilih pesan yang ingin Anda salin, lalu klik kanan pilihan dan pilih Salin (pintasan keyboard). `Ctrl + C`

Melihat Peristiwa Log

1. Dari jendela dokumen, pilih tab yang berisi daftar aliran log.
2. Klik dua kali aliran log, atau klik kanan aliran log dan pilih Lihat Aliran Log dari menu.
3. Tab peristiwa log baru akan terbuka di jendela dokumen, yang berisi tabel peristiwa log yang terkait dengan aliran log pilihan Anda.

Memfilter Peristiwa Log

Ada tiga cara bagi Anda untuk memfilter peristiwa log: berdasarkan konten, rentang waktu, atau keduanya. Untuk memfilter peristiwa log Anda berdasarkan konten dan rentang waktu, mulailah dengan memfilter pesan Anda berdasarkan konten atau rentang waktu, lalu filter hasil tersebut dengan metode lain.

Untuk memfilter peristiwa log Anda berdasarkan konten:

1. Dari tab peristiwa log, di jendela dokumen, atur kursor Anda di bilah pencarian, yang terletak di bagian atas jendela.
2. Mulai ketikkan istilah atau frasa yang terkait dengan peristiwa log yang Anda cari.
3. Saat Anda mengetik, tampilan saat ini secara otomatis mulai memfilter peristiwa log Anda.

> **Note**
>
> Pola filter peka huruf besar/kecil. Anda dapat meningkatkan hasil pencarian dengan melampirkan istilah, dan frasa yang tepat, dengan karakter non-alfanumerik dalam tanda

kutip ganda (*""*). Untuk informasi lebih rinci tentang pola filter, lihat topik Filter dan Sintaks Pola di CloudWatch panduan Amazon.

Untuk melihat peristiwa log yang dihasilkan selama rentang waktu tertentu:

1. Dari tab peristiwa log, di jendela dokumen, pilih tombol ikon Kalender, yang terletak di bilah alat.
2. Menggunakan bidang yang disediakan, tentukan rentang waktu yang ingin Anda cari.
3. Hasil yang difilter diperbarui secara otomatis saat Anda menentukan batasan tanggal dan waktu.

> **Note**
>
> Opsi Hapus Filter menghapus semua pilihan date-and-time filter Anda saat ini.

Segarkan Peristiwa Log

Untuk menyegarkan daftar peristiwa log saat ini yang ditampilkan di tab peristiwa log, pilih tombol ikon Segarkan, yang terletak di bilah alat.

Akses tambahan ke CloudWatch Log

Anda dapat mengakses CloudWatch Log yang terkait dengan AWS layanan dan sumber daya lain langsung dari AWS Toolkit di Visual Studio.

Lambda

Untuk melihat aliran log yang terkait dengan fungsi Lambda:

> **Note**
>
> Peran eksekusi Lambda Anda harus memiliki izin yang sesuai untuk mengirim log ke Log. CloudWatch Untuk informasi selengkapnya tentang izin Lambda yang diperlukan untuk CloudWatch Log, lihat https://docs.aws.amazon.com/lambda/latest/dg/monitoring-cloudwatchlogs.html#monitoring-cloudwatchlogs-prereqs

1. Dari AWS Toolkit Explorer, perluas Lambda.

2. klik kanan fungsi yang ingin Anda lihat, lalu pilih Lihat Log untuk membuka aliran log terkait di jendela dokumen.

Untuk melihat aliran log menggunakan integrasi `function view` Lambda:

1. Dari AWS Toolkit Explorer, perluas Lambda.
2. klik kanan fungsi yang ingin Anda lihat, lalu pilih View Function untuk membuka tampilan fungsi di jendela dokumen.
3. Dari`function view`, beralih ke tab Log, aliran log yang terkait dengan fungsi Lambda yang dipilih ditampilkan.

ECS

Untuk melihat sumber daya log yang terkait dengan Wadah Tugas ECS, selesaikan prosedur berikut.

> **Note**
>
> Agar layanan Amazon ECS dapat mengirim log CloudWatch, setiap kontainer untuk Tugas Amazon ECS tertentu harus memenuhi konfigurasi yang diperlukan. Untuk informasi tambahan tentang pengaturan dan konfigurasi yang diperlukan, silakan lihat panduan Menggunakan Driver Log AWS Log.

1. Dari AWS Toolkit Explorer, perluas Amazon ECS.
2. Pilih Amazon ECS Cluster yang ingin Anda lihat untuk membuka tab ECS Cluster baru, di jendela dokumen.
3. Dari menu navigasi, yang terletak di sisi kiri tab ECS Cluster, pilih Tugas untuk mencantumkan semua tugas yang terkait dengan cluster.
4. Dari tampilan Tugas, pilih tugas dan pilih tautan Lihat Log, yang terletak di sudut kiri bawah.

> **Note**
>
> Tampilan ini mencantumkan semua tugas yang terdapat dalam cluster, `View Logs` tautan hanya terlihat untuk setiap tugas yang memenuhi konfigurasi log yang diperlukan.
>
> - Jika Tugas hanya dikaitkan dengan satu kontainer, tautan Lihat Log akan membuka aliran log kontainer tersebut.

- Jika Tugas dikaitkan dengan beberapa kontainer, tautan Lihat Log membuka dialog Lihat CloudWatch Log untuk Tugas ECS, gunakan menu drop-down Container: untuk memilih wadah yang ingin Anda lihat Log, lalu pilih OK.

5. Tab baru terbuka di jendela dokumen yang menampilkan aliran log yang terkait dengan pemilihan kontainer Anda.

Mengelola Instans Amazon EC2

AWS Explorer menyediakan tampilan mendetail tentang instans Amazon Machine Images (AMI) dan Amazon Elastic Compute Cloud (Amazon EC2). Dari tampilan ini, Anda dapat meluncurkan instans Amazon EC2 dari AMI, menyambung ke instance tersebut, dan menghentikan atau menghentikan instance, semuanya dari dalam lingkungan pengembangan Visual Studio. Anda dapat menggunakan tampilan instance untuk membuat AMIs dari instance Anda. Untuk informasi selengkapnya, lihat Membuat AMI dari Instans Amazon EC2.

Gambar Mesin Amazon dan Tampilan Instans Amazon EC2

Dari AWS Explorer, Anda dapat menampilkan tampilan Amazon Machine Images (AMI) dan instans Amazon EC2. Di AWS Explorer, perluas node Amazon EC2.

Untuk menampilkan AMIs tampilan, pada subnode pertama, AMIs, buka menu konteks (klik kanan) dan kemudian pilih View.

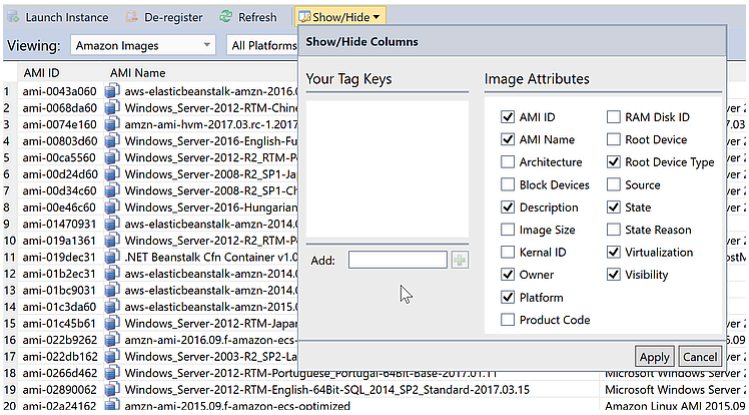
Untuk menampilkan tampilan instans Amazon EC2, pada node Instances, buka menu konteks (klik kanan) lalu pilih View.

Anda juga dapat menampilkan tampilan dengan mengklik dua kali node yang sesuai.

- Tampilan dicakup ke wilayah yang ditentukan di AWS Explorer (misalnya, wilayah AS Barat (California Utara)).
- Anda dapat mengatur ulang kolom dengan mengklik dan menyeret. Untuk mengurutkan nilai dalam kolom, klik judul kolom.
- Anda dapat menggunakan daftar drop-down dan kotak filter di Melihat untuk mengonfigurasi tampilan. Tampilan awal menampilkan semua AMIs jenis platform (Windows atau Linux) yang dimiliki oleh akun yang ditentukan dalam AWS Explorer.

Tampilkan/Sembunyikan Kolom

Anda juga dapat memilih Show/Hide drop-down di bagian atas tampilan untuk mengonfigurasi kolom mana yang ditampilkan. Pilihan kolom Anda akan tetap ada jika Anda menutup tampilan dan membukanya kembali.



Tampilkan/Sembunyikan UI Kolom untuk tampilan AMI dan Instance

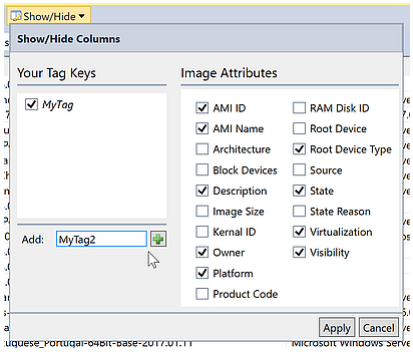
Tagging AMIs, Instans, dan Volume

Anda juga dapat menggunakan daftar drop-down Tampilkan/Sembunyikan untuk menambahkan tag untuk AMI, instans Amazon EC2, atau volume yang Anda miliki. Tag adalah pasangan nama-nilai yang memungkinkan Anda melampirkan metadata ke AMIs, instance, dan volume Anda. Nama tag dicakup baik ke akun Anda dan juga secara terpisah ke akun Anda AMIs dan instance. Misalnya, tidak akan ada konflik jika Anda menggunakan nama tag yang sama untuk instance Anda AMIs dan instance Anda. Nama tag tidak peka huruf besar/kecil.

Untuk informasi selengkapnya tentang tag, buka Menggunakan Tag di Panduan Pengguna Amazon EC2 untuk Instans Linux.

Untuk menambahkan tag

1. Di kotak Tambah, ketikkan nama untuk tag. Pilih tombol hijau dengan tanda plus (+), lalu pilih Terapkan.



Menambahkan tag ke instans AMI atau Amazon EC2

Tag baru ditampilkan dalam huruf miring, yang menunjukkan belum ada nilai yang dikaitkan dengan tag itu.

Dalam tampilan daftar, nama tag muncul sebagai kolom baru. Ketika setidaknya satu nilai telah dikaitkan dengan tag, tag akan terlihat di Konsol Manajemen AWS.

2. Untuk menambahkan nilai tag, klik dua kali sel di kolom untuk tag itu, dan ketikkan nilai. Untuk menghapus nilai tag, klik dua kali sel dan hapus teks.

   Jika Anda menghapus tag di daftar drop-down Tampilkan/Sembunyikan, kolom yang sesuai menghilang dari tampilan. Tag dipertahankan, bersama dengan nilai tag apa pun yang terkait dengan AMIs, instance, atau volume.

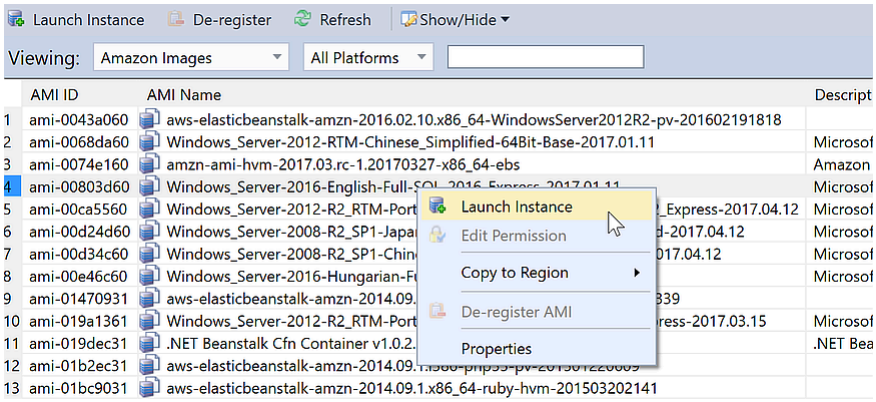
> **Note**
>
> Jika Anda menghapus tag di daftar drop-down Tampilkan/Sembunyikan yang tidak memiliki nilai terkait, AWS Toolkit akan menghapus tag sepenuhnya. Ini tidak akan lagi muncul di tampilan daftar atau di daftar drop-down Tampilkan/Sembunyikan. Untuk menggunakan tag itu lagi, gunakan kotak dialog Tampilkan/Sembunyikan untuk membuatnya kembali.

Meluncurkan Instans Amazon EC2

AWS Explorer menyediakan semua fungsionalitas yang diperlukan untuk meluncurkan instans Amazon EC2. Di bagian ini, kita akan memilih Amazon Machine Image (AMI), mengkonfigurasinya, dan kemudian memulainya sebagai instans Amazon EC2.

Untuk meluncurkan instans Windows Server Amazon EC2

1. Di bagian atas AMIs tampilan, di daftar drop-down di sebelah kiri, pilih Gambar Amazon. Di daftar drop-down di sebelah kanan, pilih Windows. Di kotak filter, ketik `ebs` untuk Penyimpanan Blok Elastis. Mungkin perlu beberapa saat agar pemandangan disegarkan.
2. Pilih AMI dalam daftar, buka menu konteks (klik kanan), lalu pilih Launch Instance.



Daftar AMI

3. Di kotak dialog Luncurkan Instans Amazon EC2 Baru, konfigurasikan AMI untuk aplikasi Anda.

   Jenis Instance

   Pilih jenis instans EC2 yang akan diluncurkan. Anda dapat menemukan daftar jenis instans dan informasi harga di halaman Harga EC2.

   Nama

   Ketik nama untuk instance Anda. Nama ini tidak boleh lebih dari 256 karakter.

   Pasangan Kunci

   Sebuah key pair digunakan untuk mendapatkan password Windows yang Anda gunakan untuk login ke instans EC2 menggunakan Remote Desktop Protocol (RDP). Pilih key pair yang memiliki akses ke private key, atau pilih opsi untuk membuat key pair. Jika Anda membuat key pair di Toolkit, Toolkit dapat menyimpan kunci pribadi untuk Anda.

   Pasangan kunci yang disimpan dalam Toolkit dienkripsi. Anda dapat menemukannya di `%LOCALAPPDATA%\AWSToolkit\keypairs` (biasanya:). `C:\Users\<user>\AppData\Local\AWSToolkit\keypairs` Anda dapat mengekspor key pair terenkripsi ke dalam file. `.pem`

   a. Di Visual Studio, pilih Lihat dan klik AWS Explorer.

   b. Klik Amazon EC2 dan pilih Pasangan Kunci.

   c. Pasangan kunci akan terdaftar, dan yang created/managed oleh Toolkit ditandai sebagai Disimpan di AWSToolkit.

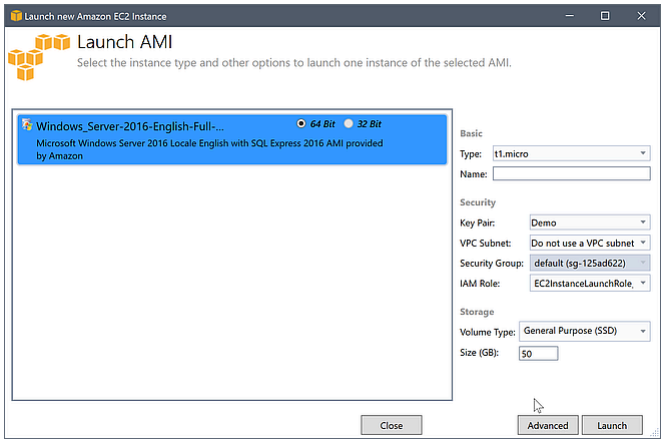
   d. Klik kanan pada key pair yang Anda buat dan pilih Export Private Key. Kunci pribadi akan tidak dienkripsi dan disimpan di lokasi yang Anda tentukan.

Grup Keamanan

Grup keamanan mengontrol jenis lalu lintas jaringan yang akan diterima instans EC2. Pilih grup keamanan yang akan memungkinkan lalu lintas masuk pada port 3389, port yang digunakan oleh RDP, sehingga Anda dapat terhubung ke instans EC2. Untuk informasi tentang cara menggunakan Toolkit untuk membuat grup keamanan, lihat Mengelola Grup Keamanan dari AWS Explorer.

Profil Instance

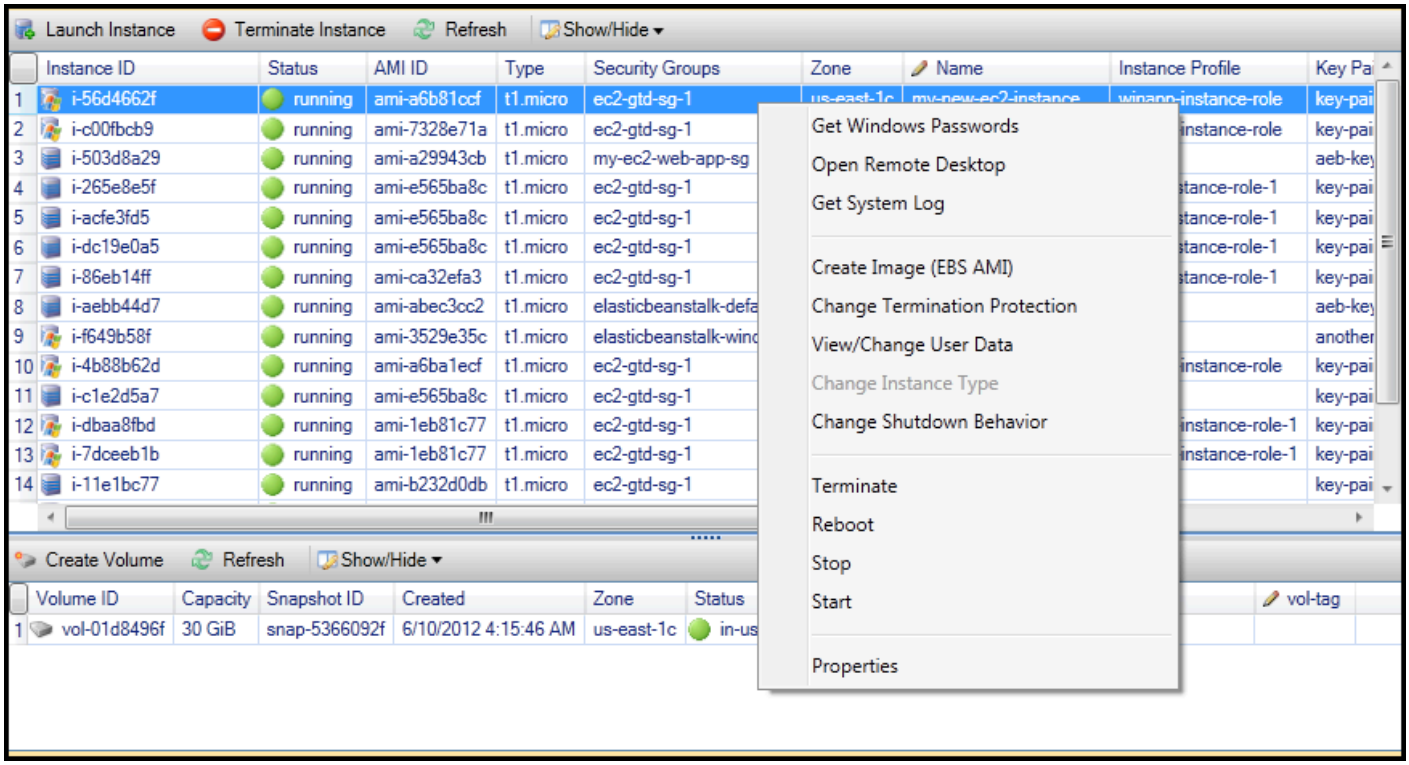
Profil instance adalah wadah logis untuk peran IAM. Saat memilih profil instans, Anda mengaitkan peran IAM yang sesuai dengan instans EC2. Peran IAM dikonfigurasi dengan kebijakan yang menentukan akses ke Amazon Web Services dan sumber daya akun. Ketika instans EC2 dikaitkan dengan peran IAM, perangkat lunak aplikasi yang berjalan pada instance berjalan dengan izin yang ditentukan oleh peran IAM. Ini memungkinkan perangkat lunak aplikasi berjalan tanpa harus menentukan AWS kredensialnya sendiri, yang membuat perangkat lunak lebih aman. Untuk informasi selengkapnya tentang peran IAM, buka Panduan Pengguna IAM.



EC2 Luncurkan kotak dialog AMI

4. Pilih Luncurkan.

Di AWS Explorer, pada subnode Instances Amazon EC2, buka menu konteks (klik kanan) lalu pilih View. AWS Toolkit menampilkan daftar instans Amazon EC2 yang terkait dengan akun aktif. Anda mungkin perlu memilih Refresh untuk melihat instance baru Anda. Ketika instance pertama kali muncul, mungkin dalam keadaan tertunda, tetapi setelah beberapa saat, ia beralih ke status berjalan.



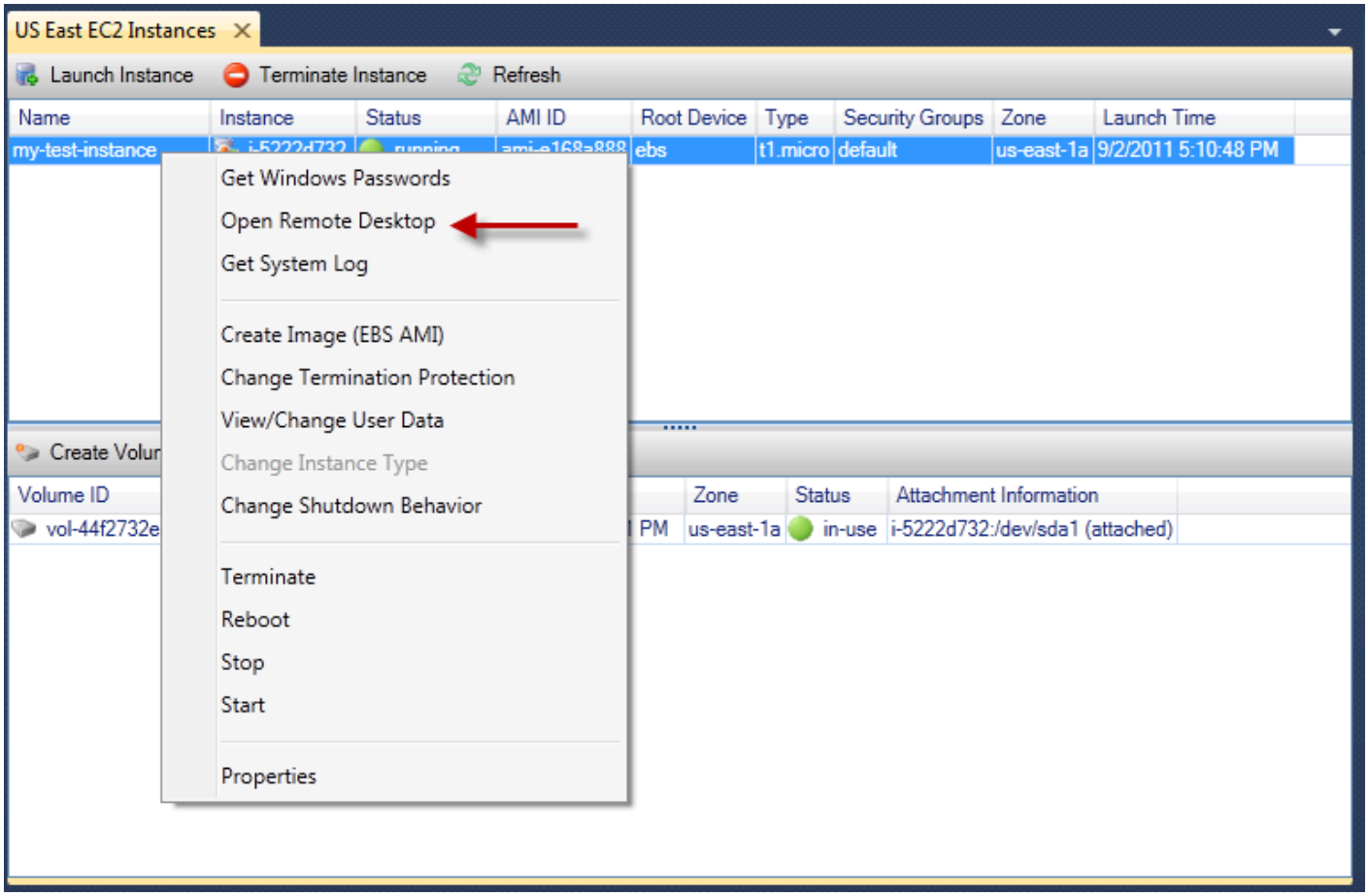
Menyambung ke Instans Amazon EC2

Anda dapat menggunakan Windows Remote Desktop untuk terhubung ke instance Windows Server. Untuk otentikasi, AWS Toolkit memungkinkan Anda untuk mengambil kata sandi administrator untuk instance, atau Anda cukup menggunakan key pair tersimpan yang terkait dengan instance. Dalam prosedur berikut, kita akan menggunakan storage key pair.

Untuk terhubung ke instance Windows Server menggunakan Windows Remote Desktop

1. Dalam daftar instans EC2, klik kanan instance Windows Server yang ingin Anda sambungkan. Dari menu konteks, pilih Buka Remote Desktop.

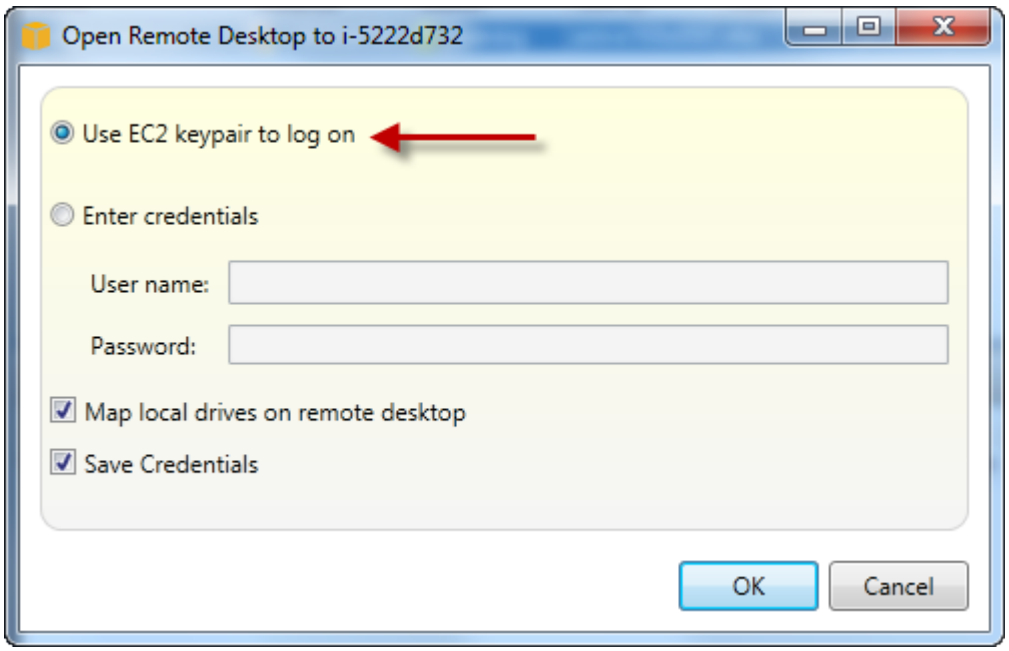
   Jika Anda ingin mengautentikasi menggunakan kata sandi administrator, Anda akan memilih Dapatkan Kata Sandi Windows.



Menu konteks Instans EC2

2. Di kotak dialog Buka Remote Desktop, pilih Gunakan keypair EC2 untuk masuk, lalu pilih OK.

   Jika Anda tidak menyimpan key pair dengan AWS Toolkit, tentukan file PEM yang berisi kunci pribadi.

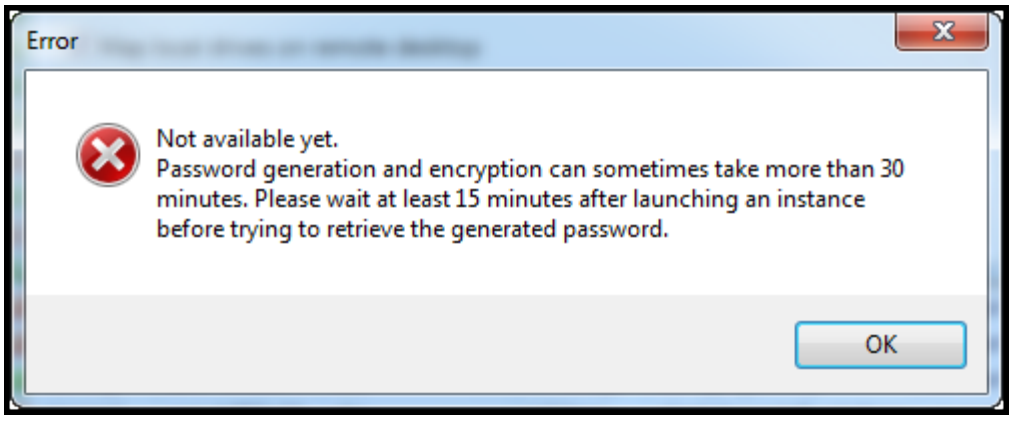


Buka kotak dialog Remote Desktop

3. Jendela Remote Desktop akan terbuka. Anda tidak perlu masuk karena otentikasi terjadi dengan key pair. Anda akan menjalankan sebagai administrator pada instans Amazon EC2.

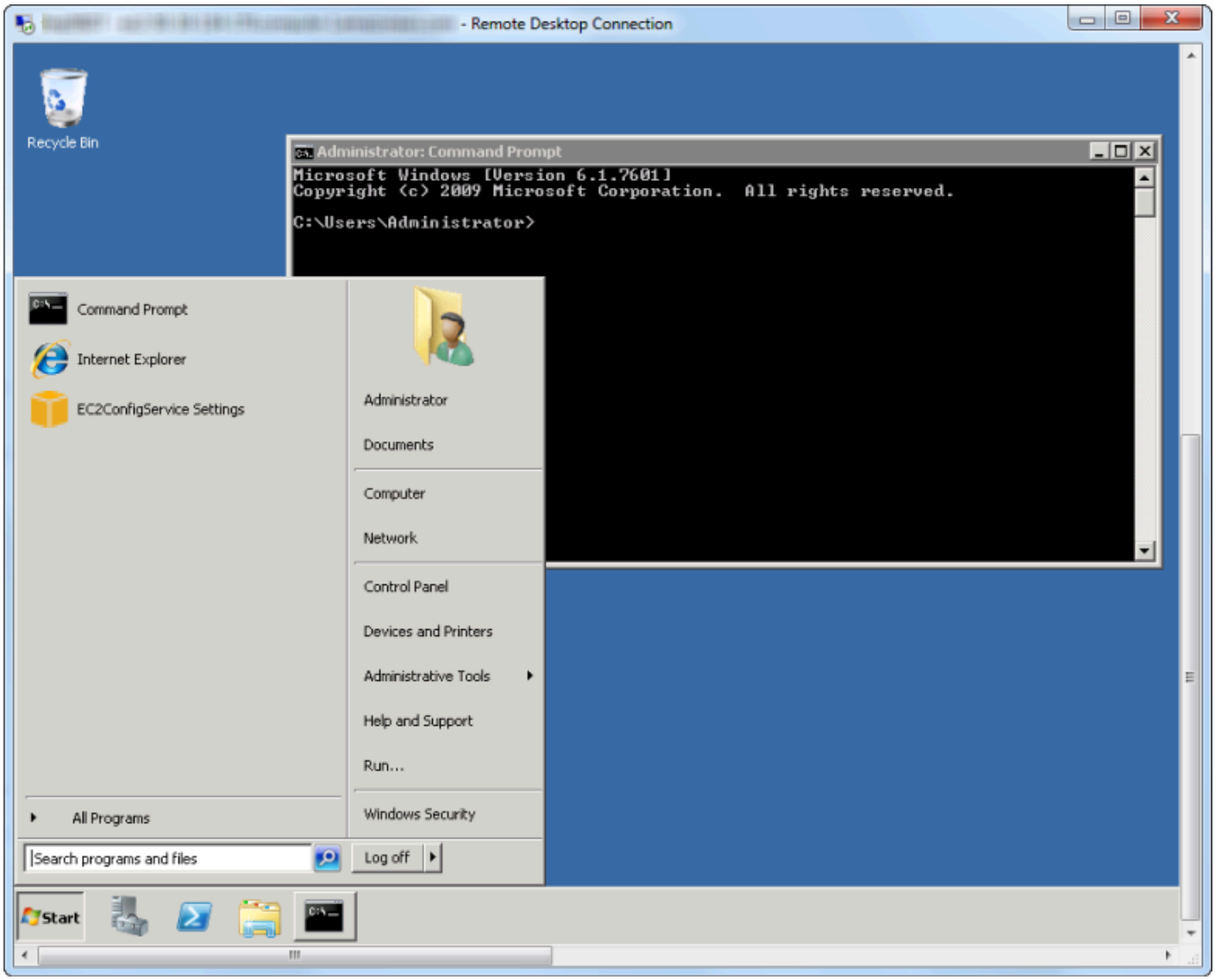
   Jika instans EC2 baru saja dimulai, Anda mungkin tidak dapat terhubung karena dua kemungkinan alasan:

   - Layanan Remote Desktop mungkin belum aktif dan berjalan. Tunggu beberapa menit dan coba lagi.
   - Informasi kata sandi mungkin belum ditransfer ke instance. Dalam hal ini, Anda akan melihat kotak pesan yang mirip dengan yang berikut ini.



Kata sandi belum tersedia

Tangkapan layar berikut menunjukkan pengguna yang terhubung sebagai administrator melalui Remote Desktop.



Desktop Jarak Jauh

Mengakhiri Instans Amazon EC2

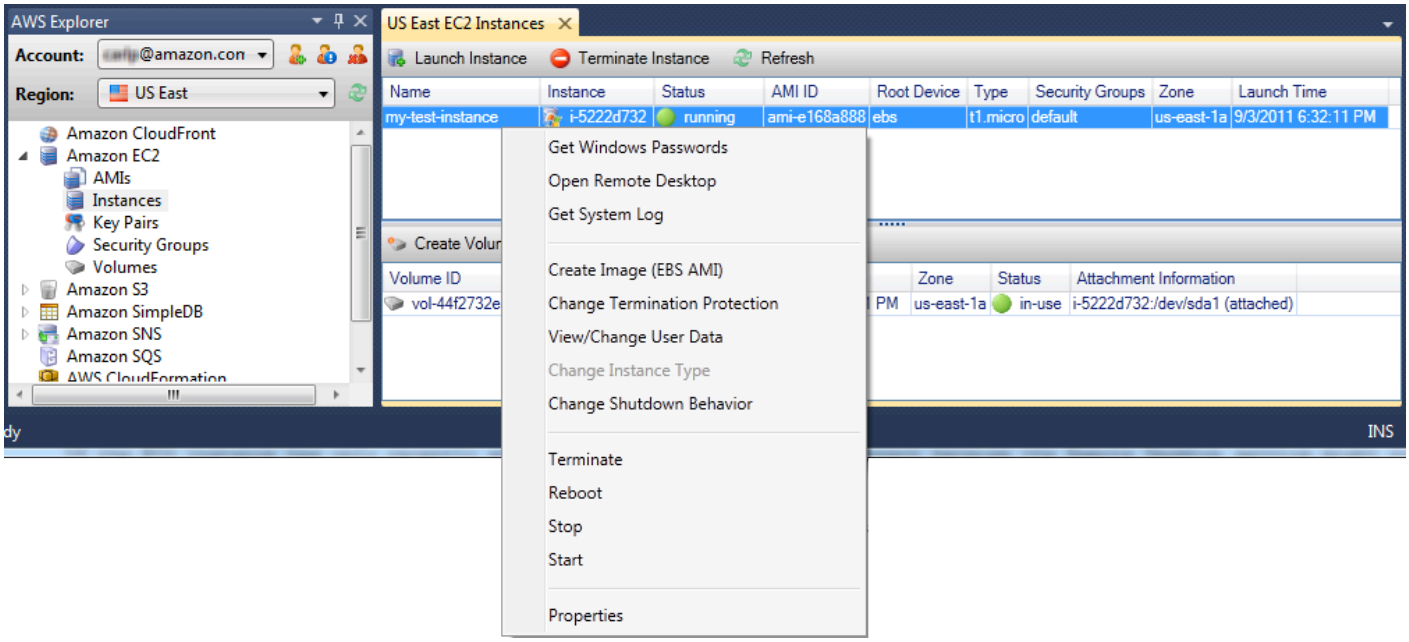
Dengan menggunakan AWS Toolkit, Anda dapat menghentikan atau menghentikan instans Amazon EC2 yang sedang berjalan dari Visual Studio. Untuk menghentikan instans, instans EC2 harus menggunakan volume Amazon EBS. Jika instans EC2 tidak menggunakan volume Amazon EBS, maka satu-satunya pilihan Anda adalah menghentikan instans.

Jika Anda menghentikan instance, data yang disimpan pada volume EBS dipertahankan. Jika Anda menghentikan instance, semua data yang disimpan di perangkat penyimpanan lokal instance akan hilang. Dalam kedua kasus, hentikan atau akhiri, Anda tidak akan terus dikenakan biaya untuk instans EC2. Namun, jika Anda menghentikan instans, Anda akan terus dikenakan biaya untuk penyimpanan EBS yang bertahan setelah instans dihentikan.

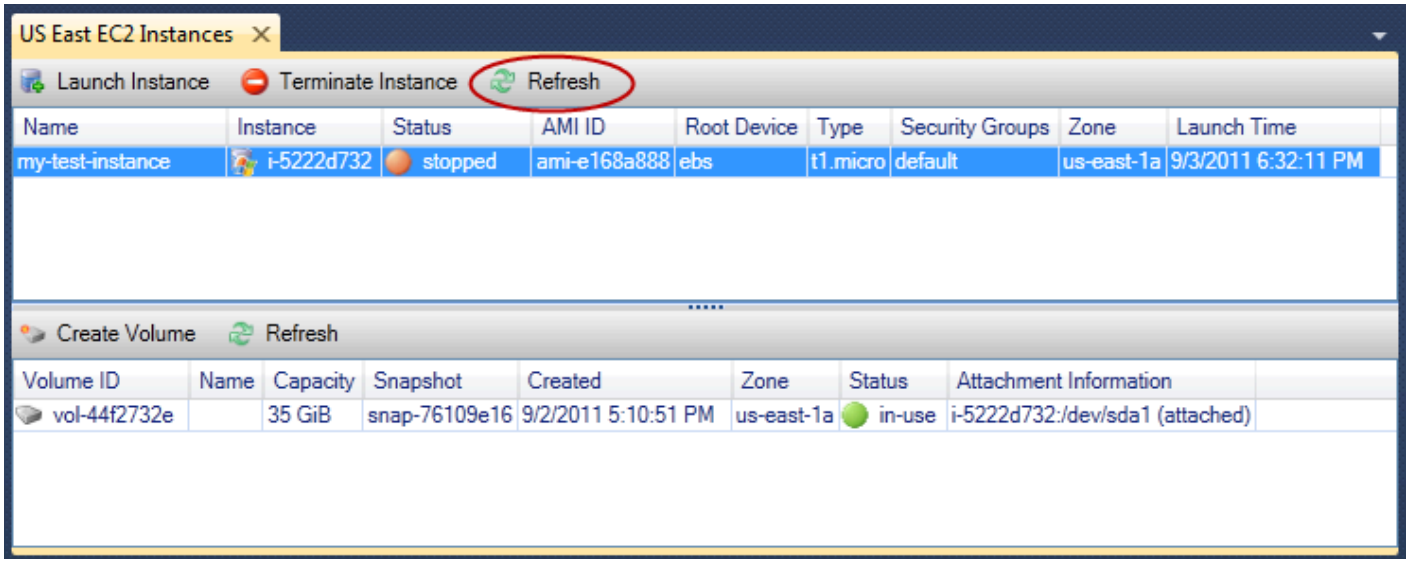
Cara lain yang mungkin untuk mengakhiri instance adalah dengan menggunakan Remote Desktop untuk terhubung ke instance, dan kemudian dari menu Start Windows, gunakan Shutdown. Anda dapat mengonfigurasi instance untuk menghentikan atau mengakhiri dalam skenario ini.

Untuk menghentikan instans Amazon EC2

1. Di AWS Explorer, perluas node Amazon EC2, buka menu konteks (klik kanan) untuk Instans, lalu pilih Lihat. Dalam daftar Instances, klik kanan instance yang ingin Anda hentikan dan pilih Stop dari menu konteks. Pilih Ya untuk mengonfirmasi bahwa Anda ingin menghentikan instance.

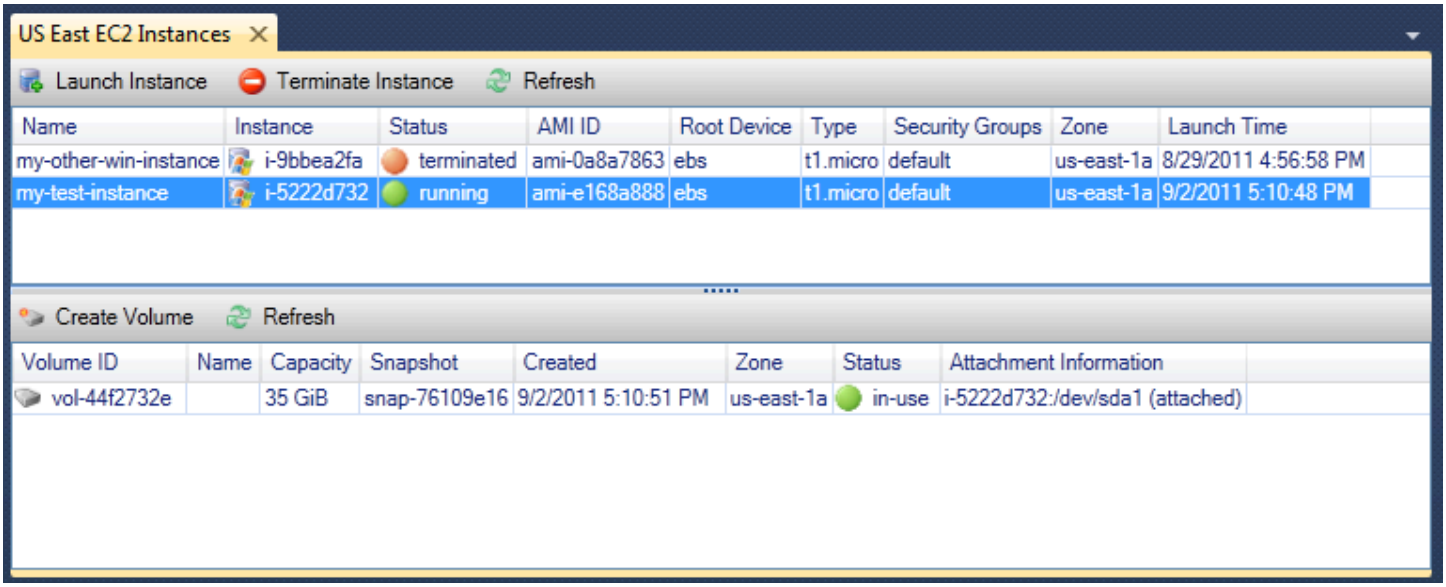


2. Di bagian atas daftar Instans, pilih Segarkan untuk melihat perubahan status instans Amazon EC2. Karena kita berhenti daripada menghentikan instance, volume EBS yang terkait dengan instance masih aktif.



Instans yang Dihentikan Tetap Terlihat

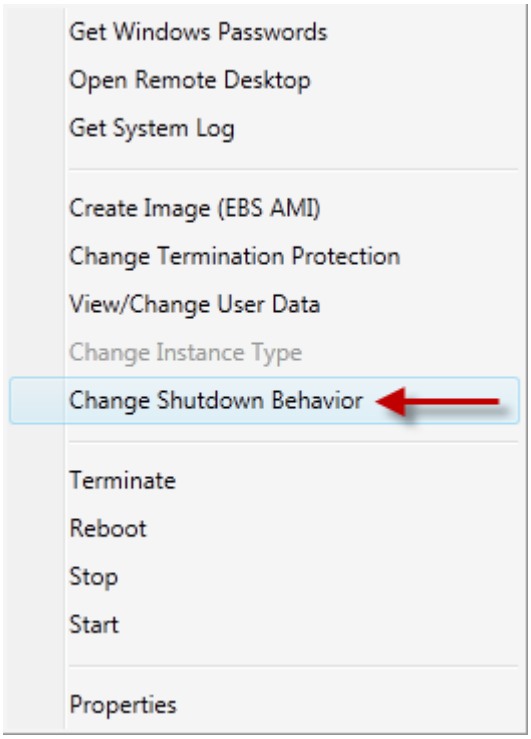
Jika Anda menghentikan sebuah instance, instance akan terus muncul di daftar Instance bersama instance yang sedang berjalan atau dihentikan. Akhirnya, AWS merebut kembali contoh-contoh ini dan mereka menghilang dari daftar. Anda tidak dikenakan biaya untuk instans dalam keadaan dihentikan.



Untuk menentukan perilaku instans EC2 saat shutdown

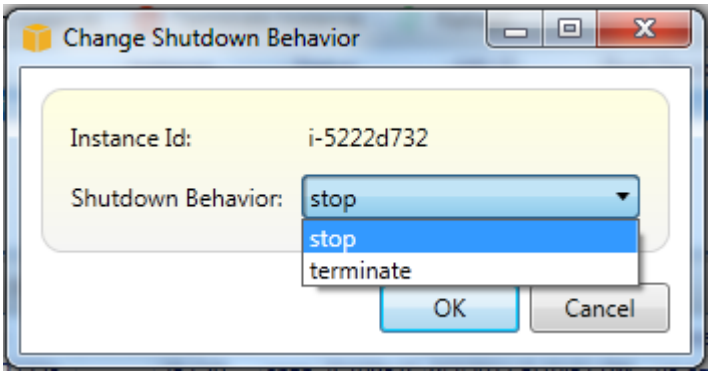
AWS Toolkit memungkinkan Anda menentukan apakah instans Amazon EC2 akan berhenti atau berakhir jika Shutdown dipilih dari menu Start.

1. Dalam daftar Instans, klik kanan instans Amazon EC2, lalu pilih Ubah perilaku shutdown.



Ubah item menu Perilaku Shutdown

2. Dalam kotak dialog Change Shutdown Behavior, dari daftar drop-down Shutdown Behavior, pilih Stop or Terminate.



Mengelola Instans Amazon ECS

AWS Explorer menyediakan tampilan rinci dari Amazon Elastic Container Service (Amazon ECS) cluster dan repositori kontainer. Anda dapat membuat, menghapus, dan mengelola detail cluster dan kontainer dari dalam lingkungan pengembangan Visual Studio.

Memodifikasi properti layanan

Anda dapat melihat detail layanan, peristiwa layanan, dan properti layanan dari tampilan cluster.

1. Di AWS Explorer, buka menu konteks (klik kanan) untuk mengelola cluster, lalu pilih Lihat.
2. Di tampilan ECS Cluster, klik Layanan di sebelah kiri, lalu klik tab Detail di tampilan detail. Anda dapat mengklik Acara untuk melihat pesan acara dan Penerapan ke status penerapan.
3. Klik Edit. Anda dapat mengubah jumlah tugas yang diinginkan dan persentase sehat minimum dan maksimum.
4. Klik Simpan untuk menerima perubahan atau Batalkan untuk kembali ke nilai yang ada.

Menghentikan tugas

Anda dapat melihat status tugas saat ini dan menghentikan satu atau lebih tugas dalam tampilan cluster.

Untuk menghentikan tugas

1. Di AWS Explorer, buka menu konteks (klik kanan) untuk cluster dengan tugas yang ingin Anda hentikan, lalu pilih Lihat.
2. Dalam tampilan ECS Cluster, klik Tugas di sebelah kiri.
3. Pastikan Status Tugas yang Diinginkan diatur ke`Running`. Pilih tugas individual untuk dihentikan dan kemudian klik Berhenti atau klik Hentikan Semua untuk memilih dan menghentikan semua tugas yang sedang berjalan.
4. Di kotak dialog Stop Tasks, pilih Ya.

Menghapus layanan

Anda dapat menghapus layanan dari klaster dari tampilan cluster.

Untuk menghapus layanan klaster

1. Di AWS Explorer, buka menu konteks (klik kanan) untuk cluster dengan layanan yang ingin Anda hapus, lalu pilih Lihat.
2. Pada tampilan ECS Cluster, klik Layanan di sebelah kiri, lalu klik Hapus.

3. Di kotak dialog Delete Cluster, jika ada penyeimbang beban dan grup target di cluster Anda, Anda dapat memilih untuk menghapusnya dengan cluster. Mereka tidak akan digunakan ketika layanan dihapus.
4. Di kotak dialog Hapus Cluster, pilih OK. Ketika cluster dihapus, itu akan dihapus dari AWS Explorer.

Menghapus klaster

Anda dapat menghapus kluster Amazon Elastic Container Service dari AWS Explorer.

Untuk menghapus klaster

1. Di AWS Explorer, buka menu konteks (klik kanan) untuk cluster yang ingin Anda hapus di bawah simpul Clusters Amazon ECS, lalu pilih Hapus.
2. Di kotak dialog Hapus Cluster, pilih OK. Ketika cluster dihapus, itu akan dihapus dari AWS Explorer.

Membuat repositori

Anda dapat membuat repositori Amazon Elastic Container Registry dari AWS Explorer.

Untuk membuat repositori

1. Di AWS Explorer, buka menu konteks (klik kanan) dari node Repositori di bawah Amazon ECS, lalu pilih Create Repository.
2. Dalam kotak dialog Create Repository, berikan nama repositori dan kemudian pilih OK.

Menghapus repositori

Anda dapat menghapus repositori Amazon Elastic Container Registry dari AWS Explorer.

Untuk menghapus repositori

1. Di AWS Explorer, buka menu konteks (klik kanan) dari node Repositori di bawah Amazon ECS, lalu pilih Delete Repository.
2. Di kotak dialog Hapus Repositori, Anda dapat memilih untuk menghapus repositori meskipun berisi gambar. Jika tidak, itu hanya akan dihapus jika kosong. Klik Ya.

Mengelola Grup Keamanan dari AWS Explorer

Toolkit for Visual Studio memungkinkan Anda membuat dan mengonfigurasi grup keamanan yang akan digunakan dengan instans Amazon Elastic Compute Cloud (Amazon EC2) dan. CloudFormation Saat meluncurkan instans Amazon EC2 atau menerapkan aplikasi ke CloudFormation, Anda menentukan grup keamanan yang akan dikaitkan dengan instans Amazon EC2. (Penerapan untuk CloudFormation membuat instans Amazon EC2.)

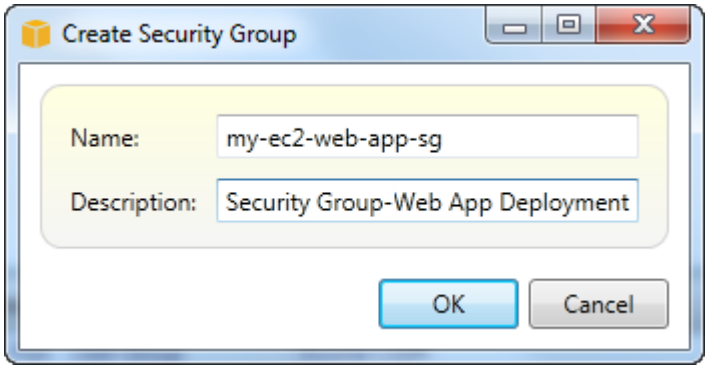
Grup keamanan bertindak seperti firewall pada lalu lintas jaringan yang masuk. Grup keamanan menentukan jenis lalu lintas jaringan yang diizinkan pada instans Amazon EC2. Hal ini juga dapat menentukan bahwa lalu lintas masuk akan diterima dari alamat IP tertentu saja atau dari pengguna tertentu atau kelompok keamanan lainnya saja.

Membuat Grup Keamanan

Di bagian ini, kita akan membuat grup keamanan. Setelah dibuat, grup keamanan tidak akan memiliki izin yang dikonfigurasi. Mengkonfigurasi izin ditangani melalui operasi tambahan.

Untuk membuat grup keamanan

1. Di AWS Explorer, di bawah node Amazon EC2, buka menu konteks (klik kanan) pada node Grup Keamanan, lalu pilih Lihat.
2. Pada tab EC2 Security Groups, pilih Create Security Group.
3. Di kotak dialog Buat Grup Keamanan, ketik nama dan deskripsi untuk grup keamanan, lalu pilih OK.

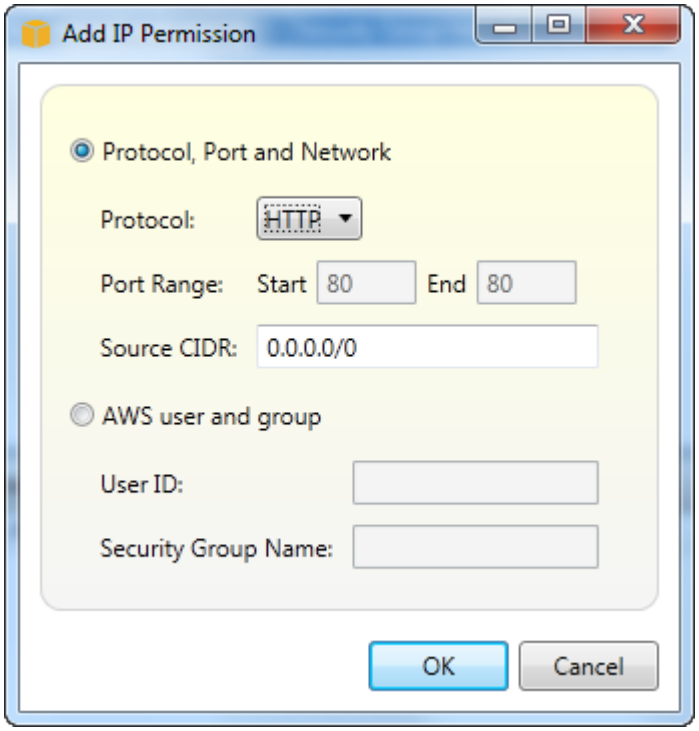


Menambahkan Izin ke Grup Keamanan

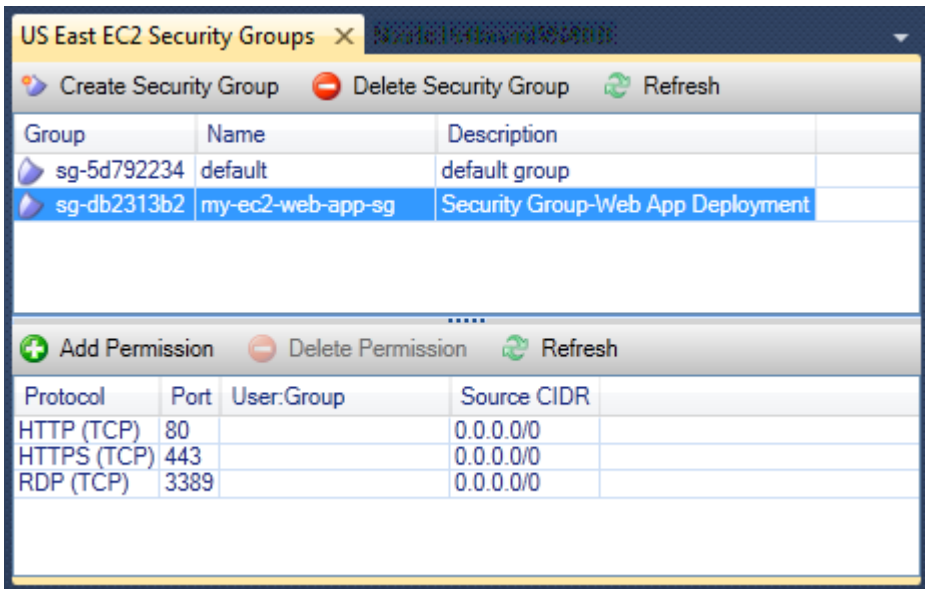
Di bagian ini, kita akan menambahkan izin ke grup keamanan untuk memungkinkan lalu lintas web melalui protokol HTTP dan HTTPS. Kami juga akan mengizinkan komputer lain untuk terhubung dengan menggunakan Windows Remote Desktop Protocol (RDP).

Untuk menambahkan izin ke grup keamanan

1. Pada tab EC2 Security Groups, pilih grup keamanan dan kemudian pilih tombol Add Permission.
2. Dalam kotak dialog Tambahkan Izin IP, pilih Protokol, Port dan Radio Jaringan tombol, dan kemudian dari Protokol daftar drop-down, pilih HTTP. Rentang port secara otomatis menyesuaikan ke port 80, port default untuk HTTP. Bidang CIDR Sumber default ke 0.0.0.0/0, yang menentukan bahwa lalu lintas jaringan HTTP akan diterima dari alamat IP eksternal apa pun. Pilih OK.



   Buka port 80 (HTTP) untuk grup keamanan ini
3. Ulangi proses ini untuk HTTPS dan RDP. Izin grup keamanan Anda sekarang akan terlihat seperti berikut ini.



Anda juga dapat mengatur izin di grup keamanan dengan menentukan ID pengguna dan nama grup keamanan. Dalam hal ini, instans Amazon EC2 dalam grup keamanan ini akan menerima semua lalu lintas jaringan masuk dari instans Amazon EC2 dalam grup keamanan yang ditentukan. Anda juga harus menentukan ID pengguna sebagai cara untuk membedakan nama grup keamanan; nama grup keamanan tidak harus unik di semua. AWS Untuk informasi lebih lanjut tentang grup keamanan, buka dokumentasi EC2.

Membuat AMI dari Instans Amazon EC2

Anda dapat membuat Amazon Machine Image (AMI) dengan file AWS Toolkit for Visual Studio. Untuk informasi selengkapnya AMIs, lihat topik Amazon Machine Images (AMI) di Amazon Elastic Compute Cloud for Windows Instances User Guide.

Untuk membuat AMI dari instans Amazon EC2 yang keluar, selesaikan prosedur berikut.

Membuat AMI dari instans Amazon EC2 yang ada

1. Dari AWS Toolkit Explorer, perluas Amazon EC2 dan pilih Instans untuk melihat daftar instans yang ada.
2. Klik kanan instance yang ingin Anda gunakan sebagai dasar untuk AMI Anda dan pilih Create Image (ABS AMI) untuk membuka jendela dialog Create Image.
3. Dari jendela dialog Buat Gambar, tambahkan nama dan deskripsi untuk gambar Anda ke dalam bidang yang disediakan, lalu pilih OK tombol untuk melanjutkan.

4. Jendela konfirmasi Gambar Dibuat terbuka di Visual Studio saat gambar dibuat, pilih tombol OK untuk melanjutkan.

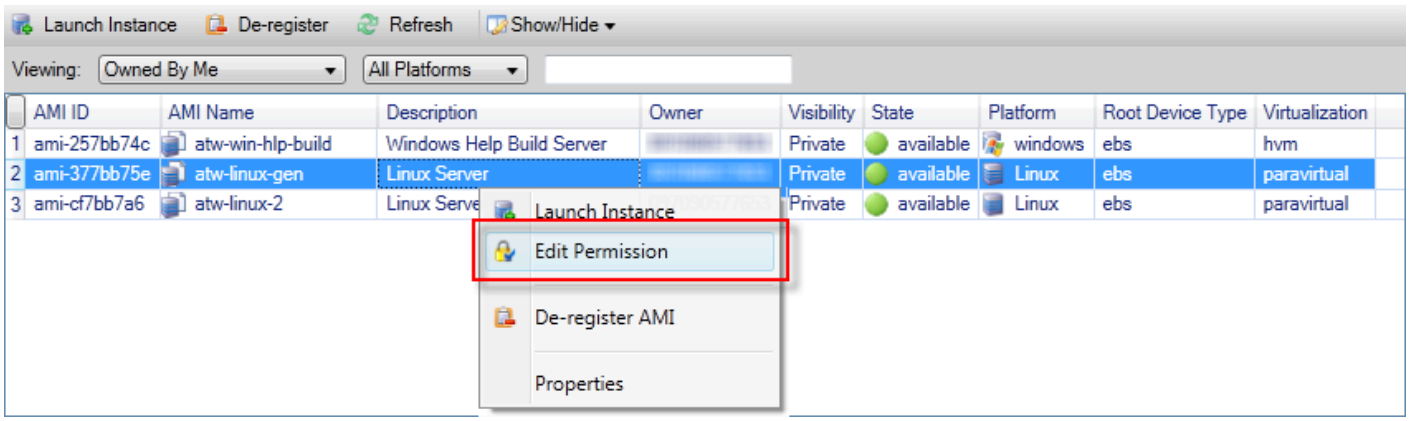
Untuk melihat AMI baru Anda dengan AWS Toolkit, perluas Amazon EC2 dan AMIsklik dua kali untuk membuka jendela di payne Visual Studio Editor yang menampilkan daftar yang sudah ada. AMIs Jika Anda tidak melihat AMI baru dalam daftar, pilih tombol Refresh yang terletak di bagian atas jendela AMI.

Mengatur Izin Peluncuran pada Gambar Mesin Amazon

Anda dapat mengatur izin peluncuran di Amazon Machine Images (AMIs) dari AMIstampilan di AWS Explorer. Anda dapat menggunakan kotak dialog Set AMI Permissions untuk menyalin izin dari. AMIs

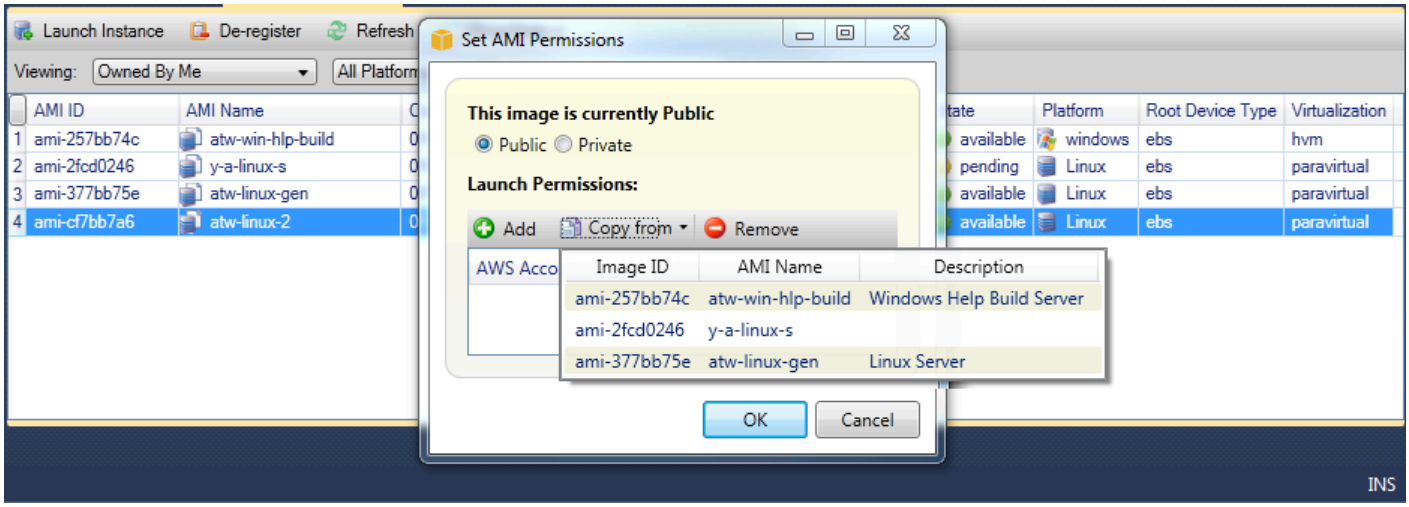
Untuk mengatur izin pada AMI

1. Dalam AMIstampilan di AWS Explorer, buka menu konteks (klik kanan) pada AMI, lalu pilih Edit Izin.



2. Ada tiga opsi yang tersedia di kotak dialog Set AMI Permissions:
   - Untuk memberikan izin peluncuran, pilih Tambah, dan ketik nomor akun untuk AWS pengguna yang Anda beri izin peluncuran.
   - Untuk menghapus izin peluncuran, pilih nomor akun untuk AWS pengguna dari siapa Anda menghapus izin peluncuran, dan pilih Hapus.
   - Untuk menyalin izin dari satu AMI ke AMI lainnya, pilih AMI dari daftar, dan pilih Salin dari. Pengguna yang memiliki izin peluncuran pada AMI yang Anda pilih akan diberikan izin peluncuran pada AMI saat ini. Anda dapat mengulangi proses ini dengan yang lain AMIs dalam daftar Salin-dari untuk menyalin izin dari beberapa AMIs ke AMI target.

Daftar Copy-from hanya berisi yang AMIs dimiliki oleh akun yang aktif saat AMIstampilan ditampilkan dari AWS Explorer. Akibatnya, daftar Copy-from mungkin tidak menampilkan apapun AMIs jika tidak ada AMIs yang dimiliki oleh akun aktif.



Salin kotak dialog izin AMI

Amazon Virtual Cloud Private Cloud (VPC)

Amazon Virtual Private Cloud (Amazon VPC) memungkinkan Anda meluncurkan sumber daya Amazon Web Services ke jaringan virtual yang telah Anda tentukan. Jaringan virtual ini menyerupai jaringan tradisional yang akan Anda operasikan di pusat data Anda sendiri, dengan manfaat menggunakan infrastruktur yang dapat diskalakan. AWS Untuk informasi lebih lanjut, buka Panduan Pengguna Amazon VPC.

Toolkit for Visual Studio memungkinkan pengembang untuk mengakses fungsionalitas VPC yang mirip dengan yang diekspos oleh Konsol Manajemen AWStetapi dari lingkungan pengembangan Visual Studio. Node Amazon VPC AWS Explorer menyertakan subnode untuk area berikut.

- VPCs
- Subnet
- Elastis IPs
- Gerbang Internet
- Jaringan ACLs
- Tabel Rute

- Grup Keamanan

Membuat VPC Publik-Pribadi untuk Deployment dengan AWS Elastic Beanstalk

Bagian ini menjelaskan cara membuat VPC Amazon yang berisi subnet publik dan pribadi. Subnet publik berisi instans Amazon EC2 yang melakukan terjemahan alamat jaringan (NAT) untuk mengaktifkan instance di subnet pribadi untuk berkomunikasi dengan internet publik. Kedua subnet harus berada di Availability Zone (AZ) yang sama.

Ini adalah konfigurasi VPC minimal yang diperlukan untuk menyebarkan AWS Elastic Beanstalk lingkungan di VPC. Dalam skenario ini, instans Amazon EC2 yang meng-host aplikasi Anda berada di subnet pribadi; penyeimbang beban Elastic Load Balancing yang merutekan lalu lintas masuk ke aplikasi Anda berada di subnet publik.

