



Panduan Pengguna

# AWS Batas Waktu Cloud



Versi latest

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

# AWS Batas Waktu Cloud: Panduan Pengguna

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara pelanggan, atau dengan cara apa pun yang merendahkan atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan hak milik masing-masing pemiliknya, yang mungkin atau tidak terafiliasi, terkait dengan, atau disponsori oleh Amazon.

---

# Table of Contents

Apa itu Deadline Cloud? .....	1
Fitur Deadline Cloud .....	1
Konsep dan terminologi .....	2
Memulai dengan Deadline Cloud .....	4
Mengakses Deadline Cloud .....	5
Layanan terkait .....	5
Bagaimana Deadline Cloud bekerja .....	6
.....	7
Izin di Deadline Cloud .....	7
Dukungan perangkat lunak dengan Deadline Cloud .....	8
Memulai .....	9
Siapkan Anda Akun AWS .....	9
Siapkan monitor Anda .....	10
Langkah 1: Siapkan monitor Anda .....	10
Langkah 2: Tentukan detail pertanian .....	13
Langkah 3: Tentukan detail antrian .....	14
Langkah 4: Tentukan detail armada .....	15
Langkah 5: Konfigurasi kemampuan pekerja .....	16
Langkah 6: Tentukan tingkat akses .....	17
Langkah 7: Tinjau dan buat .....	17
Siapkan pengirim .....	17
Langkah 1: Instal pengirim Cloud Deadline .....	18
Langkah 2: Instal dan atur monitor Deadline Cloud .....	25
Langkah 3: Luncurkan submitter Deadline Cloud .....	28
Gunakan peternakan .....	31
Menggunakan monitor .....	33
Bagikan URL monitor Cloud Deadline .....	33
Buka monitor Deadline Cloud .....	34
Lihat detail antrian dan armada .....	36
Melihat dan mengelola pekerjaan, langkah, dan tugas .....	37
Arsipkan pekerjaan .....	38
Meminta pekerjaan .....	38
Lihat detail pekerjaan .....	38
Lihat langkah .....	39

Lihat tugas .....	40
Melihat log .....	40
Unduh output jadi .....	42
Peternakan .....	43
Buat peternakan .....	43
Hapus peternakan .....	43
Edit peternakan .....	44
Antrean .....	45
Membuat antrean .....	45
Buat lingkungan antrian .....	47
Default Conda lingkungan antrian .....	48
Hapus antrian .....	49
Mengedit antrian .....	50
Kaitkan antrian dan armada .....	50
Armada .....	51
Armada yang dikelola layanan .....	51
Gunakan lisensi Anda sendiri .....	53
VFXplatform .....	69
Armada yang dikelola pelanggan .....	70
Buat CMF .....	71
Pengaturan host pekerja .....	76
Kelola akses .....	81
Instal perangkat lunak untuk pekerjaan .....	83
Konfigurasi kredensial .....	84
Buat AMI .....	86
Buat infrastruktur armada .....	89
Connect ke titik akhir lisensi .....	100
Mengelola pengguna .....	104
Kelola pengguna dan grup untuk monitor .....	104
Kelola pengguna dan grup untuk peternakan, antrian, dan armada .....	106
Tugas .....	109
Mengirimkan pekerjaan .....	110
Lebih banyak opsi untuk mengirimkan pekerjaan .....	112
Penjadwalan pekerjaan .....	114
Tentukan kompatibilitas armada .....	114
Penskalaan armada .....	116