Untuk informasi selengkapnya tentang terjemahan alamat jaringan (NAT), buka Instans NAT di Panduan Pengguna Amazon Virtual Private Cloud. Untuk contoh cara mengonfigurasi penerapan Anda untuk menggunakan VPC, lihat Menerapkan ke Elastic Beanstalk.

Untuk membuat VPC subnet publik-pribadi

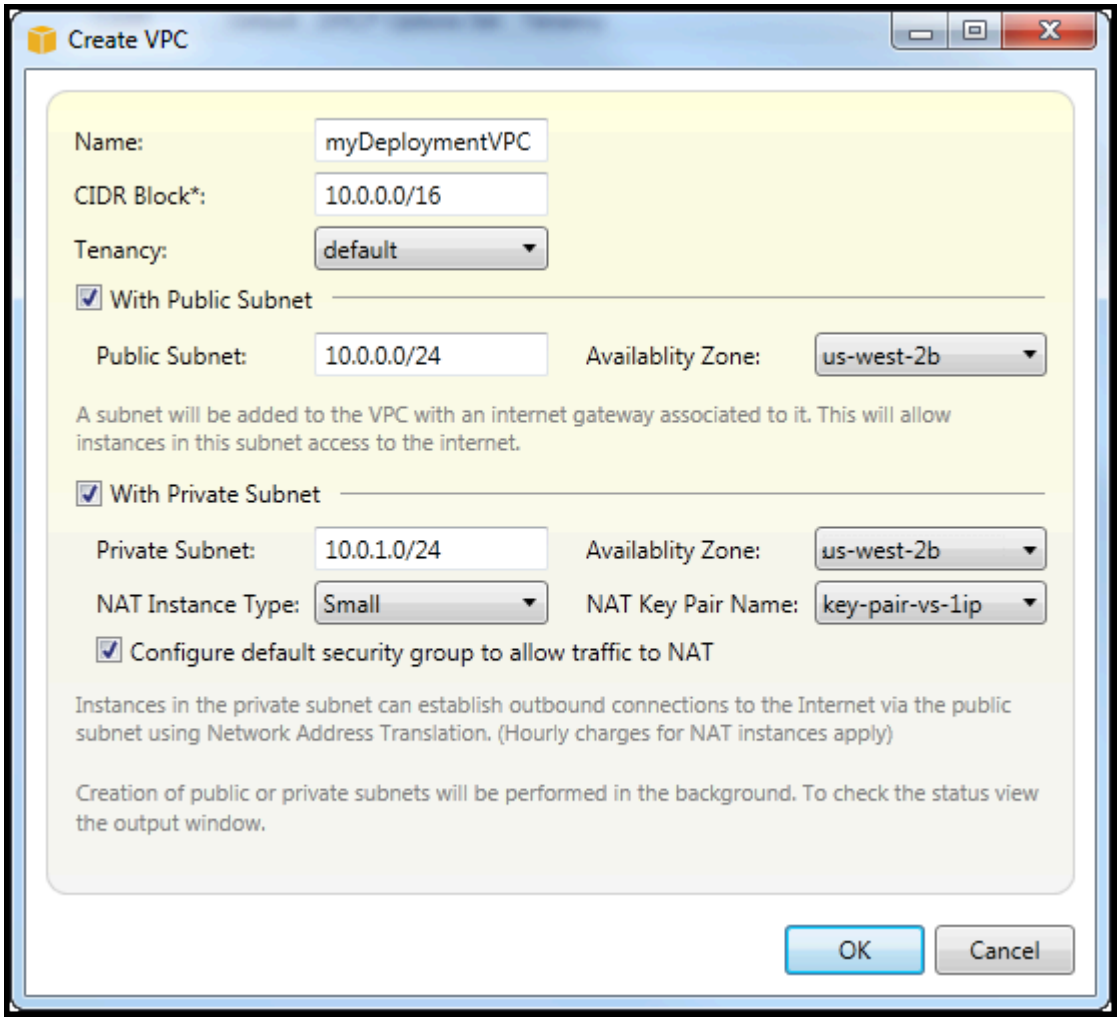
1. Di node VPC Amazon di AWS Explorer, buka VPCssubnode, lalu pilih Buat VPC.

2. Konfigurasi VPC sebagai berikut:
   - Ketik nama untuk VPC Anda.
   - Pilih kotak centang With Public Subnet dan With Private Subnet.

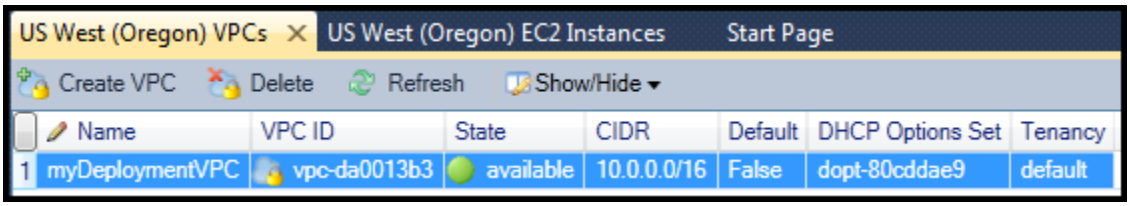
- Dari kotak daftar drop-down Availability Zone untuk setiap subnet, pilih Availability Zone. Pastikan untuk menggunakan AZ yang sama untuk kedua subnet.
- Untuk subnet pribadi, di NAT Key Pair Name, berikan key pair. Key pair ini digunakan untuk instans Amazon EC2 yang melakukan terjemahan alamat jaringan dari subnet pribadi ke Internet publik.
- Pilih kotak centang Konfigurasikan grup keamanan default untuk mengizinkan lalu lintas ke NAT.

Ketik nama untuk VPC Anda. Pilih kotak centang With Public Subnet dan With Private Subnet. Dari kotak daftar drop-down Availability Zone untuk setiap subnet, pilih Availability Zone. Pastikan untuk menggunakan AZ yang sama untuk kedua subnet. Untuk subnet pribadi, di NAT Key Pair Name, berikan key pair. Key pair ini digunakan untuk instans Amazon EC2 yang melakukan terjemahan alamat jaringan dari subnet pribadi ke Internet publik. Pilih kotak centang Konfigurasikan grup keamanan default untuk mengizinkan lalu lintas ke NAT.

Pilih OK.

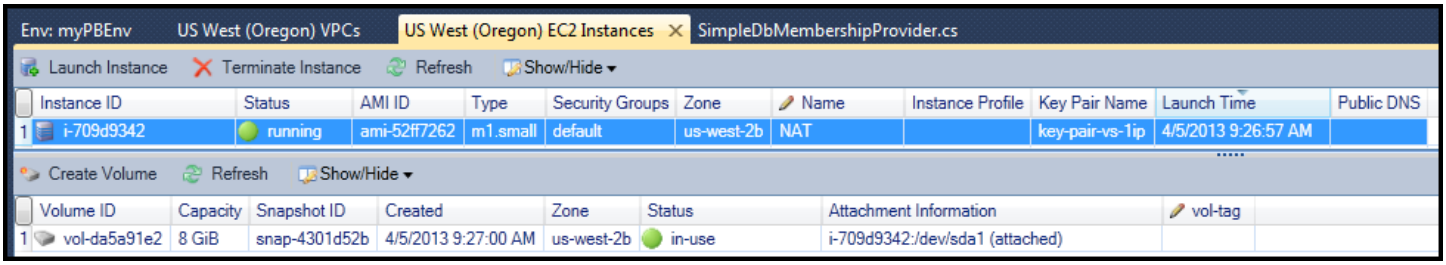


Anda dapat melihat VPC baru di VPCstab di AWS Explorer.



Instans NAT mungkin membutuhkan waktu beberapa menit untuk diluncurkan. Ketika tersedia, Anda dapat melihatnya dengan memperluas node Amazon EC2 di AWS Explorer dan kemudian membuka subnode Instances.

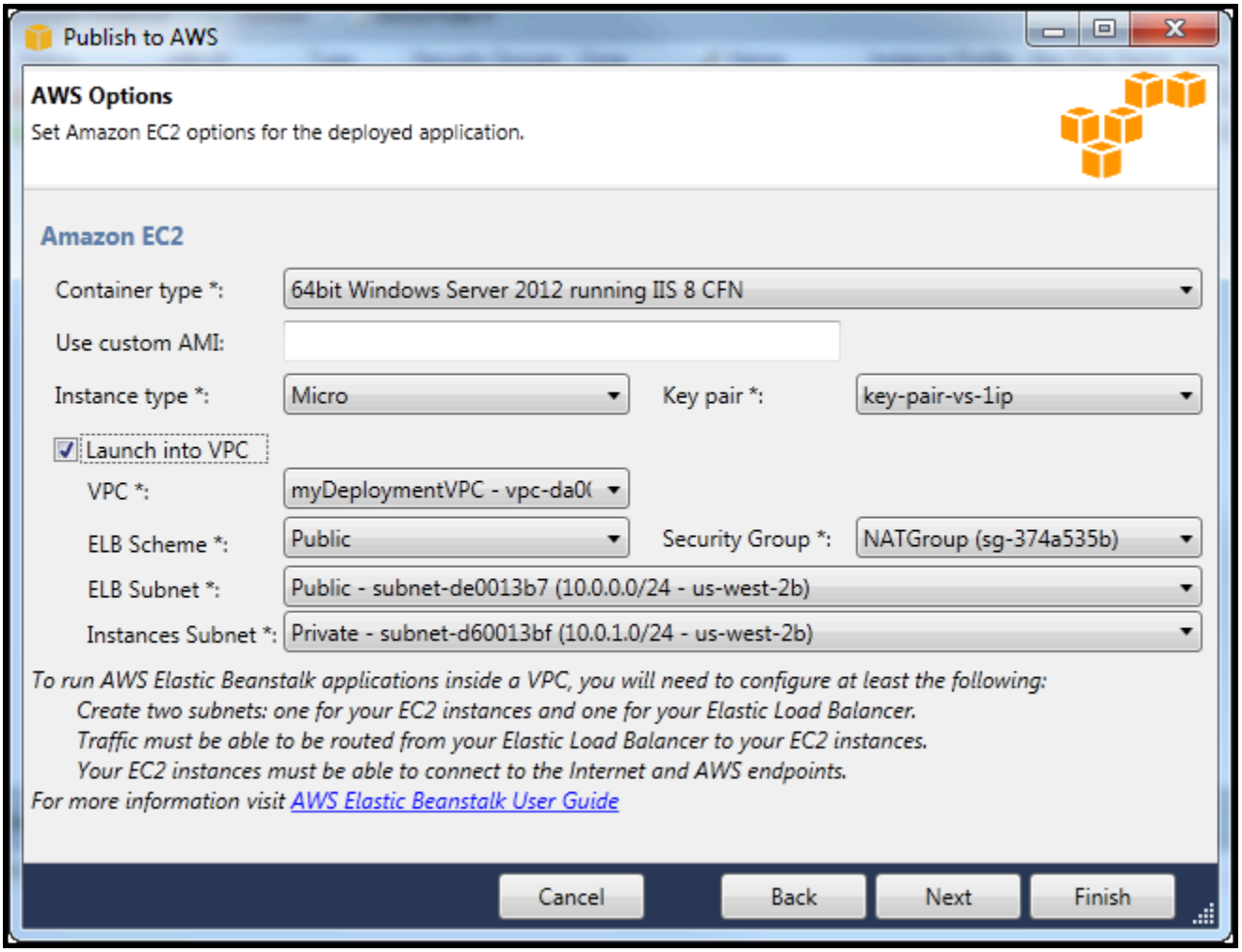
Volume Amazon Elastic Block Store (Amazon EBS) dibuat untuk instans NAT secara otomatis. Untuk informasi selengkapnya tentang Amazon EBS, lihat topik Amazon Elastic Block Store (EBS) di Panduan Pengguna Amazon EC2 untuk Instans Linux.



Jika Anda menyebarkan aplikasi ke AWS Elastic Beanstalk lingkungan dan memilih untuk meluncurkan lingkungan di VPC, Toolkit akan mengisi kotak dialog Publish Amazon Web Services to dengan informasi konfigurasi untuk VPC Anda.

Toolkit mengisi kotak dialog dengan informasi hanya dari VPCs yang dibuat di Toolkit, bukan dari VPCs dibuat menggunakan file. Konsol Manajemen AWS Ini karena ketika Toolkit membuat VPC, ia menandai komponen VPC sehingga dapat mengakses informasinya.

Screenshot berikut dari Deployment Wizard menunjukkan contoh kotak dialog yang diisi dengan nilai-nilai dari VPC yang dibuat di Toolkit.



Untuk menghapus VPC

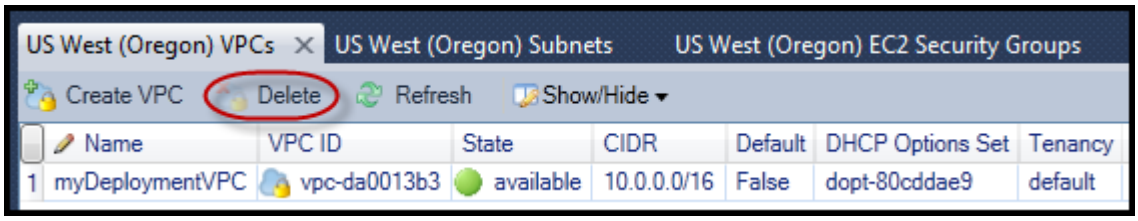
Untuk menghapus VPC, Anda harus terlebih dahulu menghentikan instans Amazon EC2 apa pun di VPC.

1. Jika Anda telah menerapkan aplikasi ke AWS Elastic Beanstalk lingkungan di VPC, hapus lingkungan. Ini akan menghentikan instans Amazon EC2 yang menghosting aplikasi Anda bersama dengan penyeimbang beban Elastic Load Balancing.

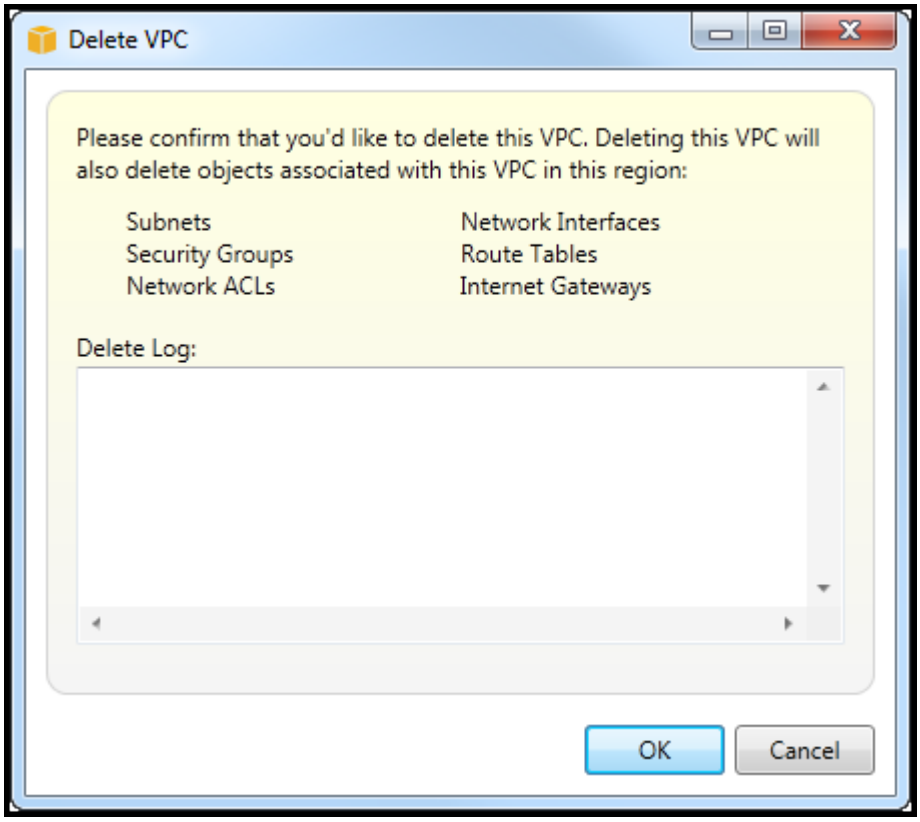
   Jika Anda mencoba menghentikan instans yang menghosting aplikasi secara langsung tanpa menghapus lingkungan, layanan Auto Scaling akan secara otomatis membuat instance baru untuk menggantikan instans yang dihapus. Untuk informasi selengkapnya, buka Panduan Pengembang Auto Scaling.
2. Hapus instance NAT untuk VPC.

Anda tidak perlu menghapus volume Amazon EBS yang terkait dengan instans NAT untuk menghapus VPC. Namun, jika Anda tidak menghapus volume, Anda akan terus dikenakan biaya untuk itu bahkan jika Anda menghapus instans NAT dan VPC.

3. Pada tab VPC, pilih tautan Hapus untuk menghapus VPC.



4. Di kotak dialog Hapus VPC, pilih OK.



Menggunakan Editor CloudFormation Template untuk Visual Studio

Toolkit for Visual Studio mencakup CloudFormation editor template CloudFormation dan proyek template untuk Visual Studio. Fitur yang didukung meliputi:

- Membuat template baru (baik kosong atau disalin dari tumpukan atau contoh template yang ada) menggunakan jenis proyek CloudFormation template yang disediakan.

- Mengedit template dengan validasi JSON otomatis, pelengkapan otomatis, pelipatan kode, dan penyorotan sintaks.
- Saran otomatis fungsi intrinsik dan parameter referensi sumber daya untuk nilai bidang di template Anda.
- Item menu untuk melakukan tindakan umum untuk template Anda dari Visual Studio.

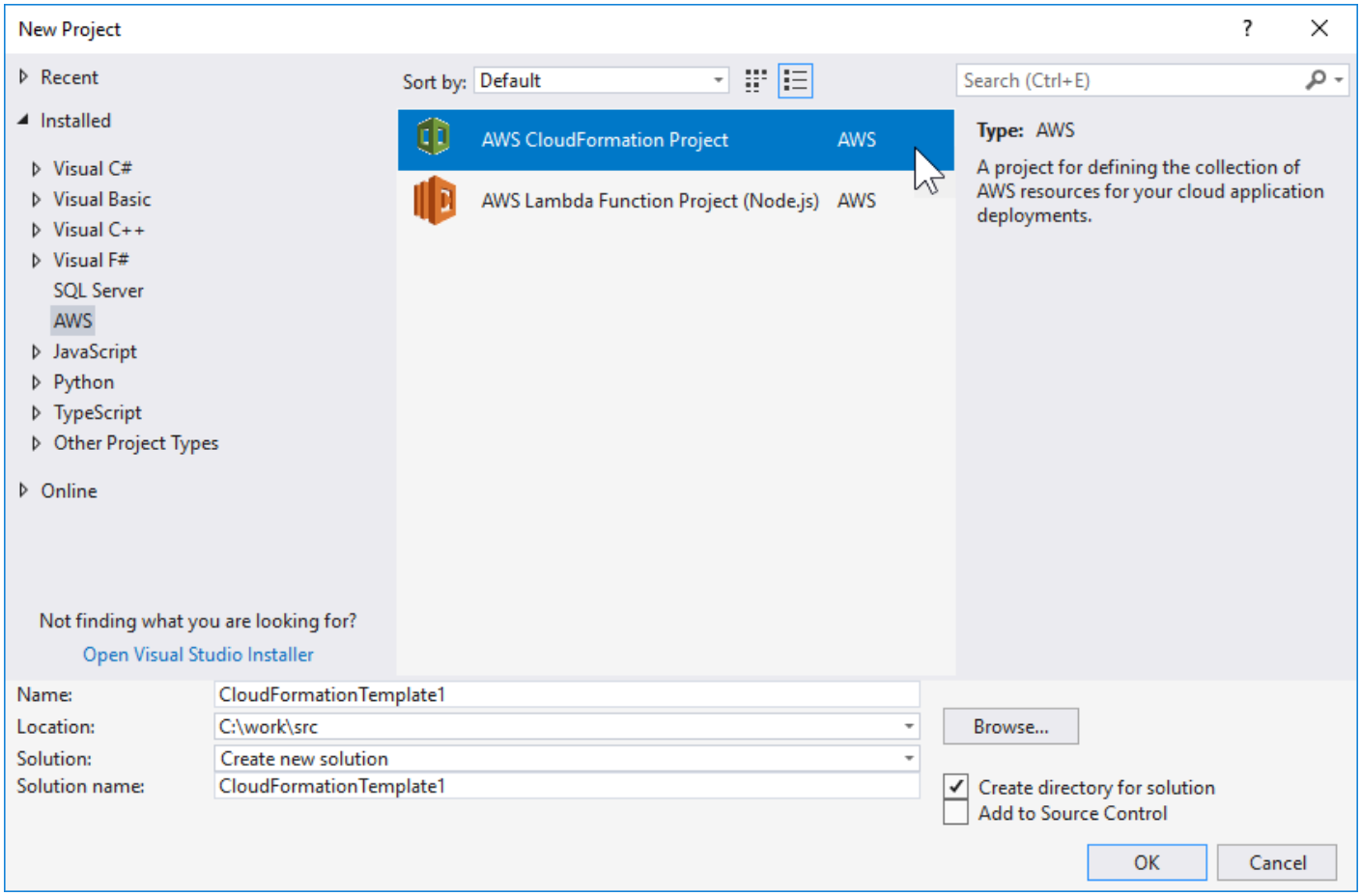
Topik

Membuat Proyek CloudFormation Template di Visual Studio

Untuk membuat proyek template

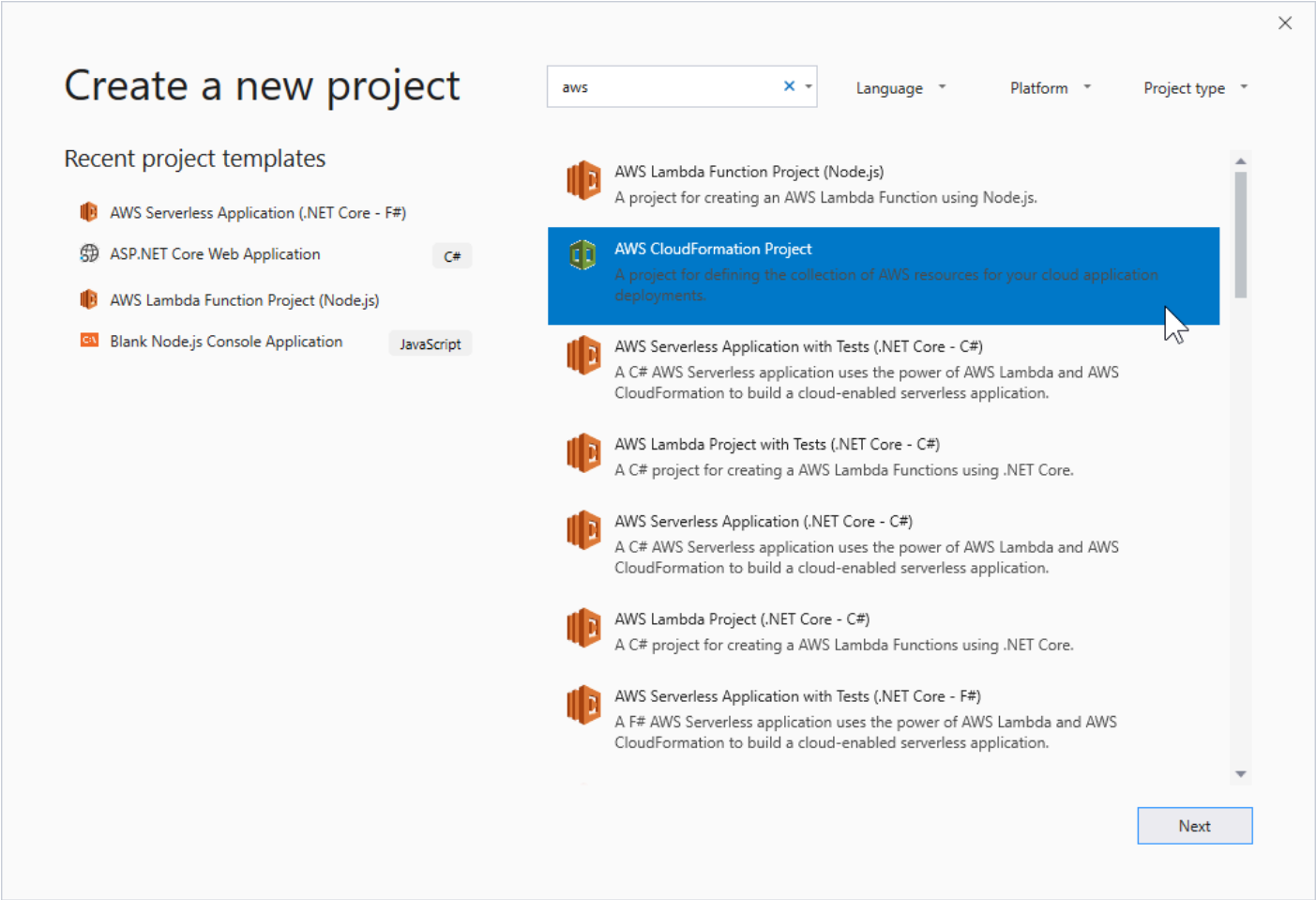
1. Di Visual Studio, pilih File, pilih New, dan kemudian pilih Project.
2. Untuk Visual Studio 2017:

   Di kotak dialog New Project, perluas Installed dan pilih AWS.



Untuk Visual Studio 2019:

Di kotak dialog New Project, pastikan bahwa kotak drop-down Bahasa, Platform, dan tipe Proyek diatur ke “Semua...” dan ketik aws di bidang Pencarian.

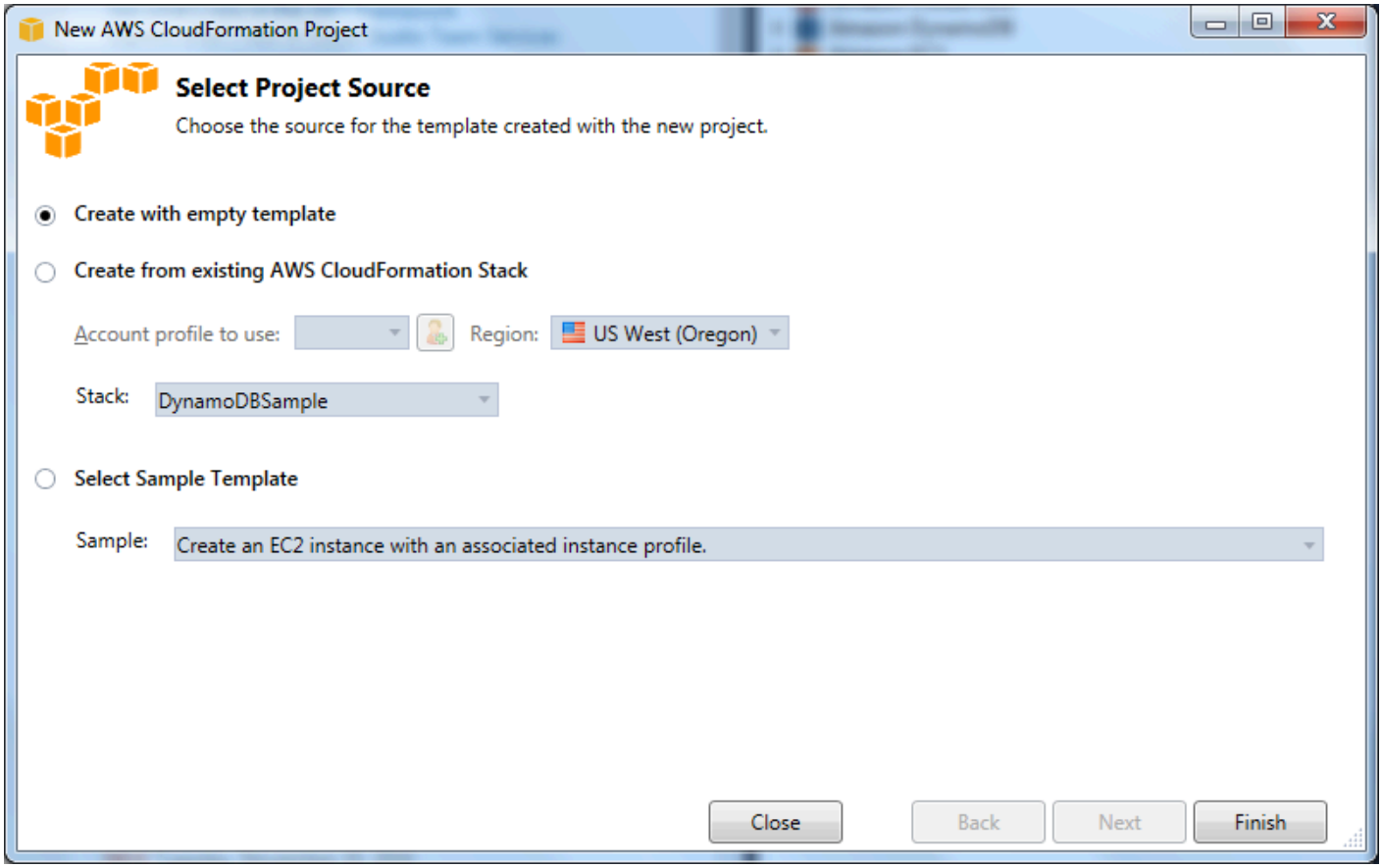


3. Pilih template AWS CloudFormation Project.
4. Untuk Visual Studio 2017:

   Masukkan Nama, Lokasi, dll yang diinginkan, untuk proyek template Anda, lalu klik OK.

   Untuk Visual Studio 2019:

   Klik Berikutnya. Pada dialog berikutnya, masukkan Nama, Lokasi, dll yang diinginkan, untuk proyek template Anda, lalu klik Buat.
5. Pada halaman Select Project Source, pilih sumber template yang akan Anda buat:
   - Buat dengan template kosong menghasilkan template baru yang kosong CloudFormation .
   - Buat dari stack AWS |CFN| yang ada menghasilkan template dari tumpukan yang ada di akun Anda. AWS (Tumpukan tidak perlu memiliki status`CREATE_COMPLETE`.)
   - Pilih contoh template menghasilkan template dari salah satu CloudFormation contoh template.

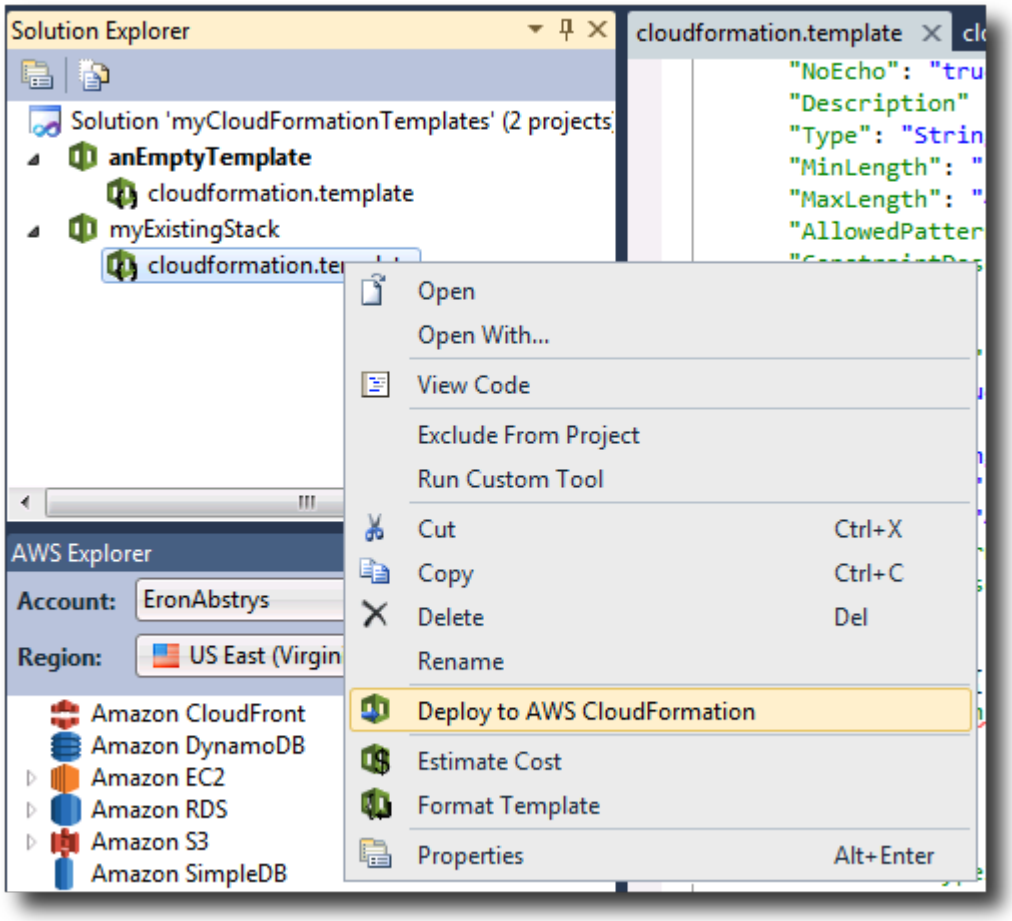


6. Untuk menyelesaikan pembuatan proyek CloudFormation template Anda, pilih Selesai.

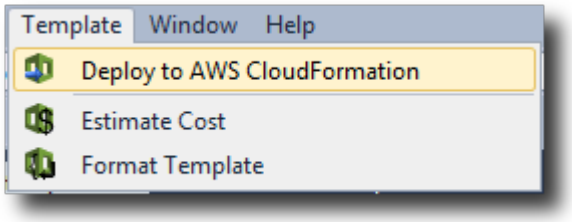
Menerapkan CloudFormation Template di Visual Studio

Untuk menyebarkan template CFN

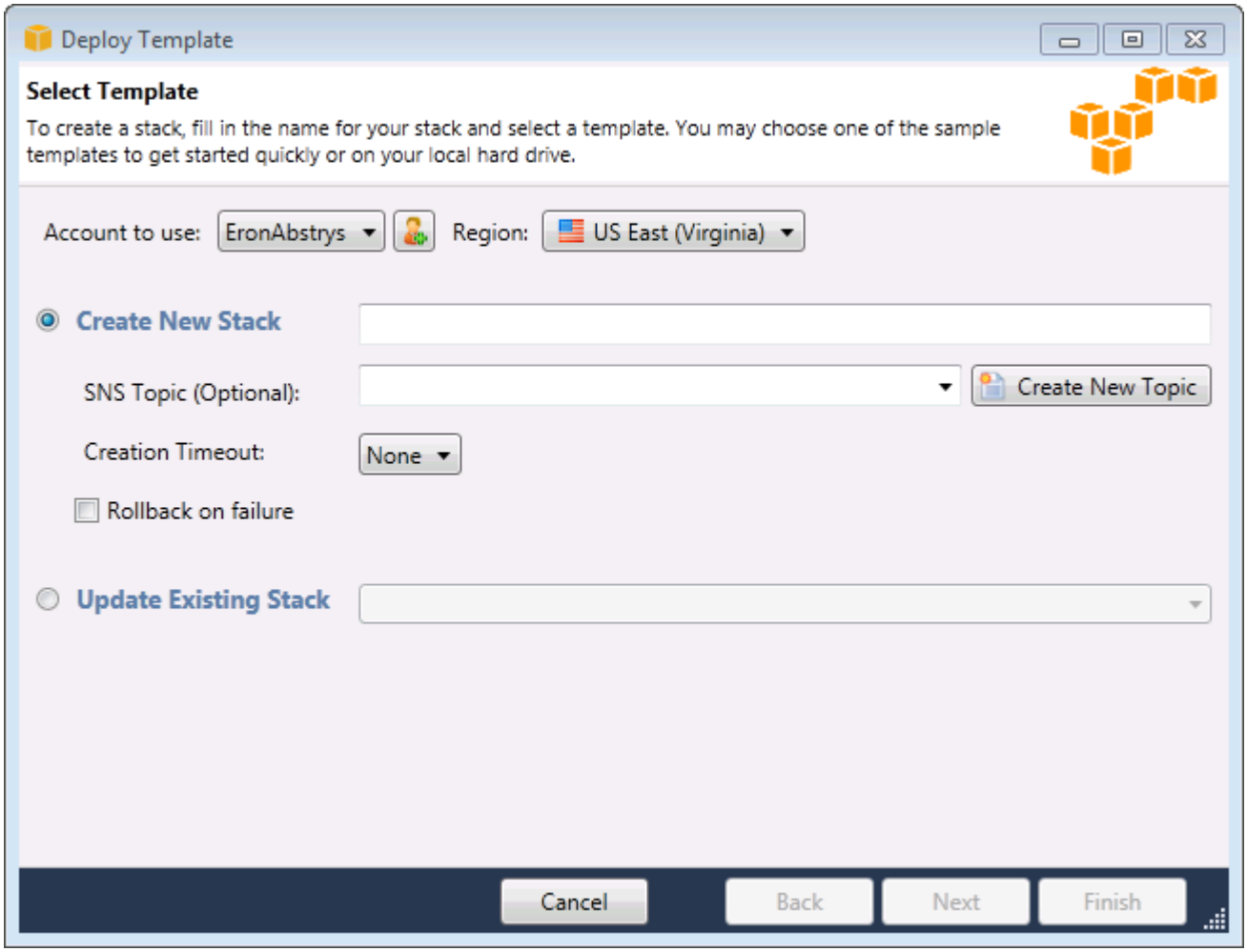
1. Di Solution Explorer, buka menu konteks (klik kanan) untuk template yang ingin Anda gunakan, dan pilih Deploy ke. AWS CloudFormation



Atau, untuk menerapkan template yang sedang Anda edit, dari menu Template, pilih Deploy to. AWS CloudFormation



2. Pada halaman Template Deploy, pilih yang akan digunakan Akun AWS untuk meluncurkan tumpukan dan wilayah di mana ia akan diluncurkan.

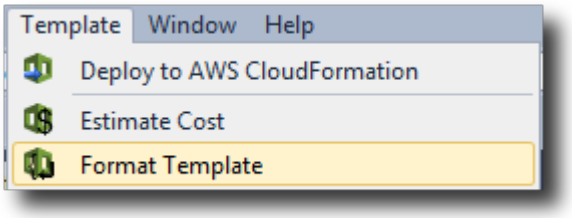


3. Pilih Create New Stack dan ketik nama untuk tumpukan Anda.
4. Pilih salah satu (atau tidak ada) dari opsi berikut:
   - Untuk menerima pemberitahuan tentang kemajuan tumpukan, dari daftar drop-down Topik SNS, pilih topik SNS. Anda juga dapat membuat topik SNS dengan memilih Buat Topik Baru dan mengetik alamat email di kotak.
   - Gunakan Creation Timeout untuk menentukan berapa lama CloudFormation seharusnya tumpukan dibuat sebelum dinyatakan gagal (dan diputar kembali, kecuali opsi Rollback on failure dihapus).
   - Gunakan Rollback pada kegagalan jika Anda ingin tumpukan berputar kembali (yaitu, hapus sendiri) pada kegagalan. Biarkan opsi ini dihapus jika Anda ingin tumpukan tetap aktif untuk tujuan debugging, bahkan jika gagal menyelesaikan peluncuran.
5. Pilih Selesai untuk meluncurkan tumpukan.

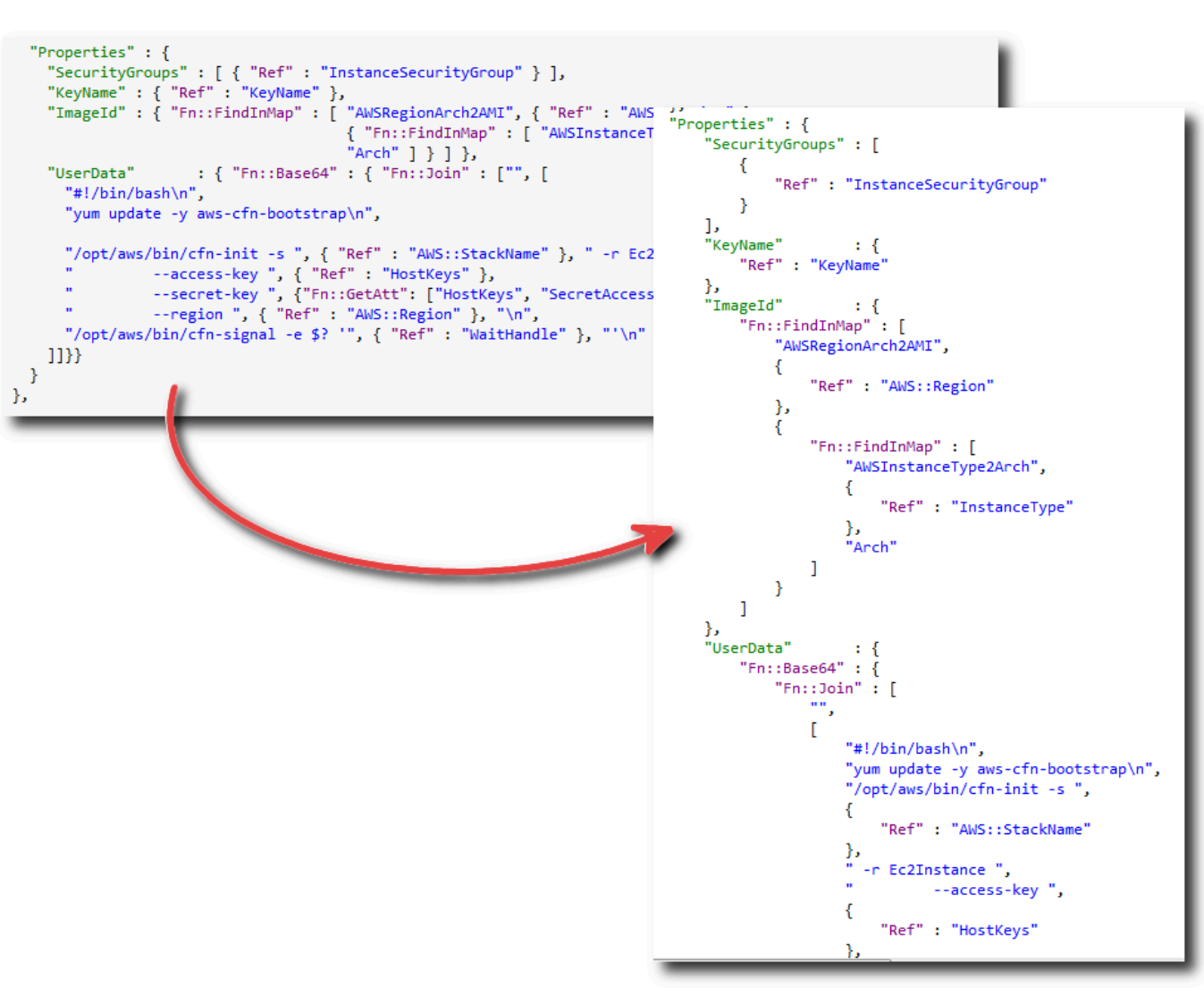
Memformat CloudFormation Template di Visual Studio

- Di Solution Explorer, buka menu konteks (klik kanan) untuk template dan pilih Format Template.

  Atau, untuk memformat template yang sedang Anda edit, dari menu Template, pilih Format Template.



  Kode JSON Anda akan diformat sehingga strukturnya disajikan dengan jelas.



Menggunakan Amazon S3 dari Explorer AWS

Amazon Simple Storage Service (Amazon S3) memungkinkan Anda menyimpan dan mengambil data dari koneksi apa pun ke Internet. Semua data yang Anda simpan di Amazon S3 dikaitkan dengan akun Anda dan, secara default, hanya dapat diakses oleh Anda. Toolkit for Visual Studio memungkinkan Anda menyimpan data di Amazon S3 dan melihat, mengelola, mengambil, dan mendistribusikan data tersebut.

Amazon S3 menggunakan konsep bucket, yang dapat Anda anggap mirip dengan sistem file atau drive logis. Bucket dapat berisi folder, yang mirip dengan direktori, dan objek, yang mirip dengan file. Di bagian ini, kita akan menggunakan konsep-konsep ini saat kita berjalan melalui fungsionalitas Amazon S3 yang diekspos oleh Toolkit for Visual Studio.

> **Note**
>
> Untuk menggunakan alat ini, kebijakan IAM Anda harus memberikan izin untuk`s3:GetBucketAcl`,`s3:GetBucket`, dan `s3:ListBucket` tindakan. Untuk informasi selengkapnya, lihat Ikhtisar Kebijakan AWS IAM.

Membuat sebuah Bucket Amazon S3

Bucket adalah unit penyimpanan paling mendasar di Amazon S3.

Untuk membuat bucket S3

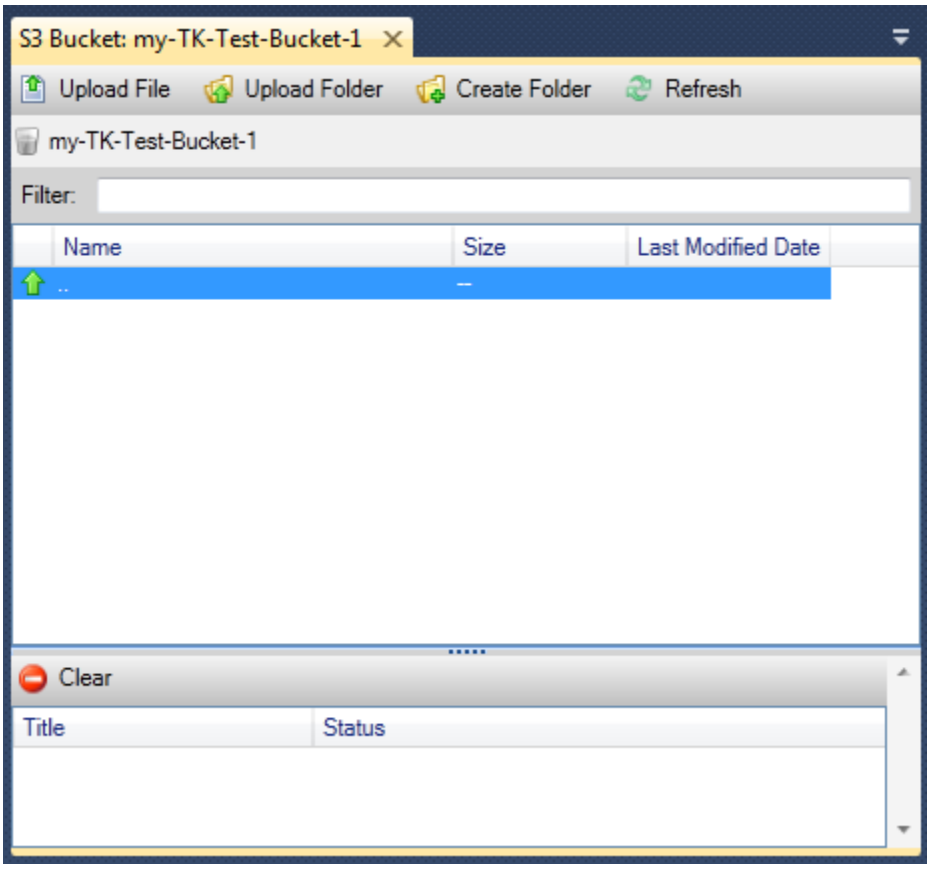
1. Di AWS Explorer, buka menu konteks (klik kanan) untuk node Amazon S3, lalu pilih Buat Bucket.
2. Di kotak dialog Create Bucket, ketikkan nama untuk bucket. Nama bucket harus unik di seberang AWS. Untuk informasi tentang kendala lainnya, buka dokumentasi Amazon S3.
3. Pilih OK.

Mengelola Bucket Amazon S3 dari Explorer AWS

Di AWS Explorer, operasi berikut tersedia saat Anda membuka menu konteks (klik kanan) untuk bucket Amazon S3.

Jelajahi

Menampilkan tampilan objek yang terkandung dalam ember. Dari sini, Anda dapat membuat folder atau mengunggah file atau seluruh direktori dan folder dari komputer lokal Anda. Panel bawah menampilkan pesan status tentang proses upload. Untuk menghapus pesan ini, pilih ikon Hapus. Anda juga dapat mengakses tampilan bucket ini dengan mengklik dua kali nama bucket di AWS Explorer.



Sifat-sifat

Menampilkan kotak dialog di mana Anda dapat melakukan hal berikut:

- Setel izin Amazon S3 yang cakupannya ke:
  - Anda sebagai pemilik ember.
  - semua pengguna yang telah diautentikasi. AWS
  - Semua orang dengan akses internet.
- Nyalakan logging untuk ember.
- Siapkan notifikasi menggunakan Amazon Simple Notification Service (Amazon SNS) sehingga jika Anda menggunakan Reduced Redundancy Storage (RRS), Anda akan diberi tahu jika terjadi kehilangan data. RRS adalah opsi penyimpanan Amazon S3 yang memberikan daya tahan lebih

sedikit daripada penyimpanan standar, tetapi dengan biaya yang lebih rendah. Untuk informasi lebih lanjut, lihat S3 FAQs.

- Buat situs web statis menggunakan data dalam ember.

Kebijakan

Memungkinkan Anda menyiapkan kebijakan AWS Identity and Access Management (IAM) untuk bucket Anda. Untuk informasi lebih lanjut, buka dokumentasi IAM dan kasus penggunaan untuk IAM dan S3.

Buat URL Pra-Tanda Tandatangani

Memungkinkan Anda menghasilkan URL terbatas waktu yang dapat Anda distribusikan untuk menyediakan akses ke konten bucket. Untuk informasi selengkapnya, lihat Cara Membuat URL Pra-Tanda Tandatangani.

Lihat Unggahan Multi-Bagian

Memungkinkan Anda untuk melihat unggahan multipart. Amazon S3 mendukung pemecahan unggahan objek besar menjadi beberapa bagian untuk membuat proses pengunggahan lebih efisien. Untuk informasi lebih lanjut, buka diskusi tentang unggahan multipart dalam dokumentasi S3.

Hapus

Memungkinkan Anda menghapus ember. Anda hanya dapat menghapus bucket kosong.

Mengunggah File dan Folder ke Amazon S3

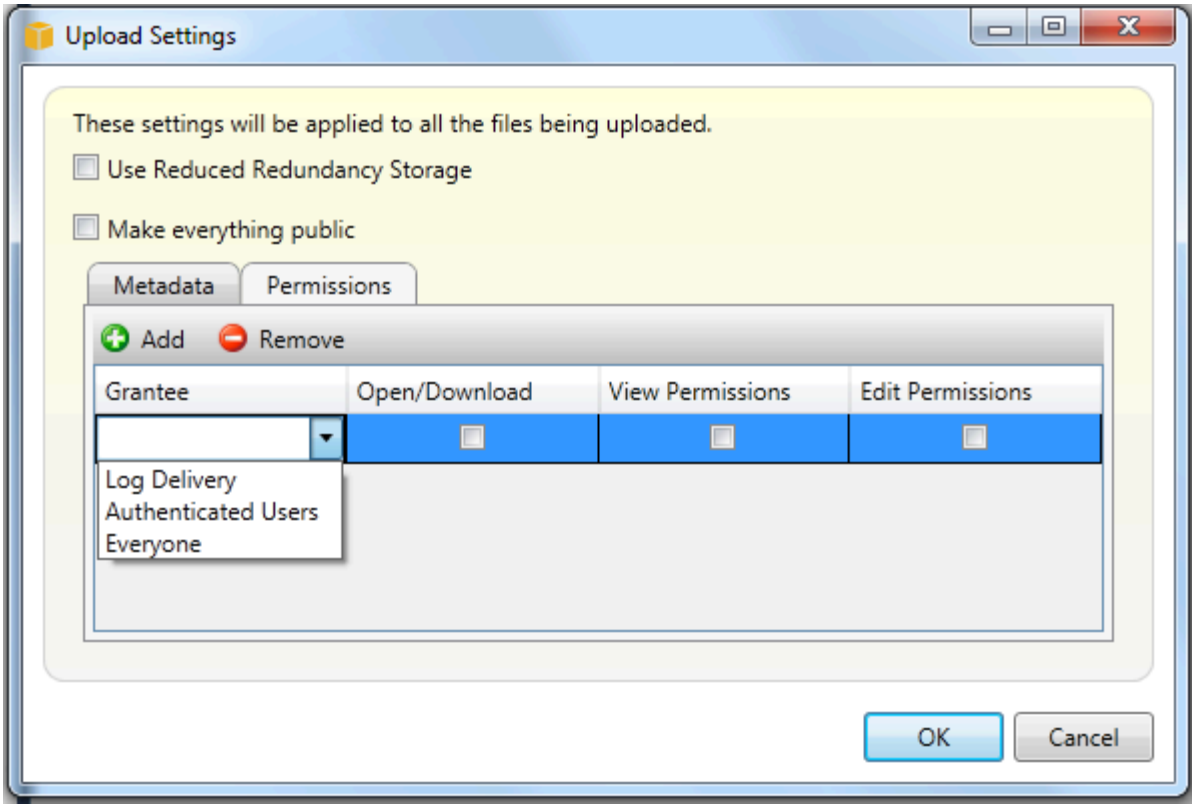
Anda dapat menggunakan AWS Explorer untuk mentransfer file atau seluruh folder dari komputer lokal Anda ke ember Anda.

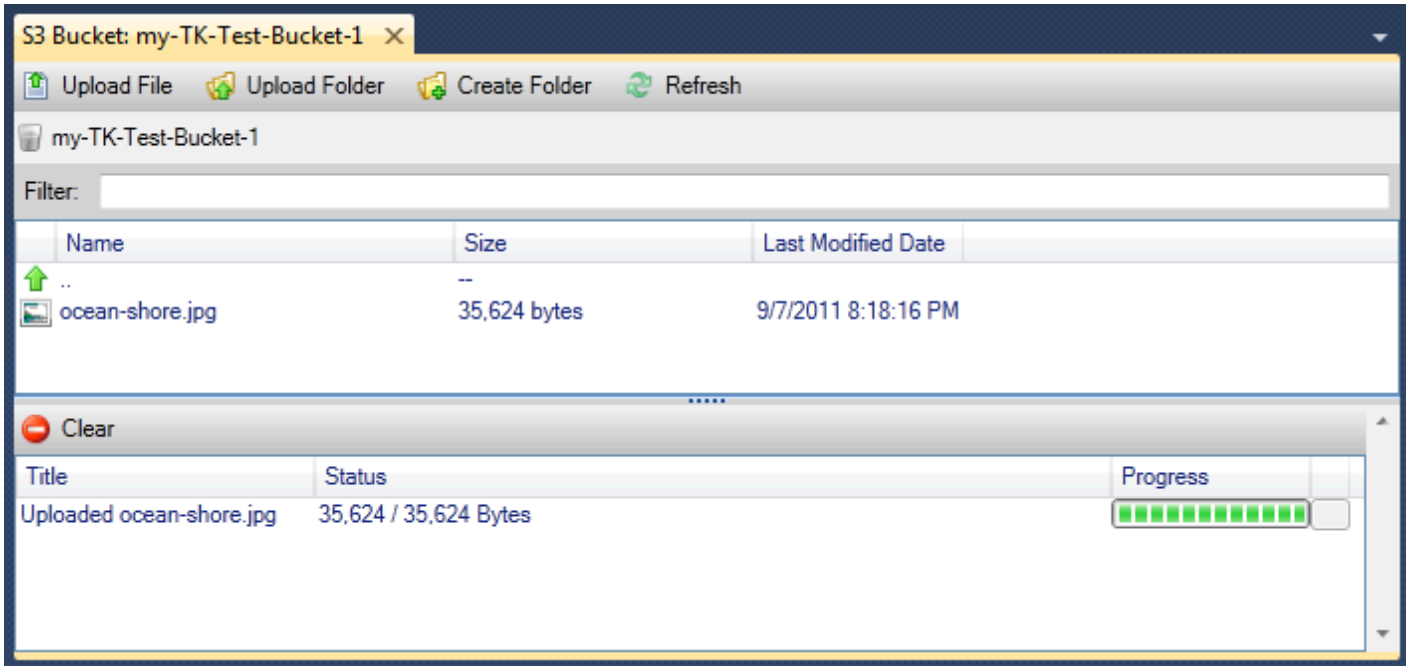
> **Note**
>
> Jika Anda mengunggah file atau folder yang memiliki nama yang sama dengan file atau folder yang sudah ada di bucket Amazon S3, file yang Anda unggah akan menimpa file yang ada tanpa peringatan.

Untuk mengunggah file ke S3

1. Di AWS Explorer, perluas node Amazon S3, dan klik dua kali bucket atau buka menu konteks (klik kanan) untuk bucket dan pilih Browse.
2. Di tampilan Browse bucket Anda, pilih Unggah File atau Unggah Folder.
3. Di kotak dialog File-Open, navigasikan ke file yang akan diunggah, pilih, lalu pilih Buka. Jika Anda mengunggah folder, navigasikan ke dan pilih folder itu, lalu pilih Buka.

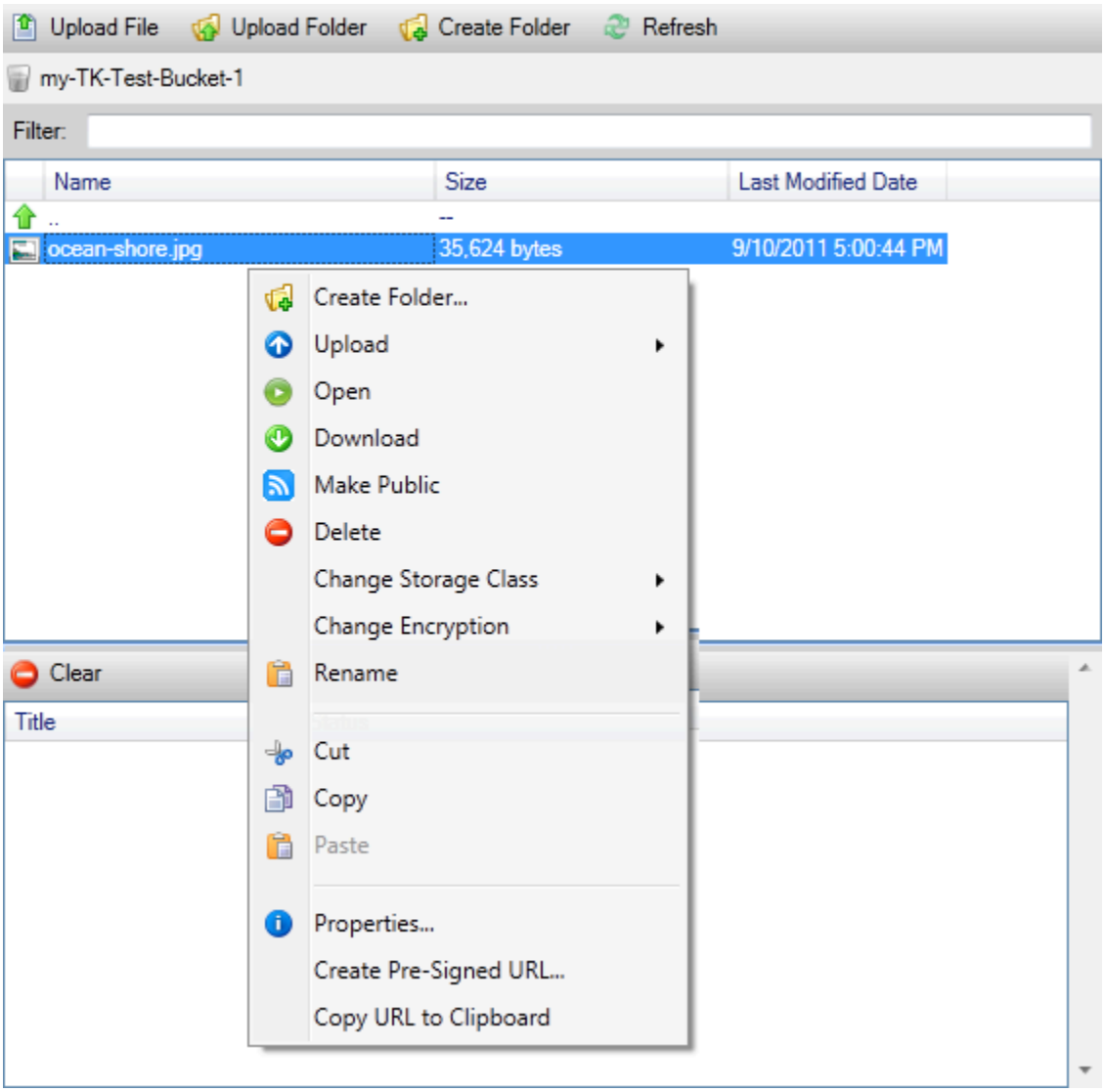
   Kotak dialog Upload Settings memungkinkan Anda untuk mengatur metadata dan izin pada file atau folder yang Anda upload. Memilih kotak centang Jadikan semuanya publik sama dengan menyetel izin Buka/Unduh ke Semua Orang. Anda dapat memilih opsi untuk menggunakan Reduced Redundancy Storage untuk file yang diunggah.





Operasi File Amazon S3 dari AWS Toolkit for Visual Studio

Jika Anda memilih file dalam tampilan Amazon S3 dan membuka menu konteks (klik kanan), Anda dapat melakukan berbagai operasi pada file tersebut.



Buat Folder

Memungkinkan Anda membuat folder di bucket saat ini. (Setara dengan memilih tautan Buat Folder.)

Unggah

Memungkinkan Anda mengunggah file atau folder. (Setara dengan memilih tautan Unggah File atau Unggah Folder.)

Terbuka

Mencoba membuka file yang dipilih di browser default Anda. Bergantung pada jenis file dan kemampuan browser default Anda, file tersebut mungkin tidak ditampilkan. Ini mungkin hanya diunduh oleh browser Anda sebagai gantinya.

Unduh

Membuka kotak dialog Folder-Tree untuk memungkinkan Anda mengunduh file yang dipilih.

Jadikan Publik

Menetapkan izin pada file yang dipilih untuk Buka/Unduh dan Semua Orang. (Setara dengan memilih Buat semuanya publik kotak centang pada kotak dialog Upload Settings.)

Hapus

Menghapus file atau folder yang dipilih. Anda juga dapat menghapus file atau folder dengan memilihnya dan menekan`Delete`.

Ubah Kelas Penyimpanan

Menetapkan kelas penyimpanan ke Standard atau Reduced Redundancy Storage (RRS). Untuk melihat pengaturan kelas penyimpanan saat ini, pilih Properties.

Ubah Enkripsi

Memungkinkan Anda mengatur enkripsi sisi server pada file. Untuk melihat setelan enkripsi saat ini, pilih Properties.

Ganti nama

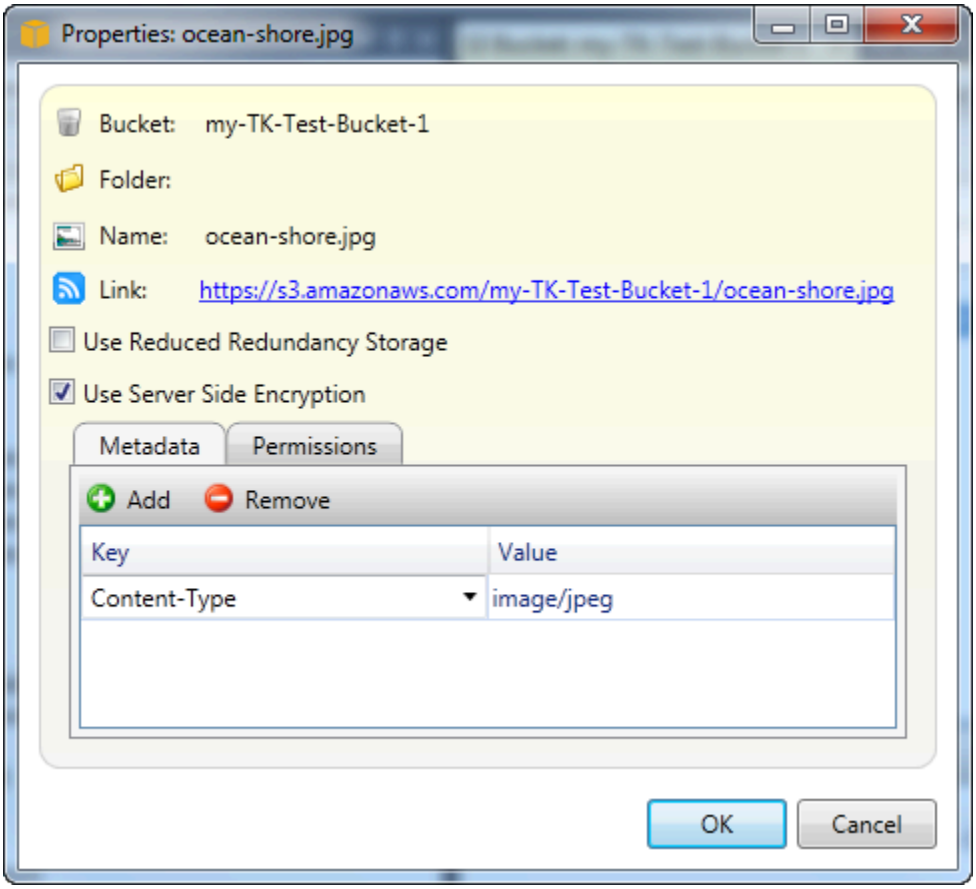
Memungkinkan Anda mengganti nama file. Anda tidak dapat mengganti nama folder.

Potong | Salin | Tempel

Memungkinkan Anda memotong, menyalin, dan menempelkan file atau folder di antara folder atau di antara ember.

Sifat-sifat

Menampilkan kotak dialog yang memungkinkan Anda mengatur metadata dan izin untuk file, serta beralih penyimpanan untuk file antara Reduced Redundancy Storage (RRS) dan Standard, dan mengatur enkripsi sisi server untuk file tersebut. Kotak dialog ini juga menampilkan tautan https ke file. Jika Anda memilih tautan ini, Toolkit for Visual Studio akan membuka file di browser default Anda. Jika Anda memiliki izin pada file yang disetel ke Open/Download dan Everyone, orang lain akan dapat mengakses file melalui tautan ini. Daripada mendistribusikan tautan ini, kami sarankan Anda membuat dan mendistribusikan URLs pra-ditandatangani.



Buat URL Pra-Tanda Tandatangani

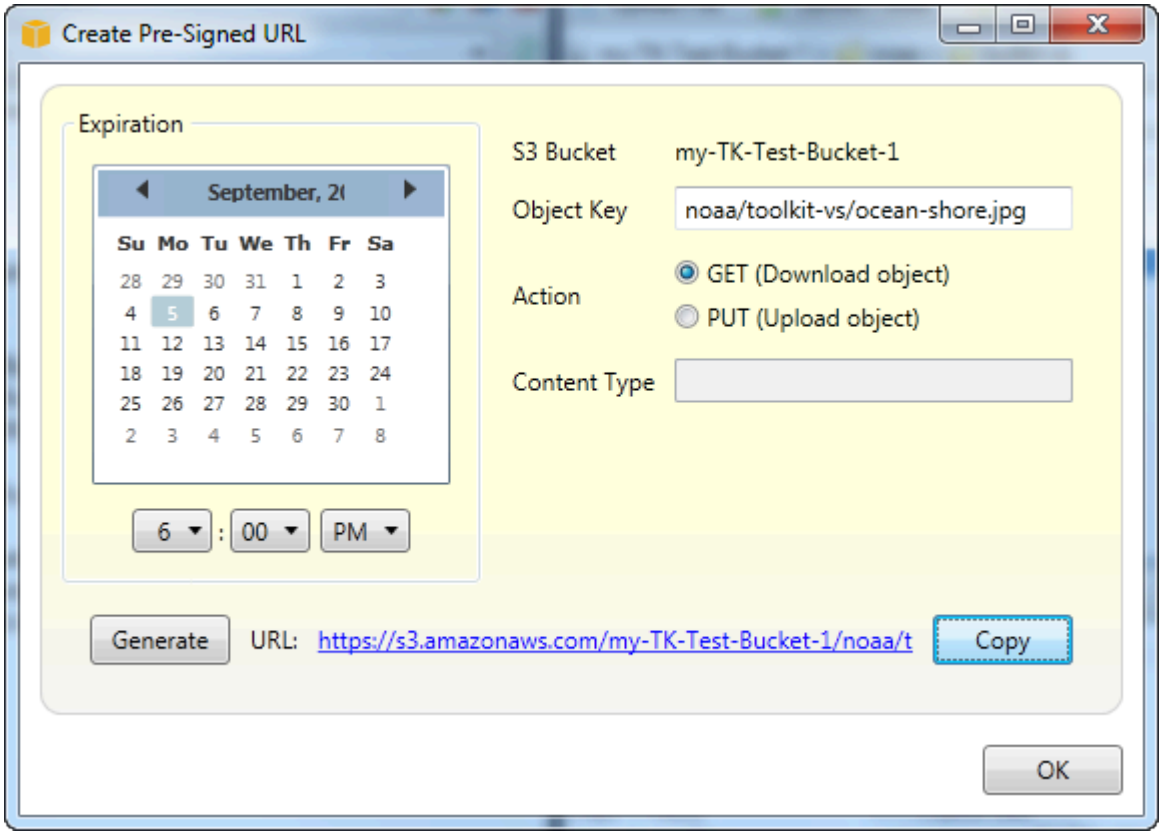
Memungkinkan Anda membuat URL pra-tanda tangan terbatas waktu yang dapat Anda distribusikan untuk memungkinkan orang lain mengakses konten yang telah Anda simpan di Amazon S3.

Cara Membuat URL Pra-Tanda Tandatangani

Anda dapat membuat URL yang telah ditandatangani sebelumnya untuk bucket atau file dalam bucket. Orang lain kemudian dapat menggunakan URL ini untuk mengakses bucket atau file. URL akan kedaluwarsa setelah periode waktu yang Anda tentukan saat Anda membuat URL.

Untuk membuat URL yang telah ditandatangani sebelumnya

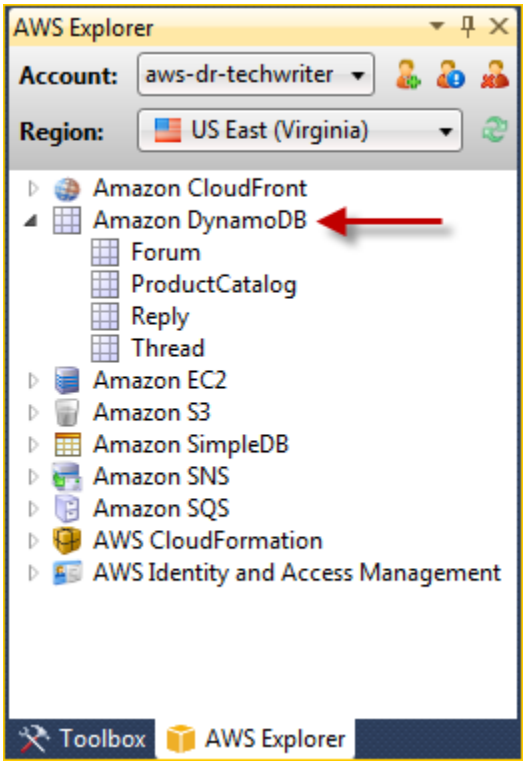
1. Di kotak dialog Buat URL Pra-Tanda Tangan, atur tanggal kedaluwarsa dan waktu URL. Pengaturan default adalah satu jam dari waktu saat ini.
2. Pilih tombol Hasilkan.
3. Untuk menyalin URL ke clipboard, pilih Salin.



Menggunakan DynamoDB dari Explorer AWS

Amazon DynamoDB adalah layanan basis data yang cepat, sangat dapat diskalakan, sangat tersedia, hemat biaya, dan bukan basis data relasional. DynamoDB menghilangkan keterbatasan skalabilitas tradisional pada penyimpanan data sekaligus mempertahankan performa latensi rendah dan dapat diprediksi. Toolkit for Visual Studio menyediakan fungsionalitas untuk bekerja dengan DynamoDB dalam konteks pengembangan. Untuk informasi selengkapnya tentang DynamoDB, lihat DynamoDB di situs web Amazon Web Services.

Di Toolkit for Visual Studio AWS , Explorer menampilkan semua tabel DynamoDB yang terkait dengan yang aktif. Akun AWS



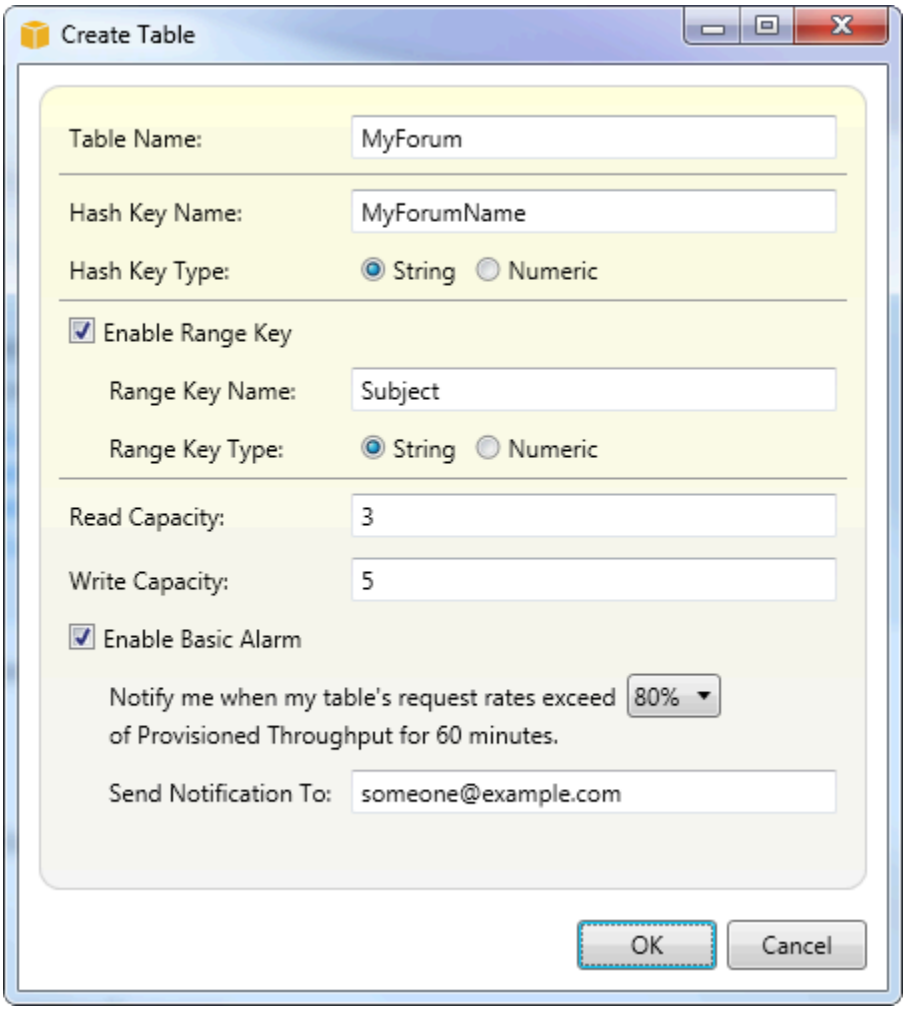
Membuat Tabel DynamoDB

Anda dapat menggunakan Toolkit for Visual Studio untuk membuat tabel DynamoDB.

Untuk membuat tabel di AWS Explorer

1. Di AWS Explorer, buka menu konteks (klik kanan) untuk Amazon DynamoDB, lalu pilih Create Table.
2. Dalam wizard Buat Tabel, di Nama Tabel, ketikkan nama untuk tabel.
3. Di bidang Hash Key Name, ketik atribut kunci hash primer dan dari tombol Hash Key Type, pilih jenis kunci hash. DynamoDB membangun indeks hash tidak berurutan menggunakan atribut kunci utama dan indeks rentang diurutkan opsional menggunakan atribut kunci utama rentang. Untuk informasi selengkapnya tentang atribut kunci hash primer, buka bagian Kunci Utama di Panduan Pengembang Amazon DynamoDB.
4. (Opsional) Pilih Aktifkan Kunci Rentang. Di bidang Range Key Name, ketik atribut kunci rentang, lalu dari tombol Range Key Type, pilih jenis kunci rentang.
5. Di bidang Kapasitas Baca, ketikkan jumlah unit kapasitas baca. Di bidang Kapasitas Tulis, ketikkan jumlah unit kapasitas tulis. Anda harus menentukan minimal tiga unit kapasitas baca dan lima unit kapasitas tulis. Untuk informasi lebih lanjut tentang unit kapasitas baca dan tulis, buka Provisioned Throughput di DynamoDB.

6. (Opsional) Pilih Aktifkan Alarm Dasar untuk mengingatkan Anda ketika tingkat permintaan tabel Anda terlalu tinggi. Pilih persentase throughput yang disediakan per 60 menit yang harus dilampaui sebelum peringatan dikirim. Di Kirim Pemberitahuan Ke, ketik alamat email.
7. Klik OK untuk membuat tabel.

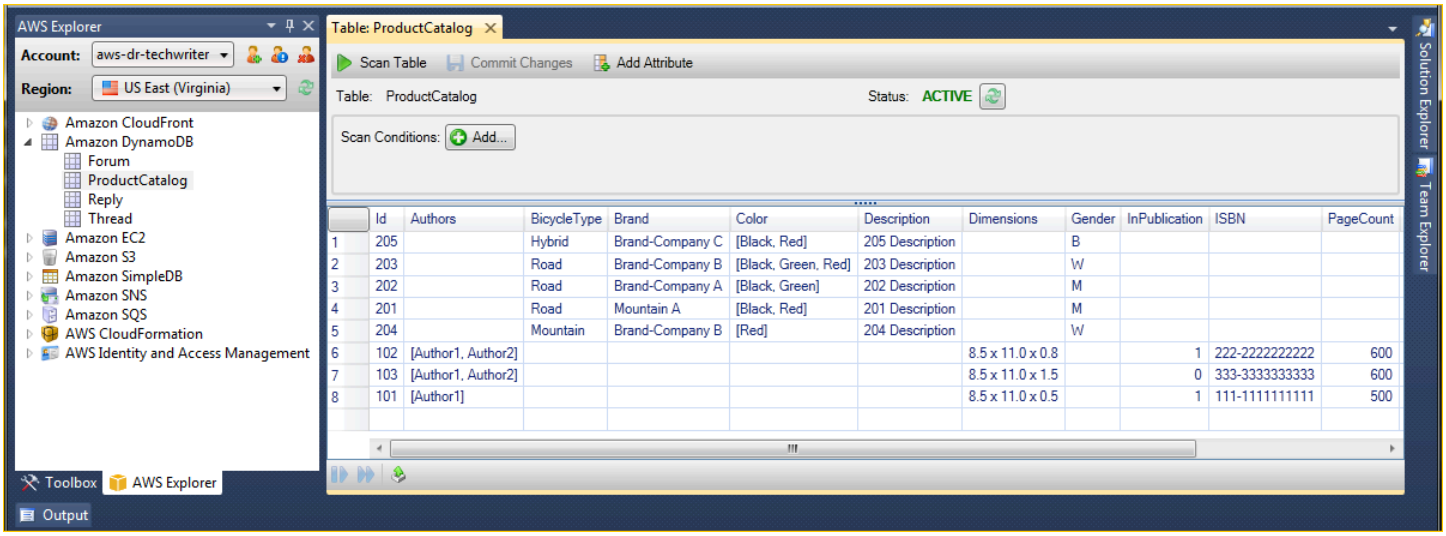


Untuk informasi selengkapnya tentang tabel DynamoDB, buka Konsep Model Data - Tabel, Item, dan Atribut.

Melihat Tabel DynamoDB sebagai Grid

Untuk membuka tampilan kisi dari salah satu tabel DynamoDB Anda, AWS di Explorer, klik dua kali subnode yang sesuai dengan tabel. Dari tampilan kisi, Anda dapat melihat item, atribut, dan nilai yang disimpan dalam tabel. Setiap baris sesuai dengan item dalam tabel. Kolom tabel sesuai dengan atribut. Setiap sel tabel menyimpan nilai yang terkait dengan atribut tersebut untuk item tersebut.

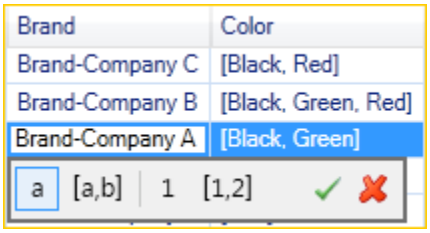
Atribut dapat memiliki nilai yang berupa string atau angka. Beberapa atribut memiliki nilai yang terdiri dari serangkaian string atau angka. Nilai yang ditetapkan ditampilkan sebagai daftar dipisahkan koma yang diapit oleh tanda kurung siku.



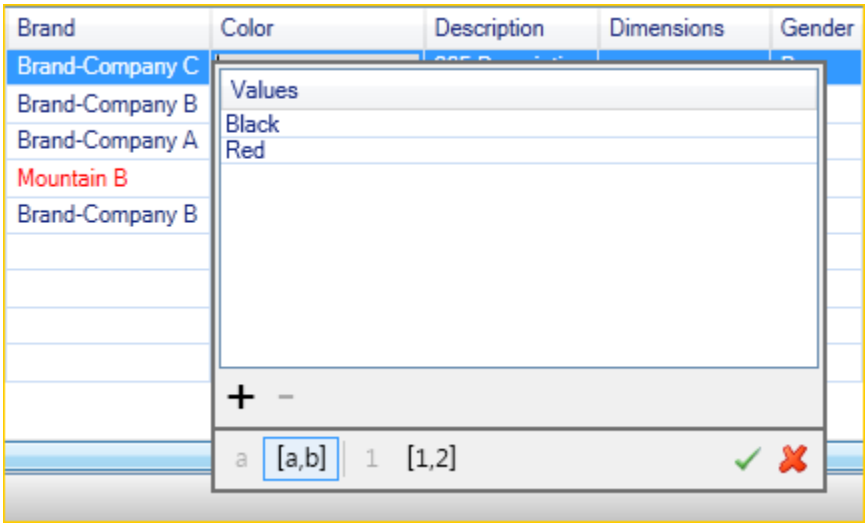
	Id	Authors	BicycleType	Brand	Color	Description	Dimensions	Gender	InPublication	ISBN	PageCount
1	205		Hybrid	Brand-Company C	[Black, Red]	205 Description		B			
2	203		Road	Brand-Company B	[Black, Green, Red]	203 Description		W			
3	202		Road	Brand-Company A	[Black, Green]	202 Description		M			
4	201		Road	Mountain A	[Black, Red]	201 Description		M			
5	204		Mountain	Brand-Company B	[Red]	204 Description		W			
6	102	[Author1, Author2]					8.5 x 11.0 x 0.8		1	222-2222222222	600
7	103	[Author1, Author2]					8.5 x 11.0 x 1.5		0	333-3333333333	600
8	101	[Author1]					8.5 x 11.0 x 0.5		1	111-1111111111	500

Mengedit dan Menambahkan Atribut dan Nilai

Dengan mengklik dua kali sel, Anda dapat mengedit nilai untuk atribut item yang sesuai. Untuk atribut set-value, Anda juga dapat menambahkan atau menghapus nilai individual dari set.



Selain mengubah nilai atribut, Anda juga dapat, dengan beberapa batasan, mengubah format nilai untuk atribut. Misalnya, nilai angka apa pun dapat diubah menjadi nilai string. Jika Anda memiliki nilai string, isinya adalah angka, seperti 125, editor sel memungkinkan Anda untuk mengonversi format nilai dari string ke angka. Anda juga dapat mengonversi nilai tunggal menjadi set-value. Namun, Anda umumnya tidak dapat mengonversi dari set-value ke nilai tunggal; pengecualian adalah ketika set-value memiliki, pada kenyataannya, hanya satu elemen dalam himpunan.

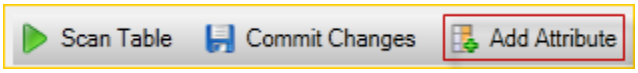


Setelah mengedit nilai atribut, pilih tanda centang hijau untuk mengonfirmasi perubahan Anda. Jika Anda ingin membuang perubahan Anda, pilih X merah.

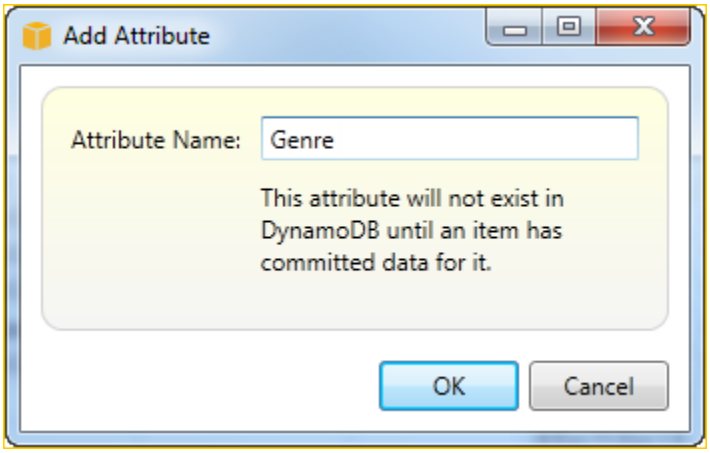
Setelah Anda mengkonfirmasi perubahan Anda, nilai atribut akan ditampilkan dengan warna merah. Ini menunjukkan atribut telah diperbarui, tetapi nilai baru belum ditulis kembali ke database DynamoDB. Untuk menulis perubahan Anda kembali ke DynamoDB, pilih Commit Changes. Untuk membuang perubahan Anda, pilih Tabel Pindai dan ketika Toolkit menanyakan apakah Anda ingin melakukan perubahan sebelum Pemindaian, pilih Tidak.

Menambahkan Atribut

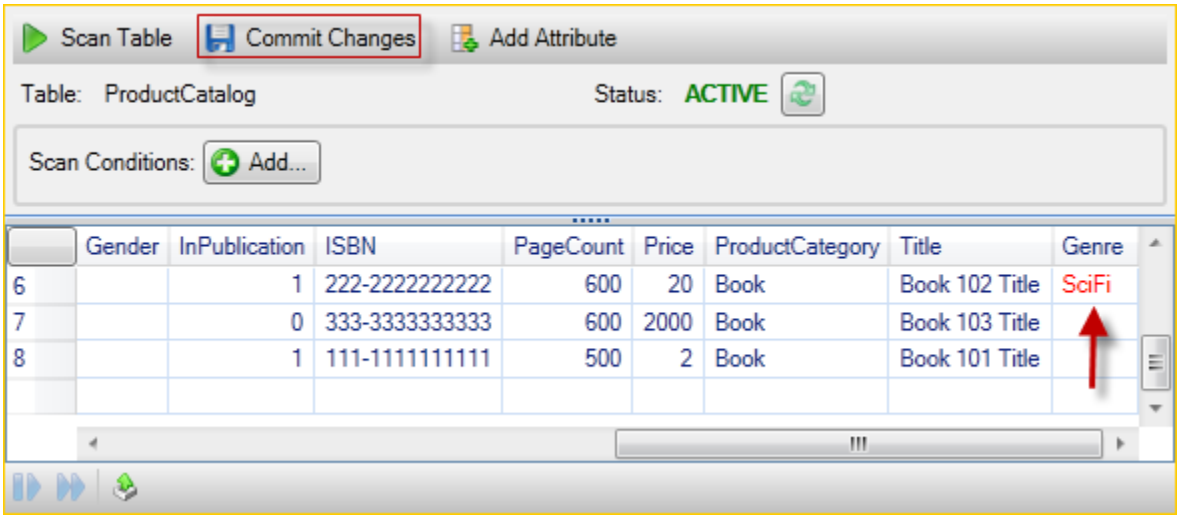
Dari tampilan grid, Anda juga dapat menambahkan atribut ke tabel. Untuk menambahkan atribut baru, pilih Tambah Atribut.



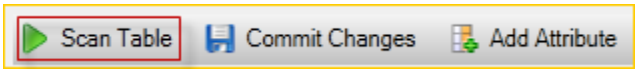
Dalam kotak dialog Add Attribute, ketikkan nama untuk atribut Anda, lalu pilih OK.



Untuk membuat atribut baru menjadi bagian dari tabel, Anda harus menambahkan nilai untuk setidaknya satu item dan kemudian memilih tombol Commit Changes. Untuk membuang atribut baru, cukup tutup tampilan kisi tabel tanpa memilih Commit Changes.



Memindai Tabel DynamoDB

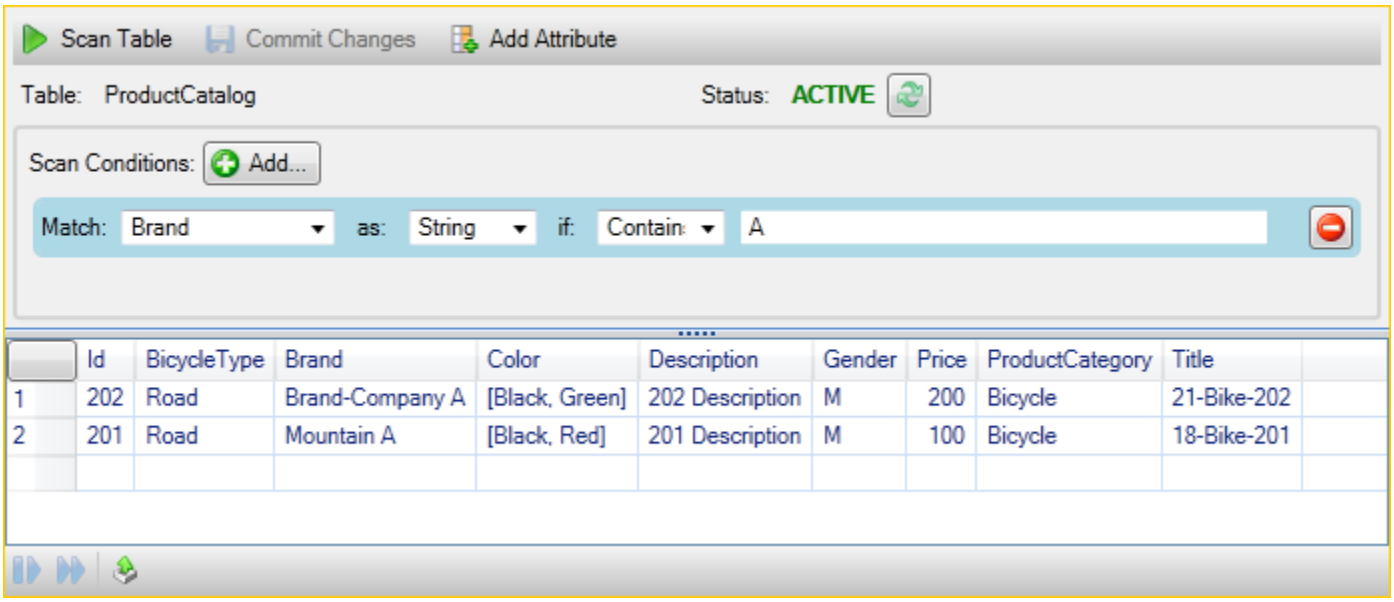


Anda dapat melakukan Scan pada tabel DynamoDB Anda dari Toolkit. Dalam Scan, Anda menentukan satu set kriteria dan Scan mengembalikan semua item dari tabel yang cocok dengan kriteria Anda. Pemindaian adalah operasi yang mahal dan harus digunakan dengan hati-hati untuk menghindari mengganggu lalu lintas produksi prioritas yang lebih tinggi di atas meja. Untuk informasi selengkapnya tentang penggunaan operasi Pindai, buka Panduan Pengembang Amazon DynamoDB.

Untuk melakukan Scan pada tabel DynamoDB dari Explorer AWS

1. Dalam tampilan kisi, pilih kondisi pemindaian: tombol tambah.
2. Dalam editor klausa Pindai, pilih atribut yang akan dicocokkan, bagaimana nilai atribut harus ditafsirkan (string, angka, nilai set), bagaimana seharusnya dicocokkan (misalnya Dimulai Dengan atau Berisi), dan nilai literal yang harus cocok.
3. Tambahkan lebih banyak klausa Pindai, sesuai kebutuhan, untuk pencarian Anda. Pemindaian hanya akan mengembalikan item yang sesuai dengan kriteria dari semua klausa Pemindaian Anda. Pemindaian akan melakukan perbandingan peka huruf besar/kecil saat mencocokkan dengan nilai string.
4. Pada bilah tombol di bagian atas tampilan kisi, pilih Tabel Pindai.

Untuk menghapus klausa Pindai, pilih tombol merah dengan garis putih di sebelah kanan setiap klausa.



Untuk kembali ke tampilan tabel yang mencakup semua item, hapus semua klausa Pindai dan pilih Pindai Tabel lagi.

Hasil Pemindaian Paginating

Di bagian bawah tampilan ada tiga tombol.

Dua tombol biru pertama memberikan pagination untuk hasil Scan. Tombol pertama akan menampilkan halaman hasil tambahan. Tombol kedua akan menampilkan tambahan sepuluh halaman hasil. Dalam konteks ini, halaman sama dengan 1 MB konten.

Ekspor Hasil Pemindaian ke CSV

Tombol ketiga mengekspor hasil dari Pindai saat ini ke file CSV.

Menggunakan AWS CodeCommit dengan Visual Studio Team Explorer

Anda dapat menggunakan akun pengguna AWS Identity and Access Management (IAM) untuk membuat kredensi Git dan menggunakannya untuk membuat dan mengkloning repositori dari dalam Team Explorer.

Jenis Kredensi untuk AWS CodeCommit

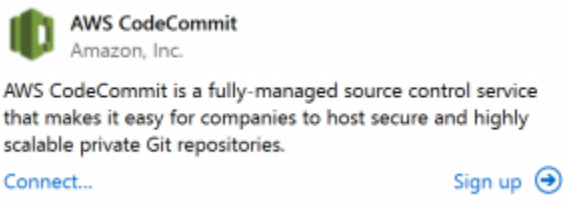
Sebagian besar AWS Toolkit for Visual Studio pengguna mengetahui pengaturan profil AWS kredensi yang berisi akses dan kunci rahasia mereka. Profil kredensi ini digunakan dalam Toolkit for Visual Studio untuk mengaktifkan panggilan ke APIs layanan, misalnya, untuk mencantumkan bucket Amazon S3 AWS di Explorer atau untuk meluncurkan instans Amazon. EC2 Integrasi AWS CodeCommit dengan Team Explorer juga menggunakan profil kredensi ini. Namun, untuk bekerja dengan Git itu sendiri, Anda memerlukan kredensi tambahan, khususnya kredensi Git untuk koneksi HTTPS. Anda dapat membaca tentang kredensil ini (nama pengguna dan kata sandi) di Pengaturan untuk Pengguna HTTPS Menggunakan Kredensial Git di Panduan Pengguna.AWS CodeCommit

Anda dapat membuat kredensi Git AWS CodeCommit hanya untuk akun pengguna IAM. Anda tidak dapat membuatnya untuk akun root. Anda dapat membuat hingga dua set kredensil ini untuk layanan dan, meskipun Anda dapat menandai satu set kredensil sebagai tidak aktif, set tidak aktif masih dihitung terhadap batas dua set Anda. Perhatikan bahwa Anda dapat menghapus dan membuat ulang kredensional kapan saja. Bila Anda menggunakan AWS CodeCommit dari dalam Visual Studio, AWS kredensi tradisional Anda digunakan untuk bekerja dengan layanan itu sendiri, misalnya, saat Anda membuat dan mencantumkan repositori. Saat bekerja dengan repositori Git aktual yang dihosting AWS CodeCommit, Anda menggunakan kredensi Git.

Sebagai bagian dari dukungan untuk AWS CodeCommit, Toolkit for Visual Studio secara otomatis membuat dan mengelola kredensi Git ini untuk Anda dan mengaitkannya dengan profil kredensi Anda. AWS Anda tidak perlu khawatir tentang memiliki kumpulan kredensional yang tepat untuk melakukan operasi Git dalam Team Explorer. Setelah Anda terhubung ke Team Explorer dengan profil AWS kredensialnya, kredensi Git yang terkait akan digunakan secara otomatis setiap kali Anda bekerja dengan remote Git.

Menghubungkan ke AWS CodeCommit

Saat Anda membuka jendela Team Explorer di Visual Studio 2015 atau yang lebih baru, Anda akan melihat AWS CodeCommit entri di bagian Penyedia Layanan yang Dihosting dari Kelola Koneksi.

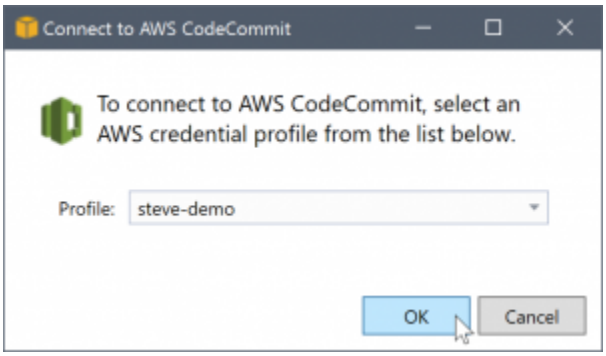


Memilih Daftar membuka halaman beranda Amazon Web Services di jendela browser. Apa yang terjadi ketika Anda memilih Connect tergantung pada apakah Toolkit for Visual Studio dapat

menemukan profil kredensi AWS dengan akses dan kunci rahasia untuk memungkinkannya melakukan panggilan AWS atas nama Anda. Anda mungkin telah menyiapkan profil kredensi dengan menggunakan halaman Memulai baru yang ditampilkan di IDE saat Toolkit for Visual Studio tidak dapat menemukan kredenal yang disimpan secara lokal. Atau Anda mungkin telah menggunakan Toolkit for Visual Studio, AWS Tools for Windows PowerShell dan sudah AWS memiliki profil AWS CLI kredensi yang tersedia untuk Toolkit for Visual Studio untuk digunakan.

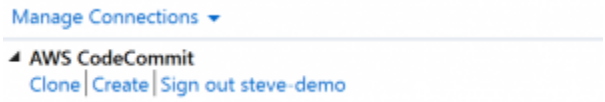
Saat Anda memilih Connect, Toolkit for Visual Studio memulai proses untuk menemukan profil kredensi yang akan digunakan dalam koneksi. Jika Toolkit for Visual Studio tidak dapat menemukan profil kredensi, itu akan membuka kotak dialog yang mengundang Anda untuk memasukkan akses dan kunci rahasia untuk Anda. Akun AWS Kami sangat menyarankan Anda menggunakan akun pengguna IAM, dan bukan kredensi root Anda. Selain itu, seperti disebutkan sebelumnya, kredensi Git yang pada akhirnya Anda butuhkan hanya dapat dibuat untuk pengguna IAM. Setelah akses dan kunci rahasia disediakan dan profil kredensi dibuat, koneksi antara Team Explorer dan AWS CodeCommit siap digunakan.

Jika Toolkit for Visual Studio menemukan lebih dari AWS satu profil kredensi, Anda diminta untuk memilih akun yang ingin Anda gunakan dalam Team Explorer.



Jika Anda hanya memiliki satu profil kredensi, Toolkit for Visual Studio melewati kotak dialog pemilihan profil dan Anda langsung terhubung:

Ketika koneksi dibuat antara Team Explorer dan AWS CodeCommit melalui profil kredensi Anda, kotak dialog undangan ditutup dan panel koneksi ditampilkan.



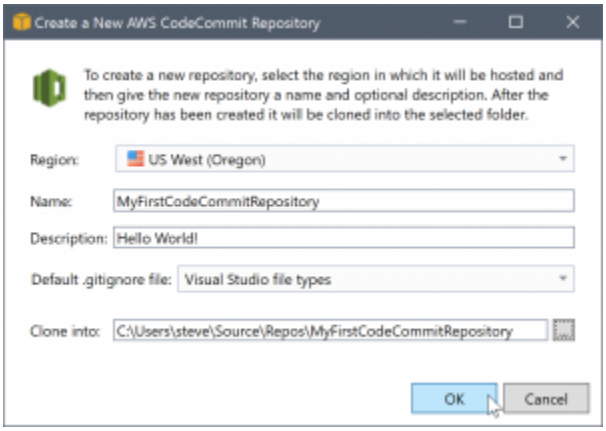
Karena Anda tidak memiliki repositori yang dikloning secara lokal, panel hanya menampilkan operasi yang dapat Anda lakukan: Kloning, Buat, dan Keluar. Seperti penyedia lainnya, AWS CodeCommit di Team Explorer hanya dapat terikat pada satu profil AWS kredensi pada waktu tertentu. Untuk beralih

akun, Anda menggunakan Keluar untuk menghapus koneksi sehingga Anda dapat memulai koneksi baru menggunakan akun lain.

Sekarang setelah Anda membuat koneksi, Anda dapat membuat repositori dengan mengklik tautan Buat.

Membuat Repositori

Ketika Anda mengklik link Create, kotak dialog Create a New AWS CodeCommit Repository terbuka.



AWS CodeCommit repositori diatur berdasarkan wilayah, jadi di Wilayah Anda dapat memilih wilayah tempat untuk meng-host repositori. Daftar ini memiliki semua wilayah yang AWS CodeCommit didukung. Anda memberikan Nama (wajib) dan Deskripsi (opsional) untuk repositori baru kami.

Perilaku default kotak dialog adalah untuk akhiran lokasi folder untuk repositori baru dengan nama repositori (saat Anda memasukkan nama, lokasi folder juga diperbarui). Untuk menggunakan nama folder yang berbeda, edit Clone ke jalur folder setelah Anda selesai memasukkan nama repositori.

Anda juga dapat memilih untuk secara otomatis membuat `.gitignore` file awal untuk repositori. AWS Toolkit for Visual Studio Ini menyediakan default bawaan untuk jenis file Visual Studio. Anda juga dapat memilih untuk tidak memiliki file atau menggunakan file kustom yang sudah ada yang ingin Anda gunakan kembali di seluruh repositori. Cukup pilih Gunakan kustom dalam daftar dan arahkan ke file kustom yang akan digunakan.

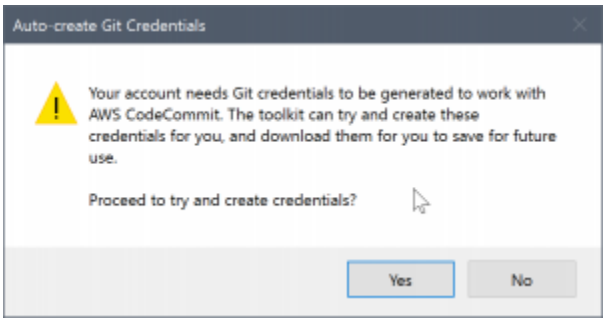
Setelah Anda memiliki nama dan lokasi repositori, Anda siap untuk mengklik OK dan mulai membuat repositori. Toolkit for Visual Studio meminta layanan membuat repositori dan kemudian mengkloning repositori baru secara lokal, menambahkan komit awal untuk file.gitignore, jika Anda menggunakannya. Pada titik inilah Anda mulai bekerja dengan remote Git, jadi Toolkit for Visual Studio sekarang membutuhkan akses ke kredenal Git yang dijelaskan sebelumnya.

Menyiapkan Kredensial Git

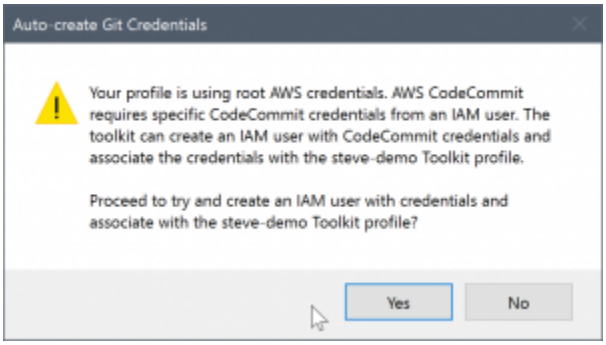
Sampai saat ini Anda telah menggunakan AWS akses dan kunci rahasia untuk meminta layanan membuat repositori Anda. Sekarang Anda perlu bekerja dengan Git itu sendiri untuk melakukan operasi klon yang sebenarnya, dan Git tidak memahami AWS akses dan kunci rahasia. Sebagai gantinya, Anda perlu memberikan nama pengguna dan kredensial kata sandi ke Git untuk digunakan pada koneksi HTTPS dengan remote.

Sebagaimana dicatat dalam Menyiapkan kredensial Git, kredensial Git yang akan Anda gunakan harus dikaitkan dengan pengguna IAM. Anda tidak dapat membuatnya untuk kredensi root. Anda harus selalu mengatur profil AWS kredensi Anda untuk berisi akses pengguna IAM dan kunci rahasia, dan bukan kunci root. Toolkit for Visual Studio dapat mencoba menyiapkan kredensi Git AWS CodeCommit untuk Anda, dan mengaitkannya dengan profil kredensi yang Anda gunakan untuk terhubung AWS di Team Explorer sebelumnya.

Ketika Anda memilih OK di kotak dialog Create a New AWS CodeCommit Repository dan berhasil membuat repositori, Toolkit for Visual Studio memeriksa profil kredensi yang terhubung AWS di Team Explorer untuk menentukan apakah kredensi Git ada dan terkait secara lokal dengan AWS CodeCommit profil. Jika demikian, Toolkit for Visual Studio menginstruksikan Team Explorer untuk memulai operasi klon pada repositori baru. Jika kredensil Git tidak tersedia secara lokal, Toolkit for Visual Studio akan memeriksa jenis kredenal akun yang digunakan dalam koneksi di Team Explorer. Jika kredensialnya untuk pengguna IAM, seperti yang kami sarankan, pesan berikut akan ditampilkan.

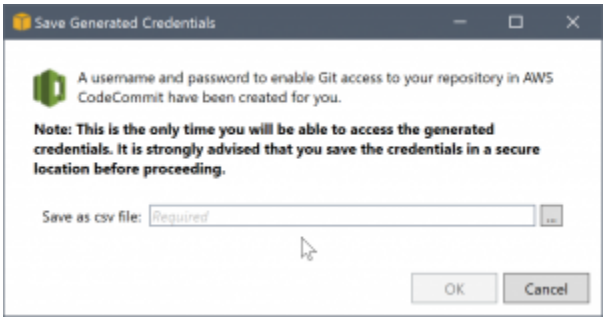


Jika kredensialnya adalah kredensial root, pesan berikut akan ditampilkan sebagai gantinya.



Dalam kedua kasus tersebut, Toolkit for Visual Studio menawarkan untuk mencoba melakukan pekerjaan untuk membuat kredensil Git yang diperlukan untuk Anda. Dalam skenario pertama, semua yang perlu dibuat adalah satu set kredensi Git untuk pengguna IAM. Saat akun root sedang digunakan, Toolkit for Visual Studio pertama kali mencoba membuat pengguna IAM dan kemudian melanjutkan untuk membuat kredensi Git untuk pengguna baru tersebut. Jika Toolkit for Visual Studio harus membuat pengguna baru, itu akan menerapkan AWS CodeCommit kebijakan terkelola Power User ke akun pengguna baru tersebut. Kebijakan ini hanya mengizinkan akses ke AWS CodeCommit dan memungkinkan semua operasi dilakukan AWS CodeCommit kecuali untuk penghapusan repositori.

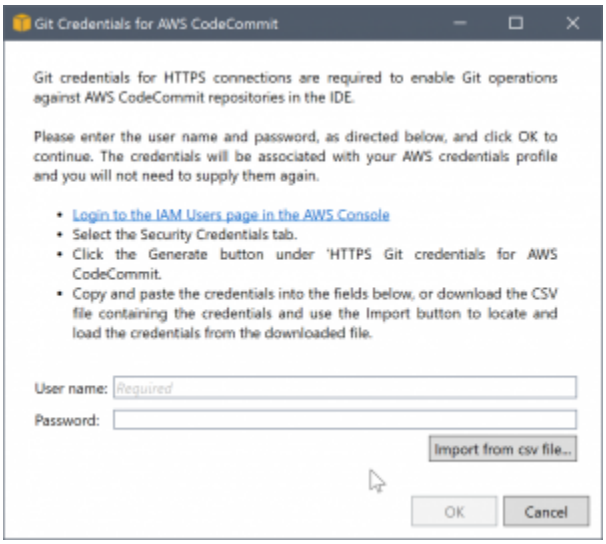
Saat Anda membuat kredensi, Anda hanya dapat melihatnya sekali. Oleh karena itu, Toolkit for Visual Studio meminta Anda untuk menyimpan kredensi yang baru dibuat sebagai `.csv` file sebelum melanjutkan.



Ini adalah sesuatu yang juga sangat kami rekomendasikan, dan pastikan untuk menyimpannya ke lokasi yang aman!

Mungkin ada kasus di mana Toolkit for Visual Studio tidak dapat secara otomatis membuat kredensi. Misalnya, Anda mungkin telah membuat jumlah maksimum set kredensil Git untuk AWS CodeCommit (dua), atau Anda mungkin tidak memiliki hak program yang memadai untuk Toolkit for Visual Studio untuk melakukan pekerjaan untuk Anda (jika Anda masuk sebagai pengguna IAM). Dalam kasus ini, Anda dapat masuk ke Konsol Manajemen AWS untuk mengelola kredensi atau mendapatkannya

dari administrator Anda. Anda kemudian dapat memasukkannya ke dalam kotak AWS CodeCommit dialog Git Credentials for, yang ditampilkan Toolkit for Visual Studio.

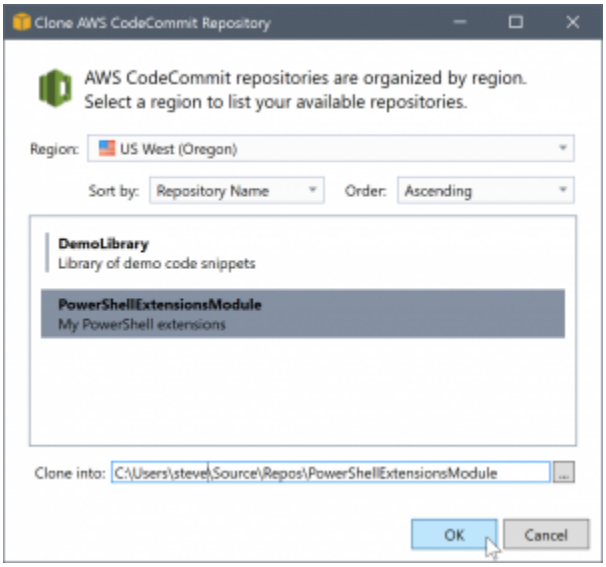


Sekarang bahwa kredensil untuk Git tersedia, operasi klon untuk repositori baru berlangsung (lihat indikasi kemajuan untuk operasi di dalam Team Explorer). Jika Anda memilih untuk menerapkan `.gitignore` file default, itu berkomitmen ke repositori dengan komentar 'Komite Awal'.

Hanya itu yang ada untuk menyiapkan kredensi dan membuat repositori dalam Team Explorer. Setelah kredensi yang diperlukan ada, yang Anda lihat saat membuat repositori baru di masa depan adalah kotak dialog Create a New AWS CodeCommit Repository itu sendiri.

Mengkloning Repositori

Untuk mengkloning repositori yang ada, kembali ke panel koneksi untuk AWS CodeCommit di Team Explorer. Klik tautan Clone untuk membuka kotak dialog Clone AWS CodeCommit Repository, lalu pilih repositori untuk dikloning dan lokasi pada disk tempat Anda ingin meletakkannya.



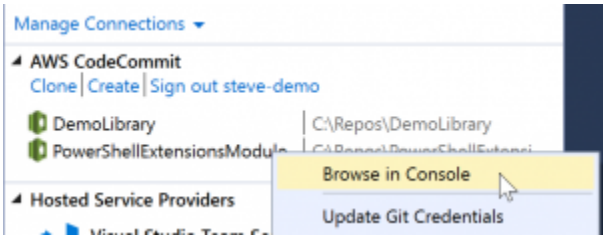
Setelah Anda memilih wilayah, Toolkit for Visual Studio menanyakan layanan untuk menemukan repositori yang tersedia di wilayah tersebut dan menampilkannya di bagian daftar pusat kotak dialog. Nama dan deskripsi opsional dari setiap repositori juga ditampilkan. Anda dapat menyusun ulang daftar untuk mengurutkannya berdasarkan nama repositori atau tanggal modifikasi terakhir, dan mengurutkan masing-masing dalam urutan naik atau turun.

Setelah Anda memilih repositori, Anda dapat memilih lokasi untuk dikloning. Ini default ke lokasi repositori yang sama yang digunakan di plugin lain ke Team Explorer, tetapi Anda dapat menelusuri atau memasukkan lokasi lain. Secara default, nama repositori diakhiran ke jalur yang dipilih. Namun, jika Anda menginginkan jalur tertentu, cukup edit kotak teks setelah Anda memilih folder. Teks apa pun yang ada di dalam kotak ketika Anda mengklik OK akan menjadi folder di mana Anda akan menemukan repositori kloning.

Setelah memilih repositori dan lokasi folder, Anda kemudian klik OK untuk melanjutkan operasi klon. Sama seperti membuat repositori, Anda dapat melihat kemajuan operasi klon yang dilaporkan di Team Explorer.

Bekerja dengan Repositori

Saat Anda mengkloning atau membuat repositori, perhatikan bahwa repositori lokal untuk koneksi terdaftar di panel koneksi di Team Explorer di bawah tautan operasi. Entri ini memberi Anda cara mudah untuk mengakses repositori untuk menelusuri konten. Cukup klik kanan repositori dan pilih Browse in Console.



Anda juga dapat menggunakan Update Git Credentials untuk memperbarui kredensi Git tersimpan yang terkait dengan profil kredensialnya. Ini berguna jika Anda telah memutar kredensialnya. Perintah membuka kotak AWS CodeCommit dialog Git Credentials untuk tempat Anda dapat memasukkan atau mengimpor kredensi baru.

Operasi Git pada repositori berfungsi seperti yang Anda harapkan. Anda dapat membuat komit lokal dan, ketika Anda siap untuk berbagi, Anda menggunakan opsi Sinkronisasi di Team Explorer. Karena kredensil Git sudah disimpan secara lokal dan terkait dengan profil AWS kredensi kami yang terhubung, kami tidak akan diminta untuk menyediakannya lagi untuk operasi terhadap remote. AWS CodeCommit

Menggunakan CodeArtifact di Visual Studio

AWS CodeArtifact adalah layanan repositori artefak yang dikelola sepenuhnya yang memudahkan organisasi untuk menyimpan dan berbagi paket perangkat lunak yang digunakan untuk pengembangan aplikasi dengan aman. Anda dapat menggunakan CodeArtifact dengan alat build populer dan manajer paket seperti NuGet dan .NET Core CLIs dan Visual Studio. Anda juga dapat mengonfigurasi CodeArtifact untuk menarik paket dari repositori publik eksternal seperti NuGet .org.

Di CodeArtifact, paket Anda disimpan dalam repositori yang kemudian disimpan dalam domain. AWS Toolkit for Visual Studio Ini menyederhanakan konfigurasi Visual Studio dengan CodeArtifact repositori Anda, sehingga mudah untuk mengkonsumsi paket di Visual Studio baik CodeArtifact secara langsung maupun .org. NuGet

Tambahkan CodeArtifact repositori Anda sebagai sumber paket NuGet

Untuk mengkonsumsi paket dari Anda CodeArtifact, Anda perlu menambahkan repositori Anda sebagai sumber packabe di Package Manager di Visual NuGet Studio

Untuk menambahkan repositori Anda sebagai sumber paket

1. Di AWS Explorer, navigasikan ke repositori Anda di node. AWS CodeArtifact

2. Buka menu konteks (klik kanan) untuk repositori yang ingin Anda tambahkan, lalu pilih Copy NuGet Source Endpoint.
3. Arahkan ke Package Sources di bawah node NuGet Package Manager di menu Tools > Options.
4. Di Package Sources, pilih tanda tambah (+), edit nama, dan tempel URL titik akhir NuGet sumber yang Anda salin sebelumnya di bidang Sumber.
5. Pilih kotak centang di sebelah sumber paket yang baru ditambahkan untuk mengaktifkannya.

> **Note**
>
> Sebaiknya tambahkan koneksi eksternal NuGetke.org ke Anda CodeArtifact dan menonaktifkan sumber paket nuget.org di Visual Studio. Saat menggunakan koneksi eksternal, semua dependensi yang ditarik dari NuGet.org disimpan di. CodeArtifact Jika NuGet.org turun karena alasan apa pun, paket yang Anda butuhkan akan tetap tersedia. Untuk informasi selengkapnya tentang koneksi eksternal, lihat Menambahkan koneksi eksternal di Panduan AWS CodeArtifact Pengguna.

6. Pilih OK untuk menutup menu.

Untuk informasi selengkapnya tentang penggunaan CodeArtifact dengan Visual Studio, lihat Menggunakan CodeArtifact dengan Visual Studio di Panduan AWS CodeArtifact Pengguna.

Amazon RDS dari Explorer AWS

Amazon Relational Database Service (Amazon RDS) adalah layanan yang memungkinkan Anda menyediakan dan mengelola sistem database relasional SQL di cloud. Amazon RDS mendukung tiga jenis sistem database:

- MySQL Community Edition
- Oracle Database Edisi Perusahaan
- Microsoft SQL Server (Edisi Ekspres, Standar, atau Web)

Untuk informasi selengkapnya, lihat Panduan Pengguna Amazon RDS.

Banyak fungsi yang dibahas di sini juga tersedia melalui AWS Management Console untuk Amazon RDS.

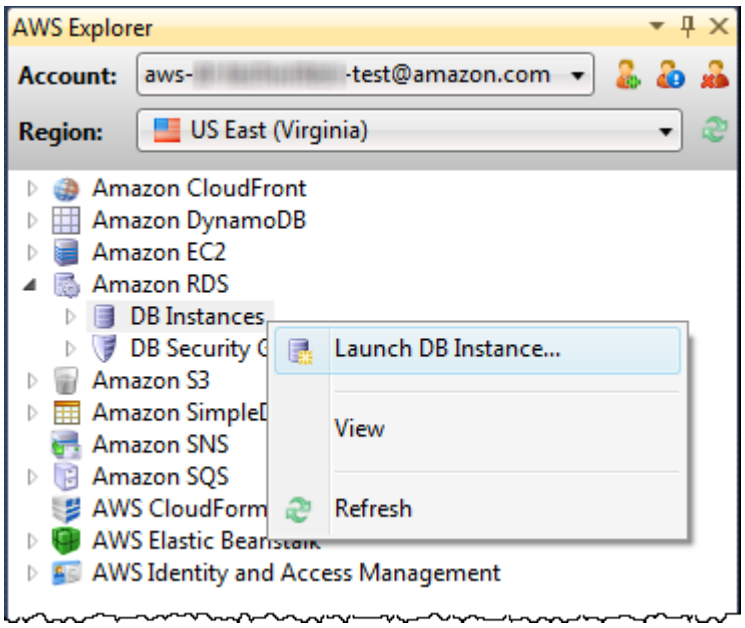
Topik

Luncurkan Instans Database Amazon RDS

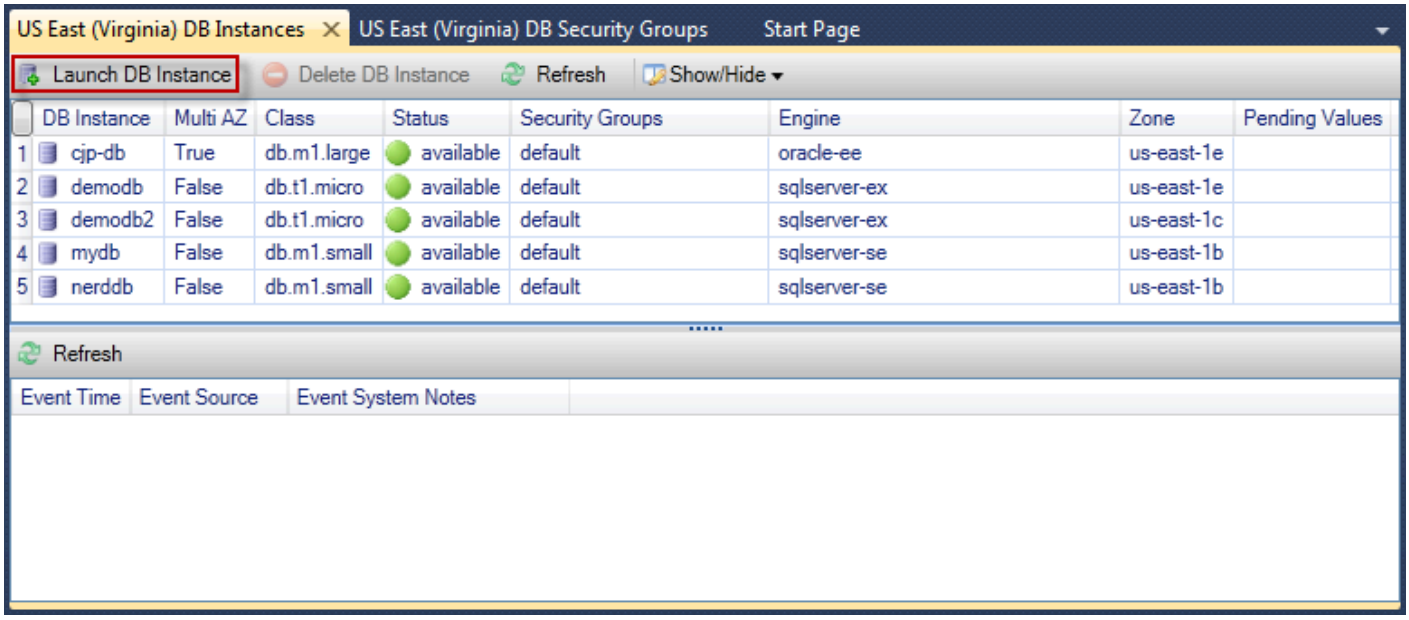
Dengan AWS Explorer, Anda dapat meluncurkan instance dari salah satu mesin database yang didukung oleh Amazon RDS. Panduan berikut menunjukkan pengalaman pengguna untuk meluncurkan instance Microsoft SQL Server Standard Edition, tetapi pengalaman pengguna serupa untuk semua mesin yang didukung.

Untuk meluncurkan instans Amazon RDS

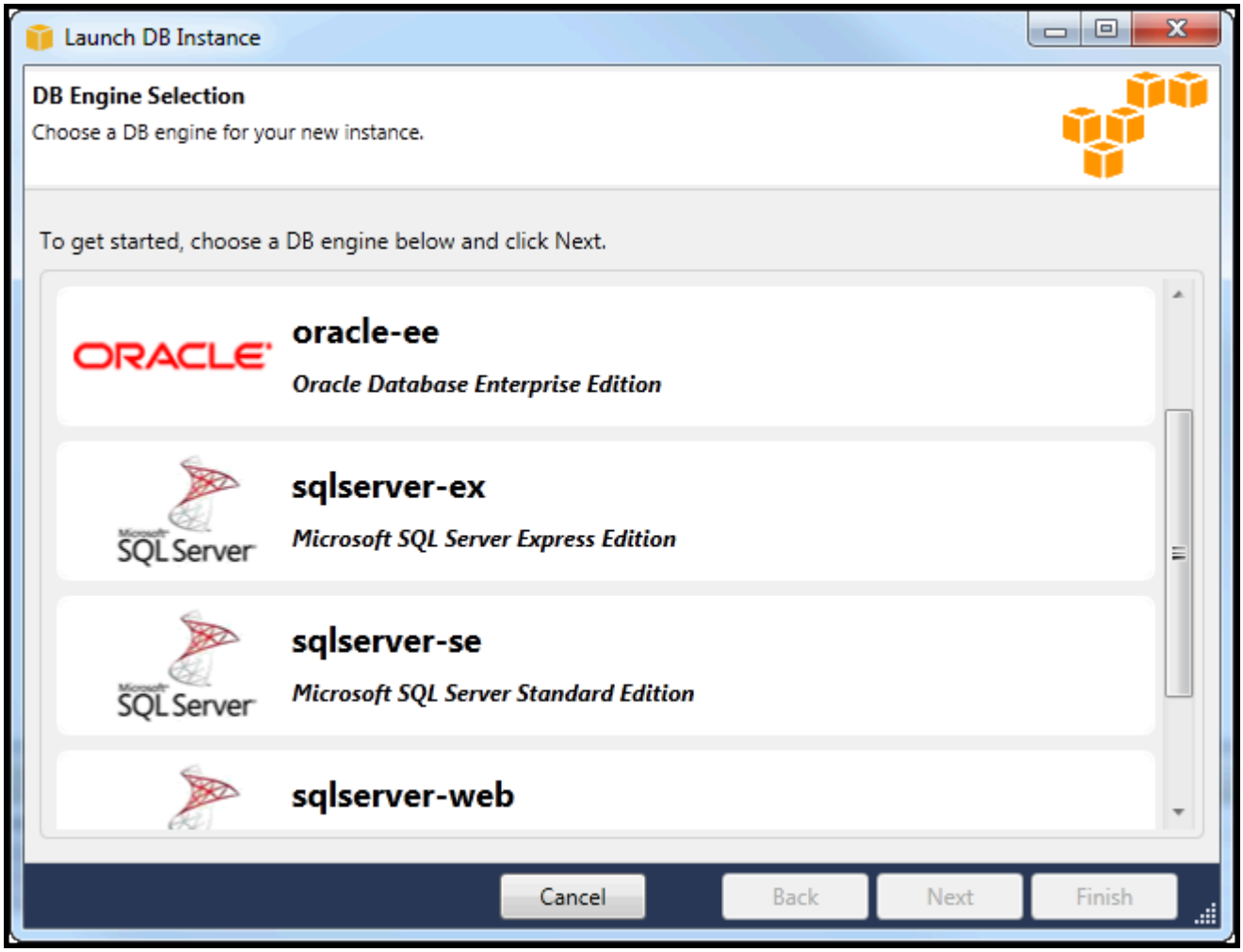
1. Di AWS Explorer, buka menu konteks (klik kanan) untuk node Amazon RDS dan pilih Launch DB Instance.



   Atau, pada tab Instans DB, pilih Luncurkan Instans DB.



2. Dalam kotak dialog DB Engine Selection, pilih jenis mesin database yang akan diluncurkan. Untuk panduan ini, pilih Microsoft SQL Server Standard Edition (sqlserver-se), lalu pilih Berikutnya.



3. Dalam kotak dialog Opsi Instans Mesin DB, pilih opsi konfigurasi.

   Di bagian DB Engine Instance Options and Class, Anda dapat menentukan pengaturan berikut.

   Model Lisensi

Tipe Mesin	Lisensi
Microsoft SQL Server	termasuk lisensi
MySql	general-public-license
Oracle	bring-your-own-license

   Model lisensi bervariasi, tergantung pada jenis mesin database. Lisensi Jenis Mesin Microsoft SQL Server termasuk lisensi Oracle MySql general-public-license bring-your-own-license

   Versi Instans DB

   Pilih versi mesin database yang ingin Anda gunakan. Jika hanya satu versi yang didukung, itu dipilih untuk Anda.

   Kelas Instans DB

   Pilih kelas instance untuk mesin database. Harga untuk kelas misalnya bervariasi. Untuk informasi selengkapnya, lihat Harga Amazon RDS.

   Lakukan penerapan multi AZ

   Pilih opsi ini untuk membuat penyebaran multi-AZ untuk meningkatkan daya tahan dan ketersediaan data. Amazon RDS menyediakan dan menyimpan salinan siaga database Anda di Availability Zone yang berbeda untuk failover otomatis jika terjadi pemadaman terjadwal atau tidak direncanakan. Untuk informasi tentang penetapan harga untuk penerapan Multi-AZ, lihat bagian harga di halaman detail Amazon RDS. Opsi ini tidak didukung untuk Microsoft SQL Server.

   Tingkatkan versi minor secara otomatis

   Pilih opsi ini untuk AWS secara otomatis melakukan pembaruan versi minor pada instans RDS Anda untuk Anda.

Di bagian RDS Database Instance, Anda dapat menentukan pengaturan berikut.

Penyimpanan yang Dialokasikan

Engine	Minimum (GB)	Maksimum (GB)
MySQL	5	1024
Oracle Enterprise Edition	10	1024
Microsoft SQL Server Edisi Ekspres	30	1024
Edisi Standar Microsoft SQL Server	250	1024
Microsoft SQL Server Edisi Web	30	1024

Minimum dan maksimum untuk penyimpanan yang dialokasikan tergantung pada jenis mesin database. Mesin Minimum (GB) Maksimum (GB) MySQL 5 1024 Oracle Enterprise Edition 10 1024 Microsoft SQL Server Express Edition 30 1024 Microsoft SQL Server Edisi Standar 250 1024 Microsoft SQL Server Web Edition 30 1024

Pengidentifikasi Instans DB

Tentukan nama untuk instance database. Nama ini tidak peka huruf besar/kecil. Ini akan ditampilkan dalam bentuk huruf kecil di AWS Explorer.

Nama Pengguna Master

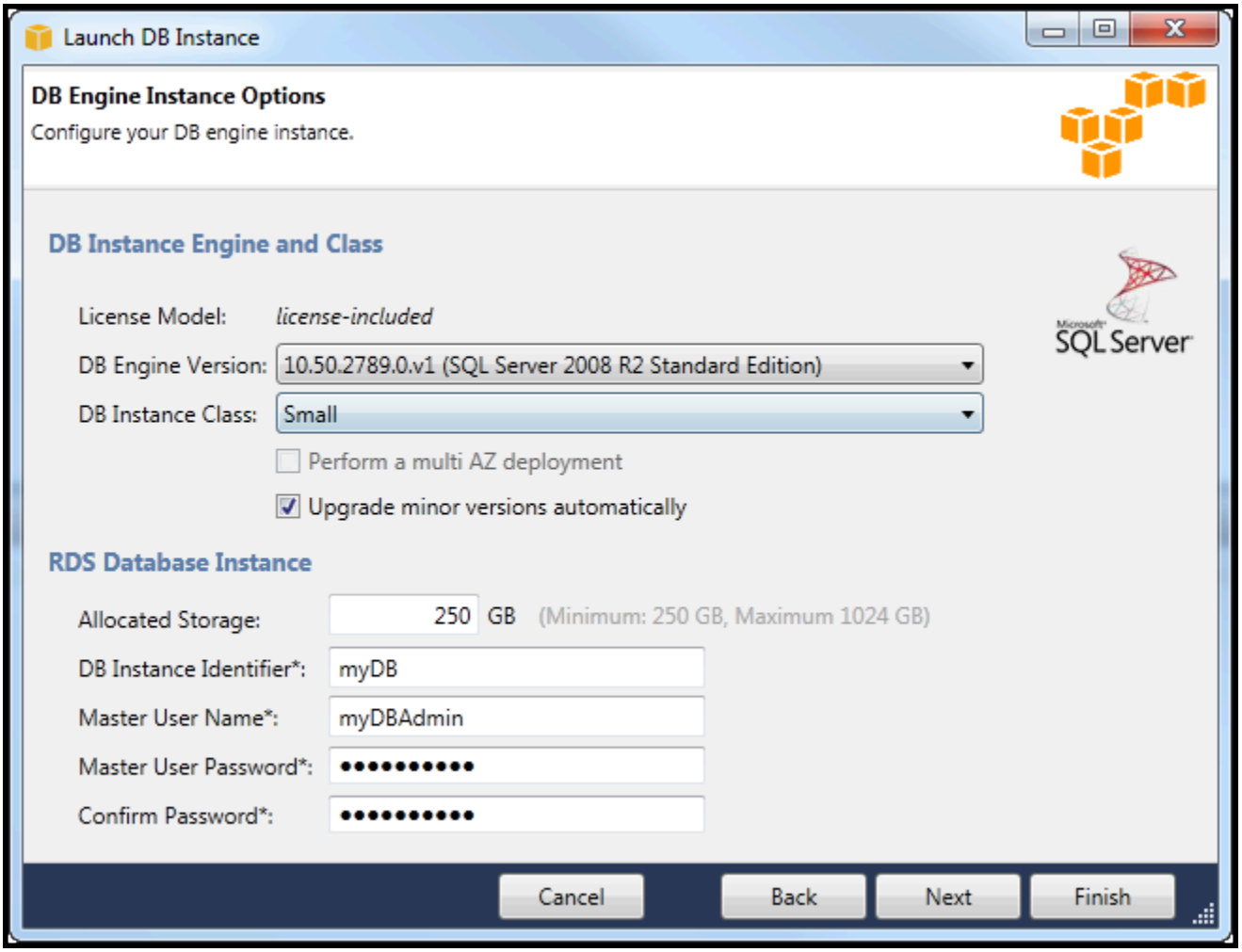
Ketik nama untuk administrator instance database.

Master Kata Sandi Pengguna

Ketik kata sandi untuk administrator instance database.

Konfirmasikan Kata Sandi

Ketik kata sandi lagi untuk memverifikasi itu benar.



1. Dalam Opsi Tambahan kotak dialog, Anda dapat menentukan pengaturan berikut.

   Pelabuhan Database

   Ini adalah port TCP yang akan digunakan instance untuk berkomunikasi di jaringan. Jika komputer Anda mengakses Internet melalui firewall, atur nilai ini ke port di mana firewall Anda memungkinkan lalu lintas.

   Zona Ketersediaan

   Gunakan opsi ini jika Anda ingin instance diluncurkan di Availability Zone tertentu di wilayah Anda. Instance database yang telah Anda tentukan mungkin tidak tersedia di semua Availability Zone di wilayah tertentu.