Sesi .....	116
Ketergantungan langkah .....	118
Status Job .....	119
Memodifikasi pekerjaan .....	122
Lowongan kerja pengolahan .....	127
Pemecahan masalah pekerjaan .....	128
Mengapa membuat pekerjaan saya gagal? .....	128
Mengapa pekerjaan saya tidak kompatibel? .....	128
Mengapa pekerjaan saya terjebak dalam siap? .....	129
Mengapa pekerjaan saya gagal? .....	129
Mengapa langkah saya tertunda? .....	129
Penyimpanan .....	130
Lampiran Job .....	130
Enkripsi untuk bucket S3 lampiran pekerjaan .....	131
Mengelola lampiran pekerjaan di bucket S3 .....	132
Sistem file virtual .....	132
Penyimpanan bersama .....	135
Profil penyimpanan di Deadline Cloud .....	135
Mengelola anggaran dan penggunaan .....	138
Asumsi biaya .....	138
Menggunakan manajer anggaran .....	139
Prasyarat .....	140
Akses pengelola anggaran .....	140
Buat anggaran .....	140
Lihat anggaran .....	142
Edit anggaran .....	142
Nonaktifkan anggaran .....	142
Menggunakan penjelajah penggunaan .....	143
Prasyarat .....	143
Buka penjelajah penggunaan .....	143
Gunakan penjelajah penggunaan .....	143
Manajemen biaya .....	146
Praktik terbaik manajemen biaya .....	147
Keamanan .....	150
Perlindungan data .....	151
Enkripsi diam .....	152

Enkripsi bergerak .....	152
Manajemen kunci .....	153
Privasi lalu lintas antar jaringan .....	162
Menyisih .....	163
Identity and Access Management .....	164
Audiens .....	164
Mengautentikasi dengan identitas .....	165
Mengelola akses menggunakan kebijakan .....	169
Bagaimana Deadline Cloud bekerja dengan IAM .....	171
Contoh kebijakan berbasis identitas .....	178
AWS kebijakan terkelola .....	182
Pemecahan Masalah .....	186
Validasi kepatuhan .....	188
Ketangguhan .....	189
Keamanan infrastruktur .....	190
Konfigurasi dan analisis kerentanan .....	191
Pencegahan confused deputy lintas layanan .....	191
AWS PrivateLink .....	193
Pertimbangan .....	193
Deadline Cloud titik akhir .....	193
Buat titik akhir .....	194
Praktik terbaik keamanan .....	195
Perlindungan data .....	195
IAMizin .....	196
Jalankan pekerjaan sebagai pengguna dan grup .....	196
Jaringan .....	197
Data Job .....	197
Struktur pertanian .....	198
Antrian lampiran pekerjaan .....	198
Bucket perangkat lunak khusus .....	200
Tuan rumah pekerja .....	201
Workstation .....	202
Pemantauan .....	204
Logging dengan CloudTrail .....	205
Informasi Batas waktu Cloud di CloudTrail .....	205
Memahami Entri file log Deadline Cloud .....	209

---

Pemantauan CloudWatch dengan .....	211
Bertindak pada EventBridge acara .....	212
Perubahan rekomendasi ukuran armada .....	212
Kuota .....	215
AWS CloudFormation sumber daya .....	216
Tenggat waktu Cloud dan template AWS CloudFormation .....	216
Pelajari lebih lanjut tentang AWS CloudFormation .....	216
Riwayat dokumen .....	218
AWS Glosarium .....	220
.....	ccxxi

# Apa itu AWS Deadline Cloud?

Deadline Cloud layanan AWS dapat digunakan untuk membuat dan mengelola proyek dan pekerjaan rendering di instans Amazon Elastic Compute Cloud (AmazonEC2) langsung dari pipeline pembuatan konten digital dan workstation.

Deadline Cloud menyediakan antarmuka konsol, aplikasi lokal, alat baris perintah, dan file. API Dengan Deadline Cloud, Anda dapat membuat, mengelola, dan memantau peternakan, armada, pekerjaan, grup pengguna, dan penyimpanan. Anda juga dapat menentukan kemampuan perangkat keras, membuat lingkungan untuk beban kerja tertentu, dan mengintegrasikan alat pembuatan konten yang diperlukan produksi Anda ke dalam pipeline Deadline Cloud Anda.

Deadline Cloud menyediakan antarmuka terpadu untuk mengelola semua proyek rendering Anda di satu tempat. Anda dapat mengelola pengguna, menetapkan proyek kepada mereka, dan memberikan izin untuk peran pekerjaan.