   Grup Keamanan RDS

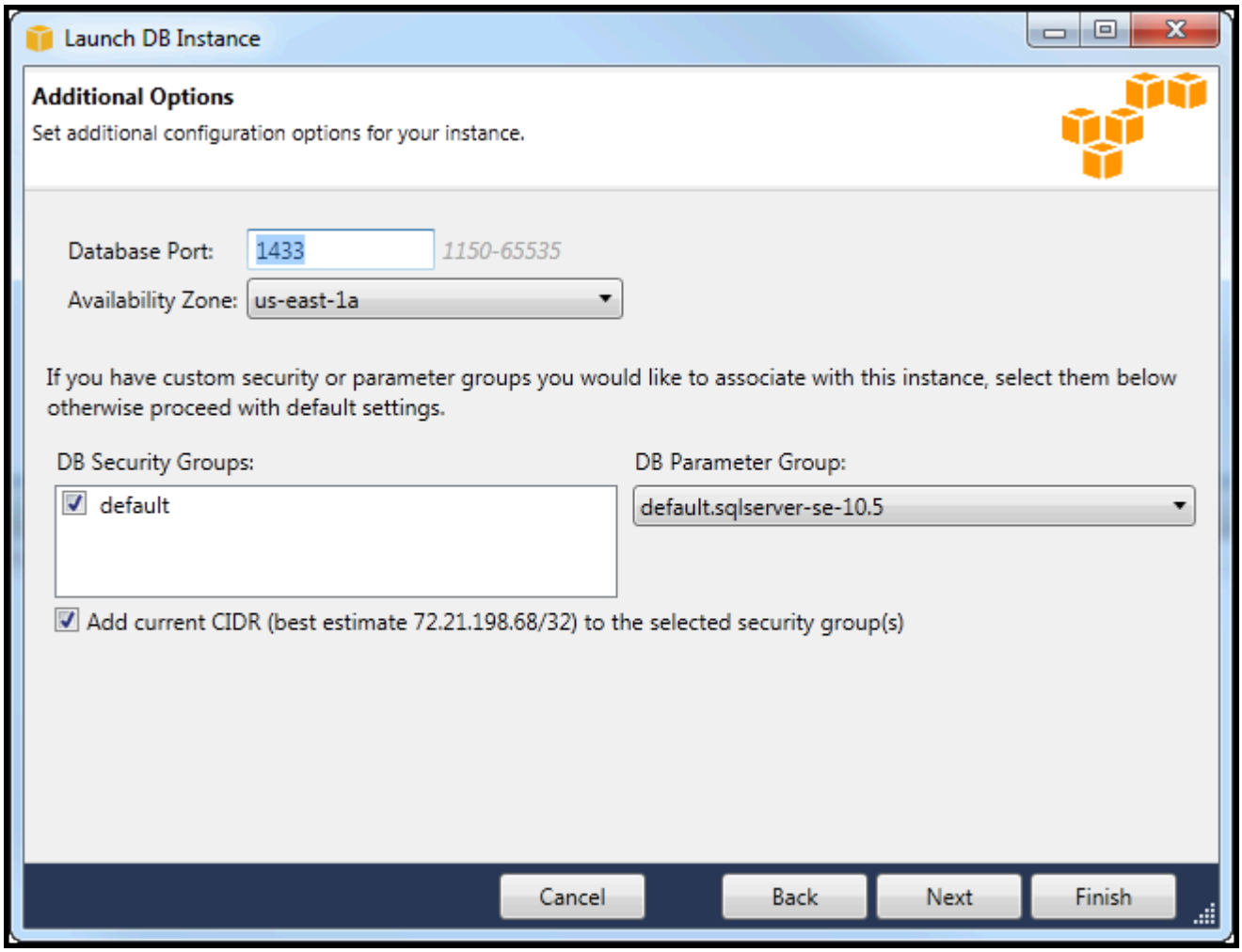
   Pilih grup keamanan RDS (atau grup) untuk dikaitkan dengan instans Anda. Grup keamanan RDS menentukan alamat IP, instans Amazon EC2, Akun AWS dan yang diizinkan untuk

mengakses instans Anda. Untuk informasi selengkapnya tentang grup keamanan RDS, lihat Grup Keamanan Amazon RDS. Toolkit for Visual Studio mencoba menentukan alamat IP Anda saat ini dan memberikan opsi untuk menambahkan alamat ini ke grup keamanan yang terkait dengan instans Anda. Namun, jika komputer Anda mengakses Internet melalui firewall, alamat IP yang dihasilkan Toolkit untuk komputer Anda mungkin tidak akurat. Untuk menentukan alamat IP mana yang akan digunakan, konsultasikan dengan administrator sistem Anda.

Grup Parameter DB

(Opsional) Dari daftar drop-down ini, pilih grup parameter DB untuk dikaitkan dengan instance Anda. Grup parameter DB memungkinkan Anda untuk mengubah konfigurasi default untuk instance. Untuk informasi lebih lanjut, buka Panduan Pengguna Layanan Amazon Relational Database Service dan artikel ini.

Ketika Anda telah menentukan pengaturan pada kotak dialog ini, pilih Berikutnya.

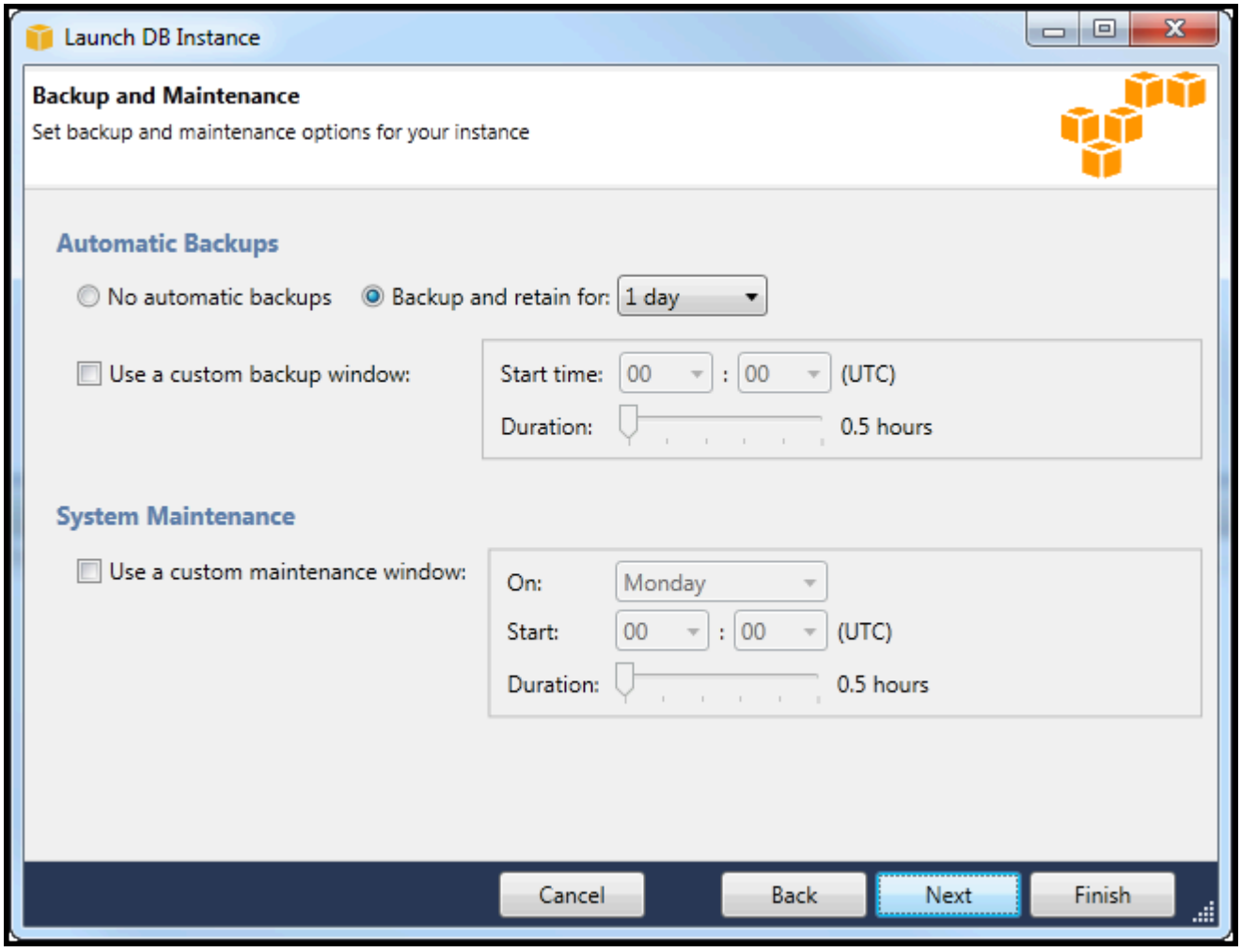


2. Kotak dialog Backup and Maintenance memungkinkan Anda menentukan apakah Amazon RDS harus mencadangkan instans Anda dan jika demikian, untuk berapa lama cadangan harus dipertahankan. Anda juga dapat menentukan jendela waktu di mana cadangan harus terjadi.

   Kotak dialog ini juga memungkinkan Anda menentukan apakah Anda ingin Amazon RDS melakukan pemeliharaan sistem pada instans Anda. Pemeliharaan mencakup tambalan rutin dan peningkatan versi minor.

   Jendela waktu yang Anda tentukan untuk pemeliharaan sistem tidak dapat tumpang tindih dengan jendela yang ditentukan untuk cadangan.

   Pilih Berikutnya.



3. Kotak dialog terakhir di wizard memungkinkan Anda untuk meninjau pengaturan untuk instance Anda. Jika Anda perlu mengubah pengaturan, gunakan tombol Kembali. Jika semua pengaturan sudah benar, pilih Luncurkan.

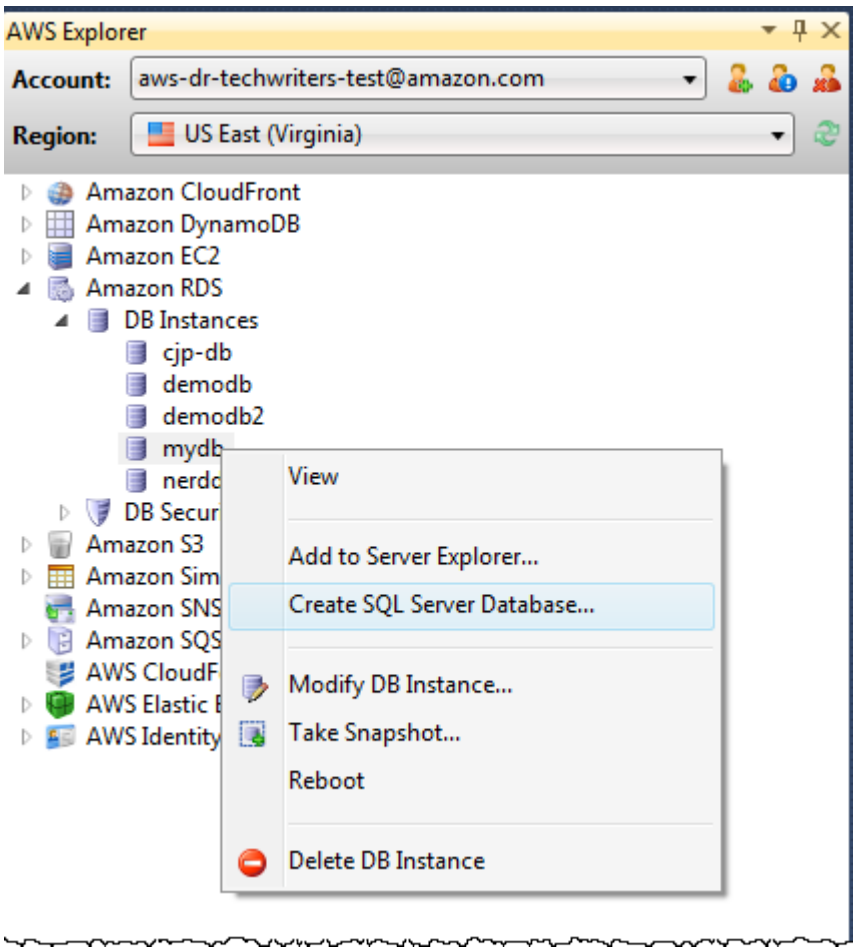
Buat Database Microsoft SQL Server dalam Instans RDS

Microsoft SQL Server dirancang sedemikian rupa sehingga, setelah meluncurkan instance Amazon RDS, Anda perlu membuat database SQL Server dalam instance RDS.

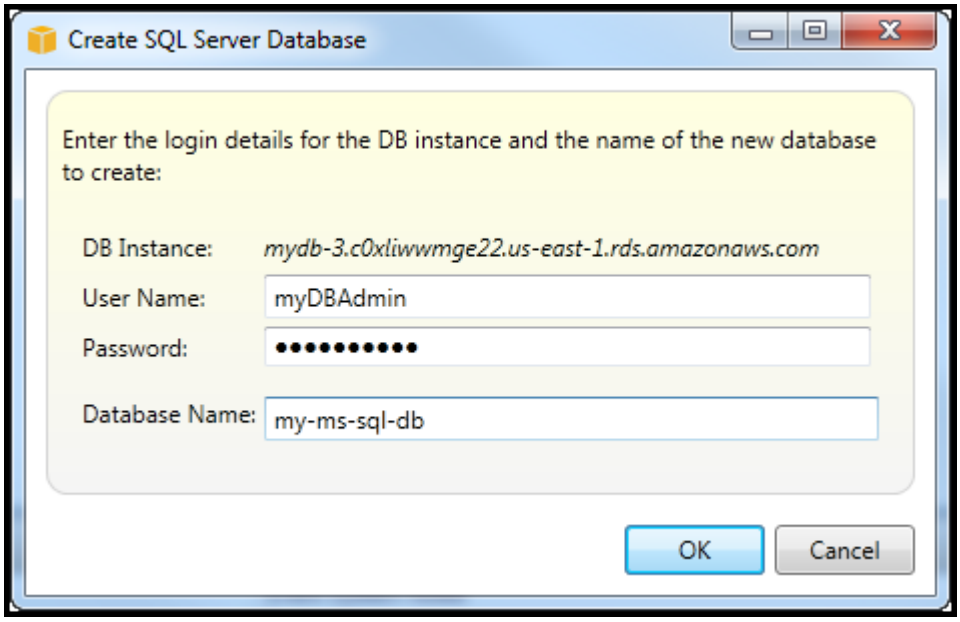
Untuk informasi tentang cara membuat instans Amazon RDS, lihat Meluncurkan Instans Basis Data Amazon RDS.

Untuk membuat database Microsoft SQL Server

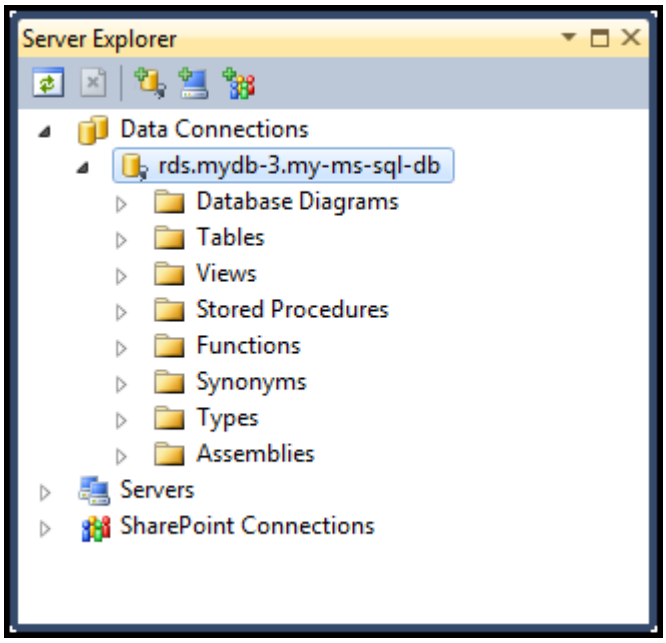
1. Di AWS Explorer, buka menu konteks (klik kanan) untuk node yang sesuai dengan instans RDS Anda untuk Microsoft SQL Server, dan pilih Buat Database SQL Server.



2. Dalam kotak dialog Buat Database SQL Server, ketik kata sandi yang Anda tentukan saat Anda membuat instance RDS, ketik nama untuk database Microsoft SQL Server, lalu pilih OK.



3. Toolkit for Visual Studio membuat database Microsoft SQL Server dan menambahkannya ke Visual Studio Server Explorer.



Grup Keamanan Amazon RDS

Grup keamanan Amazon RDS memungkinkan Anda mengelola akses jaringan ke instans Amazon RDS Anda. Dengan grup keamanan, Anda menentukan kumpulan alamat IP menggunakan notasi CIDR, dan hanya lalu lintas jaringan yang berasal dari alamat ini yang dikenali oleh instans Amazon RDS Anda.

Meskipun mereka berfungsi dengan cara yang sama, grup keamanan Amazon RDS berbeda dari grup keamanan Amazon EC2. Dimungkinkan untuk menambahkan grup keamanan EC2 ke grup keamanan RDS Anda. Setiap instans EC2 yang merupakan anggota grup keamanan EC2 kemudian dapat mengakses instans RDS yang merupakan anggota grup keamanan RDS.

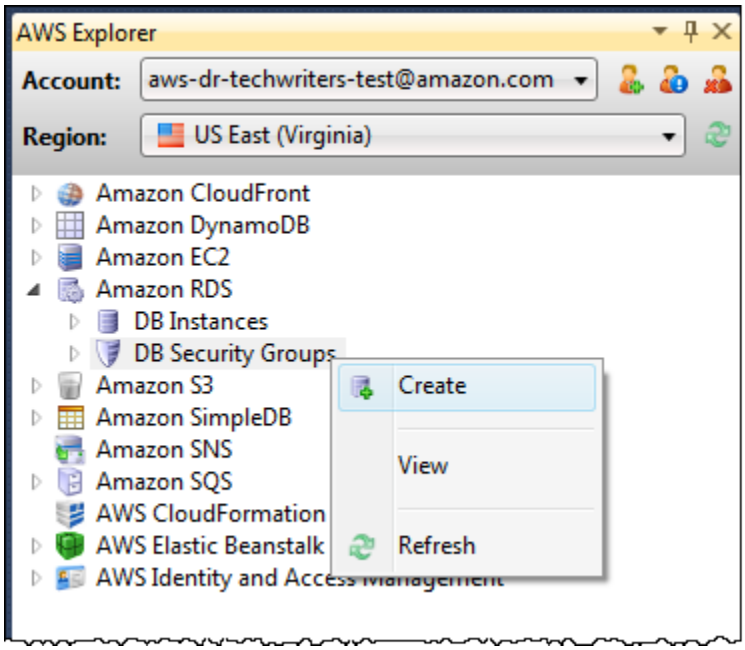
Untuk informasi selengkapnya tentang grup keamanan Amazon RDS, buka Grup Keamanan RDS. Untuk informasi selengkapnya tentang grup keamanan Amazon EC2, buka Panduan Pengguna EC2.

Buat Grup Keamanan Amazon RDS

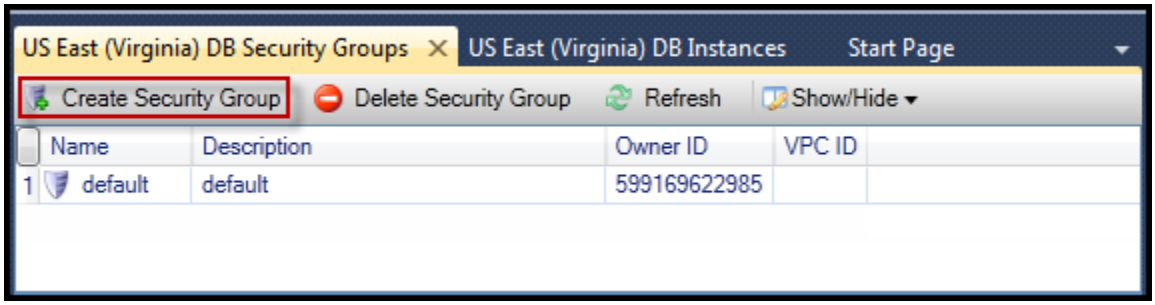
Anda dapat menggunakan Toolkit for Visual Studio untuk membuat grup keamanan RDS. Jika Anda menggunakan AWS Toolkit untuk meluncurkan instance RDS, wizard akan memungkinkan Anda untuk menentukan grup keamanan RDS untuk digunakan dengan instance Anda. Anda dapat menggunakan prosedur berikut untuk membuat grup keamanan tersebut sebelum memulai wizard.

Untuk membuat grup keamanan Amazon RDS

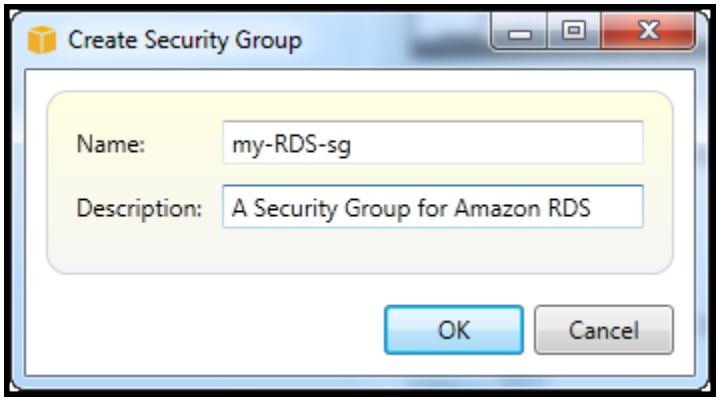
1. Di AWS Explorer, perluas node Amazon RDS, buka menu konteks (klik kanan) untuk subnode Grup Keamanan DB dan pilih Buat.



   Atau, pada tab Grup Keamanan, pilih Buat Grup Keamanan. Jika tab ini tidak ditampilkan, buka menu konteks (klik kanan) untuk subnode Grup Keamanan DB dan pilih Lihat.



2. Di kotak dialog Buat Grup Keamanan, ketik nama dan deskripsi untuk grup keamanan, lalu pilih OK.



Mengatur Izin Akses untuk Grup Keamanan Amazon RDS

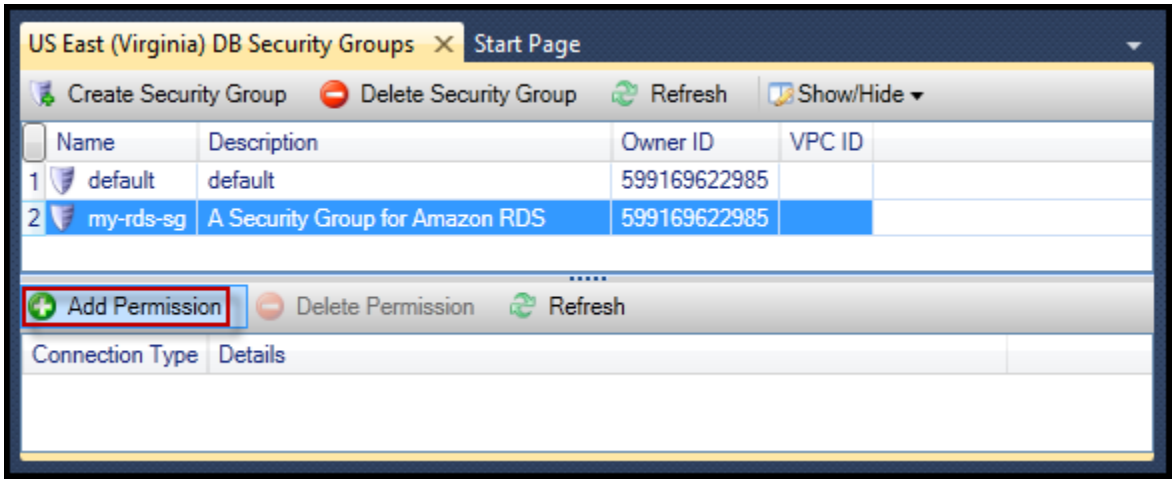
Secara default, grup keamanan Amazon RDS baru tidak menyediakan akses jaringan. Untuk mengaktifkan akses ke instans Amazon RDS yang menggunakan grup keamanan, gunakan prosedur berikut untuk mengatur izin aksesnya.

Untuk mengatur akses untuk grup keamanan Amazon RDS

1. Pada tab Grup Keamanan, pilih grup keamanan dari tampilan daftar. Jika grup keamanan Anda tidak muncul dalam daftar, pilih Segarkan. Jika grup keamanan Anda masih tidak muncul dalam daftar, pastikan Anda melihat daftar untuk AWS wilayah yang benar. Tab Grup Keamanan di AWS Toolkit bersifat spesifik wilayah.

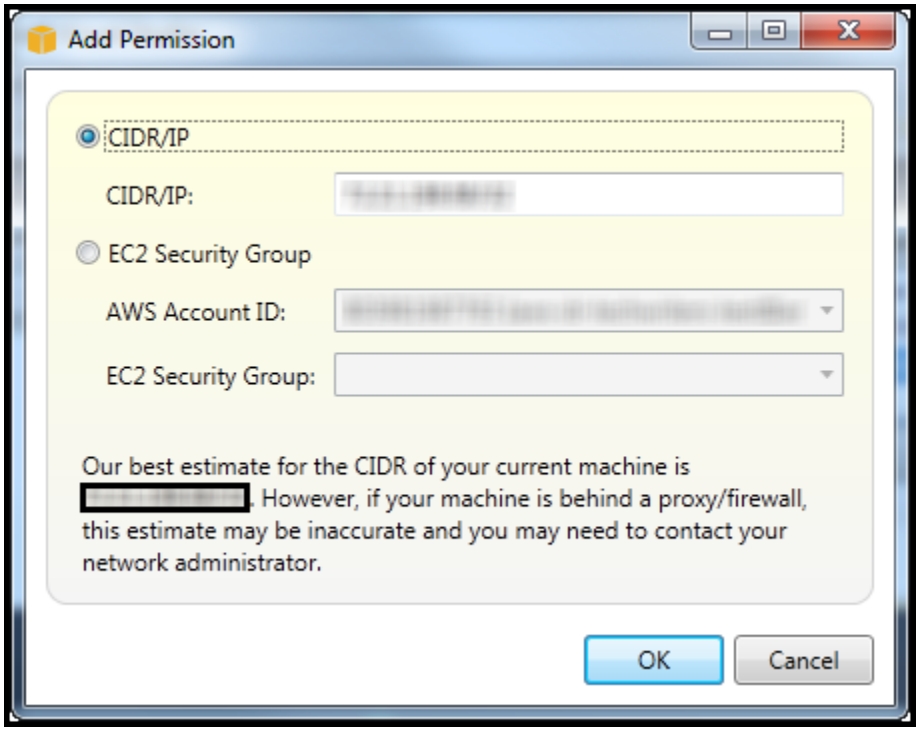
   Jika tidak ada tab Grup Keamanan yang muncul, di AWS Explorer, buka menu konteks (klik kanan) untuk subnode Grup Keamanan DB dan pilih Lihat.

2. Pilih Tambah Izin.



Tambahkan tombol Izin pada tab Grup Keamanan

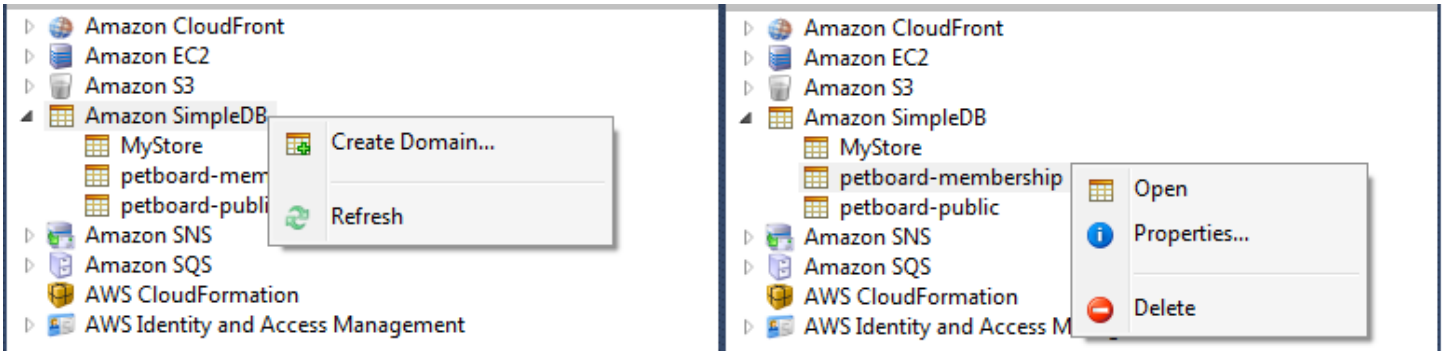
3. Dalam kotak dialog Tambah Izin, Anda dapat menggunakan notasi CIDR untuk menentukan alamat IP mana yang dapat mengakses instans RDS Anda, atau Anda dapat menentukan grup keamanan EC2 mana yang dapat mengakses instans RDS Anda. Ketika Anda memilih EC2 Security Group, Anda dapat menentukan akses untuk semua instans EC2 yang terkait dengan Akun AWS memiliki akses, atau Anda dapat memilih grup keamanan EC2 dari daftar drop-down.



AWS Toolkit mencoba menentukan alamat IP Anda dan mengisi kotak dialog secara otomatis dengan spesifikasi CIDR yang sesuai. Namun, jika komputer Anda mengakses Internet melalui firewall, CIDR yang ditentukan oleh Toolkit mungkin tidak akurat.

Menggunakan Amazon SimpleDB dari Explorer AWS

AWS Explorer menampilkan semua domain Amazon SimpleDB yang terkait dengan akun aktif. AWS Dari AWS Explorer, Anda dapat membuat atau menghapus domain Amazon SimpleDB.

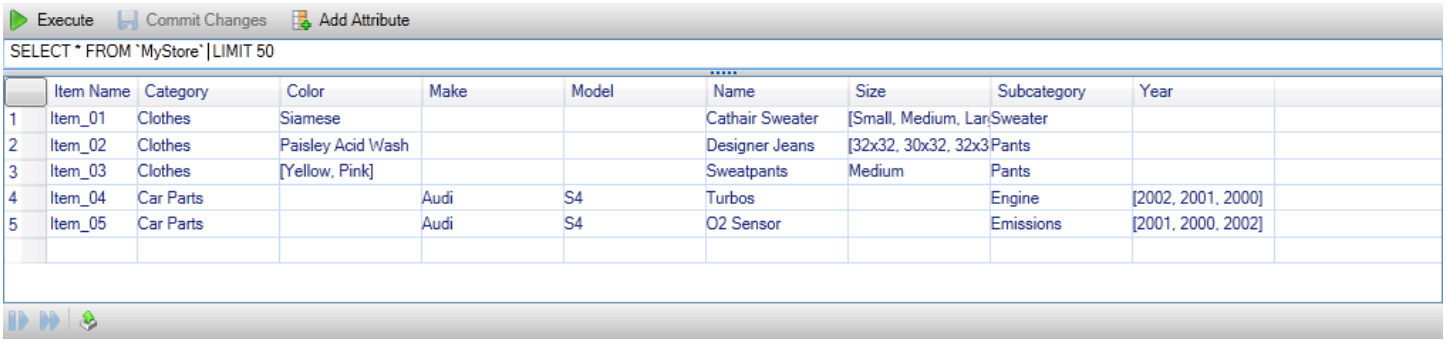


Create, delete, or open Amazon SimpleDB domains associated with your account

Menjalankan Query dan Mengedit Hasil

AWS Explorer juga dapat menampilkan tampilan kisi domain Amazon SimpleDB tempat Anda dapat melihat item, atribut, dan nilai di domain tersebut. Anda dapat menjalankan kueri sehingga hanya sebagian dari item domain yang ditampilkan. Dengan mengklik dua kali sel, Anda dapat mengedit nilai untuk atribut yang sesuai item tersebut. Anda juga dapat menambahkan atribut baru ke domain.

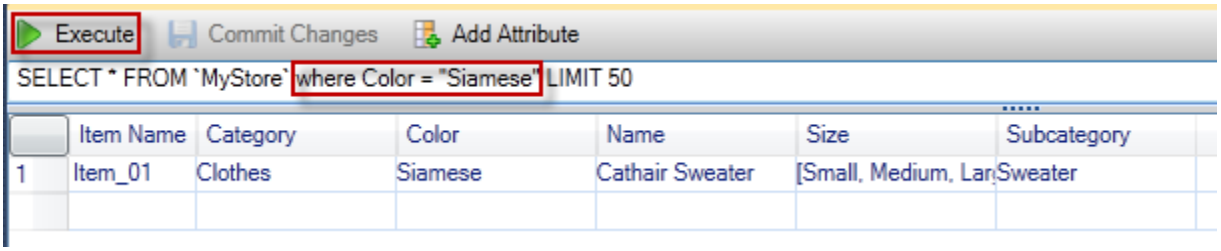
Domain yang ditampilkan di sini adalah dari sampel Amazon SimpleDB yang disertakan dengan file. AWS SDK untuk .NET



	Item Name	Category	Color	Make	Model	Name	Size	Subcategory	Year
1	Item_01	Clothes	Siamese			Cathair Sweater	[Small, Medium, Lar	Sweater	
2	Item_02	Clothes	Paisley Acid Wash			Designer Jeans	[32x32, 30x32, 32x3	Pants	
3	Item_03	Clothes	[Yellow, Pink]			Sweatpants	Medium	Pants	
4	Item_04	Car Parts		Audi	S4	Turbos		Engine	[2002, 2001, 2000]
5	Item_05	Car Parts		Audi	S4	O2 Sensor		Emissions	[2001, 2000, 2002]

Amazon SimpleDB grid view

Untuk menjalankan kueri, edit kueri di kotak teks di bagian atas tampilan kisi, lalu pilih Jalankan. Tampilan difilter untuk hanya menampilkan item yang cocok dengan kueri.

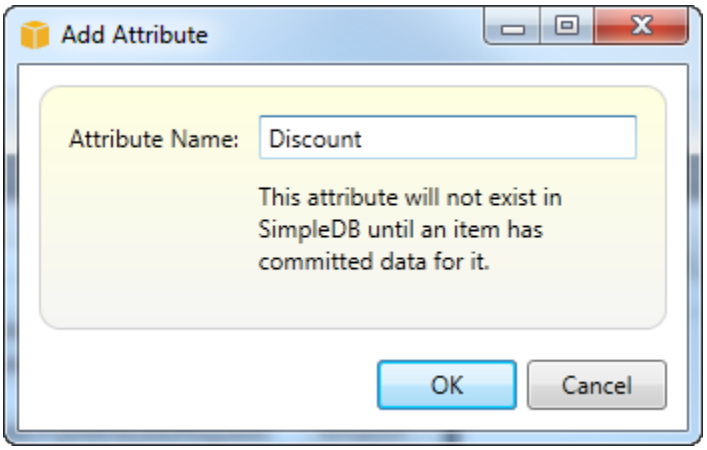


Execute query from AWS Explorer

Untuk mengedit nilai yang terkait dengan atribut, klik dua kali sel yang sesuai, edit nilainya, lalu pilih Komit Perubahan.

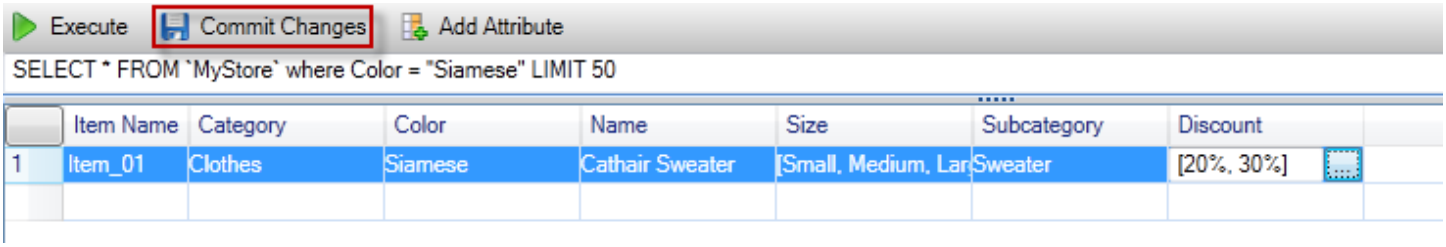
Menambahkan Atribut

Untuk menambahkan atribut, di bagian atas tampilan, pilih Tambah Atribut.



Tambahkan Atribut dialog box

Untuk membuat atribut bagian dari domain, Anda harus menambahkan nilai untuk itu ke setidaknya satu item dan kemudian memilih Commit Changes.



Commit changes for a new attribute

Hasil Kueri Paginating

Ada tiga tombol di bagian bawah tampilan.

Paginate and export buttons

Dua tombol pertama menyediakan pagination untuk hasil query. Untuk menampilkan halaman hasil tambahan, pilih tombol pertama. Untuk menampilkan sepuluh halaman hasil tambahan, pilih tombol kedua. Dalam konteks ini, halaman sama dengan 100 baris atau jumlah hasil yang ditentukan oleh nilai LIMIT, jika disertakan dalam kueri.

Ekspor ke CSV

Tombol terakhir mengekspor hasil saat ini ke file CSV.

Menggunakan Amazon SQS dari Explorer AWS

Amazon Simple Queue Service (Amazon Simple Queue Service) adalah layanan antrian fleksibel yang memungkinkan pengiriman pesan di antara berbagai proses eksekusi dalam aplikasi perangkat lunak. Antrian Amazon SQS terletak di AWS infrastruktur, tetapi proses yang mengirimkan pesan dapat ditemukan secara lokal, di instans Amazon EC2, atau pada beberapa kombinasi dari ini. Amazon SQS sangat ideal untuk mengoordinasikan distribusi pekerjaan di beberapa komputer.

Toolkit for Visual Studio memungkinkan Anda melihat antrian Amazon SQS yang terkait dengan akun aktif, membuat dan menghapus antrian, dan mengirim pesan melalui antrian. (Dengan akun aktif, yang kami maksud adalah akun yang dipilih di AWS Explorer.)

Untuk informasi selengkapnya tentang Amazon SQS, buka Pengantar SQS di dokumentasi. AWS

Membuat Antrian

Anda dapat membuat antrean Amazon SQS dari Explorer. AWS ARN dan URL untuk antrian akan didasarkan pada nomor akun untuk akun aktif dan nama antrian yang Anda tentukan saat pembuatan.

Untuk membuat antrian

1. Di AWS Explorer, buka menu konteks (klik kanan) untuk node Amazon SQS, lalu pilih Create Queue.
2. Dalam kotak dialog Buat Antrian, tentukan nama antrian, batas waktu visibilitas default, dan penundaan pengiriman default. Batas waktu visibilitas default dan penundaan pengiriman default

ditentukan dalam hitungan detik. Batas waktu visibilitas default adalah jumlah waktu pesan tidak akan terlihat oleh proses penerimaan potensial setelah proses tertentu memperoleh pesan. Penundaan pengiriman default adalah jumlah waktu dari saat pesan dikirim ke saat pertama kali terlihat oleh proses penerimaan potensial.

3. Pilih OK. Antrian baru akan muncul sebagai subnode di bawah node Amazon SQS.

Menghapus Antrian

Anda dapat menghapus antrian yang ada dari AWS Explorer. Jika Anda menghapus antrian, pesan apa pun yang terkait dengan antrian tidak lagi tersedia.

Untuk menghapus antrian

1. Di AWS Explorer, buka menu konteks (klik kanan) untuk antrian yang ingin Anda hapus, lalu pilih Hapus.

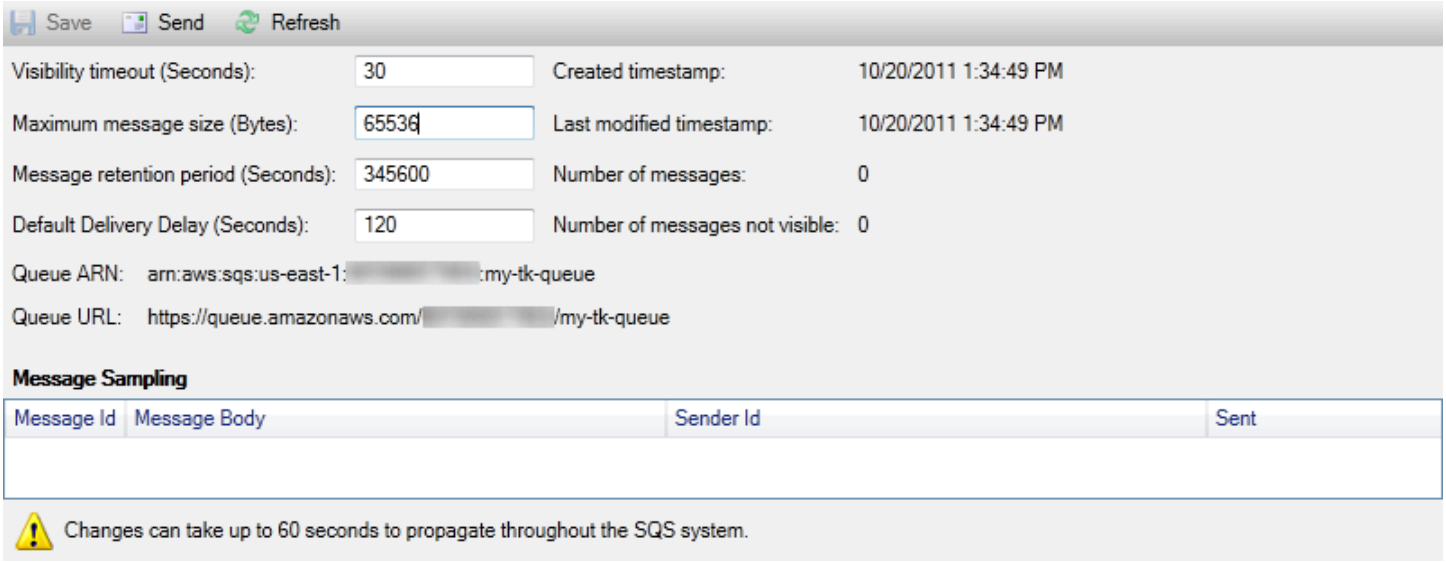
Mengelola Properti Antrian

Anda dapat melihat dan mengedit properti untuk antrian apa pun yang ditampilkan di AWS Explorer. Anda juga dapat mengirim pesan ke antrian dari tampilan properti ini.

Untuk mengelola properti antrian

- Di AWS Explorer, buka menu konteks (klik kanan) untuk antrian yang propertinya ingin Anda kelola, lalu pilih Lihat Antrian.

  Dari tampilan properti antrian, Anda dapat mengedit batas waktu visibilitas, ukuran pesan maksimum, periode penyimpanan pesan, dan penundaan pengiriman default. Penundaan pengiriman default dapat diganti saat Anda mengirim pesan. Pada tangkapan layar berikut, teks yang dikaburkan adalah komponen nomor akun dari antrian ARN dan URL.



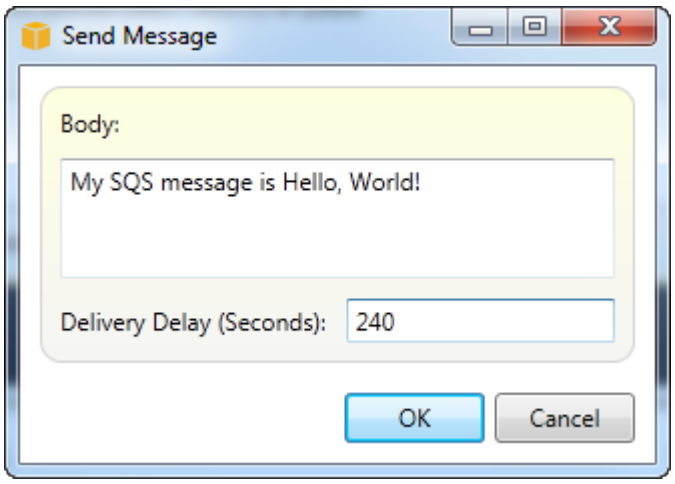
SQS queue properties view

Mengirim Pesan ke Antrian

Dari tampilan properti antrian, Anda dapat mengirim pesan ke antrian.

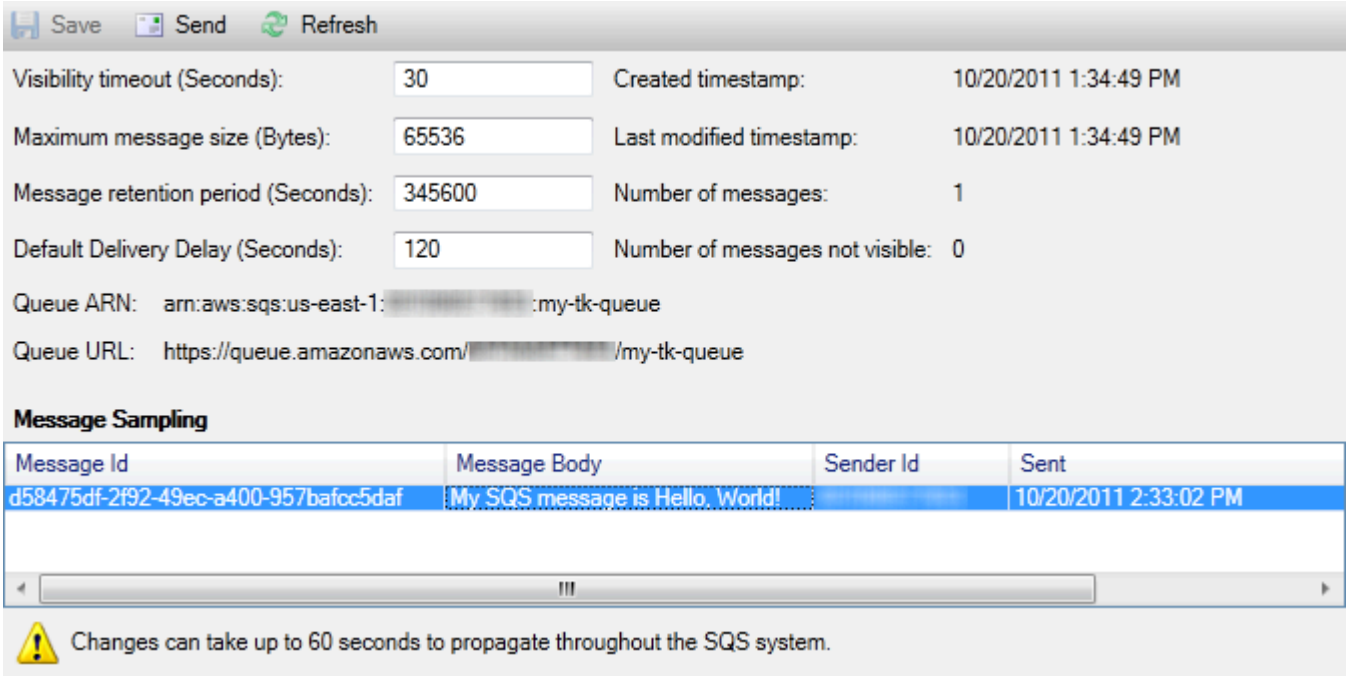
Untuk mengirim pesan

1. Di bagian atas tampilan properti antrian, pilih tombol Kirim.
2. Ketik pesan. (Opsional) Masukkan penundaan pengiriman yang akan mengganti penundaan pengiriman default untuk antrian. Dalam contoh berikut, kami telah mengganti penundaan dengan nilai 240 detik. Pilih OK.



Kirim Pesan dialog box

3. Tunggu sekitar 240 detik (empat menit). Pesan akan muncul di bagian Pengambilan Sampel Pesan dari tampilan properti antrian.



SQS properties view with sent message

Stempel waktu dalam tampilan properti antrian adalah waktu Anda memilih tombol Kirim. Itu tidak termasuk penundaan. Oleh karena itu, waktu pesan muncul dalam antrian dan tersedia untuk penerima mungkin lebih lambat dari stempel waktu ini. Stempel waktu ditampilkan dalam waktu lokal komputer Anda.

Identity and Access Management

AWS Identity and Access Management (IAM) memungkinkan Anda mengelola akses ke sumber daya Akun AWS dan Anda dengan lebih aman. Dengan IAM, Anda dapat membuat beberapa pengguna di primer Anda (root) Akun AWS. Pengguna ini dapat memiliki kredensialnya sendiri: kata sandi, ID kunci akses, dan kunci rahasia, tetapi semua pengguna IAM berbagi satu nomor akun.

Anda dapat mengelola tingkat akses sumber daya setiap pengguna IAM dengan melampirkan kebijakan IAM ke pengguna. Misalnya, Anda dapat melampirkan kebijakan ke pengguna IAM yang memberi pengguna akses ke layanan Amazon S3 dan sumber daya terkait di akun Anda, tetapi tidak menyediakan akses ke layanan atau sumber daya lain.

Untuk manajemen akses yang lebih efisien, Anda dapat membuat grup IAM, yang merupakan kumpulan pengguna. Ketika Anda melampirkan kebijakan ke grup, itu akan memengaruhi semua pengguna yang menjadi anggota grup tersebut.

Selain mengelola izin di tingkat pengguna dan grup, IAM juga mendukung konsep peran IAM. Seperti pengguna dan grup, Anda dapat melampirkan kebijakan ke peran IAM. Anda kemudian dapat mengaitkan peran IAM dengan instans Amazon EC2. Aplikasi yang berjalan pada instans EC2 dapat mengakses AWS menggunakan izin yang disediakan oleh peran IAM. Untuk informasi selengkapnya tentang penggunaan peran IAM dengan Toolkit, lihat Membuat Peran IAM. Untuk informasi lebih lanjut tentang IAM, buka Panduan Pengguna IAM.

Membuat dan Mengkonfigurasi Pengguna IAM

Pengguna IAM memungkinkan Anda untuk memberikan orang lain akses ke Anda Akun AWS. Karena Anda dapat melampirkan kebijakan ke pengguna IAM, Anda dapat dengan tepat membatasi sumber daya yang dapat diakses pengguna IAM dan operasi yang dapat mereka lakukan pada sumber daya tersebut.

Sebagai praktik terbaik, semua pengguna yang mengakses Akun AWS harus melakukannya sebagai pengguna IAM — bahkan pemilik akun. Ini memastikan bahwa jika kredensil untuk salah satu pengguna IAM dikompromikan, hanya kredensil tersebut yang dapat dinonaktifkan. Tidak perlu menonaktifkan atau mengubah kredensi root untuk akun tersebut.

Dari Toolkit for Visual Studio, Anda dapat menetapkan izin ke pengguna IAM baik dengan melampirkan kebijakan IAM ke pengguna atau dengan menetapkan pengguna ke grup. Pengguna IAM yang ditetapkan ke grup memperoleh izin mereka dari kebijakan yang dilampirkan ke grup. Untuk informasi selengkapnya, lihat Membuat Grup IAM dan Menambahkan Pengguna IAM ke Grup IAM.

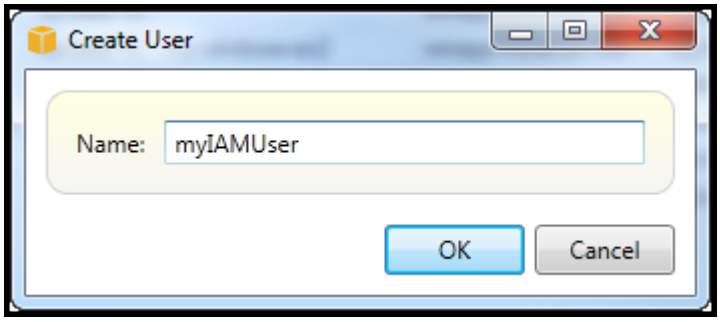
Dari Toolkit for Visual Studio, Anda juga dapat AWS menghasilkan kredensi (ID kunci akses dan kunci rahasia) untuk pengguna IAM. Untuk informasi selengkapnya, lihat Menghasilkan Kredensial untuk Pengguna IAM

Toolkit for Visual Studio mendukung menentukan kredensi pengguna IAM untuk mengakses layanan melalui Explorer. AWS Karena pengguna IAM biasanya tidak memiliki akses penuh ke semua Amazon Web Services, beberapa fungsi di AWS Explorer mungkin tidak tersedia. Jika Anda menggunakan AWS Explorer untuk mengubah sumber daya saat akun aktif adalah pengguna IAM dan kemudian mengalihkan akun aktif ke akun root, perubahan mungkin tidak terlihat sampai Anda menyegarkan tampilan di AWS Explorer. Untuk menyegarkan tampilan, pilih tombol refresh ().

Untuk informasi tentang cara mengkonfigurasi pengguna IAM dari Konsol Manajemen AWS, buka Bekerja dengan Pengguna dan Grup di Panduan Pengguna IAM.

Untuk membuat pengguna IAM

1. Di AWS Explorer, perluas AWS Identity and Access Managementnode, buka menu konteks (klik kanan) untuk Pengguna dan kemudian pilih Buat Pengguna.
2. Dalam kotak dialog Create User, ketik nama untuk pengguna IAM dan pilih OK. Ini adalah nama ramah IAM. Untuk informasi tentang batasan nama untuk pengguna IAM, buka Panduan Pengguna IAM.



Create an IAM user

Pengguna baru akan muncul sebagai subnode di bawah Pengguna di bawah AWS Identity and Access Managementnode.

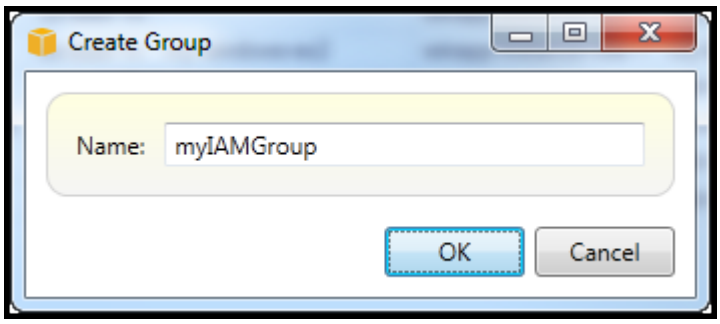
Untuk informasi tentang cara membuat kebijakan dan melampirkannya ke pengguna, lihat Membuat Kebijakan IAM.

Buat Grup IAM

Grup menyediakan cara untuk menerapkan kebijakan IAM ke kumpulan pengguna. Untuk informasi tentang cara mengelola pengguna dan grup IAM, buka Bekerja dengan Pengguna dan Grup di Panduan Pengguna IAM.

Untuk membuat grup IAM

1. Di AWS Explorer, di bawah Identity and Access Management, buka menu konteks (klik kanan) untuk Grup dan pilih Buat Grup.
2. Dalam kotak dialog Create Group, ketikkan nama untuk grup IAM dan pilih OK.



Create IAM group

Grup IAM baru akan muncul di bawah subnode Grup dari Identity and Access Management.

Untuk informasi tentang membuat kebijakan dan melampirkannya ke grup IAM, lihat Membuat Kebijakan IAM.

Menambahkan Pengguna IAM ke Grup IAM

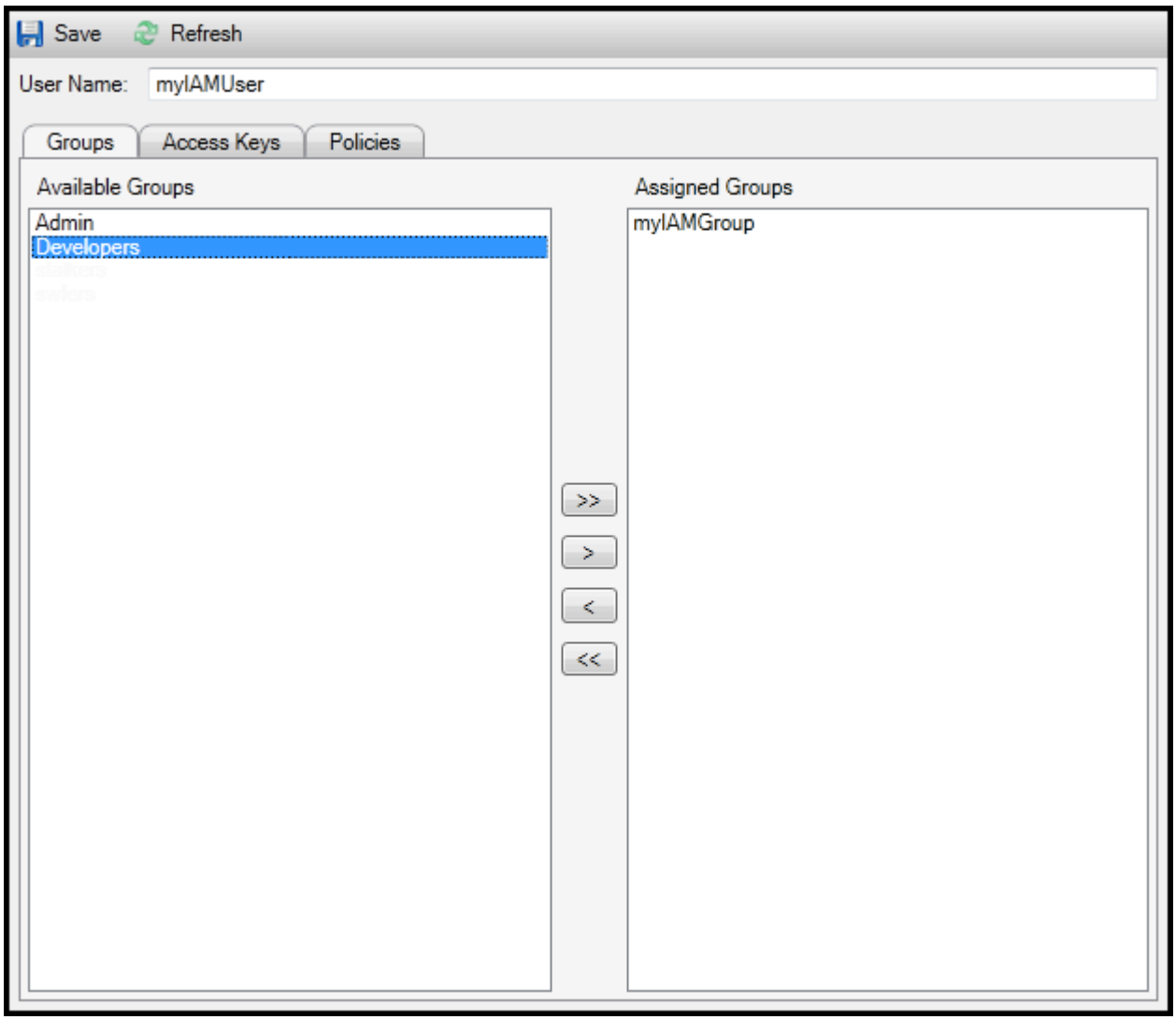
Pengguna IAM yang merupakan anggota grup IAM memperoleh izin akses dari kebijakan yang dilampirkan pada grup. Tujuan dari grup IAM adalah untuk membuatnya lebih mudah untuk mengelola izin di seluruh kumpulan pengguna IAM.

Untuk informasi tentang bagaimana kebijakan yang dilampirkan pada grup IAM berinteraksi dengan kebijakan yang dilampirkan pada pengguna IAM yang merupakan anggota grup IAM tersebut, buka Mengelola Kebijakan IAM di Panduan Pengguna IAM.

Di AWS Explorer, Anda menambahkan pengguna IAM ke grup IAM dari subnode Pengguna, bukan subnode Grup.

Untuk menambahkan pengguna IAM ke grup IAM

1. Di AWS Explorer, di bawah Identity and Access Management, buka menu konteks (klik kanan) untuk Pengguna dan pilih Edit.



Assign an IAM user to a IAM group

2. Panel kiri tab Grup menampilkan grup IAM yang tersedia. Panel kanan menampilkan grup yang pengguna IAM tertentu sudah menjadi anggota.

   Untuk menambahkan pengguna IAM ke grup, di panel kiri, pilih grup IAM dan kemudian pilih tombol **>**.

   Untuk menghapus pengguna IAM dari grup, di panel kanan, pilih grup IAM lalu pilih tombol **<**.

   Untuk menambahkan pengguna IAM ke semua grup IAM, pilih tombol **>>**. Demikian pula, untuk menghapus pengguna IAM dari semua grup, pilih tombol **<<**.

   Untuk memilih beberapa grup, pilih secara berurutan. Anda tidak perlu menahan tombol Control. Untuk menghapus grup dari pilihan Anda, cukup pilih untuk kedua kalinya.

3. Setelah selesai menetapkan pengguna IAM ke grup IAM, pilih Simpan.

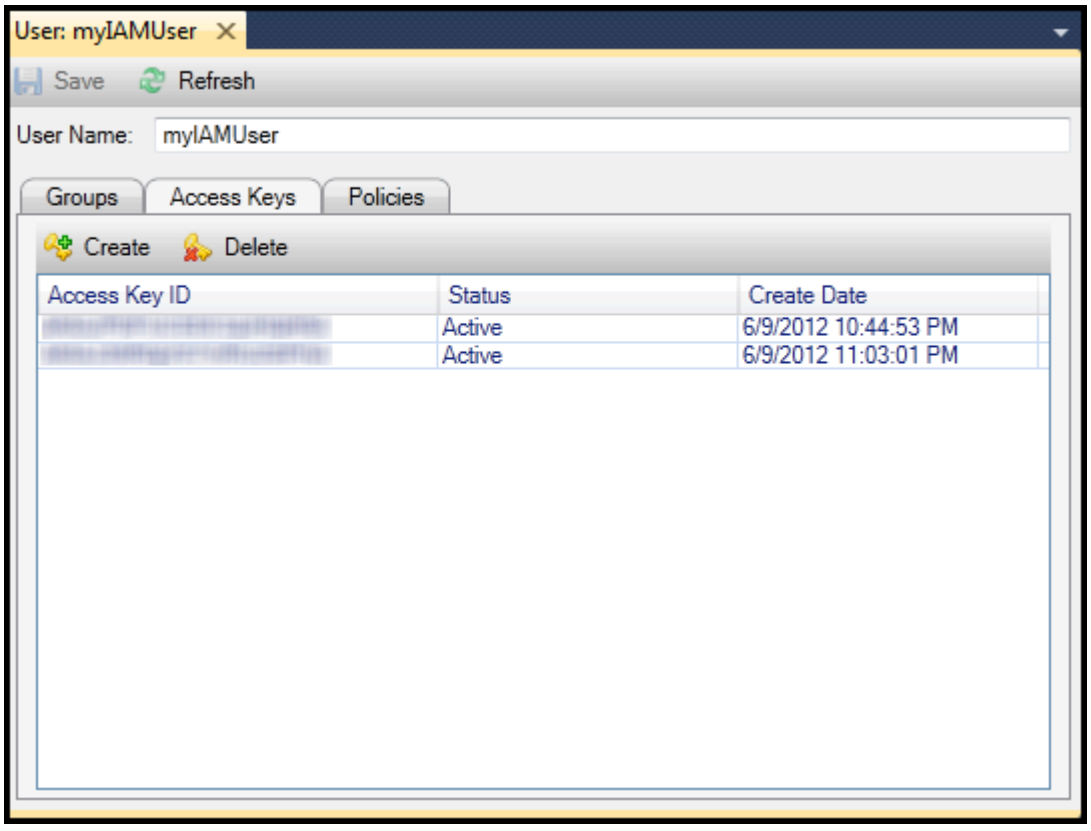
Menghasilkan Kredensi untuk Pengguna IAM

Dengan Toolkit for Visual Studio, Anda dapat menghasilkan ID kunci akses dan kunci rahasia yang digunakan untuk melakukan panggilan AWS API. Kunci ini juga dapat ditentukan untuk mengakses Amazon Web Services melalui Toolkit. Untuk informasi selengkapnya tentang cara menentukan kredensional untuk digunakan dengan Toolkit, lihat kredensi. Untuk informasi selengkapnya tentang cara menangani kredensil dengan aman, lihat Praktik Terbaik untuk Mengelola Kunci AWS Akses.

Toolkit tidak dapat digunakan untuk menghasilkan kata sandi untuk pengguna IAM.

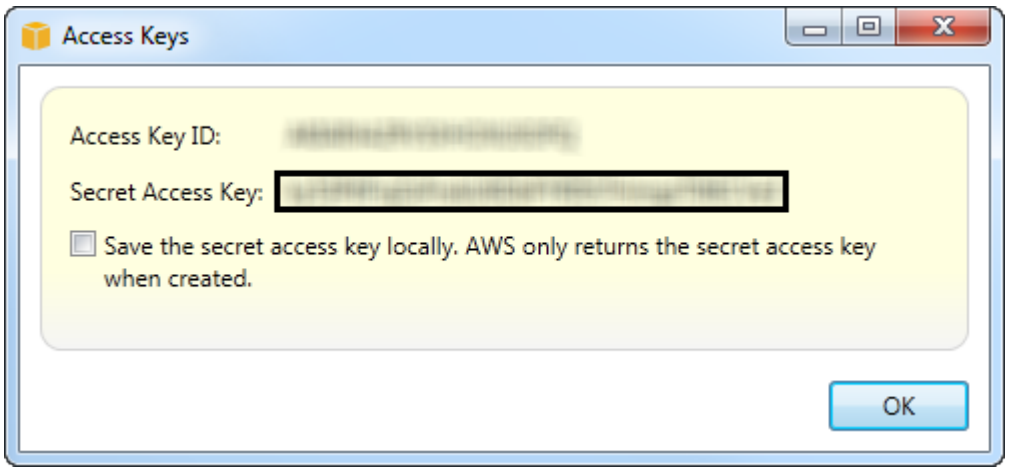
Untuk menghasilkan kredensi untuk pengguna IAM

1. Di AWS Explorer, buka menu konteks (klik kanan) untuk pengguna IAM dan pilih Edit.



2. Untuk menghasilkan kredensi, pada tab Kunci Akses, pilih Buat.

   Anda hanya dapat menghasilkan dua set kredensional per pengguna IAM. Jika Anda sudah memiliki dua set kredensil dan perlu membuat set tambahan, Anda harus menghapus salah satu set yang ada.

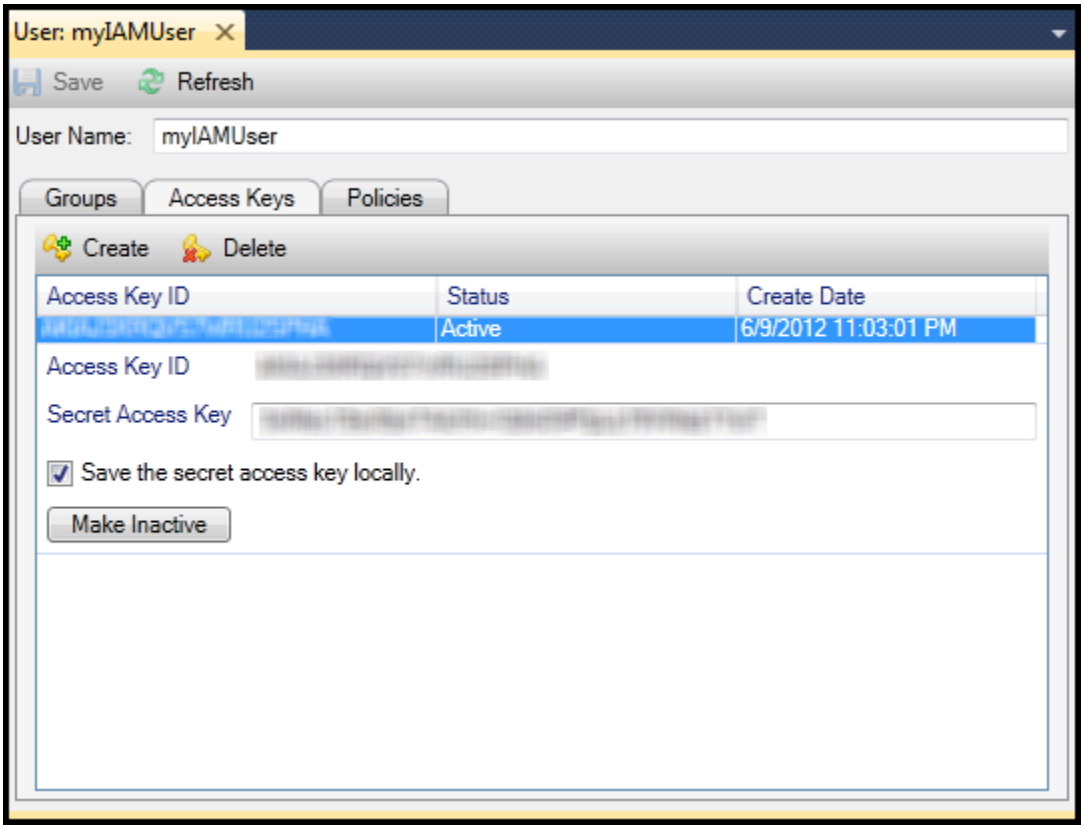


reate credentials for IAM user

Jika Anda ingin Toolkit menyimpan salinan terenkripsi kunci akses rahasia Anda ke drive lokal Anda, pilih Simpan kunci akses rahasia secara lokal. AWS hanya mengembalikan kunci akses rahasia saat dibuat. Anda juga dapat menyalin kunci akses rahasia dari kotak dialog dan menyimpannya di lokasi yang aman.

3. Pilih OK.

Setelah Anda membuat kredensialnya, Anda dapat melihatnya dari tab Kunci Akses. Jika Anda memilih opsi untuk meminta Toolkit menyimpan kunci rahasia secara lokal, itu akan ditampilkan di sini.



Create credentials for IAM user

Jika Anda menyimpan kunci rahasia sendiri dan juga ingin Toolkit menyimpannya, di kotak Kunci Akses Rahasia, ketik kunci akses rahasia, lalu pilih Simpan kunci akses rahasia secara lokal.

Untuk menonaktifkan kredensialnya, pilih Make Inactive. (Anda dapat melakukan ini jika Anda mencurigai kredensialnya telah disusupi. Anda dapat mengaktifkan kembali kredensialnya jika Anda menerima jaminan bahwa mereka aman.)

Buat IAM Role

Toolkit for Visual Studio mendukung pembuatan dan konfigurasi peran IAM. Sama seperti pengguna dan grup, Anda dapat melampirkan kebijakan ke peran IAM. Anda kemudian dapat mengaitkan peran IAM dengan instans Amazon EC2. Asosiasi dengan instans EC2 ditangani melalui profil instance, yang merupakan wadah logis untuk peran tersebut. Aplikasi yang berjalan pada instans EC2 secara otomatis diberikan tingkat akses yang ditentukan oleh kebijakan yang terkait dengan peran IAM. Ini benar bahkan ketika aplikasi belum menentukan AWS kredensi lain.

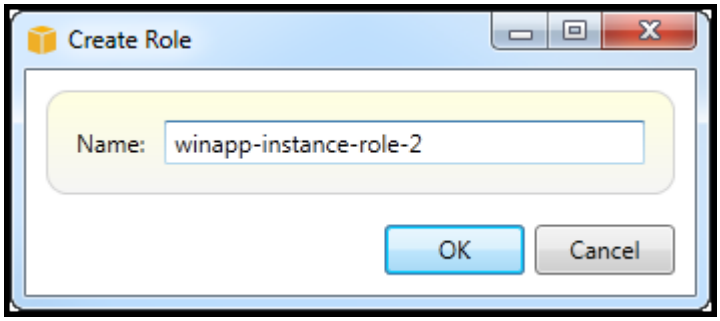
Misalnya, Anda dapat membuat peran dan melampirkan kebijakan ke peran tersebut yang membatasi akses ke Amazon S3 saja. Setelah mengaitkan peran ini dengan instans EC2, Anda kemudian dapat menjalankan aplikasi pada instance itu dan aplikasi akan memiliki akses ke Amazon

S3, tetapi tidak ke layanan atau sumber daya lain. Keuntungan dari pendekatan ini adalah bahwa Anda tidak perlu khawatir dengan aman mentransfer dan menyimpan AWS kredensional pada instans EC2.

Untuk informasi selengkapnya tentang peran IAM, buka Bekerja dengan Peran IAM di Panduan Pengguna IAM. Untuk contoh program yang mengakses AWS menggunakan peran IAM yang terkait dengan instans Amazon EC2, buka panduan pengembang untuk AWS Java, .NET, PHP , dan Ruby (Mengatur Kredensial Menggunakan IAM, Membuat Peran IAM, dan Bekerja dengan Kebijakan IAM).

Untuk membuat peran IAM

1. Di AWS Explorer, di bawah Identity and Access Management, buka menu konteks (klik kanan) untuk Peran dan kemudian pilih Buat Peran.
2. Di kotak dialog Create Role, ketikkan nama untuk peran IAM dan pilih OK.



Create IAM role

Peran IAM baru akan muncul di bawah Peran dalam Identity and Access Management.

Untuk informasi tentang cara membuat kebijakan dan melampirkannya ke peran, lihat Membuat Kebijakan IAM.

Buat Kebijakan IAM

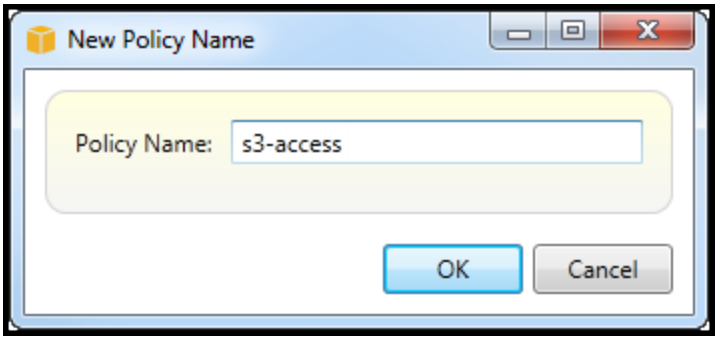
Kebijakan sangat mendasar bagi IAM. Kebijakan dapat dikaitkan dengan entitas IAM seperti pengguna, grup, atau peran. Kebijakan menentukan tingkat akses yang diaktifkan untuk pengguna, grup, atau peran.

Untuk membuat kebijakan IAM

Di AWS Explorer, perluas AWS Identity and Access Managementnode, lalu perluas node untuk jenis entitas (Grup, Peran, atau Pengguna) yang akan Anda lampirkan kebijakan. Misalnya, buka menu konteks untuk peran IAM dan pilih Edit.

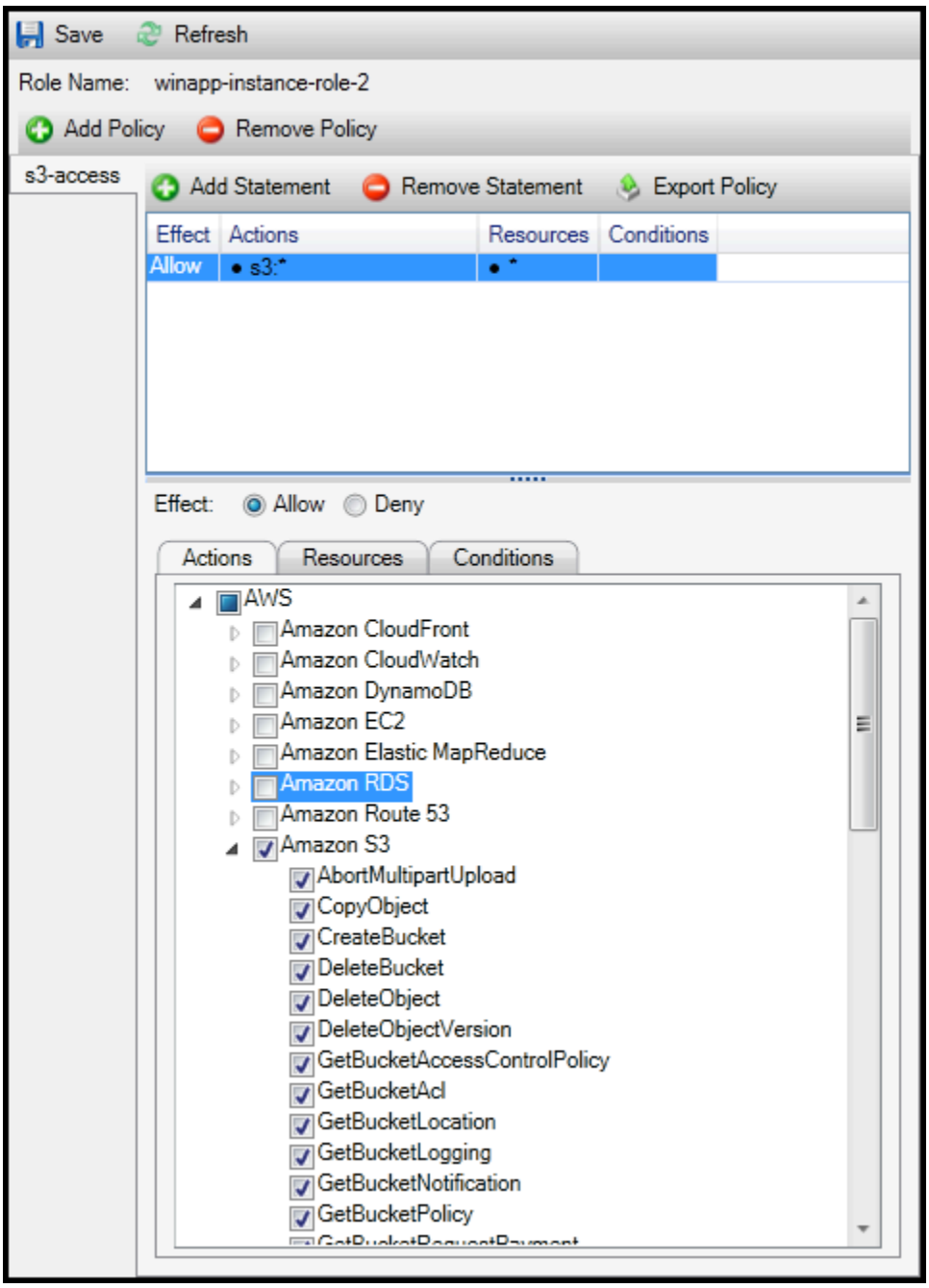
Tab yang terkait dengan peran akan muncul di AWS Explorer. Pilih tautan Tambahkan Kebijakan.

Di kotak dialog Nama Kebijakan Baru, ketikkan nama untuk kebijakan (misalnya, s3-access).



New Policy Name dialog box

Di editor kebijakan, tambahkan pernyataan kebijakan untuk menentukan tingkat akses yang akan diberikan ke peran (dalam contoh ini, winapp-instance-role -2 yang terkait dengan kebijakan. Dalam contoh ini, kebijakan menyediakan akses penuh ke Amazon S3, tetapi tidak ada akses ke sumber daya lain.



Specify IAM policy

Untuk kontrol akses yang lebih tepat, Anda dapat memperluas subnode di editor kebijakan untuk mengizinkan atau melarang tindakan yang terkait dengan Amazon Web Services.

Setelah Anda mengedit kebijakan, pilih tautan Simpan.

AWS Lambda

Kembangkan dan terapkan fungsi C# Lambda berbasis.NET Core Anda dengan. AWS Toolkit for Visual Studio AWS Lambda adalah layanan komputasi yang memungkinkan Anda menjalankan kode tanpa menyediakan atau mengelola server. Toolkit for Visual Studio AWS Lambda mencakup template proyek .NET Core untuk Visual Studio.

Untuk informasi selengkapnya AWS Lambda, lihat Panduan Pengembang AWS Lambda.

Untuk informasi selengkapnya tentang.NET Core, lihat panduan Microsoft. NET Core. Untuk prasyarat.NET Core dan petunjuk penginstalan untuk platform Windows, macOS, dan Linux, lihat .NET Core Downloads.

Topik berikut menjelaskan cara bekerja dengan AWS Lambda menggunakan Toolkit for Visual Studio.

Topik

- AWS Lambda Proyek Dasar
- AWS Lambda Proyek Dasar Membuat Gambar Docker
- Tutorial: Membangun dan Menguji Aplikasi Tanpa Server dengan AWS Lambda
- Tutorial: Membuat Aplikasi Amazon Rekognition Lambda
- Tutorial: Menggunakan Amazon Logging Frameworks dengan AWS Lambda untuk Membuat Log Aplikasi

AWS Lambda Proyek Dasar

Anda dapat membuat fungsi Lambda menggunakan template proyek Microsoft .NET Core, di file. AWS Toolkit for Visual Studio

Buat Proyek Lambda Inti Visual Studio .NET

Anda dapat menggunakan template dan cetak biru Lambda-Visual Studio untuk membantu mempercepat inisialisasi proyek Anda. Cetak biru Lambda berisi fungsi pra-tertulis yang menyederhanakan pembuatan fondasi proyek yang fleksibel.

> **Note**
>
> Layanan Lambda memiliki batas data pada jenis paket yang berbeda. Untuk informasi rinci tentang batas data, lihat topik kuota Lambda di Panduan Pengguna Lambda AWS .

Untuk membuat proyek Lambda di Visual Studio

1. Dari Visual Studio, perluas menu File, perluas Baru, lalu pilih Project.
2. Dari kotak dialog Proyek Baru, atur kotak drop-down Bahasa, Platform, dan Jenis proyek ke “Semua”, lalu ketik aws lambda di bidang Pencarian. Pilih template Proyek AWS Lambda (.NET Core - C #).
3. Di bidang Nama, masukkan**`AWSLambdaSample`**, tentukan lokasi file yang Anda inginkan, lalu pilih Buat untuk melanjutkan.
4. Dari halaman Select Blueprint, pilih cetak biru Fungsi Kosong, lalu pilih Selesai untuk membuat proyek Visual Studio.

Tinjau File Proyek

Ada dua file proyek untuk ditinjau: `aws-lambda-tools-defaults.json` dan`Function.cs`.

Contoh berikut menunjukkan `aws-lambda-tools-defaults.json` file, yang secara otomatis dibuat sebagai bagian dari proyek Anda. Anda dapat mengatur opsi build dengan menggunakan bidang dalam file ini.

> **Note**
>
> Template proyek di Visual Studio berisi banyak bidang yang berbeda, perhatikan hal-hal berikut:
>
> - function-handler: menentukan metode yang berjalan saat fungsi Lambda berjalan
> - Menentukan nilai di bidang functiontion-handler akan mengisi nilai tersebut di wizard Publish.
> - Jika Anda mengganti nama fungsi, kelas, atau perakitan maka Anda juga perlu memperbarui bidang yang sesuai dalam `aws-lambda-tools-defaults.json` file.

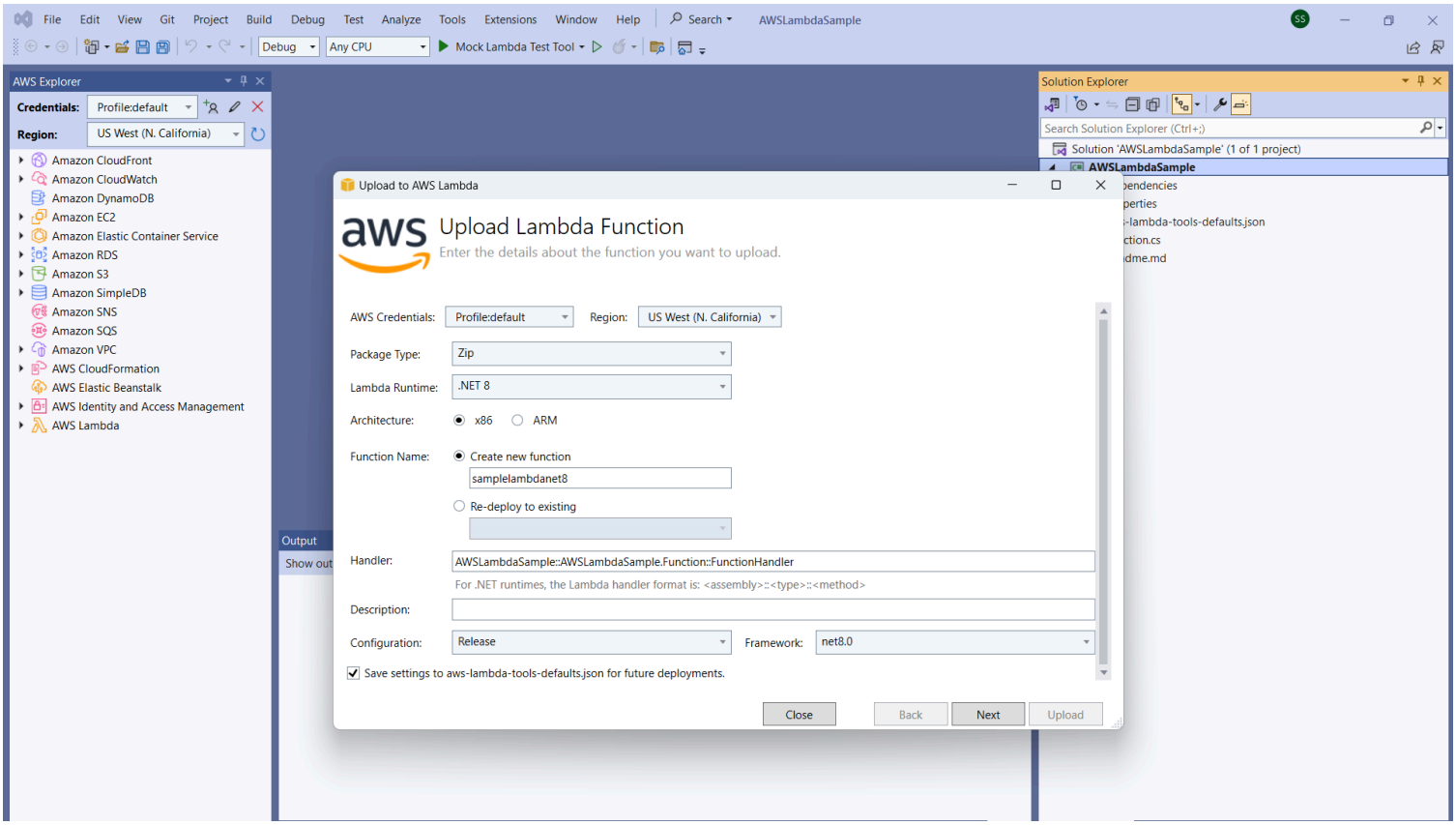
```
{
  "Information": [
    "This file provides default values for the deployment wizard inside Visual Studio
 and the AWS Lambda commands added to the .NET Core CLI.",
    "To learn more about the Lambda commands with the .NET Core CLI execute the
 following command at the command line in the project root directory.",
    "dotnet lambda help",
    "All the command line options for the Lambda command can be specified in this
 file."
  ],
  "profile": "default",
  "region": "us-west-2",
  "configuration": "Release",
  "function-architecture": "x86_64",
  "function-runtime": "dotnet8",
  "function-memory-size": 512,
  "function-timeout": 30,
  "function-handler": "AWSLambdaSample::AWSLambdaSample.Function::FunctionHandler"
}
```

Periksa `Function.cs` file. `Function.cs`mendefinisikan fungsi c # untuk mengekspos sebagai fungsi Lambda. Ini `FunctionHandler` adalah fungsi Lambda yang berjalan saat fungsi Lambda berjalan. Dalam proyek ini, ada satu fungsi yang didefinisikan:`FunctionHandler`, yang memanggil `ToUpper()` teks input.

Proyek Anda sekarang siap dipublikasikan ke Lambda.

Penerbitan ke Lambda

Prosedur dan gambar berikut menunjukkan cara mengunggah fungsi Anda ke Lambda menggunakan. AWS Toolkit for Visual Studio



Menerbitkan fungsi Anda ke Lambda

1. Arahkan ke AWS Explorer dengan memperluas View dan memilih AWS Explorer.
2. Di Solution Explorer, buka menu konteks untuk (klik kanan) proyek yang ingin Anda publikasikan, lalu pilih Publish to AWS Lambda untuk membuka jendela Upload Lambda Function.
3. Dari jendela Upload Lambda Function, lengkapi kolom berikut:

    a. Jenis Paket: Pilih**`Zip`**. File ZIP akan dibuat sebagai hasil dari proses pembuatan dan akan diunggah ke Lambda. Atau, Anda dapat memilih Package Type**`Image`**. Tutorial: Proyek Lambda Dasar Membuat Gambar Docker menjelaskan cara mempublikasikan menggunakan Package Type. **`Image`**

    b. Lambda Runtime: Pilih Lambda Runtime Anda dari menu tarik-turun.

    c. Arsitektur: Pilih radial untuk arsitektur pilihan Anda.

    d. Nama Fungsi: Pilih radial untuk Buat fungsi baru, lalu masukkan nama tampilan untuk instance Lambda Anda. Nama ini direferensikan oleh AWS Explorer dan Konsol Manajemen AWS display.

e. Handler: Gunakan bidang ini untuk menentukan fungsi handler. Sebagai contoh: **`AWSLambdaSample::AWSLambdaSample.Function::FunctionHandler`**.

f. (Opsional) Deskripsi: Masukkan teks deskriptif untuk ditampilkan dengan instance Anda, dari dalam. Konsol Manajemen AWS

g. Konfigurasi: Pilih konfigurasi pilihan Anda dari menu tarik-turun.

h. Framework: Pilih framework pilihan Anda dari menu drop-down.

i. Simpan pengaturan: Pilih kotak ini untuk menyimpan pengaturan Anda saat ini `aws-lambda-tools-defaults.json` sebagai default untuk penerapan masa depan.

j. Pilih Berikutnya untuk melanjutkan ke jendela Advanced Function Details.

4. Di jendela Advanced Function Details, lengkapi bidang-bidang berikut:

a. Nama Peran: Pilih peran yang terkait dengan akun Anda. Peran ini menyediakan kredensi sementara untuk setiap panggilan AWS layanan yang dibuat oleh kode dalam fungsi. Jika Anda tidak memiliki peran, gulir untuk menemukan Peran Baru berdasarkan Kebijakan AWS Terkelola di pemilih tarik-turun, lalu pilih. AWSLambdaBasicExecutionRole Peran ini memiliki izin akses minimal.

> **Note**
>
> Akun Anda harus memiliki izin untuk menjalankan ListPolicies tindakan IAM, atau daftar Nama Peran akan kosong dan Anda tidak akan dapat melanjutkan.

b. (Opsional) Jika fungsi Lambda Anda mengakses sumber daya di VPC Amazon, pilih subnet dan grup keamanan.

c. (Opsional) Tetapkan variabel lingkungan apa pun yang dibutuhkan fungsi Lambda Anda. Kunci secara otomatis dienkripsi oleh kunci layanan default yang gratis. Atau, Anda dapat menentukan AWS KMS kunci, yang dikenakan biaya. KMS adalah layanan terkelola yang dapat Anda gunakan untuk membuat dan mengontrol kunci enkripsi yang digunakan untuk mengenkripsi data Anda. Jika Anda memiliki AWS KMS kunci, Anda dapat memilihnya dari daftar.

5. Pilih Upload untuk membuka jendela Uploading Function dan memulai proses upload.

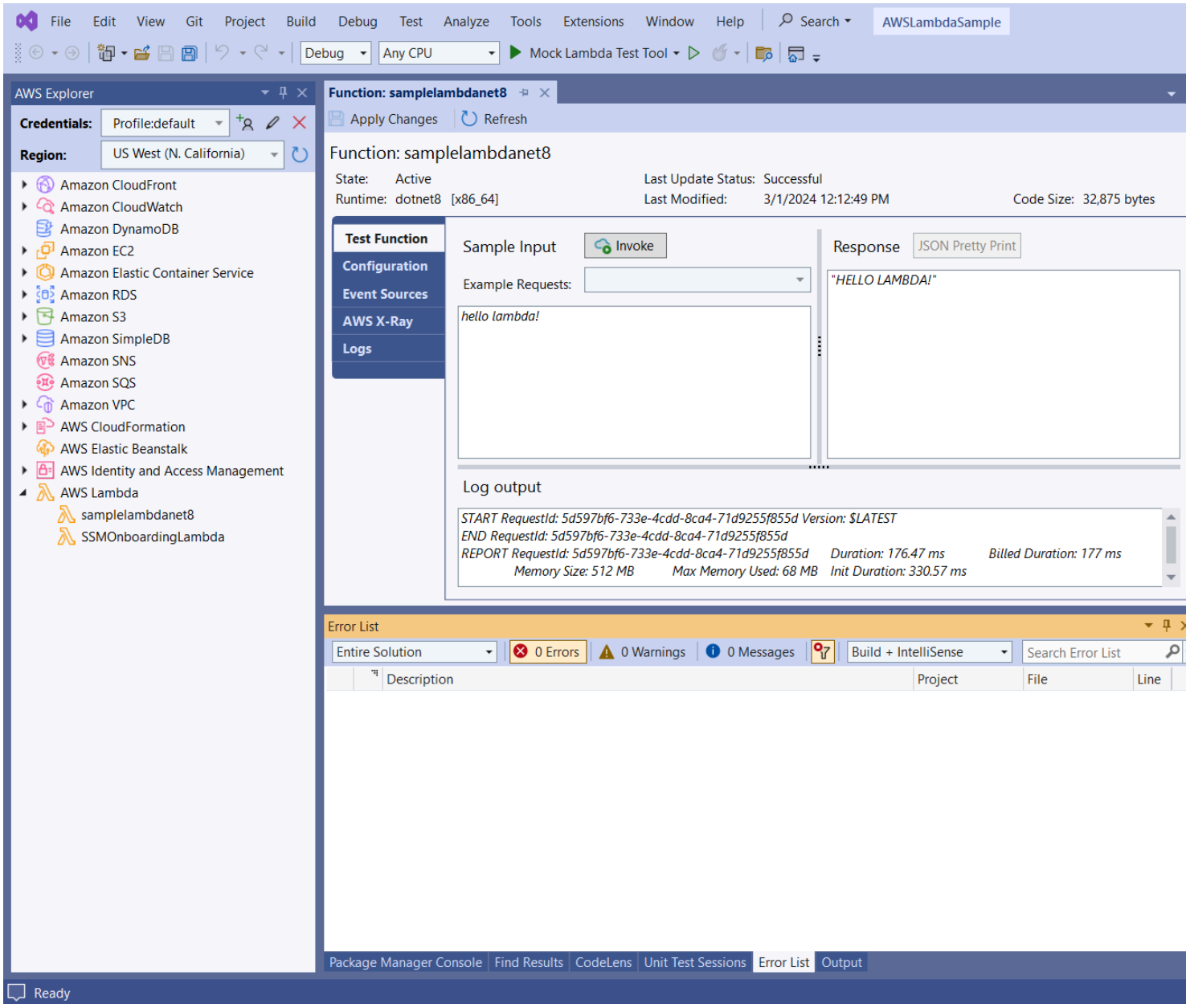
> **Note**
>
> Halaman Uploading Function ditampilkan saat fungsi diunggah ke. AWS Agar wizard tetap terbuka setelah mengunggah sehingga Anda dapat melihat laporan, hapus Wizard

tutup otomatis jika berhasil diselesaikan di bagian bawah formulir sebelum unggahan selesai.

Setelah fungsi diunggah, fungsi Lambda Anda aktif. Halaman Function: view terbuka dan menampilkan konfigurasi fungsi Lambda baru Anda.

6. Dari tab Test Function, masukkan `hello lambda!` di kolom input teks dan kemudian pilih Invoke untuk memanggil fungsi Lambda Anda secara manual. Teks Anda muncul di tab Respons, dikonversi ke huruf besar.

> **Note**
>
> Anda dapat membuka kembali tampilan Function: kapan saja dengan mengklik dua kali pada instance yang digunakan yang terletak di AWS Explorer di bawah node. AWS Lambda



7. (Opsional) Untuk mengonfirmasi bahwa Anda berhasil mempublikasikan fungsi Lambda Anda, masuk ke Konsol Manajemen AWS dan kemudian pilih Lambda. Konsol menampilkan semua fungsi Lambda yang Anda publikasikan, termasuk yang baru saja Anda buat.

Membersihkan

Jika Anda tidak akan terus mengembangkan dengan contoh ini, hapus fungsi yang Anda terapkan sehingga Anda tidak ditagih untuk sumber daya yang tidak digunakan di akun Anda.

> **Note**
>
> Lambda secara otomatis memantau fungsi Lambda untuk Anda, melaporkan metrik melalui Amazon. CloudWatch Untuk memantau dan memecahkan masalah fungsi Anda, lihat topik Pemecahan Masalah dan Pemantauan AWS Fungsi Lambda dengan Amazon CloudWatch di Panduan Pengembang. AWS Lambda

Untuk menghapus fungsi Anda

1. Dari AWS Explorer memperluas AWS Lambdanode.
2. Klik kanan instance yang Anda gunakan, lalu pilih Hapus.

AWS Lambda Proyek Dasar Membuat Gambar Docker

Anda dapat menggunakan Toolkit for Visual Studio untuk menyebarkan fungsi AWS Lambda Anda sebagai image Docker. Menggunakan Docker, Anda memiliki kontrol lebih besar atas runtime Anda. Misalnya, Anda dapat memilih runtime kustom seperti.NET 8.0. Anda menerapkan gambar Docker Anda dengan cara yang sama seperti gambar kontainer lainnya. Tutorial ini sangat mirip dengan Tutorial: Proyek Lambda Dasar, dengan dua perbedaan:

- Sebuah Dockerfile disertakan dalam proyek.
- Konfigurasi penerbitan alternatif dipilih.

Untuk informasi tentang gambar kontainer Lambda, lihat Paket Penerapan Lambda di Panduan Pengembang.AWS Lambda

Untuk informasi tambahan tentang bekerja dengan Lambda AWS Toolkit for Visual Studio, lihat Menggunakan AWS Lambda Template dalam AWS Toolkit for Visual Studio topik di Panduan Pengguna ini.

Buat Proyek Lambda Inti Visual Studio .NET

Anda dapat menggunakan template dan cetak biru Lambda Visual Studio untuk membantu mempercepat inisialisasi proyek Anda. Cetak biru Lambda berisi fungsi pra-tertulis yang menyederhanakan pembuatan fondasi proyek yang fleksibel.

Untuk membuat proyek Visual Studio .NET Core Lambda

1. Dari Visual Studio, perluas menu File, perluas Baru, lalu pilih Project.
2. Dari kotak dialog Proyek Baru, atur kotak drop-down Bahasa, Platform, dan Jenis proyek ke “Semua”, lalu ketik **`aws lambda`** di bidang Pencarian. Pilih template Proyek AWS Lambda (.NET Core - C #).
3. Di bidang Nama Proyek, masukkan**`AWSLambdaDocker`**, tentukan lokasi file Anda, lalu pilih Buat.
4. Pada halaman Select Blueprint, pilih blueprint .NET 8 (Container Image), lalu pilih Finish untuk membuat proyek Visual Studio. Anda sekarang dapat meninjau struktur dan kode proyek.

Meninjau File Proyek

Bagian berikut memeriksa tiga file proyek yang dibuat oleh cetak biru.NET 8 (Container Image):

1. `Dockerfile`
2. `aws-lambda-tools-defaults.json`
3. `Function.cs`

1. Dockerfile

A `Dockerfile` melakukan tiga tindakan utama:

- `FROM`: Menetapkan gambar dasar untuk digunakan untuk gambar ini. Gambar dasar ini menyediakan .NET Runtime, Lambda runtime, dan skrip shell yang menyediakan titik masuk untuk proses Lambda.NET.
- `WORKDIR`: Menetapkan direktori kerja internal gambar sebagai`/var/task`.
- `COPY`: Akan menyalin file yang dihasilkan dari proses pembuatan dari lokasi lokalnya ke direktori kerja gambar.

Berikut ini adalah `Dockerfile` tindakan opsional yang dapat Anda tentukan:

- `ENTRYPOINT`: Gambar dasar sudah menyertakan`ENTRYPOINT`, yang merupakan proses start-up yang dijalankan saat gambar dimulai. Jika Anda ingin menentukan sendiri, maka Anda mengesampingkan titik masuk dasar itu.

- `CMD`: Menginstruksikan kode kustom AWS mana yang ingin Anda eksekusi. Ini mengharapkan nama yang sepenuhnya memenuhi syarat untuk metode kustom Anda. Baris ini perlu disertakan langsung di Dockerfile atau dapat ditentukan selama proses publikasi.

```
# Example of alternative way to specify the Lambda target method rather than during
 the publish process.
CMD [ "AWSLambdaDocker::AWSLambdaDocker.Function::FunctionHandler"]
```

Berikut ini adalah contoh dari Dockerfile yang dibuat oleh cetak biru.NET 8 (Container Image).

```
FROM public.ecr.aws/lambda/dotnet:8

WORKDIR /var/task

# This COPY command copies the .NET Lambda project's build artifacts from the host
 machine into the image.
# The source of the COPY should match where the .NET Lambda project publishes its build
 artifacts. If the Lambda function is being built
# with the AWS .NET Lambda Tooling, the `--docker-host-build-output-dir` switch
 controls where the .NET Lambda project
# will be built. The .NET Lambda project templates default to having `--docker-host-
build-output-dir`
# set in the aws-lambda-tools-defaults.json file to "bin/Release/lambda-publish".
#
# Alternatively Docker multi-stage build could be used to build the .NET Lambda project
 inside the image.
# For more information on this approach checkout the project's README.md file.
COPY "bin/Release/lambda-publish"  .
```

2. aws-lambda-tools-defaults.json

`aws-lambda-tools-defaults.json`File ini digunakan untuk menentukan nilai default untuk wizard penyebaran Toolkit for Visual Studio dan .NET Core CLI. Daftar berikut menjelaskan bidang yang dapat Anda atur dalam `aws-lambda-tools-defaults.json` file Anda.

- `profile`: menetapkan AWS profil Anda.
- `region`: menetapkan AWS wilayah tempat sumber daya Anda disimpan.
- `configuration`: menetapkan konfigurasi yang digunakan untuk mempublikasikan fungsi Anda.
- `package-type`: menyetel tipe paket penerapan ke gambar kontainer atau arsip file.zip.

- `function-memory-size`: mengatur alokasi memori untuk fungsi Anda dalam MB.
- `function-timeout`: Timeout adalah jumlah waktu maksimum dalam hitungan detik yang dapat dijalankan oleh fungsi Lambda. Anda dapat menyesuaikan ini dengan penambahan 1 detik hingga nilai maksimum 15 menit.
- `docker-host-build-output-dir`: menyetel direktori keluaran dari proses pembuatan yang berkorelasi dengan instruksi di. `Dockerfile`
- `image-command`: adalah nama yang sepenuhnya memenuhi syarat untuk metode Anda, kode yang Anda inginkan untuk menjalankan fungsi Lambda. Sintaksnya adalah:`{Assembly}::{Namespace}.{ClassName}::{MethodName}`. Untuk informasi selengkapnya, lihat Tanda tangan Handler. Pengaturan `image-command` di sini telah mengisi nilai ini di wizard Publish Visual Studio nanti.

Berikut ini adalah contoh dari sebuah aws-lambda-tools-defaults .json yang dibuat oleh cetak biru.NET 8 (Container Image).

```
{
  "Information": [
    "This file provides default values for the deployment wizard inside Visual Studio
 and the AWS Lambda commands added to the .NET Core CLI.",
    "To learn more about the Lambda commands with the .NET Core CLI execute the
 following command at the command line in the project root directory.",
    "dotnet lambda help",
    "All the command line options for the Lambda command can be specified in this
 file."
  ],
  "profile": "default",
  "region": "us-west-2",
  "configuration": "Release",
  "package-type": "image",
  "function-memory-size": 512,
  "function-timeout": 30,
  "image-command": "AWSLambdaDocker::AWSLambdaDocker.Function::FunctionHandler",
  "docker-host-build-output-dir": "./bin/Release/lambda-publish"
}
```

3. Function.cs

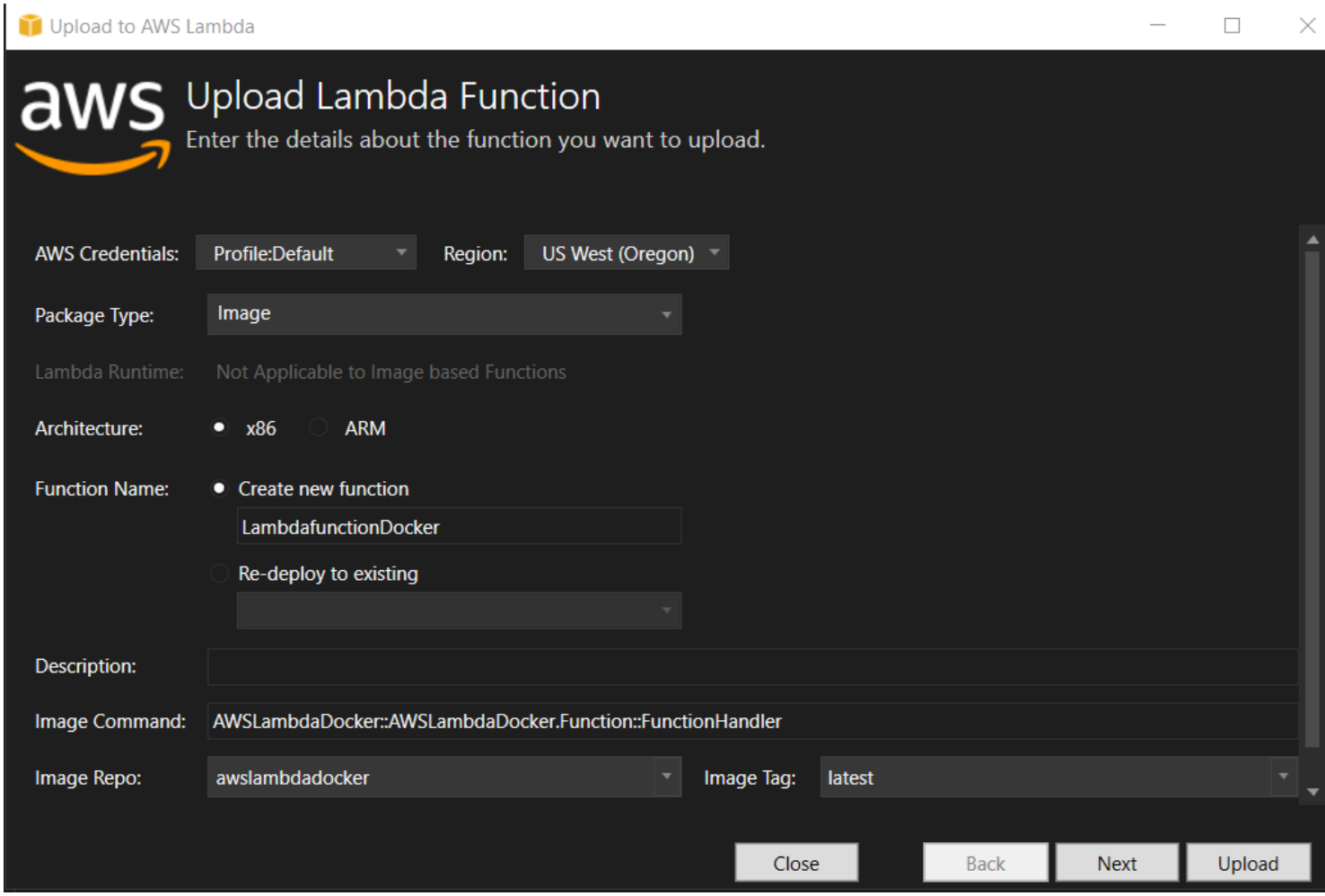
`Function.cs`File mendefinisikan fungsi c # yang akan diekspos sebagai fungsi Lambda. `FunctionHandler`Ini adalah fungsi Lambda yang berjalan saat fungsi Lambda berjalan. Dalam proyek ini, `FunctionHandler` panggilan `ToUpper()` pada teks input.

Publikasikan ke Lambda

Gambar Docker yang dihasilkan oleh proses pembuatan diunggah ke Amazon Elastic Container Registry (Amazon ECR) Registry ECR). Amazon ECR adalah registri kontainer Docker yang dikelola sepenuhnya yang Anda gunakan untuk menyimpan, mengelola, dan menyebarkan gambar kontainer Docker. Amazon ECR menghosting gambar, yang kemudian dirujuk oleh Lambda untuk menyediakan fungsionalitas Lambda yang diprogram saat dipanggil.

Untuk mempublikasikan fungsi Anda ke Lambda

1. Dari Solution Explorer, buka menu konteks untuk (klik kanan) proyek, lalu pilih Publish AWS Lambda untuk membuka jendela Upload Lambda Function.
2. Dari halaman Upload Lambda Function, lakukan hal berikut:



a. Untuk Package Type, **`Image`** telah secara otomatis dipilih sebagai Package Type Anda karena wizard publikasi mendeteksi sebuah `Dockerfile` dalam proyek Anda.

b. Untuk Nama Fungsi, masukkan nama tampilan untuk instance Lambda Anda. Nama ini adalah nama referensi yang ditampilkan di AWS Explorer di Visual Studio dan Konsol Manajemen AWS.

c. Untuk Deskripsi, masukkan teks untuk ditampilkan dengan instance Anda di Konsol Manajemen AWS.

d. Untuk Image Command, masukkan path yang sepenuhnya memenuhi syarat ke metode yang Anda inginkan untuk menjalankan fungsi Lambda: **`AWSLambdaDocker::AWSLambdaDocker.Function::FunctionHandler`**

> **Note**
>
> Nama metode apa pun yang dimasukkan di sini akan mengganti instruksi CMD apa pun di dalam Dockerfile. Memasuki Perintah Gambar hanya opsional JIKA Anda `Dockerfile` menyertakan a `CMD` untuk menginstruksikan cara meluncurkan fungsi Lambda.

e. Untuk Image Repo, masukkan nama Amazon Elastic Container Registry yang baru atau yang sudah ada. Gambar Docker yang dibuat oleh proses pembuatan diunggah ke registri ini. Definisi Lambda yang sedang diterbitkan akan merujuk pada gambar Amazon ECR itu.

f. Untuk Tag Gambar, masukkan tag Docker untuk diasosiasikan dengan gambar Anda di repositori.

g. Pilih Berikutnya.

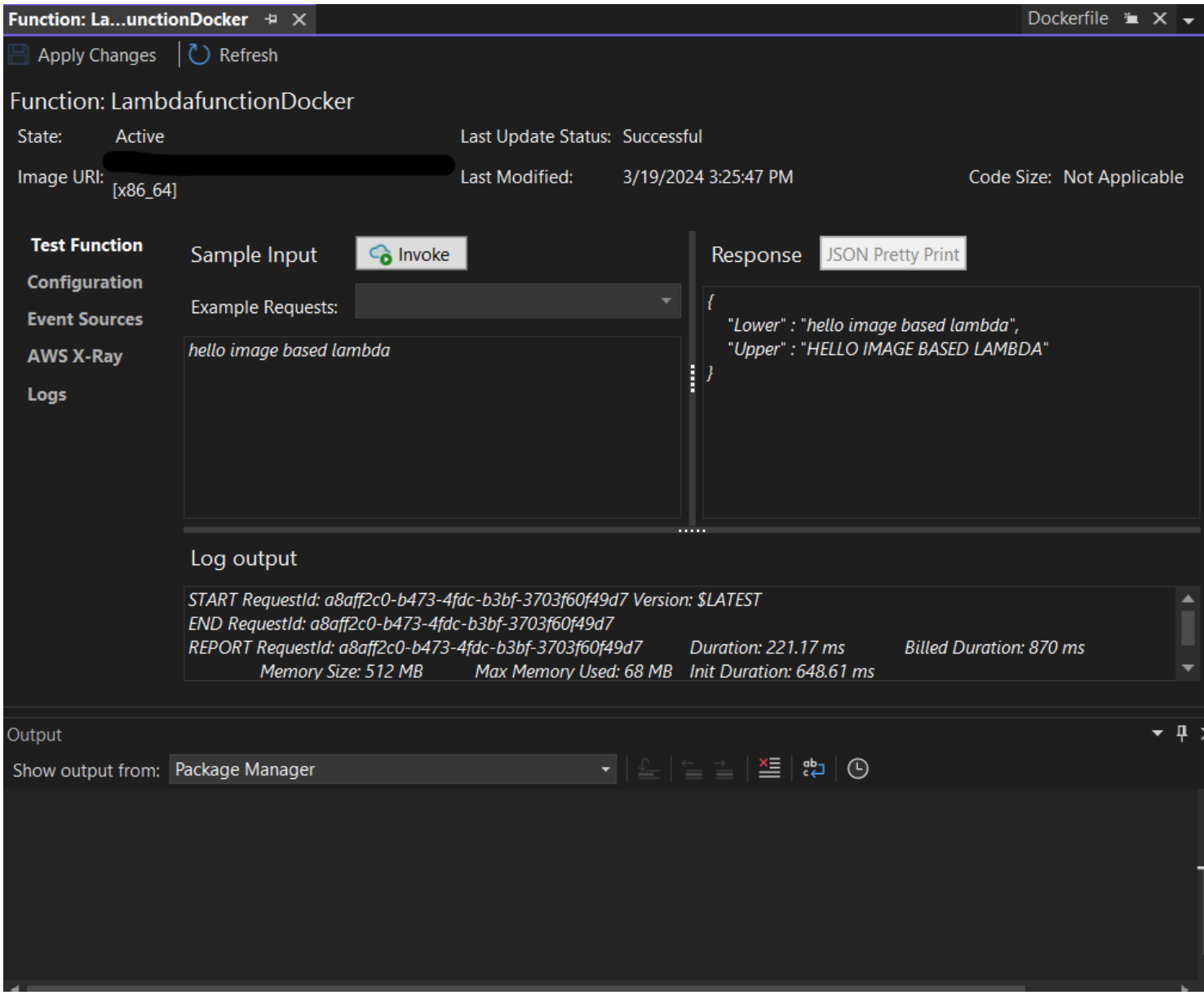
3. Pada halaman Detail Fungsi Lanjutan, di Nama Peran pilih peran yang terkait dengan akun Anda. Peran ini digunakan untuk memberikan kredensyal sementara untuk setiap panggilan Amazon Web Services yang dibuat oleh kode dalam fungsi. Jika Anda tidak memiliki peran, pilih Peran Baru berdasarkan Kebijakan AWS Terkelola, lalu pilih AWSLambdaBasicExecutionRole.

> **Note**
>
> Akun Anda harus memiliki izin untuk menjalankan ListPolicies tindakan IAM, atau daftar Nama Peran akan kosong.

4. Pilih Unggah untuk memulai proses pengunggahan dan penerbitan.

> **Note**
>
> Halaman Uploading Function ditampilkan saat fungsi sedang mengunggah. Proses publikasi kemudian membangun gambar berdasarkan parameter konfigurasi, membuat repositori Amazon ECR jika perlu, mengunggah gambar ke dalam repositori, dan membuat Lambda yang mereferensikan repo itu dengan gambar itu.
> Setelah fungsi diunggah, halaman Fungsi akan terbuka dan menampilkan konfigurasi fungsi Lambda baru Anda.

5. Untuk memanggil fungsi Lambda secara manual, pada tab Test Function, **`hello image based lambda`** masukkan ke bidang input teks bebas permintaan dan kemudian pilih Invoke. Teks Anda, dikonversi ke huruf besar, akan muncul di Respons.



6. Untuk melihat repositori, di AWS Explorer, di bawah Amazon Elastic Container Service, pilih Repositori.

   Anda dapat membuka kembali tampilan Function: kapan saja dengan mengklik dua kali pada instance yang digunakan yang terletak di AWS Explorer di bawah node. AWS Lambda

> **Note**
>
> Jika jendela AWS Explorer Anda tidak terbuka, Anda dapat merambungkannya melalui View -> AWS Explorer

7. Perhatikan opsi konfigurasi khusus gambar tambahan pada tab Konfigurasi. Tab ini menyediakan cara untuk menggantiENTRYPOINT,CMD, dan WORKDIR itu mungkin telah ditentukan dalam Dockerfile. Deskripsi adalah deskripsi yang Anda masukkan (jika ada) selama upload/publish.

Membersihkan

Jika Anda tidak akan terus mengembangkan dengan contoh ini, ingatlah untuk menghapus fungsi dan gambar ECR yang digunakan sehingga Anda tidak ditagih untuk sumber daya yang tidak digunakan di akun Anda.

- Fungsi dapat dihapus dengan mengklik kanan instance yang Anda gunakan yang terletak di AWS Explorer di bawah node. AWS Lambda
- Repositori dapat dihapus di AWS Explorer di bawah Amazon Elastic Container Service -> Repositori.

Langkah Berikutnya

Untuk informasi tentang membuat dan menguji gambar Lambda, lihat Menggunakan Gambar Kontainer dengan Lambda.

Untuk informasi tentang penerapan gambar kontainer, izin, dan pengaturan konfigurasi utama, lihat Mengonfigurasi Fungsi.

Tutorial: Membangun dan Menguji Aplikasi Tanpa Server dengan AWS Lambda

Anda dapat membangun aplikasi Lambda tanpa server dengan menggunakan template. AWS Toolkit for Visual Studio Template proyek Lambda menyertakan satu untuk Aplikasi AWS Tanpa Server, yang merupakan AWS Toolkit for Visual Studio implementasi dari Model Aplikasi AWS Tanpa Server (SAM).AWS Dengan menggunakan jenis proyek ini, Anda dapat mengembangkan kumpulan

AWS Lambda fungsi dan menyebarkannya dengan AWS sumber daya yang diperlukan sebagai keseluruhan aplikasi, menggunakan AWS CloudFormation untuk mengatur penyebaran.

Untuk prasyarat dan informasi tentang pengaturan AWS Toolkit for Visual Studio, lihat Menggunakan Template AWS Lambda di Toolkit for Visual Studio. AWS

Topik

- Buat Proyek Aplikasi AWS Tanpa Server Baru
- Meninjau file Aplikasi Tanpa Server
- Menerapkan Aplikasi Tanpa Server
- Uji Aplikasi Tanpa Server

Buat Proyek Aplikasi AWS Tanpa Server Baru

AWS Proyek Aplikasi Tanpa Server membuat fungsi Lambda dengan template tanpa server. CloudFormation CloudFormation template memungkinkan Anda untuk menentukan sumber daya tambahan seperti database, menambahkan peran IAM, dan menyebarkan beberapa fungsi pada satu waktu. Ini berbeda dari proyek AWS Lambda, yang berfokus pada pengembangan dan penerapan fungsi Lambda tunggal.

Prosedur berikut menjelaskan cara membuat Proyek Aplikasi AWS Tanpa Server baru.

1. Dari Visual Studio, perluas menu File, perluas Baru, lalu pilih Project.
2. Di kotak dialog Proyek Baru, pastikan bahwa kotak drop-down Bahasa, Platform, dan Jenis Proyek diatur ke “Semua...” dan masukkan **`aws lambda`** di bidang Pencarian.
3. Pilih template AWS Serverless Application with Tests (.NET Core - C#).

   > **Note**
   >
   > Ada kemungkinan bahwa template Aplikasi AWS Tanpa Server dengan Tes (.NET Core - C#) mungkin tidak terisi di bagian atas hasil.

4. Klik Berikutnya untuk membuka dialog Configure your new project.
5. Dari dialog Konfigurasi proyek baru Anda, masukkan **`ServerlessPowertools`** untuk Nama, lalu lengkapi bidang yang tersisa sesuai preferensi Anda. Pilih Buat tombol untuk melanjutkan ke dialog Select Blueprint.

6. Dari dialog Select Blueprint pilih Powertools untuk AWS Lambda cetak biru, lalu pilih Selesai untuk membuat proyek Visual Studio.

Meninjau file Aplikasi Tanpa Server

Bagian berikut memberikan tampilan rinci pada tiga file Aplikasi Tanpa Server yang dibuat untuk proyek Anda:

1. template tanpa server
2. Functions.cs
3. aws-lambda-tools-defaults.json

1. tanpa servers.template

`serverless.template`File adalah AWS CloudFormation template untuk mendeklarasikan fungsi Tanpa Server dan sumber daya lainnya. AWS File yang disertakan dengan proyek ini berisi deklarasi untuk satu fungsi Lambda yang akan diekspos melalui Amazon API Gateway sebagai `HTTP *Get*` operasi. Anda dapat mengedit template ini untuk menyesuaikan fungsi yang ada atau menambahkan lebih banyak fungsi dan sumber daya lain yang diperlukan oleh aplikasi Anda.

Berikut ini adalah contoh `serverless.template` file:

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Transform": "AWS::Serverless-2016-10-31",
  "Description": "An AWS Serverless Application.",
  "Resources": {
    "Get": {
      "Type": "AWS::Serverless::Function",
      "Properties": {
         "Architectures": [
            "x86_64"
            ],
         "Handler": "ServerlessPowertools::ServerlessPowertools.Functions::Get",
         "Runtime": "dotnet8",
         "CodeUri": "",
         "MemorySize": 512,
         "Timeout": 30,
         "Role": null,
         "Policies": [
```

```
            "AWSLambdaBasicExecutionRole"
            ],
         "Environment": {
            "Variables": {
               "POWERTOOLS_SERVICE_NAME": "ServerlessGreeting",
               "POWERTOOLS_LOG_LEVEL": "Info",
               "POWERTOOLS_LOGGER_CASE": "PascalCase",
               "POWERTOOLS_TRACER_CAPTURE_RESPONSE": true,
               "POWERTOOLS_TRACER_CAPTURE_ERROR": true,
               "POWERTOOLS_METRICS_NAMESPACE": "ServerlessGreeting"
               }
            },
         "Events": {
            "RootGet": {
               "Type": "Api",
               "Properties": {
                  "Path": "/",
                  "Method": "GET"
                  }
               }
            }
         }
   },
  "Outputs": {
    "ApiURL": {
      "Description": "API endpoint URL for Prod environment",
      "Value": {
        "Fn::Sub": "https://${ServerlessRestApi}.execute-api.
${AWS::Region}.amazonaws.com/Prod/"
      }
    }
  }
}
```

Perhatikan bahwa banyak bidang `...AWS:: Serverless::Function...` deklarasi mirip dengan bidang penyebaran proyek Lambda. Powertools Logging, Metrics, dan Tracing dikonfigurasi melalui variabel lingkungan berikut:

- POWERTOOLS_SERVICE_NAME= ServerlessGreeting
- PowerTools_log_level=Info
- POWERTOOLS_LOGGER_CASE = PascalCase

- PowerTools_TRACER_CAPTURE_RESPONSE=Benar
- PowerTools_TRACER_CAPTURE_ERROR=Benar
- POWERTOOLS_METRICS_NAMESPACE= ServerlessGreeting

Untuk definisi dan detail tambahan tentang variabel lingkungan, lihat situs web Powertools untuk AWS Lambda referensi.

2. Functions.cs

`Functions.cs`adalah file kelas yang berisi metode C # yang dipetakan ke satu fungsi yang dideklarasikan dalam file template. Fungsi Lambda merespons `HTTP Get` metode dari API Gateway. Berikut ini adalah contoh `Functions.cs` file:

```
public class Functions
{
    [Logging(LogEvent = true, CorrelationIdPath = CorrelationIdPaths.ApiGatewayRest)]
    [Metrics(CaptureColdStart = true)]
    [Tracing(CaptureMode = TracingCaptureMode.ResponseAndError)]
    public APIGatewayProxyResponse Get(APIGatewayProxyRequest request, ILambdaContext
 context)
    {
        Logger.LogInformation("Get Request");

        var greeting = GetGreeting();

        var response = new APIGatewayProxyResponse
        {
            StatusCode = (int)HttpStatusCode.OK,
            Body = greeting,
            Headers = new Dictionary (string, string) { { "Content-Type", "text/
plain" } }
        };

        return response;
    }

    [Tracing(SegmentName = "GetGreeting Method")]
    private static string GetGreeting()
    {
        Metrics.AddMetric("GetGreeting_Invocations", 1, MetricUnit.Count);
```

```
        return "Hello Powertools for AWS Lambda (.NET)";
    }
}
```

3. aws-lambda-tools-defaults.json

`aws-lambda-tools-defaults.json`menyediakan nilai default untuk wizard AWS penerapan di dalam Visual Studio dan AWS Lambda perintah yang ditambahkan ke.NET Core CLI. Berikut ini adalah contoh `aws-lambda-tools-defaults.json` file yang disertakan dengan proyek ini:

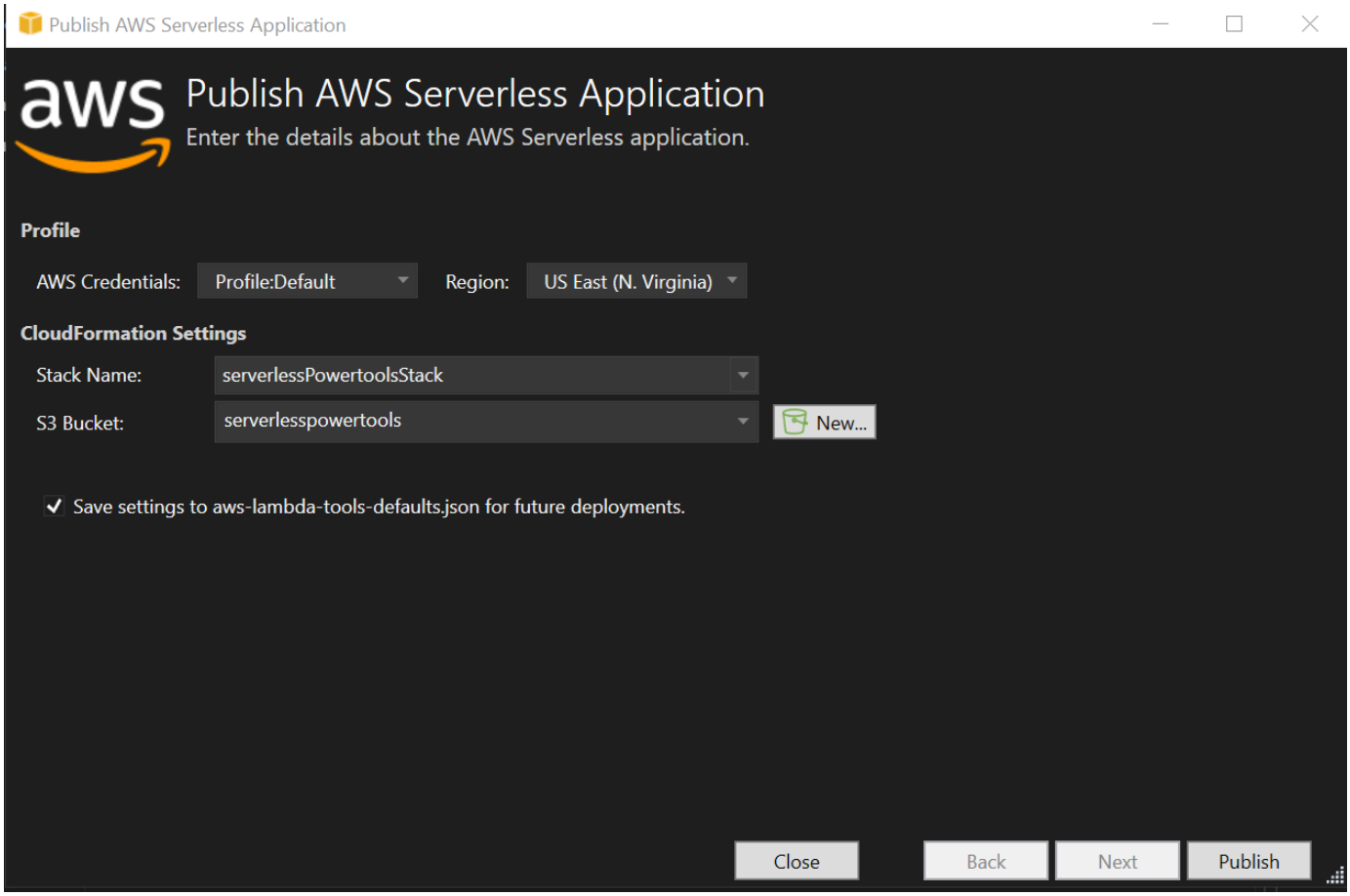
```
{
  "profile": "Default",
  "region": "us-east-1",
  "configuration": "Release",
  "s3-prefix": "ServerlessPowertools/",
  "template": "serverless.template",
  "template-parameters": ""
}
```

Menerapkan Aplikasi Tanpa Server

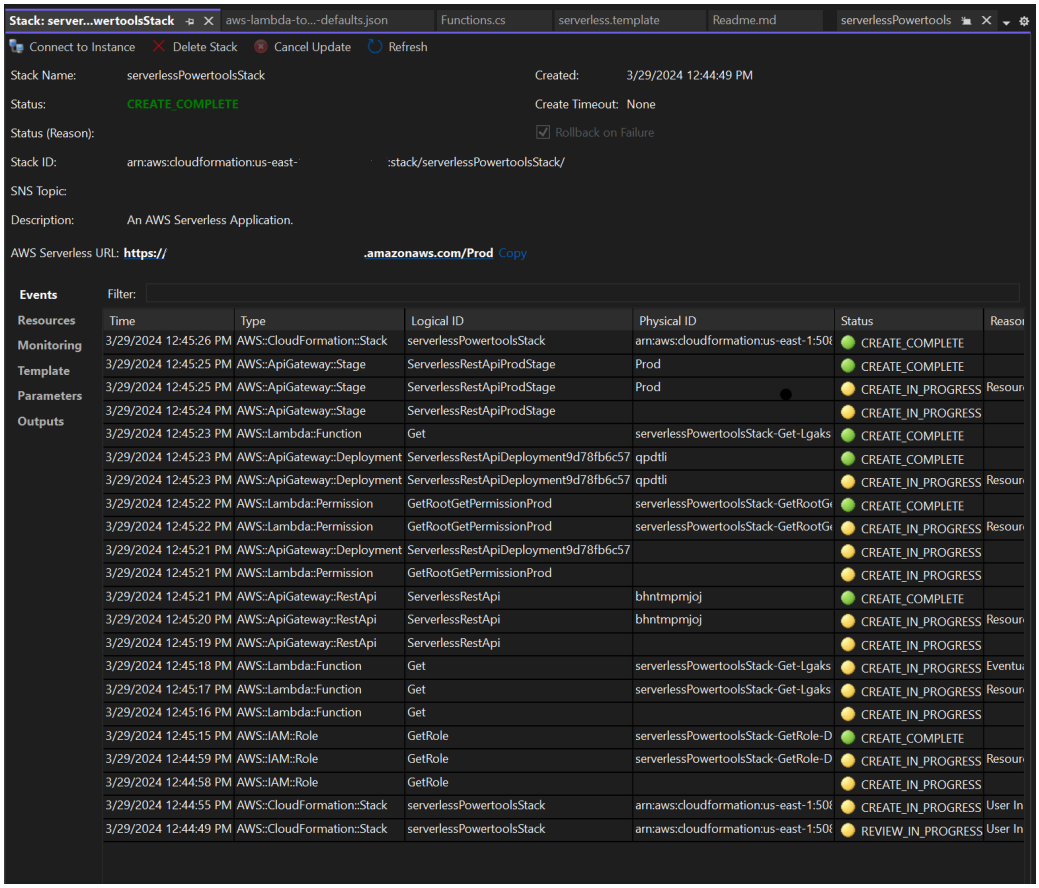
Untuk menerapkan aplikasi tanpa server Anda, selesaikan langkah-langkah berikut

1. Dari Solution Explorer, buka menu konteks untuk (klik kanan) proyek Anda dan pilih Publish to AWS Lambda untuk membuka dialog Publish AWS Serverless Application.
2. Dari dialog Publish AWS Serverless Application, masukkan nama untuk wadah CloudFormation tumpukan di bidang Stack Name.
3. Di bidang S3 Bucket, pilih bucket Amazon S3 yang akan diunggah bundel aplikasi Anda atau pilih New... tombol dan masukkan nama bucket Amazon S3 baru. Kemudian pilih Publish to publish untuk menyebarkan aplikasi Anda.

> **Note**
>
> CloudFormation Tumpukan dan Bucket Amazon S3 Anda harus ada di wilayah yang sama AWS . Pengaturan yang tersisa untuk proyek Anda ditentukan dalam `serverless.template` file.



4. Jendela tampilan Stack terbuka selama proses penerbitan, saat penerapan selesai, bidang Status menampilkan:`CREATE_COMPLETE`.



Uji Aplikasi Tanpa Server

Ketika pembuatan tumpukan selesai, Anda dapat melihat aplikasi Anda menggunakan URL AWS Tanpa Server. Jika Anda telah menyelesaikan tutorial ini tanpa menambahkan fungsi atau parameter tambahan, mengakses URL AWS tanpa server Anda akan menampilkan frasa berikut di browser web Anda: `Hello Powertools for AWS Lambda (.NET)`

Tutorial: Membuat Aplikasi Amazon Rekognition Lambda

Tutorial ini menunjukkan cara membuat aplikasi Lambda yang menggunakan Amazon Rekognition untuk menandai objek Amazon S3 dengan label yang terdeteksi.

Untuk prasyarat dan informasi tentang pengaturan AWS Toolkit for Visual Studio, lihat Menggunakan Template AWS Lambda di Toolkit for Visual Studio. AWS

Buat Proyek Rekognition Gambar Lambda Inti Visual Studio .NET

Prosedur berikut menjelaskan cara membuat aplikasi Amazon Rekognition Lambda dari aplikasi. AWS Toolkit for Visual Studio

> **Note**
>
> Setelah pembuatan, aplikasi Anda memiliki solusi dengan dua proyek: proyek sumber yang berisi kode fungsi Lambda Anda untuk diterapkan ke Lambda, dan proyek pengujian menggunakan XUnit untuk menguji fungsi Anda secara lokal.
> Terkadang Visual Studio tidak dapat menemukan semua NuGet referensi untuk proyek Anda. Ini karena cetak biru memerlukan dependensi yang harus diambil dari. NuGet Ketika proyek baru dibuat, Visual Studio hanya menarik referensi lokal dan bukan referensi jarak jauh dari NuGet. Untuk memperbaiki NuGet kesalahan: klik kanan referensi Anda dan pilih Pulihkan Paket.

1. Dari Visual Studio, perluas menu File, perluas Baru, lalu pilih Project.
2. Di kotak dialog Proyek Baru, pastikan bahwa kotak drop-down Bahasa, Platform, dan Jenis Proyek diatur ke “Semua...” dan masukkan **`aws lambda`** di bidang Pencarian.
3. Pilih template AWS Lambda with Tests (.NET Core - C #).
4. Klik Berikutnya untuk membuka dialog Configure your new project.
5. Dari dialog Konfigurasi proyek baru Anda, masukkan "ImageRekognition" untuk Nama, lalu lengkapi bidang yang tersisa sesuai preferensi Anda. Pilih Buat tombol untuk melanjutkan ke dialog Select Blueprint.
6. Dari dialog Select Blueprint, pilih cetak biru Deteksi Label Gambar, lalu pilih Selesai untuk membuat proyek Visual Studio.

> **Note**
>
> Cetak biru ini menyediakan kode untuk mendengarkan peristiwa Amazon S3 dan menggunakan Amazon Rekognition untuk mendeteksi label dan menambahkannya ke objek S3 sebagai tag.

Meninjau File Proyek

Bagian berikut memeriksa file proyek ini:

1. `Function.cs`
2. `aws-lambda-tools-defaults.json`

1. Function.cs

Di dalam `Function.cs` file, segmen kode pertama adalah atribut assembly, yang terletak di bagian atas file. Secara default, Lambda hanya menerima parameter input dan mengembalikan tipe tipe. `System.IO.Stream` Anda harus mendaftarkan serializer untuk menggunakan kelas yang diketik untuk parameter input dan tipe pengembalian. Atribut assembly mendaftarkan serializer Lambda JSON, yang `Newtonsoft.Json` digunakan untuk mengonversi aliran ke kelas yang diketik. Anda dapat mengatur serializer di tingkat perakitan atau metode.

Berikut ini adalah contoh atribut assembly:

```
// Assembly attribute to enable the Lambda function's JSON input to be converted into
 a .NET class.
[assembly:
 LambdaSerializer(typeof(Amazon.Lambda.Serialization.SystemTextJson.DefaultLambdaJsonSerializer
```

Kelas ini memiliki dua konstruktor. Yang pertama adalah konstruktor default yang digunakan saat Lambda memanggil fungsi Anda. Konstruktor ini menciptakan klien layanan Amazon S3 dan Amazon Rekognition. Konstruktor juga mengambil AWS kredensyal untuk klien ini dari peran IAM yang Anda tetapkan ke fungsi saat Anda menerapkannya. AWS Wilayah untuk klien diatur ke wilayah yang menjalankan fungsi Lambda Anda. Dalam cetak biru ini, Anda hanya ingin menambahkan tag ke objek Amazon S3 jika layanan Amazon Rekognition memiliki tingkat kepercayaan minimum tentang label tersebut. Konstruktor ini memeriksa variabel lingkungan `MinConfidence` untuk menentukan tingkat kepercayaan yang dapat diterima. Anda dapat mengatur variabel lingkungan ini saat Anda menerapkan fungsi Lambda.

Berikut ini adalah contoh konstruktor kelas pertama di`Function.cs`:

```
public Function()
{
    this.S3Client = new AmazonS3Client();
    this.RekognitionClient = new AmazonRekognitionClient();
```

```
    var environmentMinConfidence =
 System.Environment.GetEnvironmentVariable(MIN_CONFIDENCE_ENVIRONMENT_VARIABLE_NAME);
    if(!string.IsNullOrWhiteSpace(environmentMinConfidence))
    {
        float value;
        if(float.TryParse(environmentMinConfidence, out value))
        {
            this.MinConfidence = value;
            Console.WriteLine($"Setting minimum confidence to {this.MinConfidence}");
        }
        else
        {
            Console.WriteLine($"Failed to parse value {environmentMinConfidence} for
 minimum confidence. Reverting back to default of {this.MinConfidence}");
        }
    }
    else
    {
        Console.WriteLine($"Using default minimum confidence of {this.MinConfidence}");
    }
}
```

Contoh berikut menunjukkan bagaimana konstruktor kedua dapat digunakan untuk pengujian. Proyek pengujian mengonfigurasi klien S3 dan Rekognition sendiri dan meneruskannya di:

```
public Function(IAmazonS3 s3Client, IAmazonRekognition rekognitionClient, float
 minConfidence)
{
    this.S3Client = s3Client;
    this.RekognitionClient = rekognitionClient;
    this.MinConfidence = minConfidence;
}
```

Berikut ini adalah contoh `FunctionHandler` metode di dalam `Function.cs` file.