## Topik

- [Fitur Deadline Cloud](#)
- [Konsep dan terminologi untuk Deadline Cloud](#)
- [Memulai dengan Deadline Cloud](#)
- [Mengakses Deadline Cloud](#)
- [Layanan terkait](#)
- [Bagaimana Deadline Cloud bekerja](#)

## Fitur Deadline Cloud

Berikut adalah beberapa cara utama Deadline Cloud dapat membantu Anda menjalankan dan mengelola beban kerja komputasi visual:

- Buat peternakan, antrian, dan armada Anda dengan cepat. Pantau status mereka, dan dapatkan wawasan tentang pengoperasian pertanian dan pekerjaan Anda.
- Kelola pengguna dan grup Deadline Cloud secara terpusat, dan tetapkan izin.
- Kelola keamanan masuk untuk pengguna proyek dan penyedia identitas eksternal dengan AWS IAM Identity Center.

- Mengelola akses ke sumber daya proyek secara aman dengan AWS Identity and Access Management (IAM) kebijakan dan peran.
- Gunakan tag untuk mengatur dan menemukan sumber daya proyek dengan cepat.
- Kelola penggunaan sumber daya proyek dan perkiraan biaya untuk proyek Anda.
- Menyediakan berbagai pilihan manajemen komputasi untuk mendukung rendering di cloud atau secara langsung.

## Konsep dan terminologi untuk Deadline Cloud

Untuk membantu Anda memulai dengan AWS Deadline Cloud, topik ini menjelaskan beberapa konsep dan terminologi utamanya.

### Manajer anggaran

Manajer anggaran adalah bagian dari monitor Deadline Cloud. Gunakan manajer anggaran untuk membuat dan mengelola anggaran. Anda juga dapat menggunakannya untuk membatasi aktivitas agar tetap sesuai anggaran.

### Pustaka Klien Cloud Batas Waktu

Pustaka Klien menyertakan antarmuka baris perintah dan pustaka untuk mengelola Deadline Cloud. Fungsionalitas termasuk mengirimkan bundel pekerjaan berdasarkan spesifikasi Open Job Description ke Deadline Cloud, mengunduh output lampiran pekerjaan, dan memantau pertanian Anda menggunakan antarmuka baris perintah.

### Aplikasi pembuatan konten digital (DCC)

Aplikasi pembuatan konten digital (DCCs) adalah produk pihak ketiga tempat Anda membuat konten digital. Contohnya DCCs adalah Maya, Nuke, dan Houdini. Deadline Cloud menyediakan plugin terintegrasi pengirim pekerjaan untuk spesifik. DCCs

### Peternakan

Peternakan adalah tempat sumber daya proyek Anda berada. Ini terdiri dari antrian dan armada.

### Armada

Armada adalah sekelompok node pekerja yang melakukan rendering. Node pekerja memproses pekerjaan. Armada dapat dikaitkan dengan beberapa antrian, dan antrian dapat dikaitkan dengan beberapa armada.

## Pekerjaan

Pekerjaan adalah permintaan rendering. Pengguna mengirimkan pekerjaan. Pekerjaan berisi properti pekerjaan tertentu yang diuraikan sebagai langkah dan tugas.

## Lampiran Job

Lampiran pekerjaan adalah fitur Deadline Cloud yang dapat Anda gunakan untuk mengelola input dan output untuk pekerjaan. File Job diunggah sebagai lampiran pekerjaan selama proses rendering. File-file ini dapat berupa tekstur, model 3D, rig pencahayaan, dan item serupa lainnya.

## Properti Job

Properti Job adalah pengaturan yang Anda tentukan saat mengirimkan pekerjaan render. Beberapa contoh termasuk rentang bingkai, jalur keluaran, lampiran pekerjaan, kamera yang dapat dirender, dan banyak lagi. Properti bervariasi berdasarkan dari mana DCC render dikirimkan.

## Templat Job

Template pekerjaan mendefinisikan lingkungan runtime dan semua proses yang berjalan sebagai bagian dari pekerjaan Deadline Cloud.