```
public async Task FunctionHandler(S3Event input, ILambdaContext context)
{
    foreach(var record in input.Records)
    {
        if(!SupportedImageTypes.Contains(Path.GetExtension(record.S3.Object.Key)))
        {
            Console.WriteLine($"Object {record.S3.Bucket.Name}:{record.S3.Object.Key}
 is not a supported image type");
```

```
            continue;
        }

        Console.WriteLine($"Looking for labels in image {record.S3.Bucket.Name}:
{record.S3.Object.Key}");
        var detectResponses = await this.RekognitionClient.DetectLabelsAsync(new
 DetectLabelsRequest
        {
            MinConfidence = MinConfidence,
            Image = new Image
            {
                S3Object = new Amazon.Rekognition.Model.S3Object
                {
                    Bucket = record.S3.Bucket.Name,
                    Name = record.S3.Object.Key
                }
            }
        });

        var tags = new List();
        foreach(var label in detectResponses.Labels)
        {
            if(tags.Count < 10)
            {
                Console.WriteLine($"\tFound Label {label.Name} with confidence
 {label.Confidence}");
                tags.Add(new Tag { Key = label.Name, Value =
 label.Confidence.ToString() });
            }
            else
            {
                Console.WriteLine($"\tSkipped label {label.Name} with confidence
 {label.Confidence} because maximum number of tags reached");
            }
        }

        await this.S3Client.PutObjectTaggingAsync(new PutObjectTaggingRequest
        {
            BucketName = record.S3.Bucket.Name,
            Key = record.S3.Object.Key,
            Tagging = new Tagging
            {
                TagSet = tags
            }
```

```
        });
    }
    return;
}
```

`FunctionHandler`adalah metode yang dipanggil Lambda setelah membangun instance. Perhatikan bahwa parameter input adalah tipe `S3Event` dan bukan a`Stream`. Anda dapat melakukan ini karena serializer Lambda JSON yang terdaftar. `S3Event`Berisi semua informasi tentang acara yang dipicu di Amazon S3. Fungsi loop melalui semua objek S3 yang merupakan bagian dari acara dan memberitahu Rekognition untuk mendeteksi label. Setelah label terdeteksi, mereka ditambahkan sebagai tag ke objek S3.

> **Note**
>
> Kode berisi panggilan ke`Console.WriteLine()`. Saat fungsi berjalan di Lambda, semua panggilan untuk `Console.WriteLine()` mengalihkan ke Amazon Logs. CloudWatch

2. aws-lambda-tools-defaults.json

`aws-lambda-tools-defaults.json`File berisi nilai default yang telah ditetapkan cetak biru untuk mengisi beberapa bidang di wizard penerapan. Ini juga membantu dalam mengatur opsi baris perintah untuk integrasi dengan.NET Core CLI.

Untuk mengakses integrasi.NET Core CLI, navigasikan ke direktori dan ketik proyek fungsi. **`dotnet lambda help`**

> **Note**
>
> Penangan fungsi menunjukkan metode apa yang dipanggil Lambda sebagai respons terhadap fungsi yang dipanggil. Format bidang ini adalah:`<assembly-name>::<full-type-name>::<method-name>`. Namespace harus disertakan dengan nama tipe.

Menyebarkan Fungsi

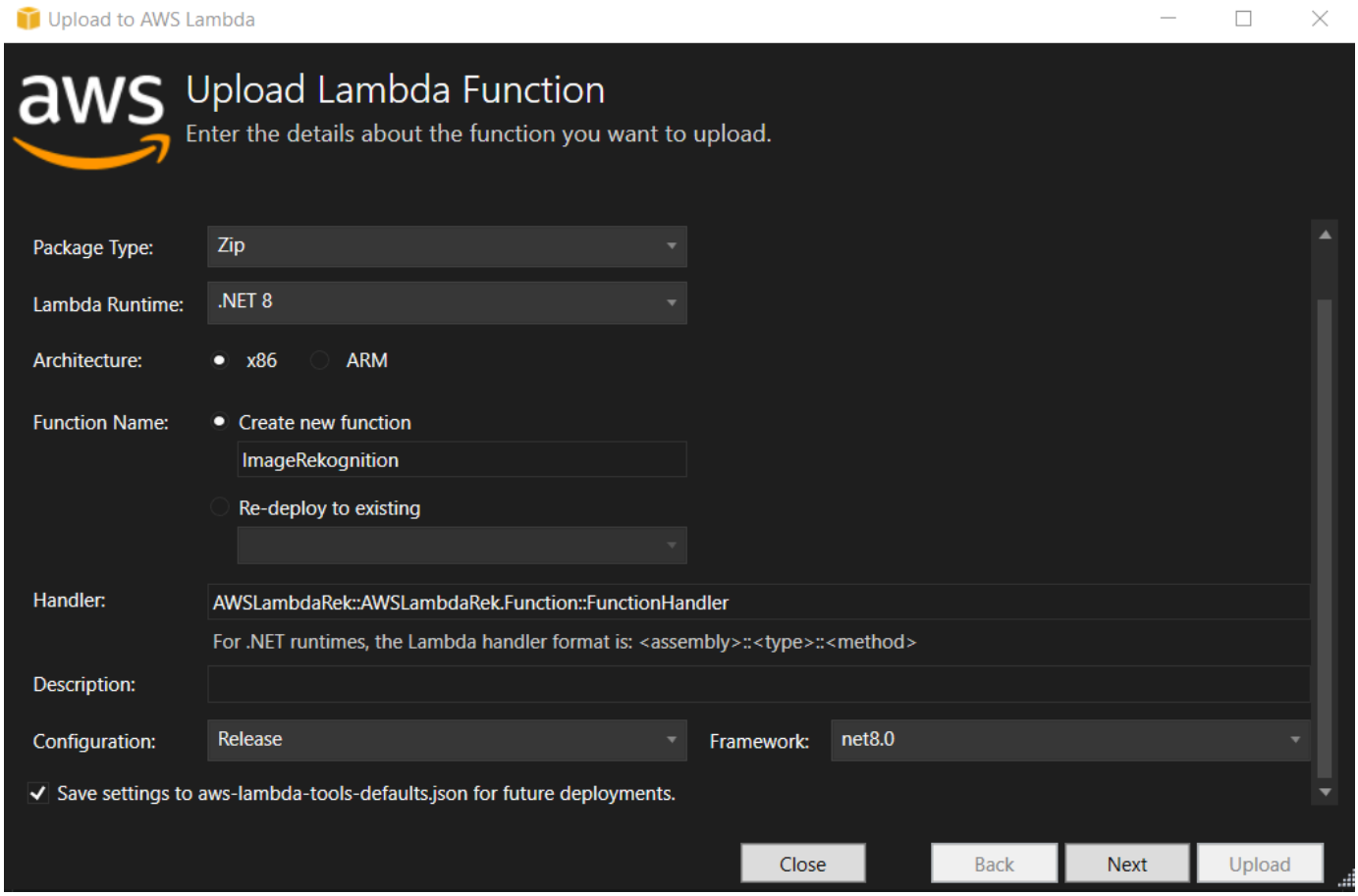
Prosedur berikut menjelaskan cara menerapkan fungsi Lambda Anda.

1. Dari Solution Explorer, klik kanan proyek Lambda dan pilih Publish to AWS Lambda untuk membuka jendela Upload to. AWS Lambda

> **Note**
>
> Nilai preset diambil dari file. `aws-lambda-tools-defaults.json`

2. Dari AWS Lambda jendela Upload to, masukkan nama ke bidang Function Name, lalu pilih tombol Next untuk maju ke jendela Advanced Function Details.

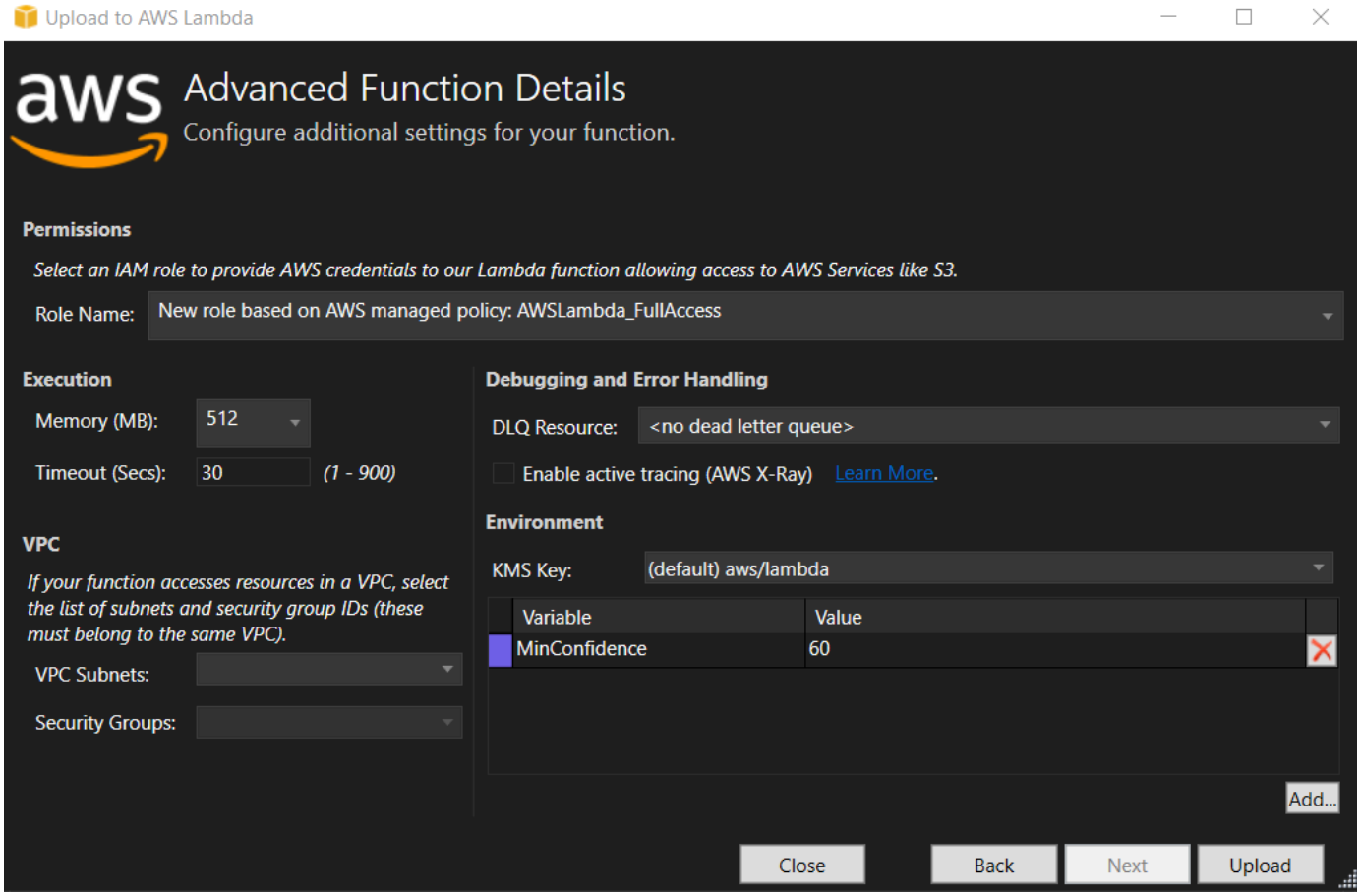
> **Note**
>
> Contoh ini, menggunakan Nama Fungsi**`ImageRekognition`**.



3. Dari jendela Detail Fungsi Lanjutan, pilih peran IAM yang memberikan izin bagi kode Anda untuk mengakses sumber daya Amazon S3 dan Amazon Rekognition Anda.

> **Note**
>
> Jika Anda mengikuti contoh ini, pilih `AWSLambda_FullAccess` peran.

4. Setel variabel lingkungan `MinConfidence` ke 60, lalu pilih Unggah untuk meluncurkan proses penerapan. Proses penerbitan selesai ketika tampilan Fungsi ditampilkan di AWS Explorer.



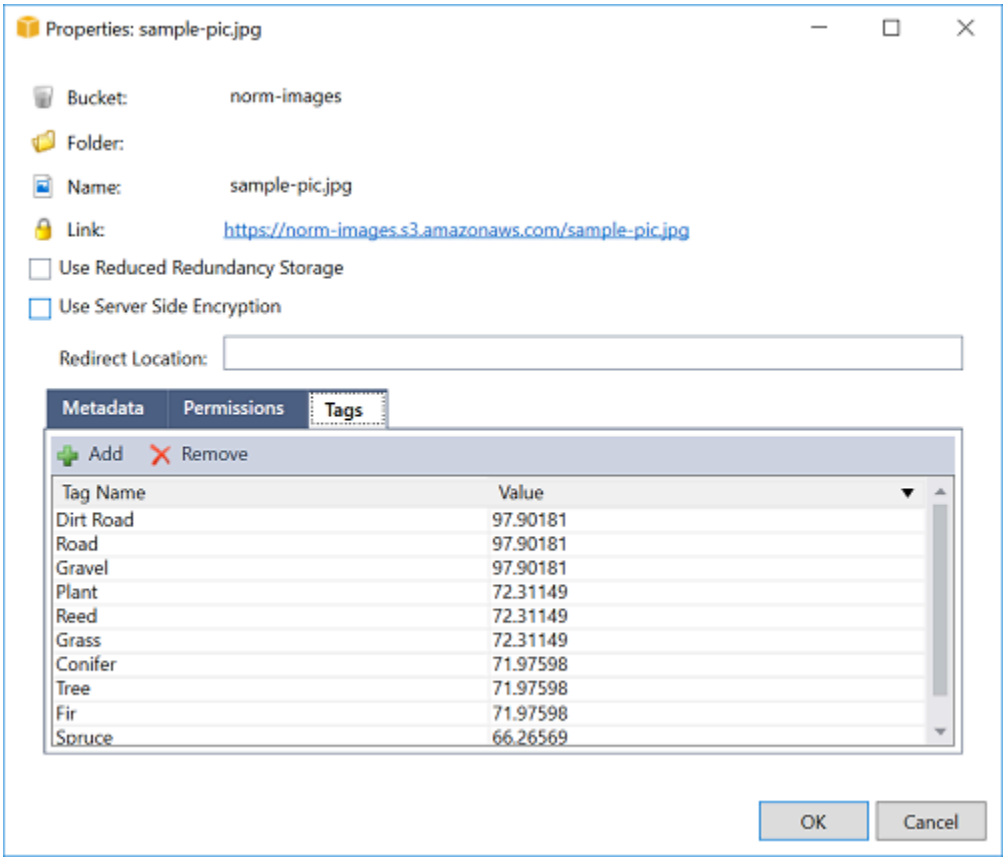
5. Setelah penerapan berhasil, konfigurasikan Amazon S3 untuk mengirim peristiwanya ke fungsi baru Anda dengan menavigasi ke tab Sumber Peristiwa.
6. Dari tab Sumber Acara, pilih tombol Tambah, lalu pilih bucket Amazon S3 untuk terhubung dengan fungsi Lambda Anda.

> **Note**
>
> Bucket harus berada di AWS wilayah yang sama dengan fungsi Lambda Anda.

Uji Fungsinya

Sekarang setelah fungsi tersebut diterapkan dan bucket S3 dikonfigurasi sebagai sumber acara untuknya, buka browser bucket S3 dari AWS Explorer untuk bucket yang Anda pilih. Kemudian unggah beberapa gambar.

Ketika unggahan selesai, Anda dapat mengonfirmasi bahwa fungsi Anda berjalan dengan melihat log dari tampilan fungsi Anda. Atau, klik kanan gambar di browser bucket dan pilih Properties. Pada tab Tag, Anda dapat melihat tag yang diterapkan ke objek Anda.



Tutorial: Menggunakan Amazon Logging Frameworks dengan AWS Lambda untuk Membuat Log Aplikasi

Anda dapat menggunakan Amazon CloudWatch Logs untuk memantau, menyimpan, dan mengakses log aplikasi Anda. Untuk mendapatkan data CloudWatch log ke Log, gunakan AWS SDK atau instal agen CloudWatch Log untuk memantau folder log tertentu. CloudWatch Log terintegrasi dengan beberapa framework logging .NET yang populer, menyederhanakan alur kerja.

Untuk mulai bekerja dengan kerangka kerja CloudWatch logging Log dan .NET, tambahkan NuGet paket yang sesuai dan sumber keluaran CloudWatch Log ke aplikasi Anda, lalu gunakan

pustaka logging Anda seperti biasa. Ini memungkinkan aplikasi Anda untuk mencatat pesan dengan framework .NET Anda, mengirimkannya ke CloudWatch Log, menampilkan pesan log aplikasi Anda di konsol CloudWatch Log. Anda juga dapat mengatur metrik dan alarm dari konsol CloudWatch Log, berdasarkan pesan log aplikasi Anda.

Kerangka kerja loging.NET yang didukung meliputi:

- NLog: Untuk melihat, lihat paket nuget.org NLog .
- Log4net: Untuk melihat, lihat paket Log4net nuget.org.
- ASP.NET Core logging Framework: Untuk melihat, lihat paket nuget.org ASP.NET Core logging Framework.

Berikut ini adalah contoh `NLog.config` file yang memungkinkan CloudWatch Log dan konsol sebagai output untuk pesan log dengan menambahkan `AWS.Logger.NLog` NuGet paket, dan AWS target ke dalam`NLog.config`.

```
<?xml version="1.0" encoding="utf-8" ?>
<nlog xmlns="http://www.nlog-project.org/schemas/NLog.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      throwExceptions="true">
  <targets>
    <target name="aws" type="AWSTarget" logGroup="NLog.ConfigExample" region="us-
east-1"/>
    <target name="logfile" xsi:type="Console" layout="${callsite} ${message}" />
  </targets>
  <rules>
    <logger name="*" minlevel="Info" writeTo="logfile,aws" />
  </rules>
</nlog>
```

Plugin logging semuanya dibangun di atas AWS SDK untuk .NET dan mengautentikasi AWS kredensional Anda dalam proses yang mirip dengan SDK. Contoh berikut merinci izin yang diperlukan oleh kredensi plugin logging untuk mengakses Log: CloudWatch

> **Note**
>
> Plugin AWS loging.NET adalah proyek open source. Untuk informasi tambahan, sampel, dan instruksi, lihat topik sampel dan instruksi di GitHubrepositori.NET AWS Logging.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "logs:DescribeLogGroups"
      ],
      "Resource": [
        "arn:aws:logs:*:*:*"
      ]
    }
  ]
}
```

Menyebarkan ke AWS

Toolkit for Visual Studio mendukung penerapan aplikasi AWS Elastic Beanstalk ke kontainer CloudFormation atau tumpukan.

> **Note**
>
> Jika Anda menggunakan Visual Studio Express Edition:
>
> - Anda dapat menggunakan CLI Docker untuk menyebarkan aplikasi ke wadah Amazon ECS.
> - Anda dapat menggunakan AWS Management Console untuk menyebarkan aplikasi ke wadah Elastic Beanstalk.
>
> Untuk penyebaran Elastic Beanstalk, Anda harus terlebih dahulu membuat paket penyebaran web. Untuk informasi selengkapnya, lihat Cara: Membuat Paket Deployment Web di Visual Studio. Untuk penerapan Amazon ECS, Anda harus memiliki image Docker. Untuk informasi selengkapnya, lihat Visual Studio Tools for Docker.

Topik

Bekerja dengan Publish to AWS di Visual Studio

Publish to AWS adalah pengalaman penyebaran interaktif yang membantu Anda memublikasikan aplikasi.NET Anda ke target AWS penerapan, mendukung aplikasi yang menargetkan .NET Core 3.1 dan yang lebih baru. Bekerja dengan Publish untuk AWS menjaga alur kerja Anda di dalam Visual Studio dengan membuat fitur penerapan ini tersedia, langsung dari IDE Anda:

- Kemampuan untuk menyebarkan aplikasi Anda dengan satu klik.

- Rekomendasi penerapan berdasarkan aplikasi Anda.
- Pembuatan Dockerfile otomatis, seperti yang relevan dan diperlukan oleh lingkungan tujuan penerapan Anda (target penerapan).
- Pengaturan yang dioptimalkan untuk membangun dan mengemas aplikasi Anda, seperti yang dipersyaratkan oleh target penerapan Anda.

> **Note**
>
> Untuk informasi tambahan tentang penerbitan aplikasi.NET Framework, lihat panduan Membuat dan menerapkan aplikasi.NET di Elastic Beanstalk
> Anda juga dapat mengakses Publish ke AWS dari .NET CLI. Untuk informasi selengkapnya, lihat AWS panduan Menerapkan aplikasi.NET.

Topik

Prasyarat

Agar berhasil mempublikasikan aplikasi.NET ke AWS layanan, instal yang berikut ini ke perangkat lokal Anda:

- .NET Core 3.1+ (yang meliputi. NET5 dan. NET6): Untuk informasi tambahan tentang produk ini dan informasi unduhan, kunjungi situs unduhan Microsoft.
- Node.js 14.x atau versi yang lebih baru: Node.js diperlukan untuk menjalankan AWS Cloud Development Kit (AWS CDK). Untuk mengunduh atau mendapatkan informasi lebih lanjut tentang Node.js, kunjungi situs unduhan Node.js.

> **Note**
>
> Publikasikan AWS untuk digunakan AWS CDK untuk menyebarkan aplikasi Anda dan semua infrastruktur penerapannya sebagai satu proyek. Untuk informasi selengkapnya, AWS CDK lihat panduan Cloud Development Kit.

- (Opsional) Docker digunakan saat menerapkan ke layanan berbasis kontainer seperti Amazon ECS. Untuk informasi lebih lanjut dan untuk mengunduh Docker, lihat situs unduhan Docker.

Jenis aplikasi yang didukung

Sebelum menerbitkan ke target baru atau keluar, mulailah dengan membuat atau membuka salah satu jenis proyek berikut di Visual Studio:

- Aplikasi ASP.NET Core
- Aplikasi Konsol .NET
- Aplikasi Blazor WebAssembly

Menerbitkan aplikasi ke AWS target

Saat menerbitkan ke target baru, Publish to AWS akan memandu Anda melalui proses dengan membuat rekomendasi dan menggunakan pengaturan umum. Jika Anda perlu mempublikasikan ke target yang telah disiapkan sebelumnya, preferensi Anda disimpan dan dapat disesuaikan, atau segera tersedia untuk penerapan satu klik.

> **Note**
>
> Integrasi toolkit dengan Server.NET CLI:
> Publishing meluncurkan proses server.NET di localhost untuk melakukan proses publikasi.

Publikasikan ke target baru

Berikut ini menjelaskan cara mengonfigurasi preferensi Publikasikan ke AWS penerapan, saat Anda memublikasikan ke target baru.

1. Dari AWS Explorer, perluas menu drop-down Kredensial, lalu pilih AWS profil yang sesuai dengan wilayah dan AWS layanan yang diperlukan untuk penyebaran Anda.
2. Perluas menu drop-down Region, lalu pilih AWS wilayah yang berisi AWS layanan yang diperlukan untuk penyebaran Anda.
3. Dari panel Visual Studio Solutions Explorer, buka menu konteks untuk (klik kanan) nama proyek, dan pilih Publish to. AWS Ini akan membuka Publikasikan ke AWS.

4. Dari Publikasikan ke AWS, pilih Publikasikan ke Target Baru untuk mengonfigurasi penerapan baru.

> **Note**
>
> Untuk mengubah kredensyal penerapan default Anda, pilih atau klik tautan Edit yang terletak di sebelah bagian Kredensial, di Publikasikan ke. AWS
> Untuk melewati proses konfigurasi target, pilih Publikasikan ke Target yang Ada, lalu pilih konfigurasi pilihan Anda dari daftar target penerapan sebelumnya.

5. Dari panel Publikasikan Target, pilih AWS layanan untuk mengelola penerapan aplikasi Anda.
6. Bila Anda puas dengan konfigurasi Anda, pilih Publish untuk memulai proses deployment.

> **Note**
>
> Setelah memulai penerapan, Publikasikan untuk AWS menampilkan pembaruan status berikut:
>
> - Selama proses penyebaran, Publikasikan untuk AWS menampilkan informasi tentang kemajuan penerapan.
> - Setelah proses penyebaran, Publikasikan untuk AWS menunjukkan apakah penerapan berhasil atau gagal.
> - Setelah penerapan berhasil, panel Resources menawarkan informasi tambahan tentang sumber daya yang dibuat. Informasi ini akan bervariasi tergantung pada jenis aplikasi dan konfigurasi penerapan.

Publikasikan ke target yang ada

Berikut ini menjelaskan cara mempublikasikan ulang aplikasi.NET Anda ke AWS target yang ada.

1. Dari AWS Explorer, perluas menu drop-down Kredensial, lalu pilih AWS profil yang sesuai dengan wilayah dan AWS layanan yang diperlukan untuk penyebaran Anda.
2. Perluas menu drop-down Region, lalu pilih AWS wilayah yang berisi AWS layanan yang diperlukan untuk penyebaran Anda.
3. Dari panel Visual Studio Solutions Explorer, klik kanan nama proyek dan pilih Publish to open Publish AWS to. AWS

4. Dari Publikasikan ke AWS, pilih Publikasikan ke Target yang Ada untuk memilih lingkungan penerapan Anda dari daftar target yang ada.

> **Note**
>
> Jika Anda baru saja menerbitkan aplikasi apa pun ke AWS Cloud, aplikasi tersebut ditampilkan di Publish to AWS.

5. Pilih target penerbitan yang ingin digunakan aplikasi, lalu klik Publikasikan untuk memulai proses penerapan.

Menerapkan AWS Lambda Proyek dengan .NET Core CLI

AWS Toolkit for Visual Studio Termasuk template proyek AWS Lambda .NET Core untuk Visual Studio. Anda dapat menerapkan fungsi Lambda yang dibangun di Visual Studio menggunakan antarmuka baris perintah .NET Core (CLI).

Topik

- Prasyarat
- Topik terkait
- Daftar Perintah Lambda yang Tersedia melalui CLI CLI.NET
- Menerbitkan Proyek Lambda N.NET Core dari .NET Core CLI

Prasyarat

Sebelum bekerja dengan.NET Core CLI untuk menyebarkan fungsi Lambda, Anda harus memenuhi prasyarat berikut:

- Pastikan Visual Studio 2015 Update 3 diinstal.
- Instal .NET Core untuk Windows.
- Siapkan CLI CLI .NET untuk bekerja dengan Lambda. Untuk informasi selengkapnya, lihat .NET Core CLI di Panduan AWS Lambda Pengembang.
- Instal Toolkit for Visual Studio. Untuk informasi selengkapnya, lihat Instalasi AWS Toolkit for Visual Studio.

Topik terkait

Topik terkait berikut dapat membantu saat Anda menggunakan .NET Core CLI untuk menyebarkan fungsi Lambda:

- Untuk informasi selengkapnya tentang fungsi Lambda, lihat Apa itu AWS Lambda? di Panduan AWS Lambda Pengembang.
- Untuk informasi tentang membuat fungsi Lambda di Visual Studio, lihat. AWS Lambda
- Untuk informasi selengkapnya tentang Microsoft. NET Core, lihat .NET Core di dokumentasi online Microsoft.

Daftar Perintah Lambda yang Tersedia melalui CLI CLI.NET

Untuk membuat daftar perintah Lambda yang tersedia melalui CLI CLI.NET Core, lakukan hal berikut.

1. Buka jendela prompt perintah, dan arahkan ke folder yang berisi proyek Visual Studio .NET Core Lambda.
2. Masukkan `dotnet lambda --help`.

```
C:\Lambda\AWSLambda1\AWSLambda1>dotnet lambda --help AWS Lambda Tools for .NET Core
 functions
    Project Home: https://github.com/aws/aws-lambda-dotnet
    .
    Commands to deploy and manage Lambda functions:
    .
            deploy-function         Deploy the project to Lambda
            invoke-function         Invoke the function in Lambda with an optional
 input
            list-functions          List all of your Lambda functions
            delete-function         Delete a Lambda function
            get-function-config     Get the current runtime configuration for a Lambda
 function
            update-function-config  Update the runtime configuration for a Lambda
 function
    .
    Commands to deploy and manage AWS serverless applications using AWS CloudFormation:
    .
            deploy-serverless       Deploy an AWS serverless application
            list-serverless         List all of your AWS serverless applications
```

```
            delete-serverless       Delete an AWS serverless application
    .
    Other Commands:
    .
            package                 Package a Lambda project into a .zip file ready for
 deployment
    .
    To get help on individual commands, run the following:

            dotnet lambda help <command>
```

Menerbitkan Proyek Lambda N.NET Core dari .NET Core CLI

Instruksi berikut mengasumsikan Anda telah membuat AWS Lambda fungsi.NET Core di Visual Studio.

1. Buka jendela prompt perintah, dan arahkan ke folder yang berisi proyek Visual Studio .NET Core Lambda Anda.
2. Masukkan `dotnet lambda deploy-function`.
3. Saat diminta, masukkan nama fungsi yang akan digunakan. Ini bisa berupa nama baru atau nama fungsi yang ada.
4. Saat diminta, masukkan AWS Wilayah (Wilayah tempat fungsi Lambda Anda akan digunakan).
5. Saat diminta, pilih atau buat peran IAM yang akan diasumsikan Lambda saat menjalankan fungsi.

Setelah berhasil diselesaikan, pesan Fungsi Lambda Baru yang dibuat ditampilkan.

```
C:\Lambda\AWSLambda1\AWSLambda1>dotnet lambda deploy-function
Executing publish command
... invoking 'dotnet publish', working folder 'C:\Lambda\AWSLambda1\AWSLambda1\bin
\Release\netcoreapp1.0\publish'
... publish: Publishing AWSLambda1 for .NETCoreApp,Version=v1.0
... publish: Project AWSLambda1 (.NETCoreApp,Version=v1.0) will be compiled because
 expected outputs are missing
... publish: Compiling AWSLambda1 for .NETCoreApp,Version=v1.0
... publish: Compilation succeeded.
... publish:     0 Warning(s)
... publish:     0 Error(s)
... publish: Time elapsed 00:00:01.2479713
... publish:
```

```
... publish: publish: Published to C:\Lambda\AWSLambda1\AWSLambda1\bin\Release
\netcoreapp1.0\publish
... publish: Published 1/1 projects successfully
Zipping publish folder C:\Lambda\AWSLambda1\AWSLambda1\bin\Release
\netcoreapp1.0\publish to C:\Lambda\AWSLambda1\AWSLamb
da1\bin\Release\netcoreapp1.0\AWSLambda1.zip
Enter Function Name: (AWS Lambda function name)
DotNetCoreLambdaTest
Enter AWS Region: (The region to connect to AWS services)
us-west-2
Creating new Lambda function
Select IAM Role that Lambda will assume when executing function:
    1) lambda_exec_LambdaCoreFunction
    2) *** Create new IAM Role ***
1
New Lambda function created
```

Jika Anda menerapkan fungsi yang ada, fungsi deploy hanya meminta Region. AWS

```
C:\Lambda\AWSLambda1\AWSLambda1>dotnet lambda deploy-function
Executing publish command
Deleted previous publish folder
... invoking 'dotnet publish', working folder 'C:\Lambda\AWSLambda1\AWSLambda1\bin
\Release\netcoreapp1.0\publish'
... publish: Publishing AWSLambda1 for .NETCoreApp,Version=v1.0
... publish: Project AWSLambda1 (.NETCoreApp,Version=v1.0) was previously compiled.
 Skipping compilation.
... publish: publish: Published to C:\Lambda\AWSLambda1\AWSLambda1\bin\Release
\netcoreapp1.0\publish
... publish: Published 1/1 projects successfully
Zipping publish folder C:\Lambda\AWSLambda1\AWSLambda1\bin\Release
\netcoreapp1.0\publish to C:\Lambda\AWSLambda1\AWSLamb
da1\bin\Release\netcoreapp1.0\AWSLambda1.zip
Enter Function Name: (AWS Lambda function name)
DotNetCoreLambdaTest
Enter AWS Region: (The region to connect to AWS services)
us-west-2
Updating code for existing function
```

Setelah fungsi Lambda Anda di-deploy, itu siap digunakan. Untuk informasi selengkapnya, lihat Contoh Cara Menggunakan AWS Lambda.

Lambda secara otomatis memonitor fungsi Lambda untuk Anda, melaporkan metrik melalui Amazon. CloudWatch Untuk memantau dan memecahkan masalah fungsi Lambda, lihat Memecahkan Masalah dan Memantau Fungsi AWS Lambda dengan Amazon. CloudWatch

Menyebarkan ke AWS Elastic Beanstalk Visual Studio menggunakan AWS Toolkit for Visual Studio dengan Amazon Q

AWS Elastic Beanstalk adalah layanan yang menyederhanakan proses penyediaan AWS sumber daya untuk aplikasi Anda. Elastic Beanstalk menyediakan semua infrastruktur yang diperlukan untuk AWS menyebarkan aplikasi Anda. Infrastruktur ini meliputi:

- Instans Amazon EC2 yang meng-host executable dan konten untuk aplikasi Anda.
- Grup Auto Scaling untuk mempertahankan jumlah instans Amazon EC2 yang sesuai untuk mendukung aplikasi Anda.
- Penyeimbang beban Elastic Load Balancing yang merutekan lalu lintas masuk ke instans Amazon EC2 dengan bandwidth terbanyak.

Topik panduan pengguna ini menjelaskan cara bekerja dengan wizard Elastic Beanstalk di AWS Toolkit dengan Amazon Q. Untuk informasi terperinci khusus tentang Elastic Beanstalk, lihat Panduan Pengembang. AWS Elastic Beanstalk Wizard Elastic Beanstalk untuk Toolkit dengan Amazon Q dijelaskan di AWS bagian topik berikut.

Topik

- Menyebarkan Aplikasi ASP.NET Tradisional ke Elastic Beanstalk
- Menyebarkan Aplikasi Inti ASP.NET ke Elastic Beanstalk (Legacy)
- Cara Menentukan Kredensi AWS Keamanan untuk Aplikasi Anda
- Cara Mempublikasikan Ulang Aplikasi Anda ke Lingkungan Elastic Beanstalk (Legacy)
- Penerapan Aplikasi Elastic Beanstalk Kustom
- Penyebaran Elastic Beanstalk Inti ASP.NET Kustom
- Dukungan Beberapa Aplikasi untuk.NET dan Elastic Beanstalk

Menyebarkan Aplikasi ASP.NET Tradisional ke Elastic Beanstalk

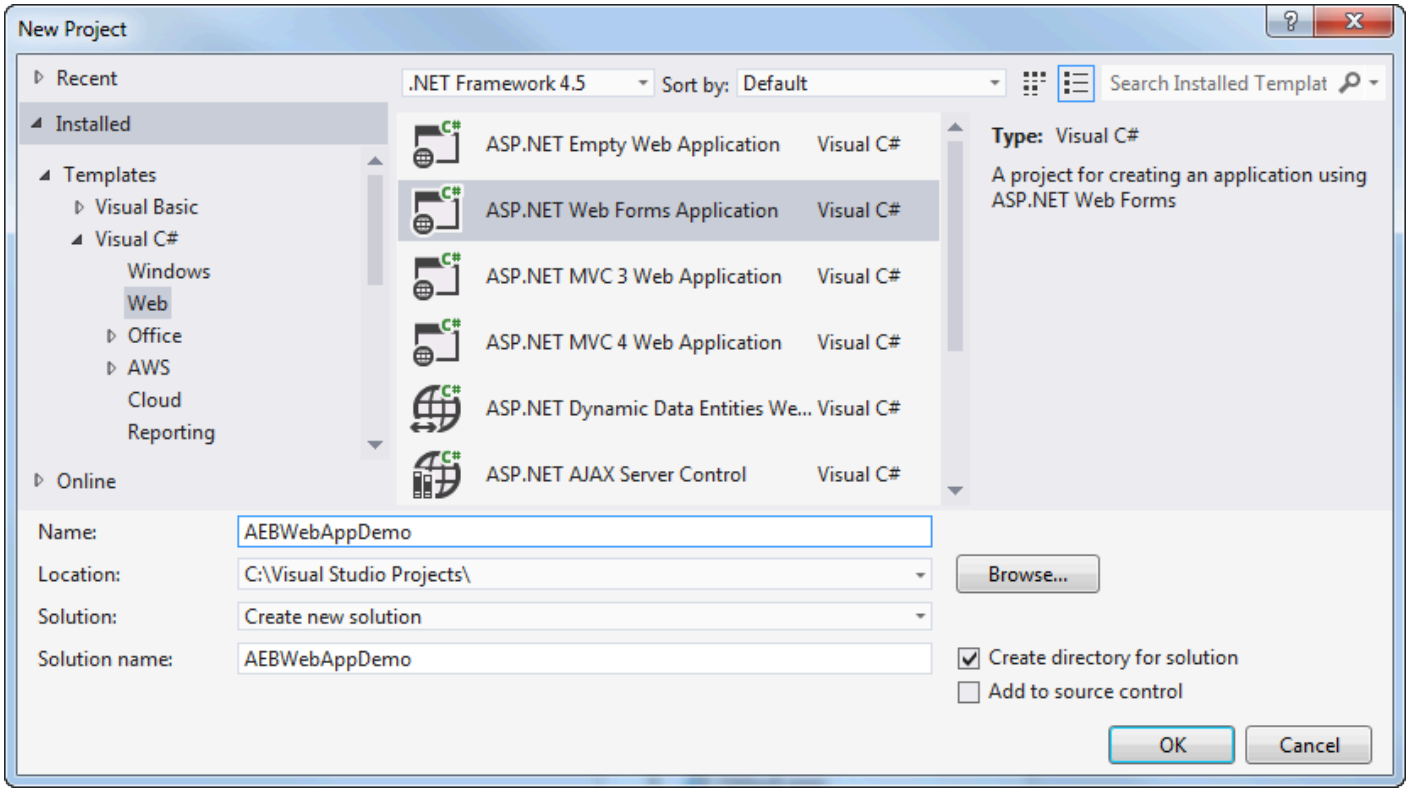
Bagian ini menjelaskan cara menggunakan wizard Publish to Elastic Beanstalk, yang disediakan sebagai bagian dari Toolkit for Visual Studio, untuk menyebarkan aplikasi melalui Elastic Beanstalk. Untuk berlatih, Anda dapat menggunakan instance proyek pemula aplikasi web yang dibangun ke Visual Studio atau Anda dapat menggunakan proyek Anda sendiri.

> **Note**
>
> Wizard juga mendukung penerapan aplikasi ASP.NET Core. Untuk informasi tentang ASP.NET Core, lihat panduan alat penyebaran AWS .NET dan AWS daftar isi Deploying to yang diperbarui.

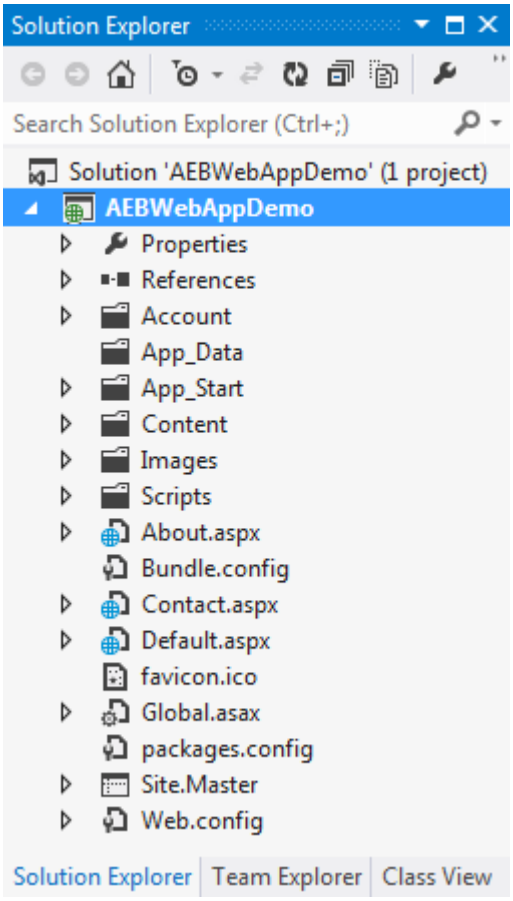
> **Note**
>
> Sebelum Anda dapat menggunakan wizard Publish to Elastic Beanstalk, Anda harus mengunduh dan menginstal Web Deploy. Wizard bergantung pada Web Deploy untuk menyebarkan aplikasi web dan situs web ke server web Internet Information Services (IIS).

Untuk membuat contoh proyek pemula aplikasi web

1. Di Visual Studio, dari menu File, pilih New, dan kemudian pilih Project.
2. Di panel navigasi kotak dialog Proyek Baru, perluas Terpasang, perluas Template, perluas Visual C #, lalu pilih Web.
3. Dalam daftar templat proyek web, pilih templat apa pun yang berisi kata-kata `Web` dan `Application` deskripsinya. Untuk contoh ini, pilih Aplikasi Formulir Web ASP.NET.

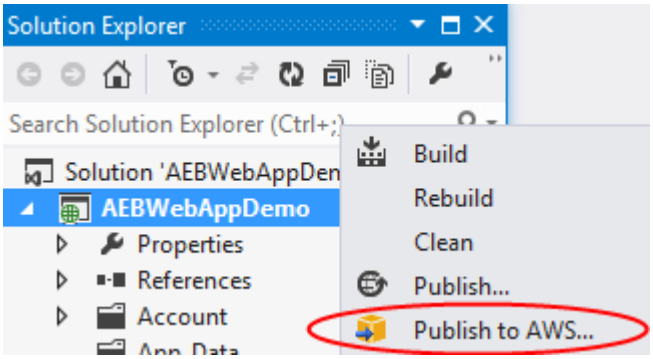


4. Di kotak Nama, ketik`AEBWebAppDemo`.
5. Di kotak Lokasi, ketik jalur ke folder solusi di mesin pengembangan Anda atau pilih Jelajahi, lalu telusuri ke dan pilih folder solusi, lalu pilih Pilih Folder.
6. Konfirmasikan kotak Buat direktori untuk solusi dipilih. Di daftar drop-down Solusi, konfirmasikan Buat solusi baru dipilih, lalu pilih OK. Visual Studio akan membuat solusi dan proyek berdasarkan template proyek Aplikasi Formulir Web ASP.NET. Visual Studio kemudian akan menampilkan Solution Explorer di mana solusi dan proyek baru muncul.

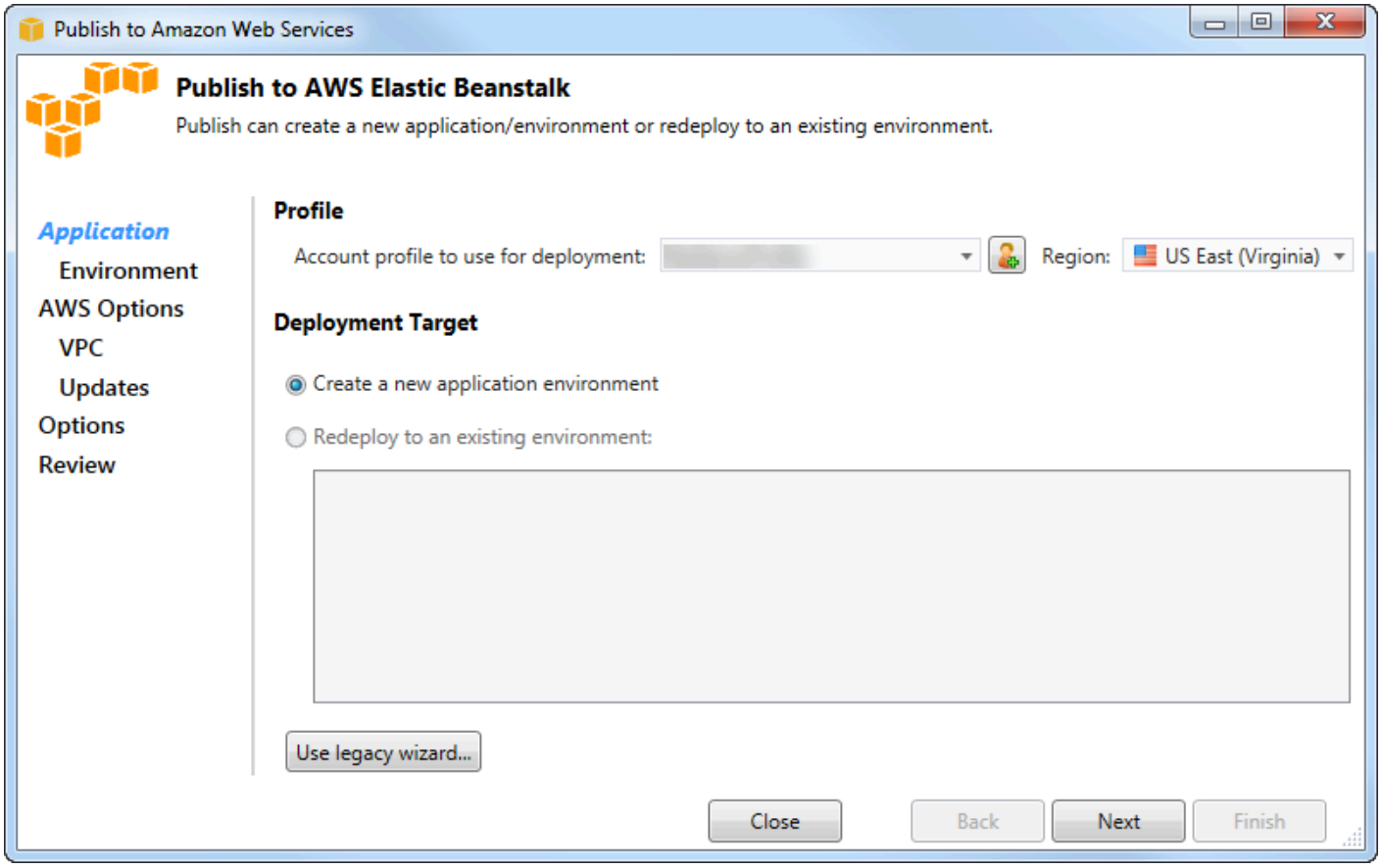


Untuk menyebarkan aplikasi dengan menggunakan wizard Publish to Elastic Beanstalk

1. Di Solution Explorer, buka menu konteks (klik kanan) untuk folder AEBWebAppDemoproyek untuk proyek yang Anda buat di bagian sebelumnya, atau buka menu konteks untuk folder proyek untuk aplikasi Anda sendiri, dan pilih Publish to AWS Elastic Beanstalk.



   Wisaya Publish to Elastic Beanstalk muncul.



2. Di Profil, dari profil Akun yang akan digunakan untuk daftar drop-down penyebaran, pilih profil AWS akun yang ingin Anda gunakan untuk penyebaran.

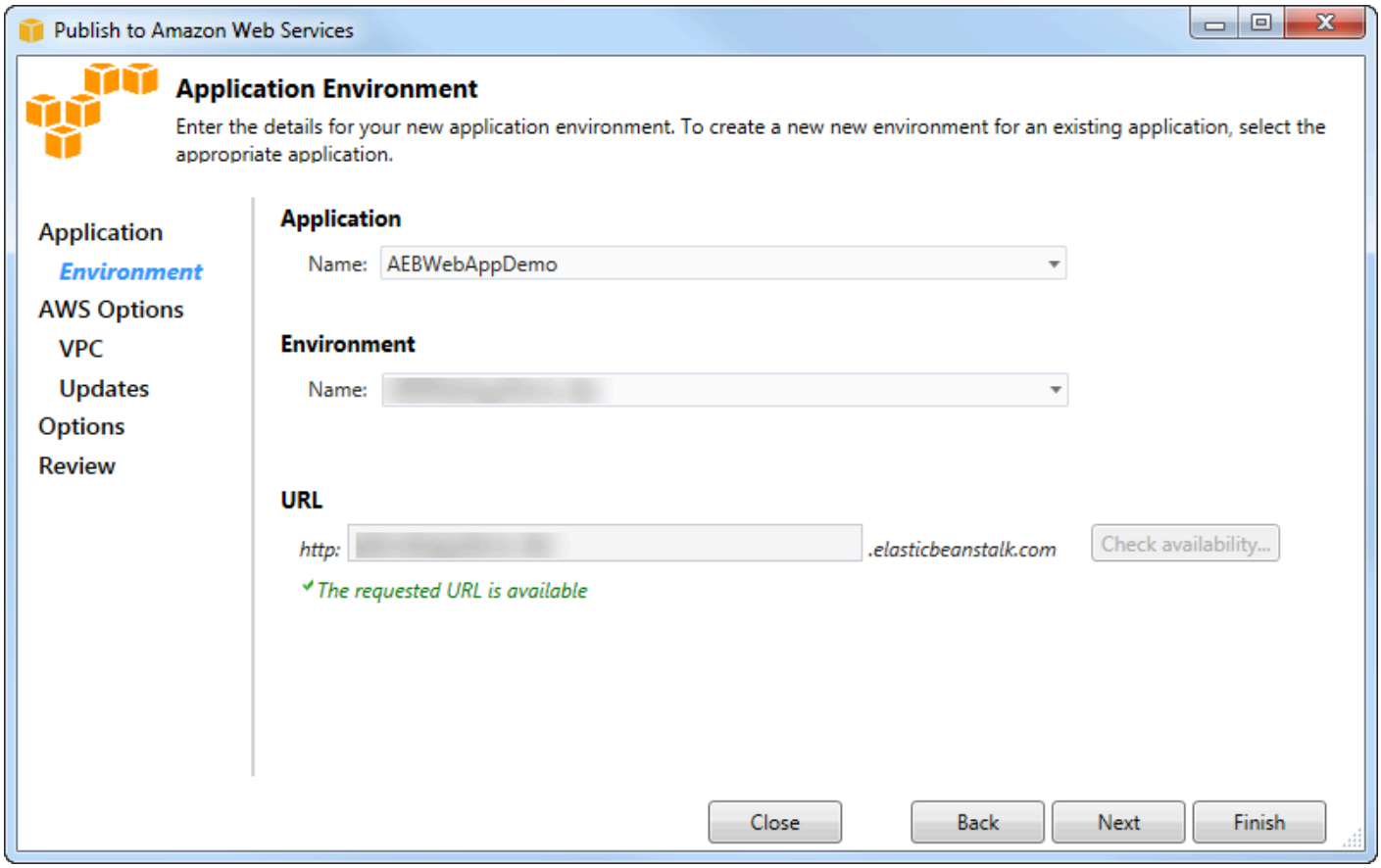
   Secara opsional, jika Anda memiliki AWS akun yang ingin Anda gunakan, tetapi Anda belum membuat profil AWS akun untuk itu, Anda dapat memilih tombol dengan simbol plus (+) untuk menambahkan profil AWS akun.

3. Dari daftar drop-down Region, pilih wilayah yang Anda inginkan Elastic Beanstalk untuk menyebarkan aplikasi.

4. Di Target Deployment, Anda dapat memilih Membuat lingkungan aplikasi baru untuk melakukan penyebaran awal aplikasi atau Redeploy ke lingkungan yang ada untuk menerapkan ulang aplikasi yang sebelumnya digunakan. (Penerapan sebelumnya mungkin telah dilakukan dengan wizard atau Standalone Deployment Tool yang tidak digunakan lagi.) Jika Anda memilih Menerapkan ulang ke lingkungan yang ada, mungkin ada penundaan saat wizard mengambil informasi dari penerapan sebelumnya yang sedang berjalan.

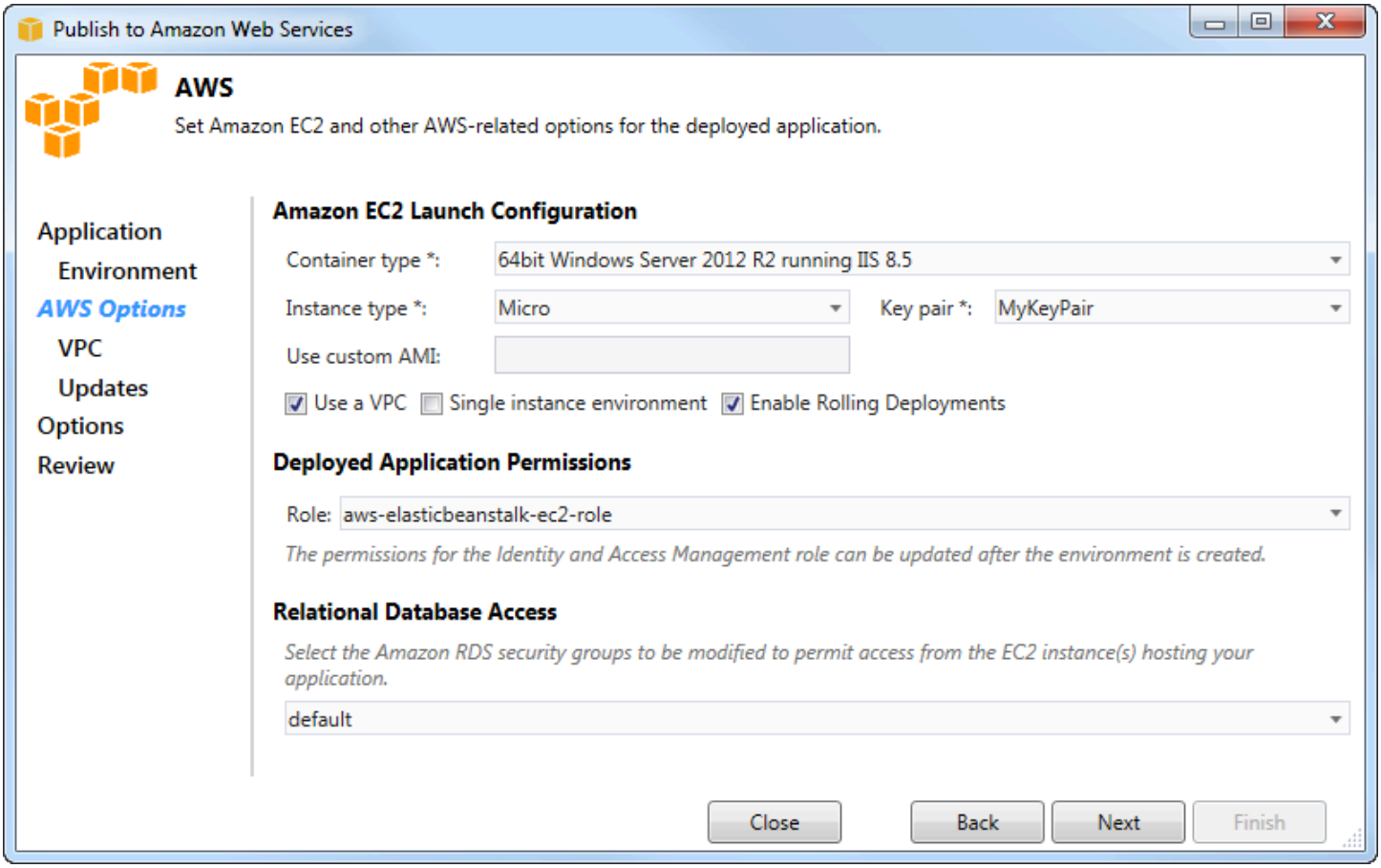
> **Note**
>
> Jika Anda memilih Redeploy ke lingkungan yang ada, pilih lingkungan dalam daftar, dan kemudian pilih Berikutnya, wizard akan membawa Anda langsung ke halaman Opsi Aplikasi. Jika Anda menempuh rute ini, lewati ke petunjuk nanti di bagian ini yang menjelaskan cara menggunakan halaman Opsi Aplikasi.

5. Pilih Berikutnya.



6. Pada halaman Lingkungan Aplikasi, di area Aplikasi, daftar drop-down Nama mengusulkan nama default untuk aplikasi. Anda dapat mengubah nama default dengan memilih nama yang berbeda dari daftar drop-down.
7. Di area Lingkungan, dalam daftar drop-down Nama, ketikkan nama untuk lingkungan Elastic Beanstalk Anda. Dalam konteks ini, istilah lingkungan mengacu pada infrastruktur ketentuan Elastic Beanstalk untuk aplikasi Anda. Nama default mungkin sudah diusulkan dalam daftar drop-down ini. Jika nama default belum diusulkan, Anda dapat mengetik satu atau memilih salah satu dari daftar drop-down, jika ada nama tambahan yang tersedia. Nama lingkungan tidak boleh lebih dari 23 karakter.

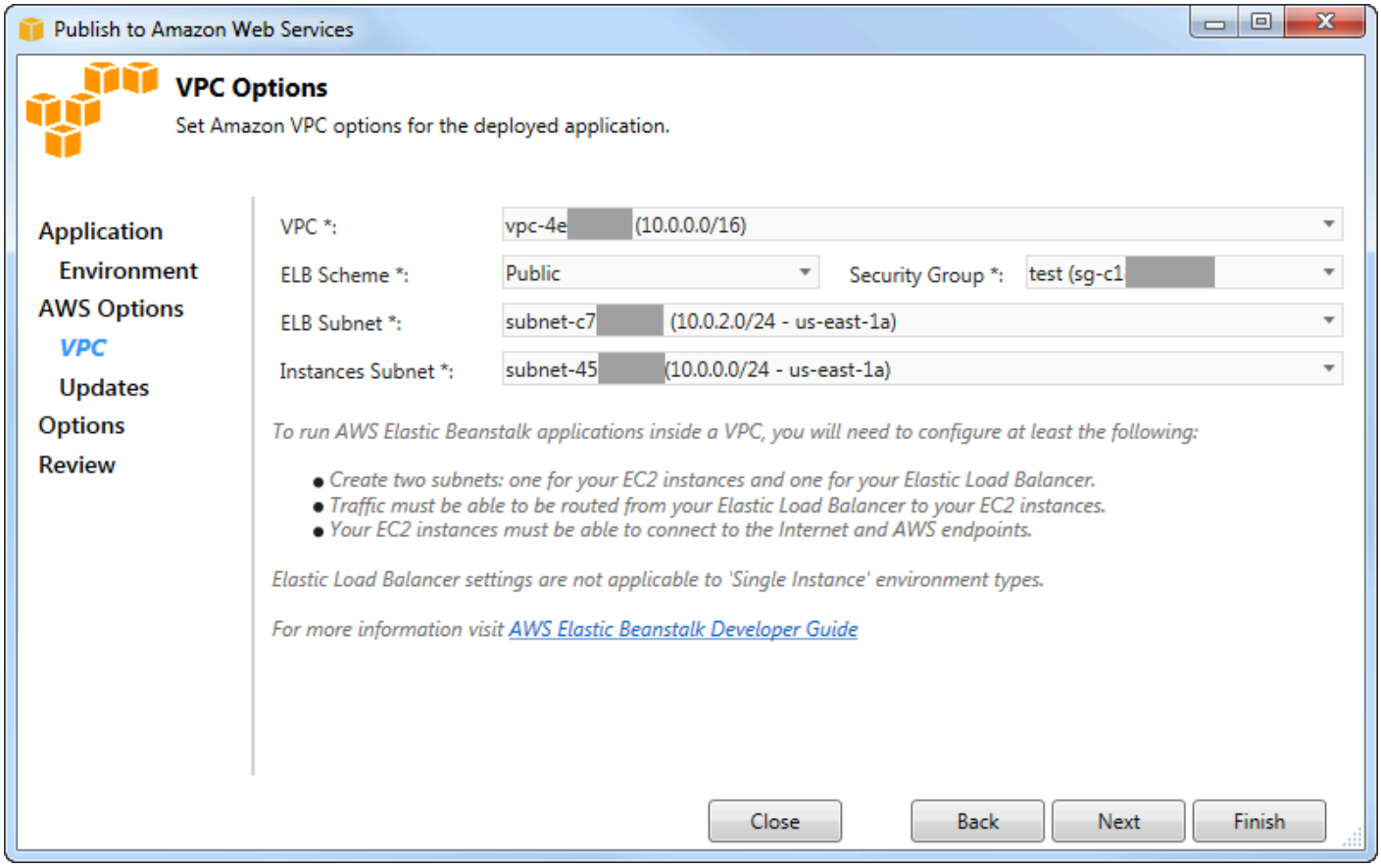
8. Di area URL, kotak mengusulkan subdomain default `.elasticbeanstalk.com` yang akan menjadi URL untuk aplikasi web Anda. Anda dapat mengubah subdomain default dengan mengetikkan nama subdomain baru.
9. Pilih Periksa ketersediaan untuk memastikan URL untuk aplikasi web Anda belum digunakan.
10. Jika URL untuk aplikasi web Anda baik-baik saja untuk digunakan, pilih Berikutnya.



1. Pada halaman AWS Opsi, di Amazon EC2 Launch Configuration, dari daftar drop-down tipe Container, pilih jenis Amazon Machine Image (AMI) yang akan digunakan untuk aplikasi Anda.
2. Dalam daftar drop-down Jenis instans, tentukan jenis instans Amazon EC2 yang akan digunakan. Untuk contoh ini, kami sarankan Anda menggunakan Micro. Ini akan meminimalkan biaya yang terkait dengan menjalankan instance. Untuk informasi lebih lanjut tentang biaya Amazon EC2, buka halaman Harga EC2.
3. Dalam daftar drop-down Key pair, pilih key pair instans Amazon EC2 yang akan digunakan untuk masuk ke instance yang akan digunakan untuk aplikasi Anda.

4. Secara opsional, di kotak Use custom AMI, Anda dapat menentukan AMI kustom yang akan mengganti AMI yang ditentukan dalam daftar drop-down tipe Container. Untuk informasi selengkapnya tentang cara membuat AMI kustom, buka Menggunakan Kustom AMIs di Panduan Pengembang AWS Elastic Beanstalk dan Buat AMI dari Instans Amazon EC2.
5. Secara opsional, jika Anda ingin meluncurkan instance Anda di VPC, pilih kotak Gunakan VPC.
6. Secara opsional, jika Anda ingin meluncurkan satu instans Amazon EC2 dan kemudian menerapkan aplikasi Anda ke instans itu, pilih kotak lingkungan instans tunggal.

   Jika Anda memilih kotak ini, Elastic Beanstalk masih akan membuat grup Auto Scaling, tetapi tidak akan mengkonfigurasinya. Jika Anda ingin mengonfigurasi grup Auto Scaling nanti, Anda dapat menggunakan. Konsol Manajemen AWS
7. Secara opsional, jika Anda ingin mengontrol kondisi di mana aplikasi Anda di-deploy ke instance, pilih kotak Aktifkan Penerapan Bergulir. Anda dapat memilih kotak ini hanya jika Anda belum memilih kotak lingkungan contoh tunggal.
8. Jika aplikasi Anda menggunakan AWS layanan seperti Amazon S3 dan DynamoDB, cara terbaik untuk memberikan kredensi adalah dengan menggunakan peran IAM. Di area Izin Aplikasi yang Diterapkan, Anda dapat memilih peran IAM yang ada atau membuat satu yang akan digunakan wizard untuk meluncurkan lingkungan Anda. Aplikasi yang menggunakan AWS SDK untuk .NET akan secara otomatis menggunakan kredensi yang disediakan oleh peran IAM ini saat membuat permintaan ke layanan. AWS
9. Jika aplikasi Anda mengakses database Amazon RDS, dalam daftar drop-down di area Akses Database Relasional, pilih kotak di sebelah grup keamanan Amazon RDS mana pun yang akan diperbarui oleh wizard sehingga instans Amazon EC2 Anda dapat mengakses database tersebut.
10. Pilih Berikutnya.
    - Jika Anda memilih Gunakan VPC, halaman Opsi VPC akan muncul.
    - Jika Anda memilih Aktifkan Penerapan Bergulir, tetapi tidak memilih Gunakan VPC, halaman Rolling Deployment akan muncul. Lewati petunjuk nanti di bagian ini yang menjelaskan cara menggunakan halaman Rolling Deployment.
    - Jika Anda tidak memilih Gunakan VPC atau Aktifkan Penyebaran Bergulir, halaman Opsi Aplikasi akan muncul. Lewati ke petunjuk nanti di bagian ini yang menjelaskan cara menggunakan halaman Opsi Aplikasi.
11. Jika Anda memilih Gunakan VPC, tentukan informasi di halaman Opsi VPC untuk meluncurkan aplikasi Anda ke dalam VPC.



VPC harus sudah dibuat. Jika Anda membuat VPC di Toolkit for Visual Studio, Toolkit for Visual Studio akan mengisi halaman ini untuk Anda. Jika Anda membuat VPC di Konsol AWS Manajemen, ketikkan informasi tentang VPC Anda ke halaman ini.

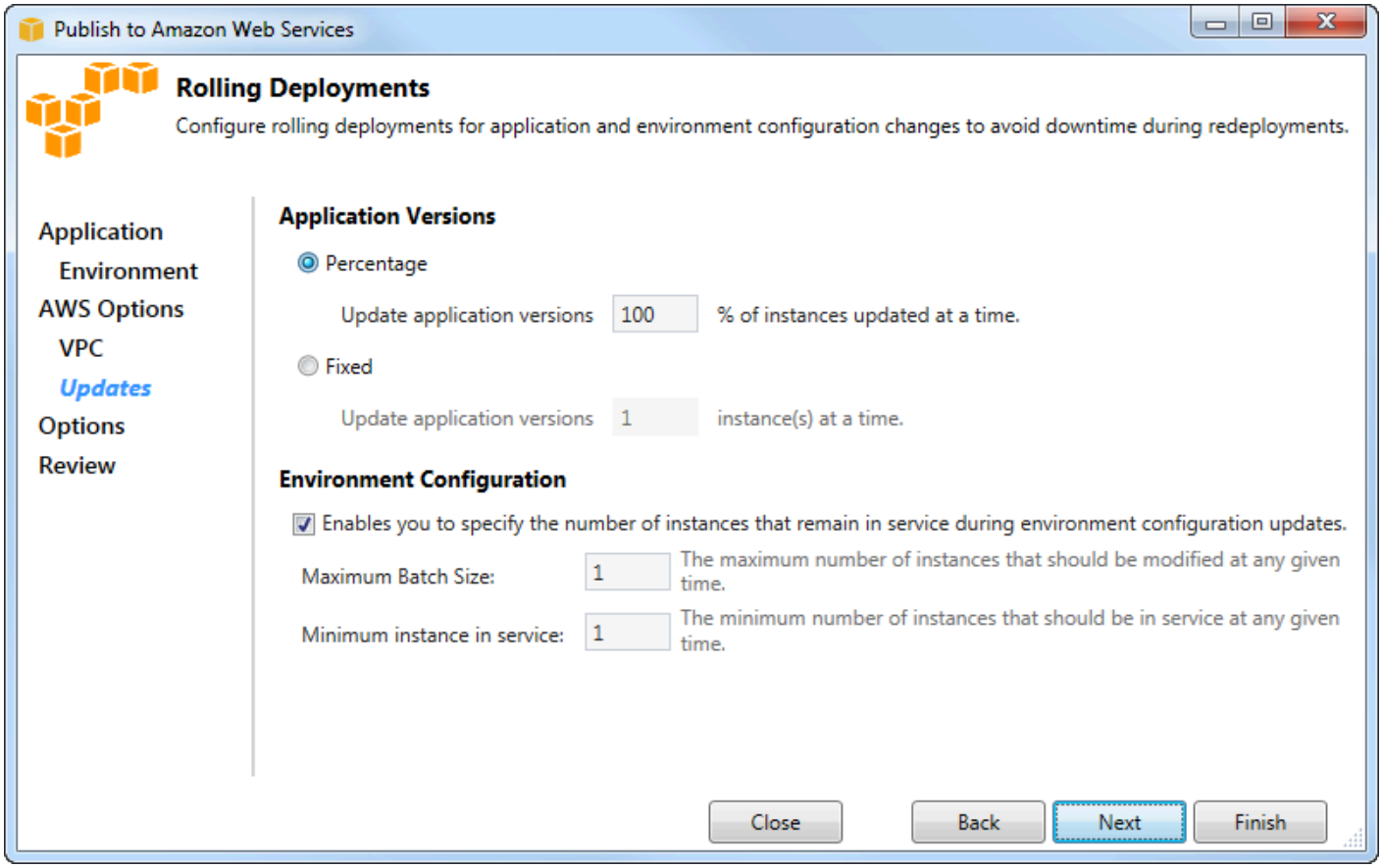
Pertimbangan utama untuk penerapan ke VPC

- VPC Anda membutuhkan setidaknya satu subnet publik dan satu subnet pribadi.
- Dalam daftar drop-down ELB Subnet, tentukan subnet publik. Toolkit for Visual Studio menyebarkan penyeimbang beban Elastic Load Balancing untuk aplikasi Anda ke subnet publik. Subnet publik dikaitkan dengan tabel routing yang memiliki entri yang mengarah ke gateway Internet. Anda dapat mengenali gateway Internet karena memiliki ID yang dimulai dengan `igw-` (misalnya,`igw-83cddaex`). Subnet publik yang Anda buat dengan menggunakan Toolkit for Visual Studio memiliki nilai tag yang mengidentifikasi mereka sebagai publik.
- Dalam daftar drop-down Instances Subnet, tentukan subnet pribadi. Toolkit for Visual Studio menyebarkan instans Amazon EC2 untuk aplikasi Anda ke subnet pribadi.

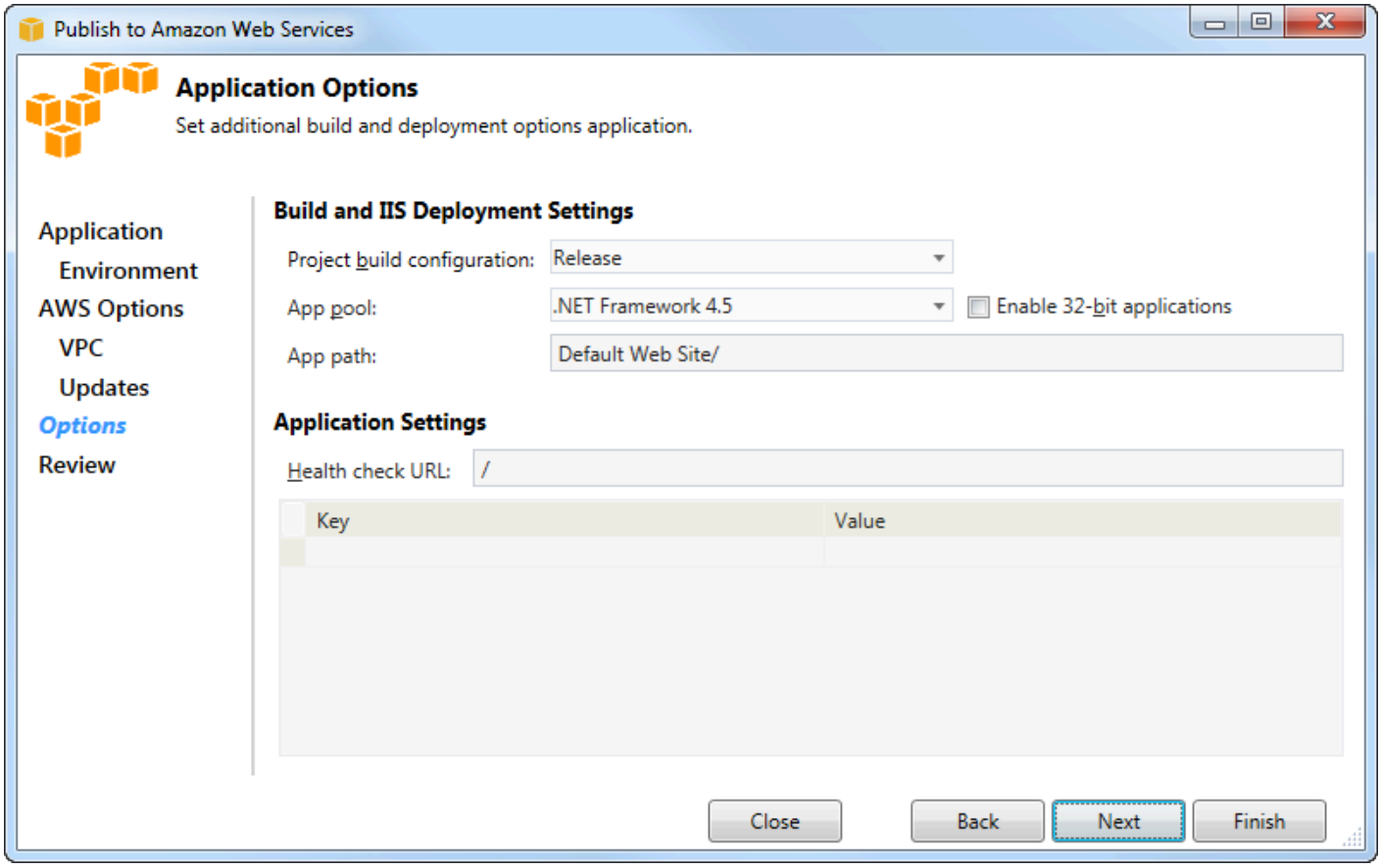
- Instans Amazon EC2 untuk aplikasi Anda berkomunikasi dari subnet pribadi ke Internet melalui instans Amazon EC2 di subnet publik yang melakukan terjemahan alamat jaringan (NAT). Untuk mengaktifkan komunikasi ini, Anda memerlukan grup keamanan VPC yang memungkinkan lalu lintas mengalir dari subnet pribadi ke instance NAT. Tentukan grup keamanan VPC ini di daftar drop-down Grup Keamanan.

Untuk informasi lebih lanjut tentang cara menerapkan aplikasi Elastic Beanstalk ke VPC, buka AWS Panduan Pengembang Elastic Beanstalk.

1. Setelah Anda mengisi semua informasi di halaman Opsi VPC, pilih Berikutnya.
   - Jika Anda memilih Aktifkan Penerapan Bergulir, halaman Rolling Deployments akan muncul.
   - Jika Anda tidak memilih Aktifkan Penerapan Bergulir, halaman Opsi Aplikasi akan muncul. Lewati ke petunjuk nanti di bagian ini yang menjelaskan cara menggunakan halaman Opsi Aplikasi.
2. Jika Anda memilih Aktifkan Penerapan Bergulir, Anda menentukan informasi di halaman Penyebaran Bergulir untuk mengonfigurasi cara versi baru aplikasi Anda diterapkan ke instans di lingkungan yang seimbang beban. Misalnya, jika Anda memiliki empat instance di lingkungan Anda dan Anda ingin mengubah jenis instance, Anda dapat mengonfigurasi lingkungan untuk mengubah dua instance sekaligus. Ini membantu memastikan aplikasi Anda masih berjalan saat perubahan sedang dilakukan.



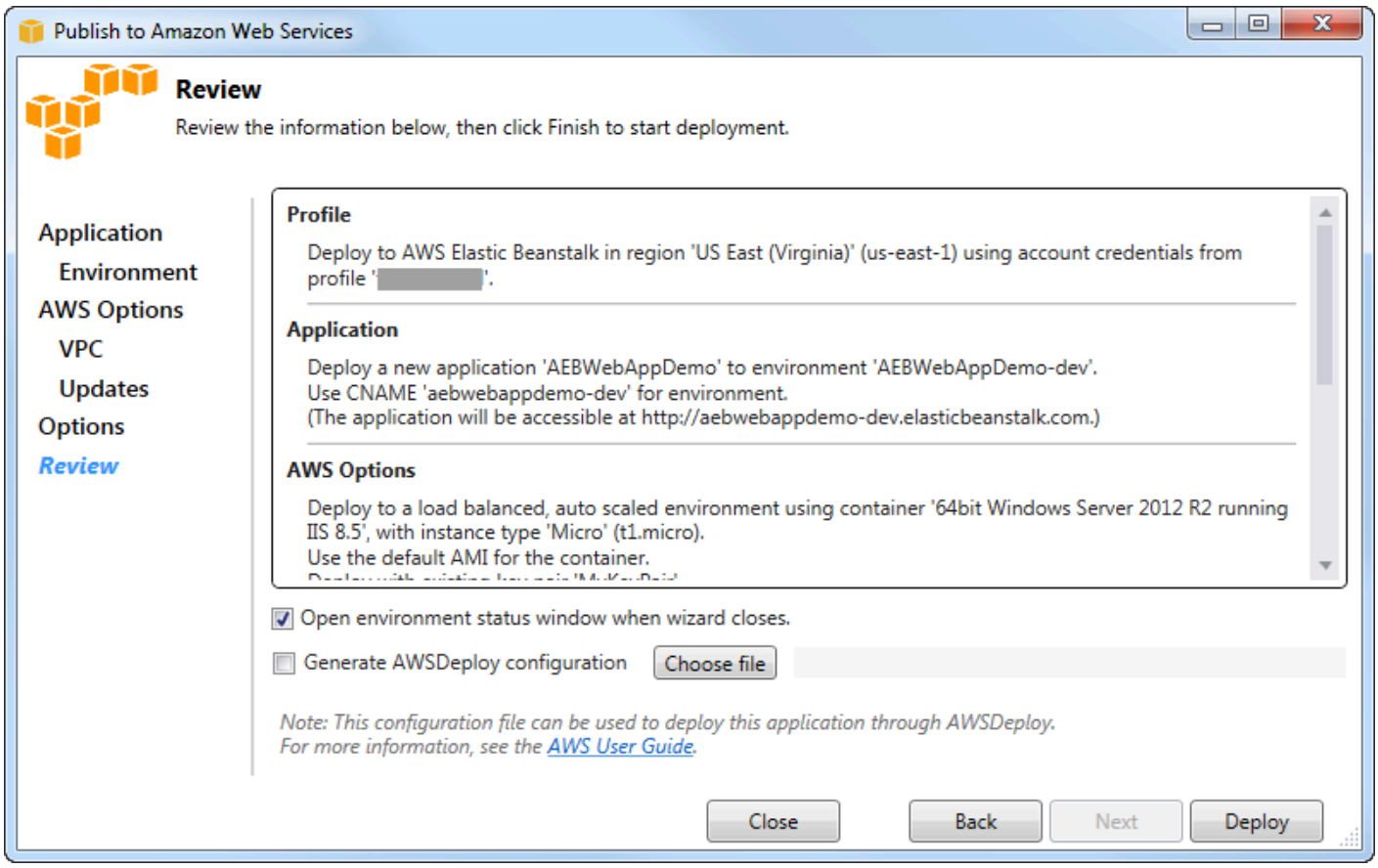
3. Di area Versi Aplikasi, pilih opsi untuk mengontrol penerapan ke persentase atau jumlah instance sekaligus. Tentukan persentase atau angka yang diinginkan.
4. Secara opsional, di area Konfigurasi Lingkungan, pilih kotak jika Anda ingin menentukan jumlah instance yang tetap dalam layanan selama penerapan. Jika Anda memilih kotak ini, tentukan jumlah maksimum instance yang harus dimodifikasi pada satu waktu, jumlah minimum instance yang harus tetap dalam layanan pada satu waktu, atau keduanya.
5. Pilih Berikutnya.
6. Pada halaman Opsi Aplikasi, Anda menentukan informasi tentang build, Internet Information Services (IIS), dan pengaturan aplikasi.



7. Di area Pengaturan Penerapan Build dan IIS, dalam daftar drop-down konfigurasi build Project, pilih konfigurasi build target. Jika wizard dapat menemukannya, Rilis muncul sebaliknya, konfigurasi aktif ditampilkan di kotak ini.
8. Dalam daftar drop-down App pool, pilih versi.NET Framework yang diperlukan oleh aplikasi Anda. Versi.NET Framework yang benar seharusnya sudah ditampilkan.
9. Jika aplikasi Anda 32-bit, pilih kotak Aktifkan aplikasi 32-bit.
10. Di kotak Jalur aplikasi, tentukan jalur yang akan digunakan IIS untuk menyebarkan aplikasi. Secara default, Situs Web Default/ditentukan, yang biasanya diterjemahkan ke jalur. `c:\inetpub\wwwroot` Jika Anda menentukan jalur selain Default Web Site/, wizard akan menempatkan redirect di Default Web Site/ path yang menunjuk ke jalur yang Anda tentukan.
11. Di area Pengaturan Aplikasi, di kotak URL centang Kesehatan, ketik URL untuk Elastic Beanstalk untuk memeriksa untuk menentukan apakah aplikasi web Anda masih responsif. URL ini relatif terhadap URL server root. URL server root ditentukan secara default. Misalnya, jika URL lengkapnya`example.com/site-is-up.html`, Anda akan mengetik`/site-is-up.html`.
12. Di area untuk Kunci dan Nilai, Anda dapat menentukan pasangan kunci dan nilai apa pun yang ingin Anda tambahkan ke `Web.config` file aplikasi Anda.

> **Note**
>
> Meskipun tidak disarankan, Anda dapat menggunakan area untuk Kunci dan Nilai, untuk menentukan AWS kredensi di mana aplikasi Anda harus berjalan. Pendekatan yang lebih disukai adalah menentukan peran IAM dalam daftar drop-down Identity and Access Management Role pada halaman AWS Options. Namun, jika Anda harus menggunakan AWS kredensi alih-alih peran IAM untuk menjalankan aplikasi Anda, di baris Kunci, pilih Kunci. AWSAccess Di baris Nilai, ketik kunci akses. Ulangi langkah-langkah ini untuk AWSSecretKey.

13 Pilih Berikutnya.



14 Pada halaman Tinjauan, tinjau opsi yang Anda konfigurasikan, dan pilih kotak Buka status lingkungan saat wizard ditutup.

15 Jika semuanya terlihat benar, pilih Deploy.

> **Note**
>
> Saat Anda menerapkan aplikasi, akun aktif akan dikenakan biaya untuk AWS sumber daya yang digunakan oleh aplikasi.

Informasi tentang penyebaran akan muncul di bilah status Visual Studio dan jendela Output. Mungkin perlu beberapa menit. Ketika penyebaran selesai, pesan konfirmasi akan muncul di jendela Output.

16. Untuk menghapus penyebaran, di AWS Explorer, perluas simpul Elastic Beanstalk, buka menu konteks (klik kanan) untuk subnode untuk penyebaran, lalu pilih Delete. Proses penghapusan mungkin memakan waktu beberapa menit.

Menyebarkan Aplikasi Inti ASP.NET ke Elastic Beanstalk (Legacy)

> **Important**
>
> Dokumentasi ini mengacu pada layanan dan fitur lama. Untuk panduan dan konten yang diperbarui, lihat panduan alat penyebaran AWS .NET dan AWS daftar isi Deploying to yang diperbarui.

AWS Elastic Beanstalk adalah layanan yang menyederhanakan proses penyediaan AWS sumber daya untuk aplikasi Anda. AWS Elastic Beanstalk menyediakan semua AWS infrastruktur yang diperlukan untuk menyebarkan aplikasi Anda.

Toolkit for Visual Studio mendukung penerapan aplikasi ASP.NET Core untuk menggunakan AWS Elastic Beanstalk. ASP.NET Core adalah desain ulang ASP.NET dengan arsitektur termodulasi yang meminimalkan overhead ketergantungan dan merampingkan aplikasi Anda untuk berjalan di cloud.

AWS Elastic Beanstalk membuatnya mudah untuk menyebarkan aplikasi dalam berbagai bahasa yang berbeda untuk AWS. Elastic Beanstalk mendukung aplikasi ASP.NET tradisional dan aplikasi ASP.NET Core. Topik ini menjelaskan penerapan aplikasi ASP.NET Core.

Menggunakan Deployment Wizard

Cara termudah untuk menyebarkan aplikasi ASP.NET Core ke Elastic Beanstalk adalah dengan Toolkit for Visual Studio.

Jika Anda telah menggunakan toolkit sebelumnya untuk menyebarkan ASP tradisional. NET, Anda akan menemukan pengalaman untuk ASP.NET Core menjadi sangat mirip. Pada langkah-langkah di bawah ini, kita akan menelusuri pengalaman penerapan.

Jika Anda belum pernah menggunakan toolkit sebelumnya, hal pertama yang harus Anda lakukan setelah menginstal toolkit adalah mendaftarkan AWS kredensil Anda dengan toolkit. Lihat Cara Menentukan Kredensial AWS Keamanan untuk Aplikasi Anda untuk dokumentasi Visual Studio untuk detail tentang cara melakukannya.

Untuk menyebarkan aplikasi web ASP.NET Core, klik kanan proyek di Solution Explorer dan pilih Publish to... AWS

Pada halaman pertama panduan Publish to AWS Elastic Beanstalk deployment, pilih untuk membuat aplikasi Elastic Beanstalk baru. Aplikasi Elastic Beanstalk adalah sebuah koleksi logis komponen Elastic Beanstalk, termasuk Lingkungan,versi, dan Konfigurasi lingkungan. Wizard penerapan menghasilkan aplikasi yang pada gilirannya berisi kumpulan versi dan lingkungan aplikasi. Lingkungan berisi AWS sumber daya aktual yang menjalankan versi aplikasi. Setiap kali Anda menerapkan aplikasi, versi aplikasi baru dibuat dan wizard mengarahkan lingkungan ke versi itu. Anda dapat mempelajari lebih lanjut tentang konsep-konsep ini di Komponen Elastic Beanstalk. .

Selanjutnya, tetapkan nama untuk aplikasi dan lingkungan pertamanya. Setiap lingkungan memiliki CNAME unik yang terkait dengannya yang dapat Anda gunakan untuk mengakses aplikasi saat penerapan selesai.

Halaman berikutnya, AWS Opsi, memungkinkan Anda mengonfigurasi jenis AWS sumber daya yang akan digunakan. Untuk contoh ini, tinggalkan nilai default, kecuali untuk bagian Pasangan kunci. Pasangan kunci memungkinkan Anda mengambil kata sandi administrator Windows sehingga Anda dapat masuk ke mesin. Jika Anda belum membuat key pair, Anda mungkin ingin memilih Create new key pair.

Izin

Halaman Izin digunakan untuk menetapkan AWS kredensional ke instans EC2 yang menjalankan aplikasi Anda. Ini penting jika aplikasi Anda menggunakan AWS SDK untuk .NET untuk mengakses

AWS layanan lain. Jika Anda tidak menggunakan layanan lain dari aplikasi Anda maka Anda dapat meninggalkan halaman ini secara default.

Opsi Aplikasi

Rincian pada halaman Opsi Aplikasi berbeda dari yang ditentukan saat menerapkan aplikasi ASP.NET tradisional. Di sini, Anda menentukan konfigurasi build dan kerangka kerja yang digunakan untuk mengemas aplikasi, dan juga menentukan jalur sumber daya IIS untuk aplikasi.

Setelah menyelesaikan halaman Opsi Aplikasi, klik Berikutnya untuk meninjau pengaturan, lalu klik Deploy untuk memulai proses penyebaran.

Memeriksa Status Lingkungan

Setelah aplikasi dikemas dan diunggah AWS, Anda dapat memeriksa status lingkungan Elastic Beanstalk dengan membuka tampilan status lingkungan dari Explorer di Visual Studio. AWS

Acara ditampilkan di bilah status saat lingkungan online. Setelah semuanya selesai, status lingkungan akan beralih ke keadaan sehat. Anda dapat mengklik URL untuk melihat situs. Dari sini, Anda juga dapat menarik log dari lingkungan atau desktop jarak jauh ke instans Amazon EC2 yang merupakan bagian dari lingkungan Elastic Beanstalk Anda.

Penerapan pertama aplikasi apa pun akan memakan waktu sedikit lebih lama daripada penerapan ulang berikutnya, karena menciptakan sumber daya baru. AWS Saat Anda mengulangi aplikasi Anda selama pengembangan, Anda dapat dengan cepat menerapkan ulang dengan kembali melalui wizard, atau memilih opsi Republish ketika Anda mengklik kanan proyek.

Publikasikan ulang paket aplikasi Anda menggunakan pengaturan dari proses sebelumnya melalui panduan penerapan dan unggah bundel aplikasi ke lingkungan Elastic Beanstalk yang ada.

Cara Menentukan Kredensi AWS Keamanan untuk Aplikasi Anda

AWS Akun yang Anda tentukan dalam wizard Publish to Elastic Beanstalk AWS adalah akun yang akan digunakan wizard untuk penyebaran ke Elastic Beanstalk.

Meskipun tidak disarankan, Anda mungkin juga perlu menentukan kredensi AWS akun yang akan digunakan aplikasi Anda untuk mengakses AWS layanan setelah digunakan. Pendekatan yang lebih disukai adalah menentukan peran IAM. Di wizard Publish to Elastic Beanstalk, Anda melakukannya melalui daftar drop-down Peran Identity and Access Management Role di halaman Opsi.AWS Di

wizard Publikasikan ke Amazon Web Services lama, Anda melakukannya melalui daftar drop-down Peran IAM di halaman Opsi.AWS

Jika Anda harus menggunakan kredensi AWS akun alih-alih peran IAM, Anda dapat menentukan kredensi AWS akun untuk aplikasi Anda dengan salah satu cara berikut:

- Referensi profil yang sesuai dengan kredensi AWS akun dalam `appSettings` elemen file proyek. `Web.config` (Untuk membuat profil, lihat Mengonfigurasi AWS Kredensial.) Contoh berikut menentukan kredensional yang nama profilnya. `myProfile`

```
<appSettings>
  <!-- AWS CREDENTIALS -->
  <add key="AWSProfileName" value="myProfile"/>
</appSettings>
```

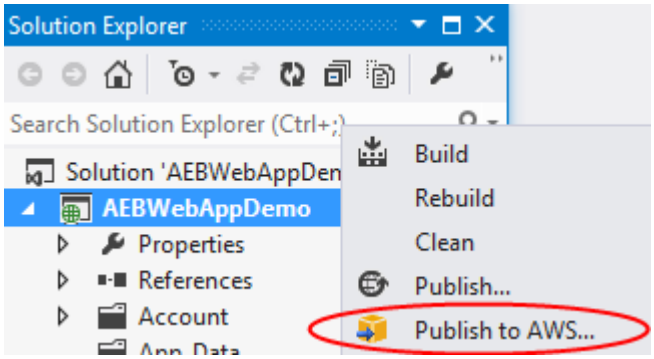
- Jika Anda menggunakan wizard Publish to Elastic Beanstalk, pada halaman Opsi Aplikasi, di baris Kunci area Kunci dan Nilai, pilih. AWS AccessKey Di baris Nilai, ketik kunci akses. Ulangi langkah-langkah ini untuk AWS SecretKey.
- Jika Anda menggunakan wizard Publish to Amazon Web Services lama, pada halaman Opsi Aplikasi, di area Kredensial Aplikasi, pilih Gunakan kredensial ini, lalu ketik kunci akses dan kunci akses rahasia ke dalam kotak Kunci Akses dan Kunci Rahasia.

Cara Mempublikasikan Ulang Aplikasi Anda ke Lingkungan Elastic Beanstalk (Legacy)

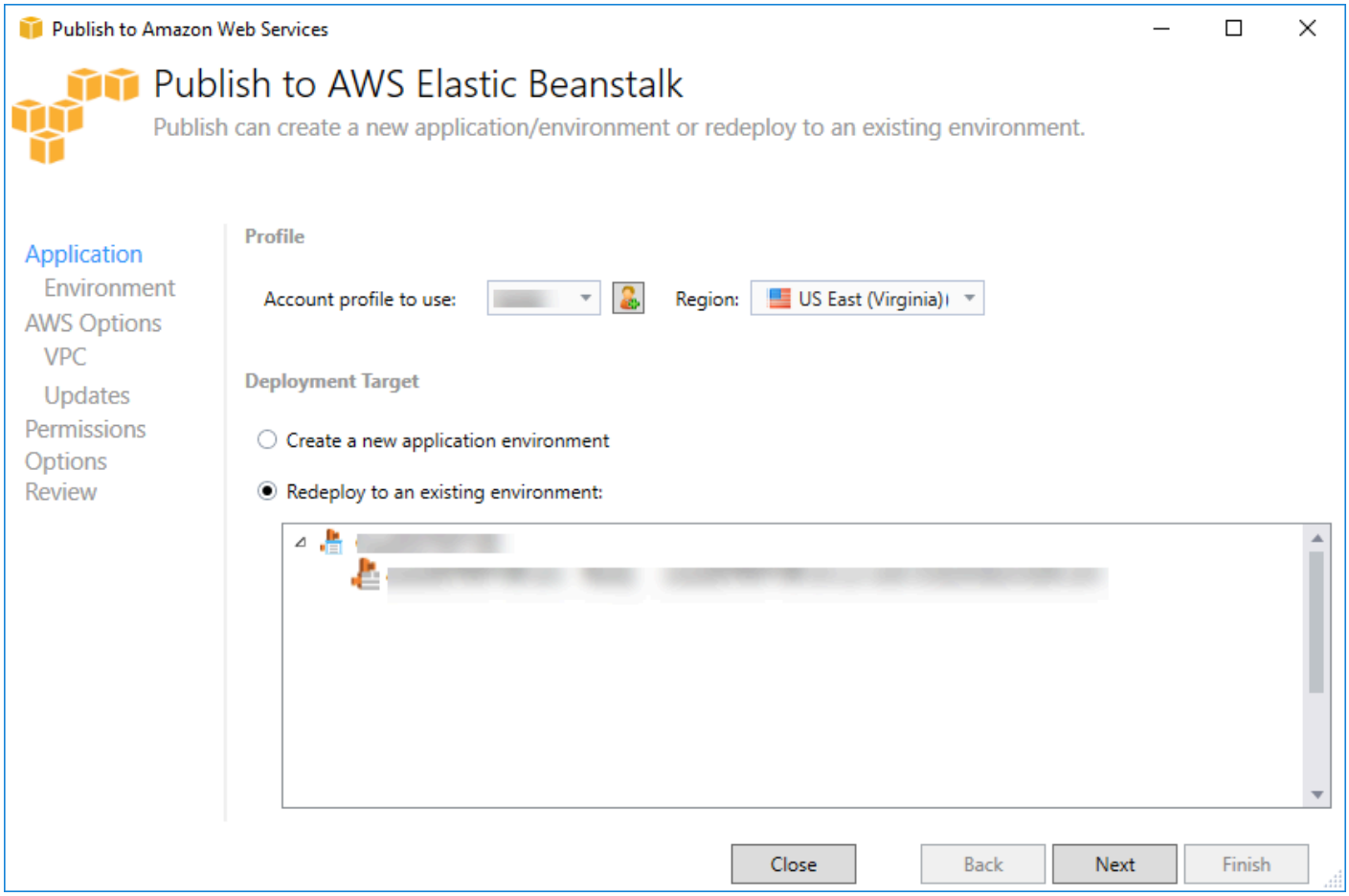
> **Important**
>
> Dokumentasi ini mengacu pada layanan dan fitur lama. Untuk panduan dan konten yang diperbarui, lihat panduan alat penyebaran AWS .NET.

Anda dapat mengulangi aplikasi Anda dengan membuat perubahan diskrit dan kemudian menerbitkan ulang versi baru ke lingkungan Elastic Beanstalk yang sudah diluncurkan.

1. Di Solution Explorer, buka menu konteks (klik kanan) untuk folder AEBWebAppDemoproyek untuk proyek yang Anda terbitkan di bagian sebelumnya, dan pilih Publish to AWS Elastic Beanstalk.

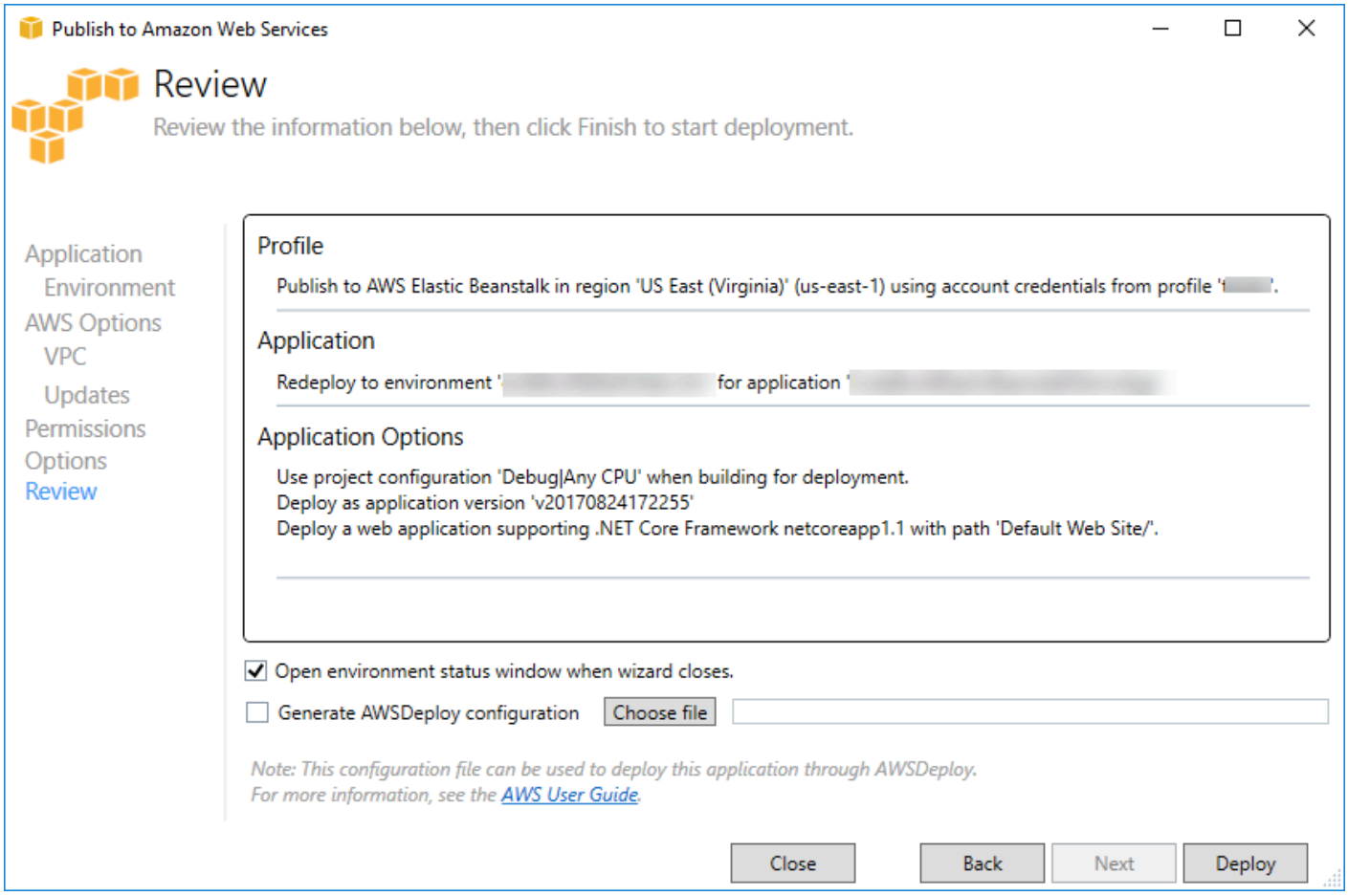


Wisaya Publish to Elastic Beanstalk muncul.



2. Pilih Menerapkan ulang ke lingkungan yang ada dan pilih lingkungan yang sebelumnya Anda publikasikan. Klik Berikutnya.

   Wizard Review muncul.



3. Klik Terapkan. Aplikasi akan dipindahkan ke lingkungan yang sama.

Anda tidak dapat mempublikasikan ulang jika aplikasi Anda sedang dalam proses peluncuran atau penghentian.

Penerapan Aplikasi Elastic Beanstalk Kustom

Topik ini menjelaskan bagaimana manifes penerapan untuk container Microsoft Windows Elastic Beanstalk mendukung penerapan aplikasi kustom.

Penerapan aplikasi khusus adalah fitur canggih untuk pengguna tingkat lanjut yang ingin memanfaatkan kekuatan Elastic Beanstalk untuk membuat dan mengelola sumber daya AWS mereka, tetapi menginginkan kontrol penuh tentang bagaimana aplikasi mereka digunakan. Untuk penerapan aplikasi khusus, Anda membuat PowerShell skrip Windows untuk tiga tindakan berbeda yang dilakukan Elastic Beanstalk. Tindakan penginstalan digunakan saat penerapan dimulai, restart digunakan saat `RestartAppServer` API dipanggil dari toolkit atau konsol web, dan uninstall yang dipanggil pada penerapan sebelumnya setiap kali penerapan baru terjadi.

Misalnya, Anda mungkin memiliki aplikasi ASP.NET yang ingin Anda terapkan saat tim dokumentasi Anda telah menulis situs web statis yang ingin mereka sertakan dengan penerapan. Anda dapat melakukannya dengan menulis manifes penerapan seperti ini:

```
{
  "manifestVersion": 1,
  "deployments": {

    "msDeploy": [
      {
        "name": "app",
        "parameters": {
          "appBundle": "CoolApp.zip",
          "iisPath": "/"
        }
      }
    ],
    "custom": [
      {
        "name": "PowerShellDocs",
        "scripts": {
          "install": {
            "file": "install.ps1"
          },
          "restart": {
            "file": "restart.ps1"
          },
          "uninstall": {
            "file": "uninstall.ps1"
          }
        }
      }
    ]
  }
}
```

Skrip yang terdaftar untuk setiap tindakan harus berada dalam bundel aplikasi relatif terhadap file manifes penerapan. Untuk contoh ini, bundel aplikasi juga akan berisi file documentation.zip yang berisi situs web statis yang dibuat oleh tim dokumentasi Anda.

`install.ps1`Script mengekstrak file zip dan mengatur IIS Path.

```
Add-Type -assembly "system.io.compression.filesystem"
[io.compression.zipfile]::ExtractToDirectory('./documentation.zip', 'c:\inetpub\wwwroot
\documentation')

powershell.exe -Command {New-WebApplication -Name documentation -PhysicalPath  c:
\inetpub\wwwroot\documentation -Force}
```

Karena aplikasi Anda berjalan di IIS, tindakan restart akan memanggil reset IIS.

```
iisreset /timeout:1
```

Untuk menghapus skrip, penting untuk membersihkan semua pengaturan dan file yang digunakan selama tahap penginstalan. Dengan begitu selama fase instalasi untuk versi baru, Anda dapat menghindari tabrakan dengan penerapan sebelumnya. Untuk contoh ini, Anda perlu menghapus aplikasi IIS untuk situs web statis dan menghapus file situs web.

```
powershell.exe -Command {Remove-WebApplication -Name documentation}
Remove-Item -Recurse -Force 'c:\inetpub\wwwroot\documentation'
```

Dengan file skrip ini dan file documentation.zip yang disertakan dalam bundel aplikasi Anda, penyebaran membuat aplikasi ASP.NET dan kemudian menyebarkan situs dokumentasi.

Untuk contoh ini, kami memilih contoh sederhana yang menyebarkan situs web statis sederhana, tetapi dengan penerapan aplikasi khusus Anda dapat menerapkan semua jenis aplikasi dan membiarkan Elastic Beanstalk mengelola sumber daya untuk itu. AWS

Penyebaran Elastic Beanstalk Inti ASP.NET Kustom

Topik ini menjelaskan cara kerja penerapan dan apa yang dapat Anda lakukan untuk menyesuaikan penerapan saat membuat aplikasi ASP.NET Core dengan Elastic Beanstalk dan Toolkit for Visual Studio.

Setelah Anda menyelesaikan wizard penerapan di Toolkit for Visual Studio, toolkit akan membundel aplikasi dan mengirimkannya ke Elastic Beanstalk. Langkah pertama Anda dalam membuat bundel aplikasi adalah dengan menggunakan CLI dotnet baru untuk mempersiapkan aplikasi untuk penerbitan dengan menggunakan perintah publish. Kerangka kerja dan konfigurasi diturunkan dari pengaturan di wizard ke perintah publish. Jadi jika Anda memilih Rilis untuk `configuration` dan netcoreapp1.0 untuk`framework`, toolkit akan menjalankan perintah berikut:

```
dotnet publish --configuration Release --framework netcoreapp1.0
```

Saat perintah publish selesai, toolkit akan menulis manifes penerapan baru ke dalam folder penerbitan. Manifes penyebaran adalah file JSON bernama aws-windows-deployment-manifest.json, yang dibaca oleh wadah Elastic Beanstalk Windows (versi 1.2 atau yang lebih baru) untuk menentukan cara menerapkan aplikasi. Misalnya, untuk aplikasi ASP.NET Core yang ingin Anda gunakan di root IIS, toolkit menghasilkan file manifes yang terlihat seperti ini:

```
{
  "manifestVersion": 1,
  "deployments": {

    "aspNetCoreWeb": [
      {
        "name": "app",
        "parameters": {
          "appBundle": ".",
          "iisPath": "/",
          "iisWebSite": "Default Web Site"
        }
      }
    ]
  }
}
```

`appBundle`Properti menunjukkan di mana bit aplikasi dalam kaitannya dengan file manifes. Properti ini dapat menunjuk ke direktori atau arsip ZIP. `iisWebSite`Properti `iisPath` dan menunjukkan di mana di IIS untuk meng-host aplikasi.

Menyesuaikan Manifest

Toolkit hanya menulis file manifes jika belum ada di folder penerbitan. Jika file memang ada, toolkit akan memperbarui `iisWebSite` propertiappBundle, `iisPath` dan di aplikasi pertama yang tercantum di bawah `aspNetCoreWeb` bagian manifes. Ini memungkinkan Anda untuk menambahkan aws-windows-deployment-manifest.json ke proyek Anda dan menyesuaikan manifes. Untuk melakukan ini untuk aplikasi ASP.NET Core Web di Visual Studio tambahkan file JSON baru ke root proyek dan beri nama .json. aws-windows-deployment-manifest

Manifes harus diberi nama aws-windows-deployment-manifest.json dan harus berada di root proyek. Wadah Elastic Beanstalk mencari manifes di root dan jika menemukannya, ia akan memanggil

perkakas penerapan. Jika file tidak ada, wadah Elastic Beanstalk kembali ke perkakas penerapan yang lebih lama, yang mengasumsikan arsip adalah arsip msdeploy.

Untuk memastikan `publish` perintah CLI dotnet menyertakan manifes, perbarui `project.json` file untuk menyertakan file manifes di bagian include di bawah. `include` `publishOptions`

```
{
   "publishOptions": {
     "include": [
       "wwwroot",
       "Views",
       "Areas/**/Views",
       "appsettings.json",
       "web.config",
       "aws-windows-deployment-manifest.json"
     ]
   }
 }
```

Sekarang setelah Anda mendeklarasikan manifes sehingga disertakan dalam bundel aplikasi, Anda dapat mengonfigurasi lebih lanjut bagaimana Anda ingin menerapkan aplikasi. Anda dapat menyesuaikan penerapan di luar apa yang didukung oleh wizard penerapan. AWS telah mendefinisikan skema JSON untuk aws-windows-deployment-manifestfile.json, dan ketika Anda menginstal Toolkit for Visual Studio, penyiapan mendaftarkan URL untuk skema tersebut.

Saat Anda membuka`windows-deployment-manifest.json`, Anda akan melihat URL skema yang dipilih di kotak tarik-turun Skema. Anda dapat menavigasi ke URL untuk mendapatkan deskripsi lengkap tentang apa yang dapat diatur dalam manifes. Dengan skema yang dipilih, Visual Studio akan menyediakan IntelliSense saat Anda mengedit manifes.

Salah satu penyesuaian yang dapat Anda lakukan adalah mengkonfigurasi kumpulan aplikasi IIS di mana aplikasi akan berjalan. Contoh berikut menunjukkan bagaimana Anda dapat menentukan kumpulan Aplikasi IIS (“CustomPool”) yang mendaur ulang proses setiap 60 menit, dan menetapkannya ke aplikasi menggunakan. `"appPool": "customPool"`

```
{
  "manifestVersion": 1,
  "iisConfig": {
    "appPools": [
      {
```

```
        "name": "customPool",
        "recycling": {
          "regularTimeInterval": 60
        }
      }
    ]
  },
  "deployments": {
    "aspNetCoreWeb": [
      {
        "name": "app",
        "parameters": {
          "appPool": "customPool"
        }
      }
    ]
  }
}
```

Selain itu, manifes dapat mendeklarasikan PowerShell skrip Windows untuk dijalankan sebelum dan sesudah tindakan penginstalan, restart, dan hapus instalan. Misalnya, manifes berikut menjalankan PowerShell skrip Windows `PostInstallSetup.ps1` untuk melakukan pekerjaan penyiapan lebih lanjut setelah aplikasi ASP.NET Core dikerahkan ke IIS. Saat menambahkan skrip seperti ini, pastikan skrip ditambahkan ke bagian include di bawah PublisHoptions dalam file, seperti yang Anda lakukan dengan `project.json` file tersebut. `aws-windows-deployment-manifest.json` Jika tidak, skrip tidak akan disertakan sebagai bagian dari perintah publikasi CLI dotnet.

```
{
  "manifestVersion": 1,
  "deployments": {
    "aspNetCoreWeb": [
      {
        "name": "app",
        "scripts": {
          "postInstall": {
            "file": "SetupScripts/PostInstallSetup.ps1"
          }
        }
      }
    ]
  }
}
```

Bagaimana dengan.ebextensions?

File konfigurasi Elastic Beanstalk .ebextensions didukung seperti semua wadah Elastic Beanstalk lainnya. Untuk menyertakan.ebextensions dalam aplikasi ASP.NET Core, tambahkan `.ebextensions` direktori ke bagian di bawah file. `include` `publishOptions` `project.json` Untuk informasi lebih lanjut tentang.ebextensions, lihat Panduan Pengembang Elastic Beanstalk.

Dukungan Beberapa Aplikasi untuk.NET dan Elastic Beanstalk

Menggunakan manifes penyebaran, Anda memiliki kemampuan untuk menyebarkan beberapa aplikasi ke lingkungan Elastic Beanstalk yang sama.

Manifes penyebaran mendukung aplikasi web ASP.NET Core serta arsip msdeploy untuk aplikasi ASP.NET tradisional. Bayangkan sebuah skenario di mana Anda telah menulis aplikasi luar biasa baru menggunakan ASP.NET Core untuk frontend dan proyek API Web untuk API ekstensi. Anda juga memiliki aplikasi admin yang Anda tulis menggunakan ASP.NET tradisional.

Wizard penerapan toolkit berfokus pada penerapan satu proyek. Untuk memanfaatkan beberapa penerapan aplikasi, Anda harus membuat bundel aplikasi dengan tangan. Untuk memulai, tulis manifes. Untuk contoh ini, Anda akan menulis manifes di akar solusi Anda.

Bagian penyebaran dalam manifes memiliki dua anak: array aplikasi web ASP.NET Core untuk diterapkan, dan array arsip msdeploy untuk diterapkan. Untuk setiap aplikasi, Anda mengatur jalur IIS dan lokasi bit aplikasi relatif terhadap manifes.

```
{
  "manifestVersion": 1,
  "deployments": {

    "aspNetCoreWeb": [
      {
        "name": "frontend",
        "parameters": {
          "appBundle": "./frontend",
          "iisPath": "/frontend"
        }
      },
      {
        "name": "ext-api",
        "parameters": {
          "appBundle": "./ext-api",
```

```
          "iisPath": "/ext-api"
        }
      }
    ],
    "msDeploy": [
      {
        "name": "admin",
        "parameters": {
          "appBundle": "AmazingAdmin.zip",
          "iisPath": "/admin"
        }
      }
    ]
  }
}
```

Dengan manifes tertulis, Anda akan menggunakan Windows PowerShell untuk membuat bundel aplikasi dan memperbarui lingkungan Elastic Beanstalk yang ada untuk menjalankannya. Script ditulis dengan asumsi bahwa itu akan dijalankan dari folder yang berisi solusi Visual Studio Anda.

Hal pertama yang perlu Anda lakukan dalam skrip adalah menyiapkan folder ruang kerja untuk membuat bundel aplikasi.

```
$publishFolder = "c:\temp\publish"

$publishWorkspace = [System.IO.Path]::Combine($publishFolder, "workspace")
$appBundle = [System.IO.Path]::Combine($publishFolder, "app-bundle.zip")

If (Test-Path $publishWorkspace){
  Remove-Item $publishWorkspace -Confirm:$false -Force
}
If (Test-Path $appBundle){
  Remove-Item $appBundle -Confirm:$false -Force
}
```

Setelah Anda membuat folder, sekarang saatnya untuk menyiapkan frontend. Seperti halnya wizard penerapan, gunakan CLI dotnet untuk mempublikasikan aplikasi.

```
Write-Host 'Publish the ASP.NET Core frontend'
$publishFrontendFolder = [System.IO.Path]::Combine($publishWorkspace, "frontend")
dotnet publish .\src\AmazingFrontend\project.json -o $publishFrontendFolder -c Release
 -f netcoreapp1.0
```

Perhatikan bahwa subfolder “frontend” digunakan untuk folder keluaran, cocok dengan folder yang Anda atur dalam manifes. Sekarang Anda perlu melakukan hal yang sama untuk proyek API Web.

```
Write-Host 'Publish the ASP.NET Core extensibility API'
$publishExtAPIFolder = [System.IO.Path]::Combine($publishWorkspace, "ext-api")
dotnet publish .\src\AmazingExtensibleAPI\project.json -o $publishExtAPIFolder -c
 Release -f netcoreapp1.0
```

Situs admin adalah aplikasi ASP.NET tradisional, jadi Anda tidak dapat menggunakan CLI dotnet. Untuk aplikasi admin, Anda harus menggunakan msbuild, meneruskan paket target build untuk membuat arsip msdeploy. Secara default target paket membuat arsip msdeploy di bawah `obj \Release\Package` folder, jadi Anda harus menyalin arsip ke ruang kerja publikasi.

```
Write-Host 'Create msdeploy archive for admin site'
msbuild .\src\AmazingAdmin\AmazingAdmin.csproj /t:package /p:Configuration=Release
Copy-Item .\src\AmazingAdmin\obj\Release\Package\AmazingAdmin.zip $publishWorkspace
```

Untuk memberi tahu lingkungan Elastic Beanstalk apa yang harus dilakukan dengan semua aplikasi ini, salin manifes dari solusi Anda ke ruang kerja publikasi dan kemudian zip folder.

```
Write-Host 'Copy deployment manifest'
Copy-Item .\aws-windows-deployment-manifest.json $publishWorkspace

Write-Host 'Zipping up publish workspace to create app bundle'
Add-Type -assembly "system.io.compression.filesystem"
[io.compression.zipfile]::CreateFromDirectory( $publishWorkspace, $appBundle)
```

Sekarang setelah Anda memiliki bundel aplikasi, Anda dapat pergi ke konsol web dan mengunggah arsip ke lingkungan Elastic Beanstalk. Atau, Anda dapat terus menggunakan AWS PowerShell cmdlet untuk memperbarui lingkungan Elastic Beanstalk dengan bundel aplikasi. Pastikan Anda telah mengatur profil dan wilayah saat ini ke profil dan wilayah yang berisi lingkungan Elastic Beanstalk Anda dengan menggunakan dan cmdlet. `Set-AWSCredentials` `Set-DefaultAWSRegion`

```
Write-Host 'Write application bundle to S3'
# Determine S3 bucket to store application bundle
$s3Bucket = New-EBStorageLocation
Write-S3Object -BucketName $s3Bucket -File $appBundle


$applicationName = "ASPNETCoreOnAWS"
```

```
$environmentName = "ASPNETCoreOnAWS-dev"
$versionLabel = [System.DateTime]::Now.Ticks.ToString()

Write-Host 'Update Beanstalk environment for new application bundle'
New-EBApplicationVersion -ApplicationName $applicationName -VersionLabel $versionLabel
 -SourceBundle_S3Bucket $s3Bucket -SourceBundle_S3Key app-bundle.zip
Update-EBEnvironment -ApplicationName $applicationName -EnvironmentName
 $environmentName -VersionLabel $versionLabel
```

Sekarang, periksa status pembaruan menggunakan halaman status lingkungan Elastic Beanstalk di toolkit atau konsol web. Setelah selesai, Anda akan dapat menavigasi ke setiap aplikasi yang Anda gunakan di jalur IIS yang diatur dalam manifes penerapan.

Menyebarkan ke Amazon EC2 Container Service

> **Important**
>
> AWS Fitur Publish to yang baru dirancang untuk menyederhanakan cara Anda mempublikasikan aplikasi.NET. AWS Anda mungkin ditanya apakah Anda ingin beralih ke pengalaman penerbitan ini setelah Anda memilih Publish Container ke AWS. Untuk informasi selengkapnya, lihat Bekerja dengan Publish to AWS di Visual Studio.

Amazon Elastic Container Service adalah layanan manajemen kontainer yang sangat skalabel dan berkinerja tinggi yang mendukung kontainer Docker dan memungkinkan Anda menjalankan aplikasi dengan mudah di kluster instans Amazon EC2 yang dikelola.

Untuk menerapkan aplikasi di Amazon Elastic Container Service, komponen aplikasi Anda harus dikembangkan agar berjalan di container Docker. Container Docker adalah unit standar pengembangan perangkat lunak, yang berisi semua yang perlu dijalankan oleh aplikasi perangkat lunak Anda: kode, runtime, alat sistem, pustaka sistem, dll.

Toolkit for Visual Studio menyediakan wizard yang menyederhanakan penerbitan aplikasi melalui Amazon ECS. Wizard ini dijelaskan di bagian berikut.

Untuk informasi selengkapnya tentang Amazon ECS, buka dokumentasi Elastic Container Service. Ini mencakup ikhtisar dasar-dasar Docker dan membuat cluster.

Topik

Tentukan AWS Kredensi untuk Aplikasi ASP.NET Core 2 Anda

Ada dua jenis kredenal yang sedang dimainkan saat Anda menerapkan aplikasi ke container Docker: kredensial penerapan dan kredensial instance.

Kredensial penerapan digunakan oleh AWS wizard Publish Container to untuk membuat lingkungan di Amazon ECS. Ini termasuk hal-hal seperti tugas, layanan, peran IAM, repositori kontainer Docker, dan jika Anda mau, penyeimbang beban.

Kredensi instans digunakan oleh instans (termasuk aplikasi Anda) untuk mengakses layanan yang berbeda AWS . Misalnya, jika aplikasi ASP.NET Core 2.0 Anda membaca dan menulis ke objek Amazon S3, itu akan memerlukan izin yang sesuai. Anda dapat memberikan kredensi yang berbeda menggunakan metode yang berbeda berdasarkan lingkungan. Misalnya, aplikasi ASP.NET Core 2 Anda mungkin menargetkan lingkungan Pengembangan dan Produksi. Anda dapat menggunakan instance Docker lokal dan kredensional untuk pengembangan dan peran yang ditentukan dalam produksi.

Menentukan kredenal penerapan

AWS Akun yang Anda tentukan di AWS wizard Publish Container to adalah AWS akun yang akan digunakan wizard untuk penyebaran ke Amazon ECS. Profil akun harus memiliki izin ke Amazon Elastic Compute Cloud, Amazon Elastic Container Service, dan. AWS Identity and Access Management

Jika Anda melihat opsi hilang dari daftar drop-down, itu mungkin karena Anda tidak memiliki izin. Misalnya, jika Anda membuat klaster untuk aplikasi Anda tetapi tidak melihatnya di halaman Publish Container to AWS wizard Cluster. Jika ini terjadi, tambahkan izin yang hilang dan coba wizard lagi.

Menentukan kredensial instance pengembangan

Untuk lingkungan non-produksi, Anda dapat mengonfigurasi kredensional Anda di pengaturan aplikasi. <environment>.json. Misalnya, untuk mengonfigurasi kredensional Anda di file AppSettings.Development.json di Visual Studio 2017:

1. Tambahkan AWSSDK .Extensions. NETCore.Setup NuGet paket untuk proyek Anda.

2. Tambahkan AWS pengaturan ke AppSettings.development.json. Konfigurasi di bawah ini menetapkan `Profile` dan`Region`.

```
{
    "AWS": {
        "Profile": "local-test-profile",
        "Region": "us-west-2"
    }
}
```

Menentukan kredensial instance produksi

Untuk instans produksi, kami sarankan Anda menggunakan peran IAM untuk mengontrol apa yang dapat diakses oleh aplikasi (dan layanan) Anda. Misalnya, untuk mengonfigurasi peran IAM dengan Amazon ECS sebagai prinsipal layanan dengan izin ke Amazon Simple Storage Service dan Amazon DynamoDB dari: Konsol Manajemen AWS

1. Masuk ke Konsol Manajemen AWS dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi konsol IAM, pilih Peran, lalu pilih Buat peran.
3. Pilih jenis peran AWS Layanan, lalu pilih EC2 Container Service.
4. Pilih kasus penggunaan EC2 Container Service Task. Kasus penggunaan ditentukan oleh layanan untuk menyertakan kebijakan kepercayaan yang diperlukan layanan. Kemudian pilih Selanjutnya: Izin.
5. Pilih kebijakan izin AmazonS3 FullAccess dan AmazonDynamoDBFullAccess. Centang kotak di samping setiap kebijakan, lalu pilih Berikutnya: Tinjau,
6. Untuk nama Peran, ketikkan nama peran atau akhiran nama peran untuk membantu Anda mengidentifikasi tujuan peran ini. Nama peran harus unik di akun AWS Anda. Grup tidak dibedakan berdasarkan huruf besar-kecil. Misalnya, Anda tidak dapat membuat peran dengan nama `PRODROLE` dan `prodrole`. Anda tidak dapat mengubah nama peran setelah dibuat karena berbagai entitas mungkin mereferensikan peran tersebut.
7. (Opsional) Untuk Deskripsi peran, ketikkan deskripsi untuk peran baru tersebut.
8. Tinjau peran dan kemudian pilih Buat peran.

Anda dapat menggunakan peran ini sebagai peran tugas di halaman Definisi Tugas ECS dari AWS panduan Publish Container to.

Untuk informasi selengkapnya, lihat Menggunakan Peran Berbasis Layanan.

Menyebarkan Aplikasi ASP.NET Core 2.0 ke Amazon ECS (Fargate) (Warisan)

> **Important**
>
> Dokumentasi ini mengacu pada layanan dan fitur lama. Untuk panduan dan konten yang diperbarui, lihat panduan alat penyebaran AWS .NET dan AWS daftar isi Deploying to yang diperbarui.

Bagian ini menjelaskan cara menggunakan AWS wizard Publish Container to, yang disediakan sebagai bagian dari Toolkit for Visual Studio, untuk menyebarkan aplikasi ASP.NET Core 2.0 dalam kontainer yang menargetkan Linux melalui Amazon ECS menggunakan tipe peluncuran Fargate. Karena aplikasi web dimaksudkan untuk berjalan terus menerus, itu akan digunakan sebagai layanan.

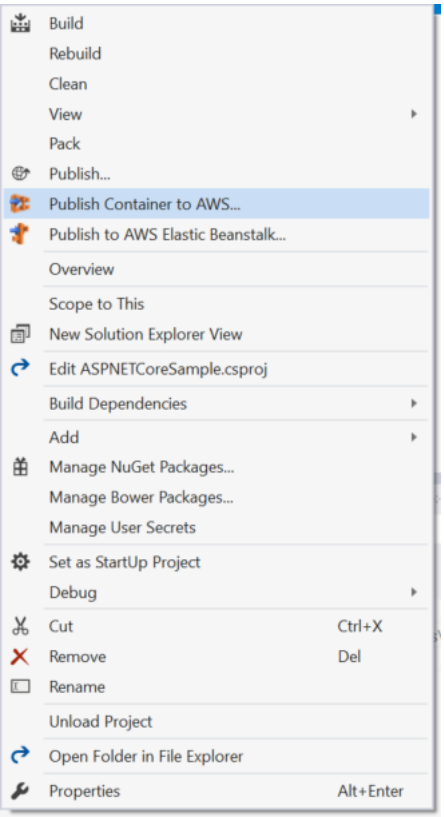
Sebelum Anda mempublikasikan kontainer Anda

Sebelum menggunakan Publish Container to AWS wizard untuk menyebarkan aplikasi ASP.NET Core 2.0 Anda:

- Tentukan AWS kredensil Anda dan dapatkan penyiapan dengan Amazon ECS.
- Instal Docker. Anda memiliki beberapa opsi instalasi yang berbeda termasuk Docker untuk Windows.
- Di Visual Studio, buat (atau buka) proyek untuk aplikasi kontainer ASP.NET Core 2.0 yang menargetkan Linux.

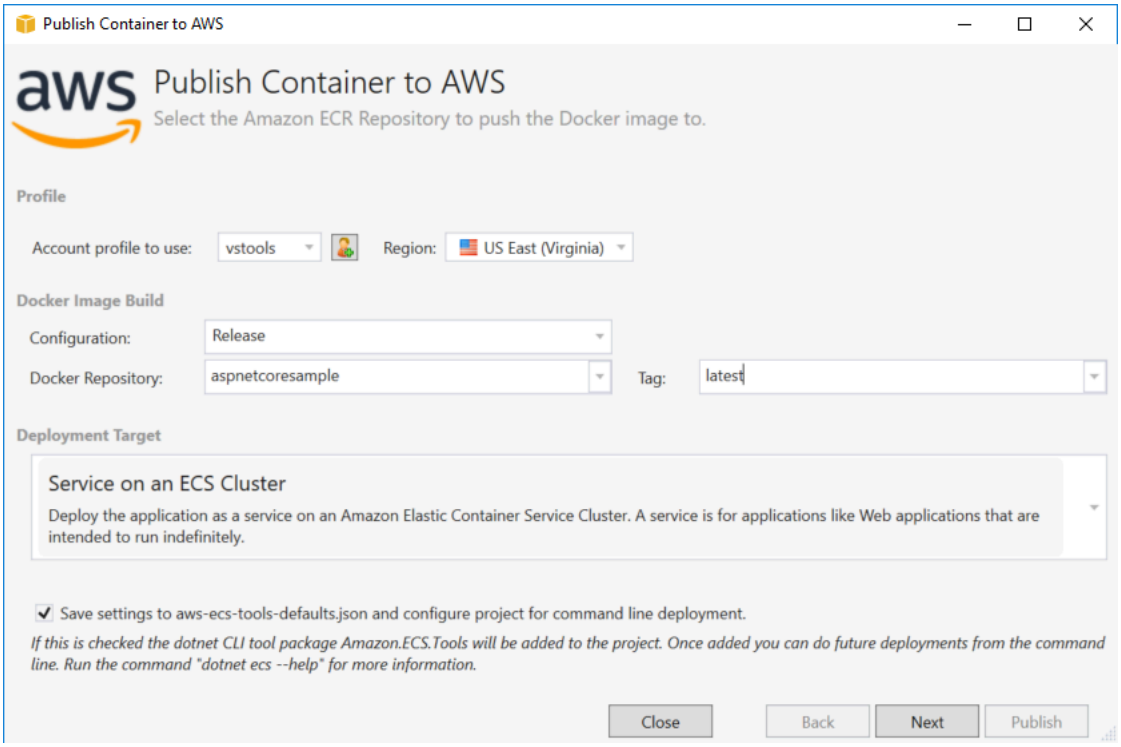
Mengakses Kontainer Publikasikan ke wizard AWS

Untuk menyebarkan aplikasi kontainer ASP.NET Core 2.0 yang menargetkan Linux, klik kanan proyek di Solution Explorer dan pilih Publish Container ke. AWS



Anda juga dapat memilih Publish Container ke AWS menu Visual Studio Build.

Publikasikan Kontainer ke AWS Wizard



Profil akun yang akan digunakan - Pilih profil akun yang akan digunakan.

Wilayah - Pilih wilayah penyebaran. Profil dan wilayah digunakan untuk mengatur sumber daya lingkungan penyebaran Anda dan untuk memilih registri Docker default.

Konfigurasi - Pilih konfigurasi build image Docker.

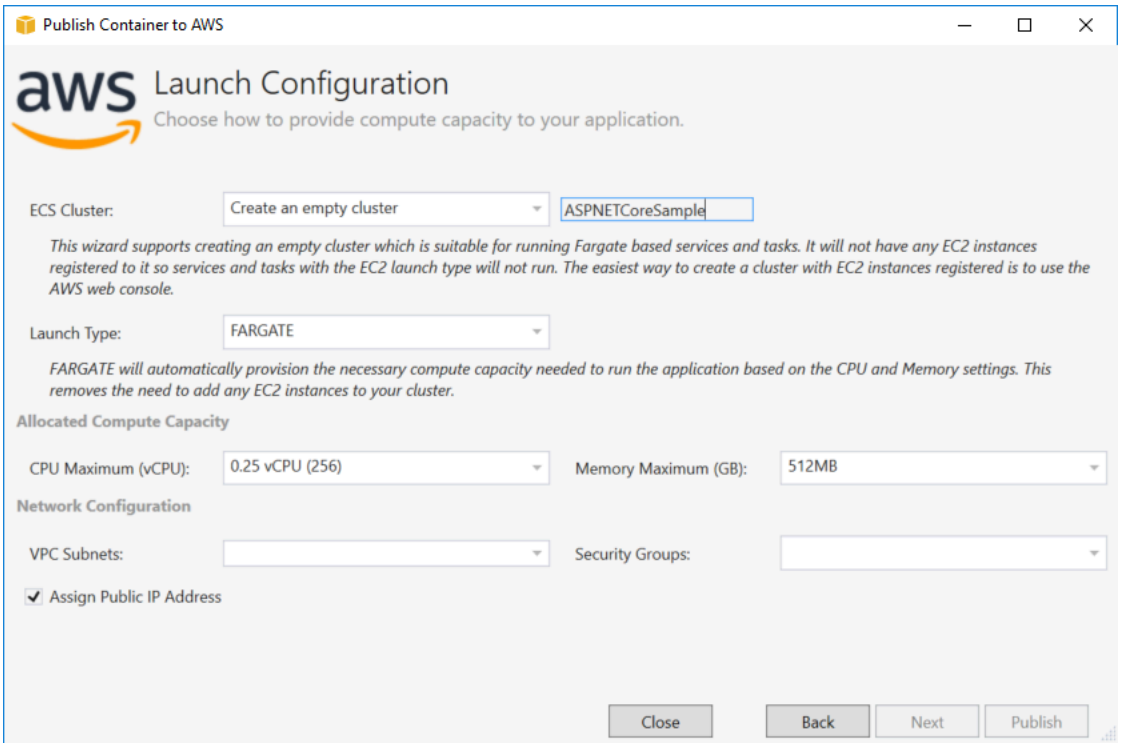
Docker Repository - Pilih repositori Docker yang ada atau ketik nama repositori baru dan itu akan dibuat. Ini adalah repositori tempat wadah build didorong.

Tag - Pilih tag yang ada atau ketik nama tag baru. Tag dapat melacak detail penting seperti versi, opsi, atau elemen konfigurasi unik lainnya dari wadah Docker.

Target Deployment - Pilih Layanan pada Cluster ECS. Gunakan opsi penyebaran ini ketika aplikasi Anda dimaksudkan untuk berjalan lama (seperti aplikasi web ASP.NET).

Simpan pengaturan ke **`aws-docker-tools-defaults.json`** dan konfigurasikan proyek untuk penyebaran baris perintah - Periksa opsi ini jika Anda ingin fleksibilitas penerapan dari baris perintah. Gunakan `dotnet ecs deploy` dari direktori proyek Anda untuk menyebarkan dan `dotnet ecs publish` wadah.

Luncurkan halaman Konfigurasi



ECS Cluster - Pilih cluster yang akan menjalankan image Docker Anda. Jika Anda memilih untuk membuat cluster kosong, berikan nama untuk cluster baru Anda.

Jenis Peluncuran - Pilih FARGATE.

CPU Maksimum (vCPU) - Pilih jumlah maksimum kapasitas komputasi yang diperlukan untuk aplikasi Anda. Untuk melihat rentang nilai CPU dan Memori yang diizinkan, lihat ukuran tugas.

Memory Maximum (GB) - Pilih jumlah maksimum memori yang tersedia untuk aplikasi Anda.

Subnet VPC - Pilih satu atau lebih subnet di bawah satu VPC. Jika Anda memilih lebih dari satu subnet, tugas Anda akan didistribusikan di seluruh subnet. Hal ini dapat meningkatkan ketersediaan. Untuk informasi selengkapnya, lihat VPC default dan subnet default.

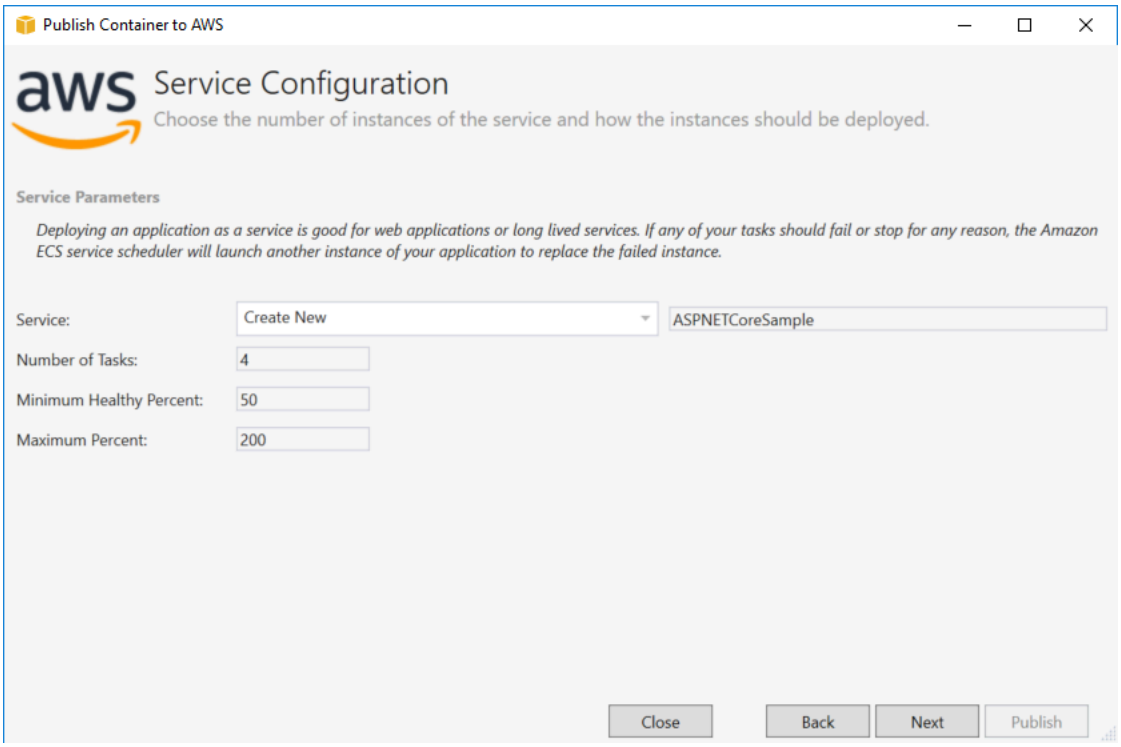
Grup Keamanan - Pilih grup keamanan.

Grup keamanan bertindak sebagai firewall untuk instans Amazon EC2 terkait, mengendalikan lalu lintas masuk dan keluar pada tingkat instans.

Grup keamanan default dikonfigurasi untuk memungkinkan lalu lintas masuk dari instans yang ditetapkan ke grup keamanan yang sama dan semua lalu lintas keluar IPv4 . Anda perlu keluar diizinkan sehingga layanan dapat mencapai repositori kontainer.

Tetapkan Alamat IP Publik - Periksa ini untuk membuat tugas Anda dapat diakses dari internet.

Halaman Konfigurasi Layanan



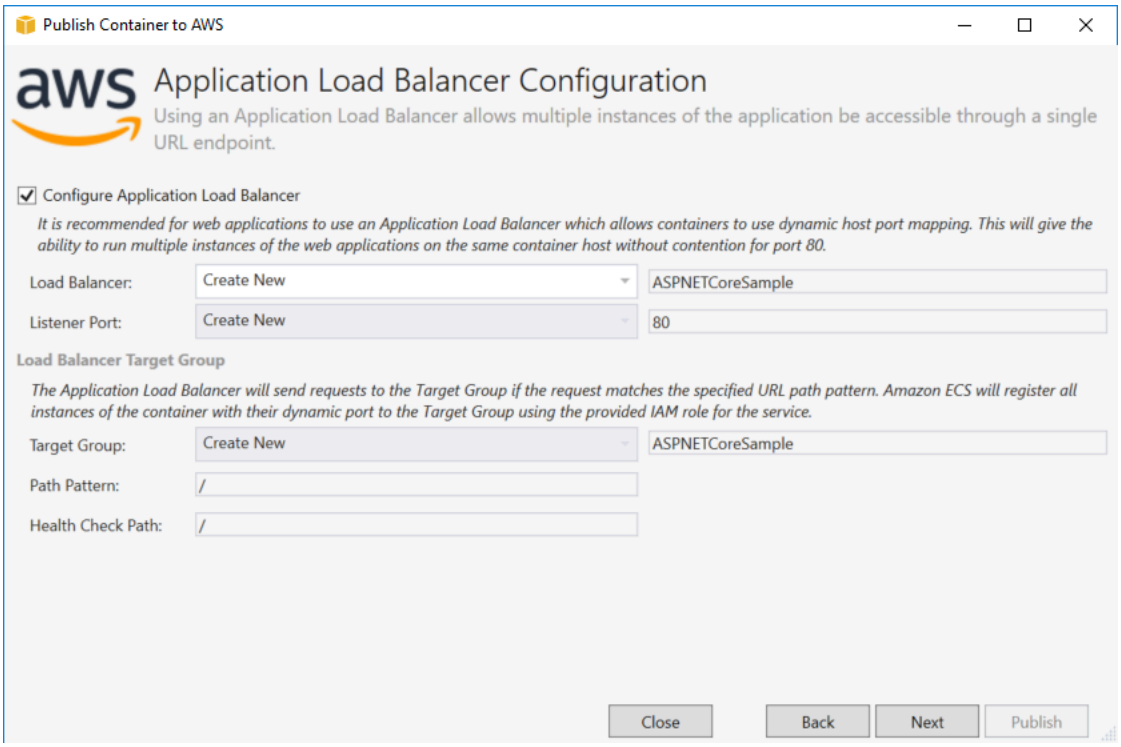
Layanan - Pilih salah satu layanan di drop-down untuk menyebarkan kontainer Anda ke layanan yang ada. Atau pilih Buat Baru untuk membuat layanan baru. Nama layanan harus unik dalam klaster, tetapi Anda dapat memiliki layanan bernama serupa di beberapa cluster dalam suatu wilayah atau di beberapa wilayah.

Jumlah Tugas - Jumlah tugas yang akan diterapkan dan terus berjalan di klaster Anda. Setiap tugas adalah salah satu contoh dari wadah Anda.

Persen Sehat Minimum - Persentase tugas yang harus tetap dalam `RUNNING` status selama penerapan dibulatkan ke bilangan bulat terdekat.

Persentase Maksimum - Persentase tugas yang diizinkan dalam `PENDING` status `RUNNING` atau selama penerapan dibulatkan ke bilangan bulat terdekat.

Halaman Application Load Balancer



Konfigurasikan Application Load Balancer - Periksa untuk mengonfigurasi penyeimbang beban aplikasi.

Load Balancer - Pilih penyeimbang beban yang ada atau pilih Buat Baru dan ketik nama untuk penyeimbang beban baru.

Port Listener - Pilih port listener yang ada atau pilih Buat Baru dan ketik nomor port. Default, port`80`, sesuai untuk sebagian besar aplikasi web.

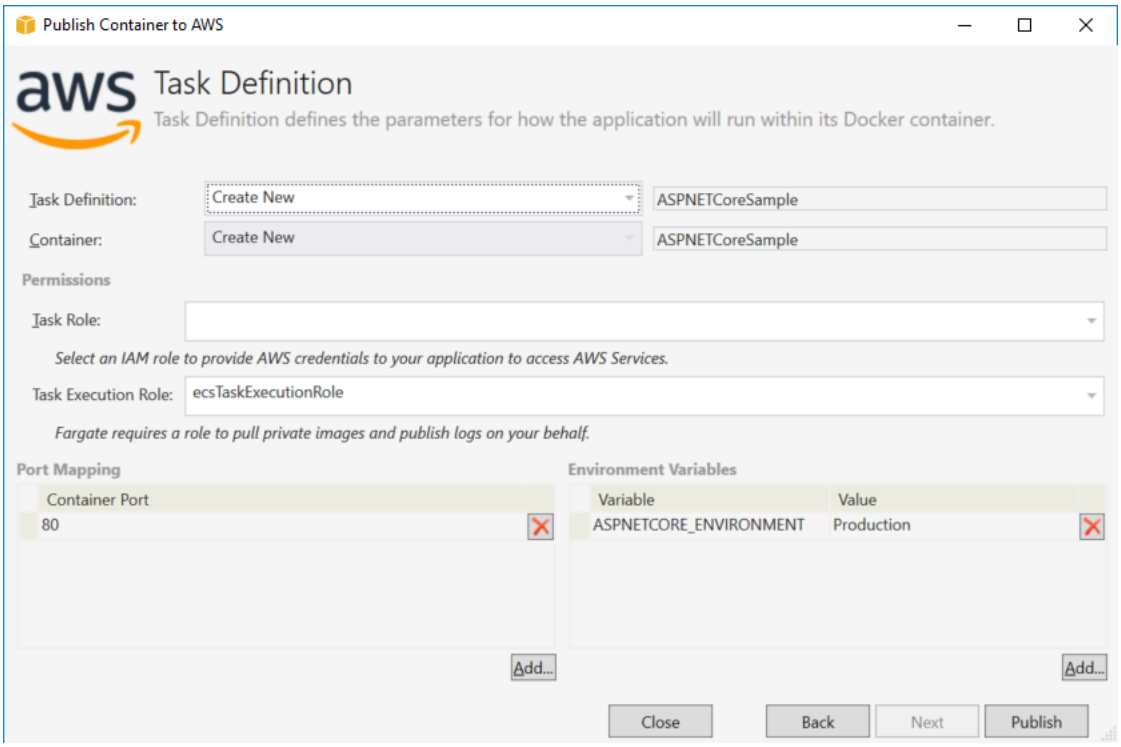
Grup Target - Pilih grup target Amazon ECS akan mendaftarkan tugas ke layanan.

Pola Jalur - Penyeimbang beban akan menggunakan routing berbasis jalur. Terima default `/` atau berikan pola yang berbeda. Pola jalur peka huruf besar/kecil, panjangnya bisa mencapai 128 karakter, dan berisi serangkaian karakter tertentu.

Health Check Path - Jalur ping yang merupakan tujuan pada target pemeriksaan kesehatan. Secara default, itu adalah `/`. Masukkan jalur yang berbeda jika diperlukan. Jika jalur yang Anda masukkan tidak valid, pemeriksaan kesehatan akan gagal dan akan dianggap tidak sehat.

Jika Anda menerapkan beberapa layanan, dan setiap layanan akan diterapkan ke jalur atau lokasi yang berbeda, Anda akan memerlukan jalur pemeriksaan khusus.

Halaman Definisi Tugas



Definisi Tugas - Pilih definisi tugas yang ada atau pilih Buat Baru dan ketik nama definisi tugas baru.

Kontainer - Pilih wadah yang ada atau pilih Buat Baru dan ketik nama kontainer baru.

Peran Tugas - Pilih peran IAM yang memiliki kredensil yang dibutuhkan aplikasi Anda untuk mengakses Layanan. AWS Ini adalah bagaimana kredensil diteruskan ke aplikasi Anda. Lihat cara menentukan AWS kredensil keamanan untuk aplikasi Anda.

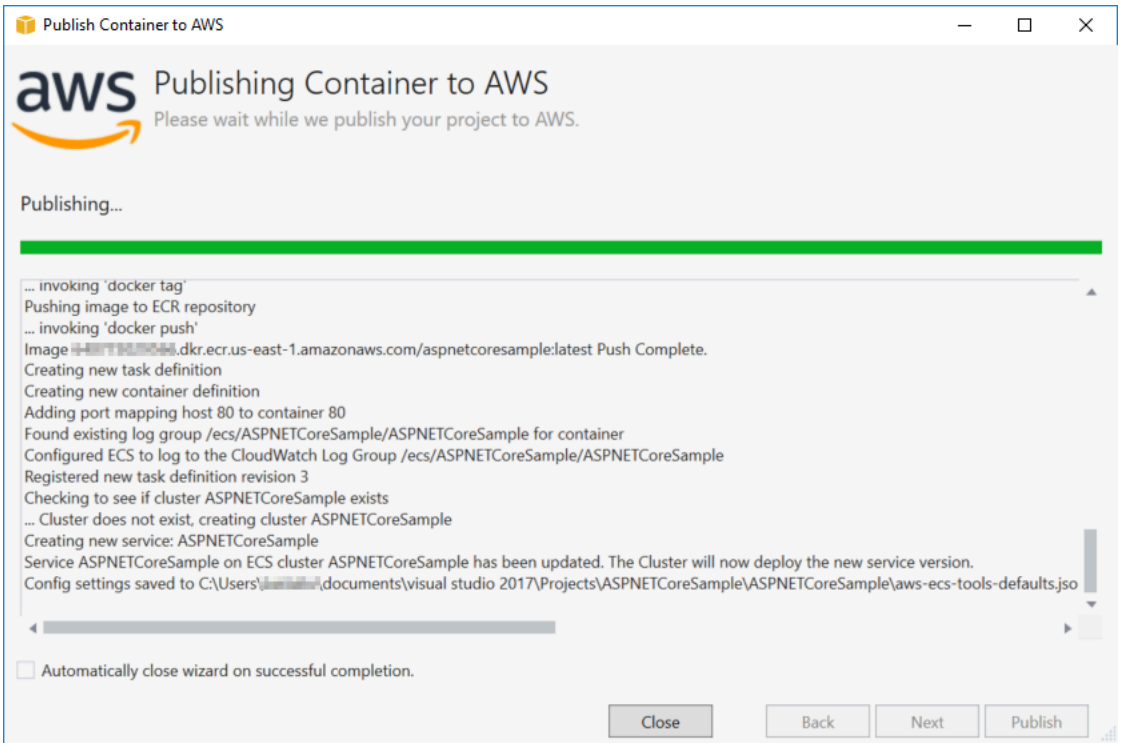
Peran Eksekusi Tugas - Pilih peran dengan izin untuk menarik gambar pribadi dan menerbitkan log. AWS Fargate akan menggunakannya atas nama Anda.

Pemetaan Port - Pilih nomor port pada wadah yang terikat ke port host yang ditetapkan secara otomatis.

Variabel Lingkungan - Menambahkan, memodifikasi, atau menghapus variabel lingkungan untuk wadah. Anda dapat memodifikasinya agar sesuai dengan penerapan Anda.

Ketika Anda puas dengan konfigurasi, klik Publikasikan untuk memulai proses penerapan.

Wadah Penerbitan ke AWS



Acara ditampilkan selama penyebaran. Wizard secara otomatis ditutup pada penyelesaian yang berhasil. Anda dapat mengganti ini dengan menghapus centang pada kotak di bagian bawah halaman.

Anda dapat menemukan URL instance baru Anda di AWS Explorer. Perluas Amazon ECS dan Cluster, lalu klik klaster Anda.

Menyebarkan Aplikasi ASP.NET Core 2.0 ke Amazon ECS (EC2)

Bagian ini menjelaskan cara menggunakan AWS wizard Publish Container to, yang disediakan sebagai bagian dari Toolkit for Visual Studio, untuk menyebarkan aplikasi ASP.NET Core 2.0 dalam kontainer yang menargetkan Linux melalui Amazon ECS menggunakan tipe peluncuran EC2. Karena aplikasi web dimaksudkan berjalan terus menerus, itu akan digunakan sebagai layanan.

Sebelum Anda mempublikasikan wadah Anda

Sebelum menggunakan Publish Container AWS untuk menyebarkan aplikasi ASP.NET Core 2.0 Anda:

- Tentukan AWS kredensil Anda dan dapatkan penyiapan dengan Amazon ECS.

- Instal Docker. Anda memiliki beberapa opsi instalasi yang berbeda termasuk Docker untuk Windows.
- Buat cluster Amazon ECS berdasarkan kebutuhan aplikasi web Anda. Hanya butuh beberapa langkah.
- Di Visual Studio, buat (atau buka) proyek untuk aplikasi kontainer ASP.NET Core 2.0 yang menargetkan Linux.

Mengakses Kontainer Publikasikan ke wizard AWS

Untuk menyebarkan aplikasi kontainer ASP.NET Core 2.0 yang menargetkan Linux, klik kanan proyek di Solution Explorer dan pilih Publish Container ke. AWS

Anda juga dapat memilih Publish Container ke AWS menu Visual Studio Build.

Publikasikan Kontainer ke AWS Wizard

Profil akun yang akan digunakan - Pilih profil akun yang akan digunakan.

Wilayah - Pilih wilayah penyebaran. Profil dan wilayah digunakan untuk mengatur sumber daya lingkungan penyebaran Anda dan memilih registri Docker default.

Konfigurasi - Pilih konfigurasi build image Docker.

Docker Repository - Pilih repositori Docker yang ada atau ketik nama repositori baru dan itu akan dibuat. Ini adalah repositori yang didorong oleh gambar kontainer yang dibangun.

Tag - Pilih tag yang ada atau ketik nama tag baru. Tag dapat melacak detail penting seperti versi, opsi, atau elemen konfigurasi unik lainnya dari wadah Docker.

Deployment - Pilih Layanan pada Cluster ECS. Gunakan opsi penyebaran ini ketika aplikasi Anda dimaksudkan untuk berjalan lama (seperti aplikasi web ASP.NET Core 2.0).

Simpan pengaturan ke **`aws-docker-tools-defaults.json`** dan konfigurasikan proyek untuk penyebaran baris perintah - Periksa opsi ini jika Anda ingin fleksibilitas penerapan dari baris perintah. Gunakan `dotnet ecs deploy` dari direktori proyek Anda untuk menyebarkan dan `dotnet ecs publish` wadah.

Luncurkan halaman Konfigurasi

ECS Cluster - Pilih cluster yang akan menjalankan image Docker Anda. Anda dapat membuat cluster ECS menggunakan AWS Management Console.

Jenis Peluncuran - Pilih EC2. Untuk menggunakan tipe peluncuran Fargate, lihat Menerapkan Aplikasi ASP.NET Core 2.0 ke Amazon ECS (Fargate).

Halaman Konfigurasi Layanan

Layanan - Pilih salah satu layanan di drop-down untuk menyebarkan kontainer Anda ke layanan yang ada. Atau pilih Buat Baru untuk membuat layanan baru. Nama layanan harus unik dalam klaster, tetapi Anda dapat memiliki layanan bernama serupa di beberapa cluster dalam suatu wilayah atau di beberapa wilayah.

Jumlah Tugas - Jumlah tugas yang akan diterapkan dan terus berjalan di cluster Anda. Setiap tugas adalah salah satu contoh dari wadah Anda.

Persen Sehat Minimum - Persentase tugas yang harus tetap dalam `RUNNING` status selama penerapan dibulatkan ke bilangan bulat terdekat.

Persen Maksimum - Persentase tugas yang diizinkan dalam `PENDING` status `RUNNING` atau selama penerapan dibulatkan ke bilangan bulat terdekat.

Template Penempatan - Pilih templat penempatan tugas.

Saat Anda meluncurkan tugas ke dalam klaster, Amazon ECS harus menentukan tempat menempatkan tugas berdasarkan persyaratan yang ditentukan dalam definisi tugas, seperti CPU dan memori. Demikian pula, saat Anda menurunkan jumlah tugas, Amazon ECS harus menentukan tugas mana yang akan dihentikan.

Template penempatan mengontrol cara tugas diluncurkan ke dalam klaster:

- AZ Balanced Spread - mendistribusikan tugas di seluruh Availability Zone dan di seluruh instance container di Availability Zone.
- AZ Balanced BinPack - mendistribusikan tugas di seluruh Availability Zone dan di seluruh instans kontainer dengan memori yang paling sedikit tersedia.
- BinPack - mendistribusikan tugas berdasarkan jumlah CPU atau memori yang paling sedikit tersedia.
- Satu Tugas Per Host - tempatkan, paling banyak, satu tugas dari layanan pada setiap instance kontainer.

Untuk informasi selengkapnya, lihat Penempatan Tugas Amazon ECS.

Halaman Application Load Balancer

Konfigurasikan Application Load Balancer - Periksa untuk mengonfigurasi penyeimbang beban aplikasi.

Pilih peran IAM untuk layanan - Pilih peran yang ada atau pilih Buat Baru dan peran baru akan dibuat.

Load Balancer - Pilih penyeimbang beban yang ada atau pilih Buat Baru dan ketik nama untuk penyeimbang beban baru.

Port Listener - Pilih port listener yang ada atau pilih Buat Baru dan ketik nomor port. Default, port`80`, sesuai untuk sebagian besar aplikasi web.

Grup Target - Secara default, penyeimbang beban mengirimkan permintaan ke target terdaftar menggunakan port dan protokol yang Anda tentukan untuk grup target. Anda dapat mengganti port ini ketika Anda mendaftar setiap target dengan kelompok target.

Pola Jalur - Penyeimbang beban akan menggunakan routing berbasis jalur. Terima default `/` atau berikan pola yang berbeda. Pola jalur peka huruf besar/kecil, panjangnya bisa mencapai 128 karakter, dan berisi serangkaian karakter tertentu.

Health Check Path - Jalur ping yang merupakan tujuan pada target pemeriksaan kesehatan. Secara default, ini `/` dan sesuai untuk aplikasi web. Masukkan jalur yang berbeda jika diperlukan. Jika jalur yang Anda masukkan tidak valid, pemeriksaan kesehatan akan gagal dan akan dianggap tidak sehat.

Jika Anda menerapkan beberapa layanan, dan setiap layanan akan disebarkan ke jalur atau lokasi yang berbeda, Anda mungkin memerlukan jalur pemeriksaan khusus.

Halaman Definisi Tugas ECS

Definisi Tugas - Pilih definisi tugas yang ada atau pilih Buat Baru dan ketik nama definisi tugas baru.

Kontainer - Pilih wadah yang ada atau pilih Buat Baru dan ketik nama wadah baru.

Memory (MiB) - Berikan nilai untuk Soft Limit atau Hard Limit atau keduanya.

Batas lunak (dalam MiB) memori untuk cadangan untuk wadah. Docker mencoba menyimpan memori kontainer di bawah batas lunak. Wadah dapat mengkonsumsi lebih banyak memori, hingga batas keras yang ditentukan dengan parameter memori (jika ada), atau semua memori yang tersedia pada instance kontainer, mana yang lebih dulu.

Batas keras (dalam MiB) memori untuk disajikan ke wadah. Jika kontainer Anda mencoba untuk melebihi memori yang ditentukan di sini, kontainer akan dimatikan.

Peran Tugas - Pilih peran tugas untuk peran IAM yang memungkinkan izin kontainer memanggil AWS APIs yang ditentukan dalam kebijakan terkait atas nama Anda. Ini adalah bagaimana kredensil diteruskan ke aplikasi Anda. Lihat cara menentukan AWS kredensil keamanan untuk aplikasi Anda.

Pemetaan Port - Menambahkan, memodifikasi atau menghapus pemetaan port untuk wadah. Jika load balancer aktif, port host akan default ke 0 dan penetapan port akan dinamis.

Variabel Lingkungan - Menambahkan, memodifikasi, atau menghapus variabel lingkungan untuk wadah.

Ketika Anda puas dengan konfigurasi, klik Publikasikan untuk memulai proses penerapan.

Wadah Penerbitan ke AWS

Acara ditampilkan selama penyebaran. Wizard secara otomatis ditutup pada penyelesaian yang berhasil. Anda dapat mengganti ini dengan menghapus centang pada kotak di bagian bawah halaman.

Anda dapat menemukan URL instance baru Anda di AWS Explorer. Perluas Amazon ECS dan Cluster, lalu klik klaster Anda.

Memecahkan masalah AWS Toolkit for Visual Studio

Bagian berikut berisi informasi pemecahan masalah umum tentang AWS Toolkit for Visual Studio dan bekerja dengan AWS layanan dari toolkit.

> **Note**
>
> Informasi set-up-specific penginstalan dan pemecahan masalah tersedia di topik Masalah penginstalan Pemecahan Masalah, yang terdapat di Panduan Pengguna ini.

Topik

Memecahkan masalah praktik terbaik

Berikut ini adalah praktik terbaik yang disarankan saat memecahkan masalah AWS Toolkit for Visual Studio .

- Perbaiki Visual Studio dan restart sistem Anda
- Cobalah untuk membuat ulang masalah atau kesalahan Anda sebelum mengirim laporan.
- Buat catatan rinci dari setiap langkah, pengaturan, dan pesan kesalahan selama proses rekreasi.
- Kumpulkan AWS Log Toolkit. Untuk penjelasan rinci tentang cara menemukan log AWS Toolkit Anda, lihat prosedur Cara menemukan AWS log Anda, yang terletak di topik panduan ini.
- Periksa permintaan terbuka, solusi yang diketahui, atau laporkan masalah Anda yang belum terselesaikan di bagian AWS Toolkit for Visual Studio Masalah di AWS Toolkit for Visual Studio GitHub repositori.

Perbaiki Visual Studio dan Restart sistem Anda

1. Tutup semua instance Visual Studio yang sedang berjalan.

2. Dari menu mulai Windows, Luncurkan Visual Studio Installer.
3. Jalankan Perbaikan pada instalasi Visual Studio yang terpengaruh. Ini memungkinkan Visual Studio untuk membangun kembali indeks ekstensi yang diinstal.
4. Mulai ulang Windows sebelum meluncurkan kembali Visual Studio.

Cara menemukan log AWS Toolkit Anda

1. Dari menu utama Visual Studio, perluas Ekstensi.
2. Pilih AWS Toolkit untuk memperluas menu AWS Toolkit, lalu pilih View Toolkit Logs.
3. Saat folder log AWS Toolkit terbuka di Sistem Operasi Anda, urutkan file berdasarkan tanggal dan temukan file log apa pun yang berisi informasi yang relevan dengan masalah Anda saat ini.

Melihat dan memfilter pemindaian keamanan Amazon Q

Untuk melihat pemindaian keamanan Amazon Q Anda di Visual Studio, buka Daftar Kesalahan Visual Studio dengan memperluas judul Tampilan di menu utama Visual Studio dan memilih Daftar Kesalahan.

Secara default, Daftar Kesalahan Visual Studio menampilkan semua peringatan dan kesalahan untuk basis kode Anda. Untuk memfilter temuan pemindaian keamanan Amazon Q Anda dari Daftar Kesalahan Visual Studio, buat filter dengan menyelesaikan prosedur berikut.

> **Note**
>
> Temuan pemindaian keamanan Amazon Q hanya terlihat setelah pemindaian keamanan berjalan dan mendeteksi masalah.
> Temuan pemindaian keamanan Amazon Q muncul sebagai peringatan di Visual Studio. Untuk melihat temuan pemindaian keamanan Amazon Q dari Daftar Kesalahan Anda, opsi Peringatan di judul Daftar Kesalahan harus dipilih.

1. Dari menu utama Visual Studio, perluas judul Tampilan dan pilih Daftar Kesalahan untuk membuka panel Daftar Kesalahan.
2. Dari panel Daftar Kesalahan, klik kanan baris header untuk membuka menu konteks.
3. Dari menu konteks, perluas Tampilkan Kolom, lalu pilih Alat di menu yang diperluas.

4. Kolom Alat ditambahkan ke Daftar Kesalahan Anda.
5. Dari header kolom Alat, pilih ikon Filter dan pilih Amazon Q untuk memfilter temuan pemindaian keamanan Amazon Q.

AWS Toolkit tidak diinstal dengan benar

Masalah:

Dalam satu menit setelah memulai Visual Studio, pesan berikut muncul di panel output dan bilah info, masing-masing: AWS Toolkit for Visual Studio

```
Some Toolkit components could not be initialized. Some functionality may
not work during this IDE session.

The AWS Toolkit is not properly installed.
```

Solusi:

Ada kemungkinan bahwa memperbarui atau menginstal ekstensi menyebabkan beberapa file cache internal Visual Studio pergi out-of-sync. Prosedur berikut menjelaskan cara membuat file-file ini dibangun kembali saat Anda meluncurkan Visual Studio berikutnya.

> **Note**
>
> Ada kemungkinan bahwa solusi ini dapat mempengaruhi kustomisasi Visual Studio Anda. Setelah menyelesaikan prosedur ini, ekstensi AWS Toolkit harus terdaftar sebagai diinstal dan tidak lagi melaporkan pesan kesalahan. Jika Anda terus mengalami masalah ini setelah menyelesaikan langkah-langkah berikut, silakan lihat Masalah #452 di AWS Toolkit for Visual Studio GitHub repositori untuk informasi tambahan.

1. Instal versi terbaru Visual Studio 2022.

   > **Note**
   >
   > Versi minimum yang diperlukan adalah 17.11.5.

2. Tutup semua instance Visual Studio yang sedang berjalan.

3. Dari Windows, buka Prompt Perintah Pengembang sebagai Administrator.
4. Dari Developer Command Prompt, jalankan perintah berikut:`devenv /updateconfiguration /resetExtensions`, lalu tunggu sampai perintah selesai.
5. Setelah perintah selesai, restart Visual Studio.
6. Di Visual Studio AWS ekstensi sekarang terdaftar sebagai diinstal dan tidak lagi melaporkan pesan kesalahan yang tercantum di bagian atas masalah ini.

Pengaturan firewall dan proxy

Memecahkan masalah firewall dan pengaturan proxy

Perangkat lunak pemindaian keamanan dapat mengganggu kemampuan Anda untuk mengunduh file dari server bahasa AWS Toolkit dengan menghapus file dari unduhan atau mencegah unduhan sama sekali.

Untuk memeriksa pengaturan firewall dan proxy Anda, navigasikan ke https://aws-toolkit-language-servers.amazonaws.com/codewhisperer/0/manifest.json dari browser internet yang diinstal pada sistem yang sama dengan instance Visual Studio Anda. Jika Anda menemukan kesalahan atau halaman tidak dapat memuat, maka mungkin ada firewall atau filter proxy yang mencegah Anda menjangkau`aws-toolkit-language-servers.amazonaws.com`.

Sertifikat kustom

AWS Toolkit for Visual Studio Menggunakan server bahasa yang berjalan pada runtime Node.js. Untuk informasi terperinci tentang cara memeriksa apakah jaringan Anda menggunakan sertifikat kustom, lihat pengaturan Konfigurasi dan file kredensi dalam AWS CLI topik di Panduan AWS Command Line InterfacePengguna untuk Versi 1.

Untuk mengonfigurasi pengaturan proxy Anda dan menentukan sertifikat, Anda harus mengonfigurasi variabel `HTTPS_PROXY` env Anda dan membuat Variabel Lingkungan Windows untuk `NODE_EXTRA_CA_CERTS` kunci `NODE_OPTIONS` dan.

Untuk mengonfigurasi variabel `HTTPS_PROXY` env Anda, selesaikan langkah-langkah berikut.

1. Dari menu utama Visual Studio, pilih Tools, lalu pilih Options.
2. Dari menu Opsi, perluas AWS Toolkit, lalu pilih Proxy.
3. Dari menu Proxy, tentukan Host dan Port Anda.

> **Note**
>
> Untuk informasi tentang mengonfigurasi HTTPS_PROXY dari AWS CLI, lihat Menggunakan proxy HTTP untuk AWS CLI topik di Panduan AWS Command Line InterfacePengguna.

Buat Variabel Lingkungan Windows untuk kunci berikut.

- `NODE_OPTIONS = --use-openssl-ca`
- `NODE_EXTRA_CA_CERTS = Path/To/Corporate/Certs`

> **Note**
>
> Untuk informasi selengkapnya tentang mengekstrak sertifikat akar perusahaan, lihat artikel Ekspor sertifikat dengan kunci privatnya di learn.microsoft.com. Untuk informasi rinci tentang kunci Variabel Lingkungan Windows, lihat dokumentasi Node.js v23.3.0 di nodejs.org.

Izinkan daftar dan langkah-langkah tambahan

Selain mengganggu server bahasa AWS Toolkit, pengaturan firewall dapat mencegah Amazon Q mengunggah ke Amazon S3 dan memanggil API layanan. Untuk meminimalkan potensi kesalahan ini, sebaiknya izinkan akses internet keluar pada port 443 (HTTPS) untuk titik akhir berikut:

- `https://codewhisperer.us-east-1.amazonaws.com/`
- `https://amazonq-code-transformation-us-east-1-c6160f047e0.s3.amazonaws.com/`
- `https://aws-toolkit-language-servers.amazonaws.com/`
- `https://q.us-east-1.amazonaws.com`
- `https://client-telemetry.us-east-1.amazonaws.com`
- `https://cognito-identity.us-east-1.amazonaws.com`
- `https://oidc.us-east-1.amazonaws.com`

Untuk daftar detail titik akhir, lihat topik Memperbarui firewall dan gateway untuk mengizinkan akses dalam Panduan Pengguna ini. Untuk informasi terperinci tentang mengonfigurasi proxy perusahaan

untuk Amazon Q, lihat topik Mengonfigurasi proxy perusahaan di Amazon Q di Panduan Pengguna Pengembang Amazon Q. Jika Anda terus mengalami masalah firewall dan proxy, kumpulkan Log AWS Toolkit Anda dan hubungi AWS Toolkit for Visual Studio tim melalui bagian AWS Toolkit for Visual Studio masalah di AWS Toolkit for Visual Studio GitHub repositori. Untuk detail tentang mengumpulkan Log AWS Toolkit Anda, tinjau informasi di bagian Pemecahan Masalah praktik terbaik dari topik Panduan Pengguna ini.

Keamanan untuk AWS Toolkit for Visual Studio

Keamanan cloud di Amazon Web Services (AWS) merupakan prioritas tertinggi. Sebagai seorang pelanggan AWS , Anda mendapatkan manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan dari organisasi yang paling sensitif terhadap keamanan. Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. Model Tanggung Jawab Bersama menggambarkan ini sebagai Keamanan dari Cloud dan Keamanan dalam Cloud.

Security of the Cloud - AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan semua layanan yang ditawarkan di AWS Cloud dan memberi Anda layanan yang dapat Anda gunakan dengan aman. Tanggung jawab keamanan kami adalah prioritas tertinggi di AWS, dan efektivitas keamanan kami secara teratur diuji dan diverifikasi oleh auditor pihak ketiga sebagai bagian dari Program AWS Kepatuhan.

Keamanan di Cloud — Tanggung jawab Anda ditentukan oleh AWS layanan yang Anda gunakan, dan faktor-faktor lain termasuk sensitivitas data Anda, persyaratan organisasi Anda, serta undang-undang dan peraturan yang berlaku.

AWS Produk atau layanan ini mengikuti model tanggung jawab bersama melalui layanan Amazon Web Services (AWS) tertentu yang didukungnya. Untuk informasi keamanan AWS layanan, lihat halaman dokumentasi keamanan AWS layanan dan AWS layanan yang berada dalam lingkup upaya AWS kepatuhan oleh program kepatuhan.

Topik

Perlindungan Data di AWS Toolkit for Visual Studio

Model tanggung jawab AWS bersama model berlaku untuk perlindungan data di AWS Toolkit for Visual Studio dengan Amazon Q. Seperti yang dijelaskan dalam model ini AWS , bertanggung jawab

untuk melindungi infrastruktur global yang menjalankan semua. AWS Cloud Anda bertanggung jawab untuk mempertahankan kendali atas konten yang di-host pada infrastruktur ini. Anda juga bertanggung jawab atas tugas-tugas konfigurasi dan manajemen keamanan untuk Layanan AWS yang Anda gunakan. Lihat informasi yang lebih lengkap tentang privasi data dalam Pertanyaan Umum Privasi Data. Lihat informasi tentang perlindungan data di Eropa di pos blog Model Tanggung Jawab Bersama dan GDPR AWS di Blog Keamanan AWS .

Untuk tujuan perlindungan data, kami menyarankan Anda melindungi Akun AWS kredensil dan mengatur pengguna individu dengan AWS IAM Identity Center atau AWS Identity and Access Management (IAM). Dengan cara itu, setiap pengguna hanya diberi izin yang diperlukan untuk memenuhi tanggung jawab tugasnya. Kami juga menyarankan supaya Anda mengamankan data dengan cara-cara berikut:

- Gunakan autentikasi multi-faktor (MFA) pada setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan AWS sumber daya. Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail. Untuk informasi tentang penggunaan CloudTrail jejak untuk menangkap AWS aktivitas, lihat Bekerja dengan CloudTrail jejak di AWS CloudTrail Panduan Pengguna.
- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola tingkat lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-3 saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Lihat informasi selengkapnya tentang titik akhir FIPS yang tersedia di Standar Pemrosesan Informasi Federal (FIPS) 140-3.

Kami sangat merekomendasikan agar Anda tidak pernah memasukkan informasi identifikasi yang sensitif, seperti nomor rekening pelanggan Anda, ke dalam tanda atau bidang isian bebas seperti bidang Nama. Ini termasuk saat Anda bekerja dengan AWS Toolkit dengan Amazon Q atau lainnya Layanan AWS menggunakan konsol, API AWS CLI, atau AWS SDKs. Data apa pun yang Anda masukkan ke dalam tanda atau bidang isian bebas yang digunakan untuk nama dapat digunakan untuk log penagihan atau log diagnostik. Saat Anda memberikan URL ke server eksternal, kami sangat menganjurkan supaya Anda tidak menyertakan informasi kredensial di dalam URL untuk memvalidasi permintaan Anda ke server itu.

Identity and Access Management

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. Administrator IAM mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya. AWS IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Topik

Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda, tergantung pada pekerjaan yang Anda lakukan. AWS

Pengguna layanan — Jika Anda menggunakan Layanan AWS untuk melakukan pekerjaan Anda, maka administrator Anda memberi Anda kredensi dan izin yang Anda butuhkan. Saat Anda menggunakan lebih banyak AWS fitur untuk melakukan pekerjaan Anda, Anda mungkin memerlukan izin tambahan. Memahami cara akses dikelola dapat membantu Anda meminta izin yang tepat dari administrator Anda. Jika Anda tidak dapat mengakses fitur AWS, lihat Memecahkan masalah AWS identitas dan akses atau panduan pengguna yang Layanan AWS Anda gunakan.

Administrator layanan — Jika Anda bertanggung jawab atas AWS sumber daya di perusahaan Anda, Anda mungkin memiliki akses penuh ke AWS. Tugas Anda adalah menentukan AWS fitur dan sumber daya mana yang harus diakses pengguna layanan Anda. Kemudian, Anda harus mengirimkan permintaan kepada administrator IAM untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep dasar IAM. Untuk mempelajari lebih lanjut tentang bagaimana perusahaan Anda dapat menggunakan IAM AWS, lihat panduan pengguna yang Layanan AWS Anda gunakan.

Administrator IAM – Jika Anda adalah administrator IAM, Anda sebaiknya mempelajari detail tentang cara menulis kebijakan untuk mengelola akses ke AWS. Untuk melihat contoh kebijakan AWS

berbasis identitas yang dapat Anda gunakan di IAM, lihat panduan pengguna yang Anda gunakan. Layanan AWS

Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredensi identitas Anda. Anda harus diautentikasi sebagai Pengguna root akun AWS, pengguna IAM, atau dengan mengasumsikan peran IAM.

Anda dapat masuk sebagai identitas federasi menggunakan kredensional dari sumber identitas seperti AWS IAM Identity Center (Pusat Identitas IAM), autentikasi masuk tunggal, atau kredensional. Google/Facebook Untuk informasi selengkapnya tentang cara masuk, lihat Cara masuk ke Akun AWS Anda dalam Panduan Pengguna AWS Sign-In .

Untuk akses terprogram, AWS sediakan SDK dan CLI untuk menandatangani permintaan secara kriptografis. Untuk informasi selengkapnya, lihat AWS Signature Version 4 untuk permintaan API dalam Panduan Pengguna IAM.

Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang disebut pengguna Akun AWS root yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Untuk tugas yang memerlukan kredensial pengguna root, lihat Tugas yang memerlukan kredensial pengguna root dalam Panduan Pengguna IAM.

Identitas terfederasi

Sebagai praktik terbaik, mewajibkan pengguna manusia untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS menggunakan kredensi sementara.

Identitas federasi adalah pengguna dari direktori perusahaan Anda, penyedia identitas web, atau Directory Service yang mengakses Layanan AWS menggunakan kredensi dari sumber identitas. Identitas terfederasi mengambil peran yang memberikan kredensial sementara.

Untuk manajemen akses terpusat, kami menyarankan AWS IAM Identity Center. Untuk informasi selengkapnya, lihat Apa itu Pusat Identitas IAM? dalam Panduan Pengguna AWS IAM Identity Center .

Pengguna dan grup IAM

Pengguna IAM adalah identitas dengan izin khusus untuk satu orang atau aplikasi. Sebaiknya gunakan kredensial sementara alih-alih pengguna IAM dengan kredensial jangka panjang. Untuk informasi selengkapnya, lihat Mewajibkan pengguna manusia untuk menggunakan federasi dengan penyedia identitas untuk mengakses AWS menggunakan kredensi sementara di Panduan Pengguna IAM.

Grup IAM menentukan kumpulan pengguna IAM dan mempermudah pengelolaan izin untuk pengguna dalam jumlah besar. Untuk mempelajari selengkapnya, lihat Kasus penggunaan untuk pengguna IAM dalam Panduan Pengguna IAM.

Peran IAM

Peran IAM adalah identitas dengan izin khusus yang menyediakan kredensial sementara. Anda dapat mengambil peran dengan beralih dari pengguna ke peran IAM (konsol) atau dengan memanggil operasi AWS CLI atau AWS API. Untuk informasi selengkapnya, lihat Metode untuk mengambil peran dalam Panduan Pengguna IAM.

Peran IAM berguna untuk akses pengguna terfederasi, izin pengguna IAM sementara, akses lintas akun, akses lintas layanan, dan aplikasi yang berjalan di Amazon EC2. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan menentukan izin saat dikaitkan dengan identitas atau sumber daya. AWS mengevaluasi kebijakan ini ketika kepala sekolah membuat permintaan. Sebagian besar kebijakan disimpan AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang dokumen kebijakan JSON, lihat Gambaran umum kebijakan JSON dalam Panduan Pengguna IAM.

Menggunakan kebijakan, administrator menentukan siapa yang memiliki akses ke apa dengan mendefinisikan principal mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Administrator IAM membuat kebijakan IAM dan menambahkannya ke peran, yang kemudian dapat diambil oleh pengguna. Kebijakan IAM mendefinisikan izin terlepas dari metode yang Anda gunakan untuk melakukan operasinya.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang Anda lampirkan ke identitas (pengguna, grup, atau peran). Kebijakan ini mengontrol tindakan apa yang bisa dilakukan oleh identitas tersebut, terhadap sumber daya yang mana, dan dalam kondisi apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan yang dikelola pelanggan dalam Panduan Pengguna IAM.

Kebijakan berbasis identitas dapat berupa kebijakan inline (disematkan langsung ke dalam satu identitas) atau kebijakan terkelola (kebijakan mandiri yang dilampirkan pada banyak identitas). Untuk mempelajari cara memilih antara kebijakan terkelola dan kebijakan inline, lihat Pilih antara kebijakan terkelola dan kebijakan inline dalam Panduan Pengguna IAM.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contohnya termasuk kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Anda harus menentukan principal dalam kebijakan berbasis sumber daya.

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan AWS terkelola dari IAM dalam kebijakan berbasis sumber daya.

Daftar kontrol akses (ACLs)

Access control lists (ACLs) mengontrol prinsipal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACLs mirip dengan kebijakan berbasis sumber daya, meskipun mereka tidak menggunakan format dokumen kebijakan JSON.

Amazon S3, AWS WAF, dan Amazon VPC adalah contoh layanan yang mendukung. ACLs Untuk mempelajari selengkapnya ACLs, lihat Ringkasan daftar kontrol akses (ACL) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang dapat menetapkan izin maksimum yang diberikan oleh jenis kebijakan yang lebih umum:

- Batasan izin – Menetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas IAM. Untuk informasi selengkapnya, lihat Batasan izin untuk entitas IAM dalam Panduan Pengguna IAM.
- Kebijakan kontrol layanan (SCPs) — Tentukan izin maksimum untuk organisasi atau unit organisasi di AWS Organizations. Untuk informasi selengkapnya, lihat Kebijakan kontrol layanan dalam Panduan Pengguna AWS Organizations .
- Kebijakan kontrol sumber daya (RCPs) — Tetapkan izin maksimum yang tersedia untuk sumber daya di akun Anda. Untuk informasi selengkapnya, lihat Kebijakan kontrol sumber daya (RCPs) di Panduan AWS Organizations Pengguna.
- Kebijakan sesi – Kebijakan lanjutan yang diteruskan sebagai parameter saat membuat sesi sementara untuk peran atau pengguna terfederasi. Untuk informasi selengkapnya, lihat Kebijakan sesi dalam Panduan Pengguna IAM.

Berbagai jenis kebijakan

Ketika beberapa jenis kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan saat beberapa jenis kebijakan terlibat, lihat Logika evaluasi kebijakan di Panduan Pengguna IAM.

Bagaimana Layanan AWS bekerja dengan IAM

Untuk mendapatkan tampilan tingkat tinggi tentang cara Layanan AWS bekerja dengan sebagian besar fitur IAM, lihat AWS layanan yang bekerja dengan IAM di Panduan Pengguna IAM.

Untuk mempelajari cara menggunakan spesifik Layanan AWS dengan IAM, lihat bagian keamanan dari Panduan Pengguna layanan yang relevan.

Memecahkan masalah AWS identitas dan akses

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan AWS dan IAM.

Topik

- Saya tidak berwenang untuk melakukan tindakan di AWS
- Saya tidak berwenang untuk melakukan iam: PassRole
- Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses AWS sumber daya saya

Saya tidak berwenang untuk melakukan tindakan di AWS

Jika Anda menerima pesan kesalahan bahwa Anda tidak memiliki otorisasi untuk melakukan tindakan, kebijakan Anda harus diperbarui agar Anda dapat melakukan tindakan tersebut.

Contoh kesalahan berikut terjadi ketika pengguna IAM `mateojackson` mencoba menggunakan konsol untuk melihat detail tentang suatu sumber daya *`my-example-widget`* rekaan, tetapi tidak memiliki izin `awes:`*`GetWidget`* rekaan.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 awes:GetWidget on resource: my-example-widget
```

Dalam hal ini, kebijakan untuk pengguna `mateojackson` harus diperbarui untuk mengizinkan akses ke sumber daya *`my-example-widget`* dengan menggunakan tindakan `awes:`*`GetWidget`*.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya tidak berwenang untuk melakukan iam: PassRole

Jika Anda menerima kesalahan yang tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran AWS.

Beberapa Layanan AWS memungkinkan Anda untuk meneruskan peran yang ada ke layanan tersebut alih-alih membuat peran layanan baru atau peran terkait layanan. Untuk melakukannya, Anda harus memiliki izin untuk meneruskan peran ke layanan.

Contoh kesalahan berikut terjadi ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol tersebut untuk melakukan tindakan di AWS. Namun, tindakan tersebut memerlukan layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses AWS sumber daya saya

Anda dapat membuat peran yang dapat digunakan pengguna di akun lain atau orang-orang di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa saja yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACLs), Anda dapat menggunakan kebijakan tersebut untuk memberi orang akses ke sumber daya Anda.

Untuk mempelajari selengkapnya, periksa referensi berikut:

- Untuk mempelajari apakah AWS mendukung fitur ini, lihatBagaimana Layanan AWS bekerja dengan IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda di seluruh sumber daya Akun AWS yang Anda miliki, lihat Menyediakan akses ke pengguna IAM di pengguna lain Akun AWS yang Anda miliki di Panduan Pengguna IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda kepada pihak ketiga Akun AWS, lihat Menyediakan akses yang Akun AWS dimiliki oleh pihak ketiga dalam Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses melalui federasi identitas, lihat Menyediakan akses ke pengguna terautentikasi eksternal (federasi identitas) dalam Panduan Pengguna IAM.
- Untuk mempelajari perbedaan antara menggunakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Akses sumber daya lintas akun di IAM di Panduan Pengguna IAM.

Validasi Kepatuhan untuk AWS Produk atau Layanan ini

Untuk mempelajari apakah an Layanan AWS berada dalam lingkup program kepatuhan tertentu, lihat Layanan AWS di Lingkup oleh Program Kepatuhan Layanan AWS dan pilih program kepatuhan yang Anda minati. Untuk informasi umum, lihat Program AWS Kepatuhan Program AWS .

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat Mengunduh Laporan di AWS Artifact .

Tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, dan hukum dan peraturan yang berlaku. Untuk informasi selengkapnya tentang tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS, lihat Dokumentasi AWS Keamanan.

AWS Produk atau layanan ini mengikuti model tanggung jawab bersama melalui layanan Amazon Web Services (AWS) tertentu yang didukungnya. Untuk informasi keamanan AWS layanan, lihat halaman dokumentasi keamanan AWS layanan dan AWS layanan yang berada dalam lingkup upaya AWS kepatuhan oleh program kepatuhan.

Ketahanan untuk AWS Produk atau Layanan ini

Infrastruktur AWS global dibangun di sekitar Region AWS dan Availability Zones.

Region AWS menyediakan beberapa Availability Zone yang terpisah secara fisik dan terisolasi, yang terhubung dengan latensi rendah, throughput tinggi, dan jaringan yang sangat redundan.

Dengan Zona Ketersediaan, Anda dapat merancang serta mengoperasikan aplikasi dan basis data yang secara otomatis melakukan fail over di antara zona tanpa gangguan. Zona Ketersediaan memiliki ketersediaan dan toleransi kesalahan yang lebih baik, dan dapat diskalakan dibandingkan infrastruktur pusat data tunggal atau multi tradisional.

Untuk informasi selengkapnya tentang AWS Wilayah dan Availability Zone, lihat Infrastruktur AWS Global.

AWS Produk atau layanan ini mengikuti model tanggung jawab bersama melalui layanan Amazon Web Services (AWS) tertentu yang didukungnya. Untuk informasi keamanan AWS layanan, lihat halaman dokumentasi keamanan AWS layanan dan AWS layanan yang berada dalam lingkup upaya AWS kepatuhan oleh program kepatuhan.

Keamanan Infrastruktur untuk AWS Produk atau Layanan ini

AWS Produk atau layanan ini menggunakan layanan terkelola, dan karenanya dilindungi oleh keamanan jaringan AWS global. Untuk informasi tentang layanan AWS keamanan dan cara AWS melindungi infrastruktur, lihat Keamanan AWS Cloud. Untuk mendesain AWS lingkungan Anda menggunakan praktik terbaik untuk keamanan infrastruktur, lihat Perlindungan Infrastruktur dalam Kerangka Kerja yang AWS Diarsiteksikan dengan Baik Pilar Keamanan.

Anda menggunakan panggilan API yang AWS dipublikasikan untuk mengakses AWS Produk atau Layanan ini melalui jaringan. Klien harus mendukung hal-hal berikut:

- Keamanan Lapisan Pengangkutan (TLS). Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.

- Sandi cocok dengan sistem kerahasiaan maju sempurna (perfect forward secrecy, PFS) seperti DHE (Ephemeral Diffie-Hellman) atau ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Sebagian besar sistem modern seperti Java 7 dan versi lebih baru mendukung mode-mode ini.

Selain itu, permintaan harus ditandatangani menggunakan ID kunci akses dan kunci akses rahasia yang terkait dengan principal IAM. Atau Anda dapat menggunakan AWS Security Token Service (AWS STS) untuk menghasilkan kredensial keamanan sementara untuk menandatangani permintaan.

AWS Produk atau layanan ini mengikuti model tanggung jawab bersama melalui layanan Amazon Web Services (AWS) tertentu yang didukungnya. Untuk informasi keamanan AWS layanan, lihat halaman dokumentasi keamanan AWS layanan dan AWS layanan yang berada dalam lingkup upaya AWS kepatuhan oleh program kepatuhan.

Analisis Konfigurasi dan Kerentanan di AWS Toolkit for Visual Studio

Toolkit for Visual Studio dirilis ke Marketplace Visual Studio saat fitur atau perbaikan baru dikembangkan. Pembaruan ini terkadang menyertakan pembaruan keamanan, jadi penting untuk selalu memperbarui AWS Toolkit dengan Amazon Q.

Untuk memverifikasi bahwa pembaruan otomatis untuk ekstensi diaktifkan

1. Buka pengelola ekstensi dengan memilih Alat, Ekstensi, dan Pembaruan (Visual Studio 2017), atau Ekstensi, Kelola Ekstensi (Visual Studio 2019).
2. Pilih Ubah pengaturan Ekstensi dan Pembaruan Anda (Visual Studio 2017), atau Ubah pengaturan Anda untuk Ekstensi (Visual Studio 2019).
3. Sesuaikan pengaturan untuk lingkungan Anda.

Jika Anda memilih untuk menonaktifkan pembaruan otomatis untuk ekstensi, pastikan untuk memeriksa pembaruan AWS Toolkit dengan Amazon Q pada interval yang sesuai untuk lingkungan Anda.

Riwayat dokumen Panduan AWS Toolkit for Visual Studio Pengguna

Riwayat dokumen

Tabel berikut menjelaskan perubahan penting terbaru dari Panduan AWS Toolkit for Visual Studio Pengguna. Untuk notifikasi tentang pembaruan dokumentasi ini, Anda dapat berlangganan ke umpan RSS.

Perubahan	Deskripsi	Tanggal
Pembaruan untuk Memulai Konten	Pembaruan yang dilakukan untuk Memulai dan Menghubungkan ke AWS konten untuk mencerminkan perubahan yang dibuat di UI.	April 24, 2025
Memperbarui firewall dan gateway untuk memungkinkan akses	Daftar titik akhir dan sumber daya yang harus diizinkan terdaftar untuk mengakses semua layanan dan fitur di AWS Toolkit for Visual Studio Amazon Q untuk ekstensi.	Maret 20, 2025
Memecahkan masalah Firewall dan pengaturan proxy	Menambahkan topik pemecahan masalah baru yang menangani firewall dan pengaturan proxy untuk dan AWS Toolkit for Visual Studio Amazon Q.	Desember 15, 2024
Pemecahan masalah pembaruan instalasi	Memperbarui konten masalah instalasi ke akun untuk pembaruan dari Microsoft.	November 20, 2024

Pembaruan untuk Memulai Konten	Pembaruan yang dilakukan untuk Memulai dan Menghubungkan ke AWS konten untuk mencerminkan perubahan yang dibuat di UI.	Oktober 24, 2024
Pembaruan untuk Menghubungkan ke AWS	Pembaruan dilakukan untuk Menghubungkan ke AWS konten.	September 26, 2024
Pembaruan untuk konten Amazon EC2 AMI	Pembaruan konten telah dilakukan untuk mendokumentasikan perubahan pada proses dan prosedur Amazon EC2 AMI.	September 13, 2024
AWS Komponen toolkit tidak dapat diinisialisasi	Menambahkan topik pemecahan masalah untuk mengatasi masalah dengan AWS Toolkit for Visual Studio komponen yang tidak diinisialisasi.	September 13, 2024
Melihat dan memfilter pemindaian keamanan Amazon Q	Menambahkan topik pemecahan masalah untuk membantu melihat dan memfilter pemindaian keamanan Amazon Q.	Juli 31, 2024
Amazon Q untuk AWS Toolkit for Visual Studio	Amazon Q sekarang tersedia untuk AWS Toolkit for Visual Studio.	Juni 30, 2024
Pembaruan dan pemeliharaan konten	Memperbarui konten untuk perubahan pada UI dan pedoman AWS gaya.	Maret 6, 2024

Pembaruan dan pemeliharaan konten	Memperbarui konten untuk perubahan pada UI dan pedoman AWS gaya.	Maret 6, 2024
Pembaruan dan pemeliharaan konten	Memperbarui konten untuk perubahan pada UI dan pedoman AWS gaya.	Maret 6, 2024
Pembaruan dan pemeliharaan konten	Memperbarui konten untuk perubahan pada UI dan pedoman AWS gaya.	Maret 6, 2024
Pembaruan dan pemeliharaan konten	Memperbarui konten untuk perubahan pada UI dan pedoman AWS gaya.	Maret 6, 2024
Pembaruan untuk mengatur dan otentikasi	Topik penyiapan dan otentikasi telah diperbarui untuk meningkatkan keamanan dan pengalaman orientasi toolkit. Lihat topik Memulai dan Otentikasi dan akses TOCs untuk melihat perubahan.	22 Juni 2023
Otentikasi dan akses	Memberikan AWS kredensi sekarang adalah Otentikasi dan akses. Refactoring TOC dan subtopik untuk memenuhi persyaratan AWS gaya dan keamanan.	4 Mei 2023
Pembaruan pada bagian dan topik Menyiapkan	Menyiapkan AWS Toolkit for Visual Studio bagian dan topik dalam Panduan Pengguna ini telah diperbarui untuk meningkatkan pengalaman naik pesawat untuk AWS Toolkit for Visual Studio.	30 Januari 2023

Pembaruan pada bagian dan topik Menyiapkan	Menyiapkan AWS Toolkit for Visual Studio bagian dan topik dalam Panduan Pengguna ini telah diperbarui untuk meningkatkan pengalaman naik pesawat untuk AWS Toolkit for Visual Studio.	30 Januari 2023
Menambahkan AWS Toolkit for Visual Studio informasi 2022	Support untuk Visual Studio 2022 telah ditambahkan ke AWS Toolkit for Visual Studio.	Desember 20, 2022
Pembaruan untuk Publikasikan ke AWS panduan	Pembaruan dokumentasi untuk mencerminkan perubahan yang dibuat pada layanan untuk peluncuran GA.	6 Juli 2022
Pembaruan judul dan relokasi	Perubahan judul kecil dilakukan untuk mencerminkan konten dengan lebih baik. Panduan sekarang terletak di Publishing to AWS guide.	6 Juli 2022

Menyebarkan ke AWS: pembaruan judul dan konten	Bagian panduan secara resmi berjudul: Deployment Using the AWS Toolkit, memiliki daftar isi (TOC) yang diperbarui dan sekarang berjudul: Deploying to. AWS Panduan berikut telah menyelesaikan penghentian dan tidak lagi dapat diakses: Deploying to Elastic Beanstalk (Legacy) dan Deploying to (Legacy). AWS CloudFormation Konten yang diperbarui mengenai penyebaran ke Elastic Beanstalk dan Cloudformation dapat ditemukan dari TOC yang diperbarui dalam panduan ini.	6 Juli 2022
Menerapkan Aplikasi ASP.NET Core 2.0 (Fargate) sekarang menjadi panduan lama	Dokumentasi ini mengacu pada layanan dan fitur lama. Untuk panduan dan konten yang diperbarui, lihat panduan AWS alat.NET Deployment dan AWS daftar isi Deploying to yang diperbarui.	6 Juli 2022
Menerapkan Aplikasi ASP.NET sekarang menjadi panduan lama	Dokumentasi ini mengacu pada layanan dan fitur lama. Untuk panduan dan konten yang diperbarui, lihat panduan alat penyebaran AWS .NET dan AWS daftar isi Deploying to yang diperbarui.	6 Juli 2022

Menerapkan Aplikasi ASP.NET sekarang menjadi panduan lama	Dokumentasi ini mengacu pada layanan dan fitur lama. Untuk panduan dan konten yang diperbarui, lihat panduan alat penyebaran AWS .NET dan AWS daftar isi Deploying to yang diperbarui.	6 Juli 2022
Topik panduan baru: Bekerja dengan CloudWatch Log di Visual Studio	Membuat topik ikhtisar baru untuk integrasi Amazon CloudWatch Logs dalam panduan Visual Studio.	Juni 29, 2022
Topik panduan baru: Menyiapkan integrasi CloudWatch Log untuk Visual Studio	Membuat bagian pengaturan baru untuk integrasi Amazon CloudWatch Logs dalam panduan Visual Studio.	Juni 29, 2022
CloudWatch Integrasi log untuk Visual Studio	Membuat panduan baru untuk integrasi Amazon CloudWatch Log di Visual Studio, termasuk topik panduan: Menyiapkan CloudWatch Log untuk Visual Studio dan Bekerja dengan CloudWatch Log di Visual Studio.	Juni 29, 2022
Publikasikan ke AWS	Publikasikan ke AWS tidak lagi dalam pratinjau. Pembaruan untuk mencermin kan perubahan pada UI dan peningkatan saran penerbitan.	1 Juni 2022

Publikasikan baru untuk AWS tersedia untuk pratinjau	Pengalaman penerapan yang disempurnakan yang memberikan panduan tentang AWS layanan mana yang tepat untuk aplikasi Anda.	21 Oktober 2021
Dukungan SSO dan MFA untuk kredensi AWS	Diperbarui untuk mendokumentasikan dukungan baru untuk AWS Single Sign-On (IAM Identity Center) dan otentikasi multi-faktor dalam kredensi. AWS	21 April 2021
AWS Lambda Proyek Dasar Membuat Gambar Docker	Menambahkan dukungan untuk citra kontainer Lambda.	1 Desember 2020
Konten Keamanan	Menambahkan konten keamanan.	6 Februari 2020
Memberikan AWS kredensi	Diperbarui dengan informasi tentang membuat profil kredensi di file AWS kredensial bersama.	20 Juni 2019
Menggunakan Proyek AWS Lambda di AWS Toolkit for Visual Studio	Support untuk Visual Studio 2019 telah ditambahkan ke AWS Toolkit for Visual Studio.	28 Maret 2019
Tutorial: Membuat Aplikasi Amazon Rekognition Lambda	Support untuk Visual Studio 2019 telah ditambahkan ke AWS Toolkit for Visual Studio.	28 Maret 2019
Tutorial: Membangun dan Menguji Aplikasi Tanpa Server dengan Lambda AWS	Support untuk Visual Studio 2019 telah ditambahkan ke AWS Toolkit for Visual Studio.	28 Maret 2019

Menyiapkan AWS Toolkit for Visual Studio	Support untuk Visual Studio 2019 telah ditambahkan ke AWS Toolkit for Visual Studio.	28 Maret 2019
Menerapkan Aplikasi ASP.NET Core 2.0 (Fargate)	Support untuk Visual Studio 2019 telah ditambahkan ke AWS Toolkit for Visual Studio.	28 Maret 2019
Menerapkan Aplikasi ASP.NET Core 2.0 () EC2	Support untuk Visual Studio 2019 telah ditambahkan ke AWS Toolkit for Visual Studio.	28 Maret 2019
Membuat Proyek AWS CloudFormation Template di Visual Studio	Support untuk Visual Studio 2019 telah ditambahkan ke AWS Toolkit for Visual Studio.	28 Maret 2019
Tampilan Detil dari Layanan Kontainer	Menambahkan informasi tentang tampilan mendetail klaster Amazon Elastic Container Service dan repositori kontainer yang disediakan oleh Explorer. AWS	16 Februari 2018
Menyebarkan ke Amazon EC2 Container Service	Menambahkan informasi tentang penerapan ke layanan EC2 kontainer Amazon.	16 Februari 2018
Menyebarkan Layanan Kontainer menggunakan Fargate	Menambahkan informasi tentang cara menerapkan aplikasi ASP.NET Core 2.0 kontainer yang menargetkan Linux melalui Amazon ECS menggunakan tipe peluncuran Fargate.	16 Februari 2018

Menyebarkan Layanan Kontainer menggunakan EC2	Menambahkan informasi tentang cara menerapkan aplikasi ASP.NET Core 2.0 kontainer yang menargetkan Linux melalui Amazon ECS menggunakan tipe peluncuran. EC2	16 Februari 2018
Kredensi untuk Menerapkan ke Amazon Container Service EC2	Menambahkan informasi tentang cara menentukan kredensional saat menerapkan ke layanan penampung Amazon EC2.	16 Februari 2018

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.