## Antrean

Antrian adalah tempat pekerjaan yang diajukan berada dan dijadwalkan akan diberikan. Antrian harus dikaitkan dengan armada untuk membuat render yang berhasil. Antrian dapat dikaitkan dengan beberapa armada.

## Asosiasi antrian armada

Ketika antrian dikaitkan dengan armada, ada asosiasi antrian-armada. Gunakan asosiasi untuk menjadwalkan pekerja dari armada ke pekerjaan dalam antrian itu. Anda dapat memulai dan menghentikan asosiasi untuk mengontrol penjadwalan kerja.

## Langkah

Langkah adalah salah satu proses khusus untuk dijalankan dalam pekerjaan.

## Batas waktu pengirim Cloud

Submitter Deadline Cloud adalah plugin pembuatan konten digital (DCC). Artis menggunakannya untuk mengirimkan pekerjaan dari DCC antarmuka pihak ketiga yang mereka kenal.

## Tanda

Tag adalah label yang dapat Anda tetapkan ke AWS sumber daya. Setiap tag terdiri dari kunci dan nilai opsional yang Anda tentukan.

Dengan tag, Anda dapat mengkategorikan AWS sumber daya Anda dengan berbagai cara. Misalnya, Anda dapat menentukan satu set tag untuk EC2 instans Amazon akun Anda yang membantu Anda melacak setiap pemilik instans dan tingkat tumpukan.

Anda juga dapat mengkategorikan AWS sumber daya Anda berdasarkan tujuan, pemilik, atau lingkungan. Pendekatan ini berguna ketika Anda memiliki banyak sumber daya dari jenis yang sama. Anda dapat dengan cepat mengidentifikasi sumber daya tertentu berdasarkan tag yang telah Anda tetapkan padanya.

## Tugas

Tugas adalah komponen tunggal dari langkah render.

## Lisensi berbasis penggunaan (U) UBL

Lisensi berbasis penggunaan (UBL) adalah model lisensi sesuai permintaan yang tersedia untuk produk pihak ketiga tertentu. Model ini dibayar sesuai keinginan Anda, dan Anda dikenakan biaya untuk jumlah jam dan menit yang Anda gunakan.

## Penjelajah penggunaan

Penjelajah penggunaan adalah fitur monitor Deadline Cloud. Ini memberikan perkiraan perkiraan biaya dan penggunaan Anda.

## Pekerja

Pekerja termasuk dalam armada dan menjalankan tugas yang diberikan Deadline Cloud untuk menyelesaikan langkah dan pekerjaan. Pekerja menyimpan log dari operasi tugas di Amazon CloudWatch Logs. Pekerja juga dapat menggunakan fitur lampiran pekerjaan untuk menyinkronkan input dan output ke bucket Amazon Simple Storage Service (Amazon S3).

# Memulai dengan Deadline Cloud

Gunakan Deadline Cloud untuk membuat farm render dengan cepat dengan pengaturan dan sumber daya default, seperti konfigurasi EC2 instans Amazon dan bucket Amazon Simple Storage Service (Amazon S3).

Anda juga dapat menentukan pengaturan dan sumber daya saat membuat render farm. Metode ini membutuhkan lebih banyak waktu daripada menggunakan pengaturan dan sumber daya default tetapi memberi Anda lebih banyak kontrol.

Setelah Anda terbiasa dengan [Konsep dan terminologi](#) Deadline Cloud, lihat [Memulai](#) step-by-step petunjuk untuk membuat farm, menambahkan pengguna, dan tautan ke informasi bermanfaat.

## Mengakses Deadline Cloud

Anda dapat mengakses Deadline Cloud dengan salah satu cara berikut:

- **Konsol Cloud Deadline** — Akses konsol di browser untuk membuat pertanian dan sumber dayanya, dan mengelola akses pengguna. Untuk informasi selengkapnya, lihat [Memulai](#).
- **Monitor Cloud Deadline** — Kelola pekerjaan render Anda, termasuk memperbarui prioritas dan status pekerjaan. Pantau pertanian Anda dan lihat log dan status pekerjaan. Untuk pengguna dengan izin Pemilik, monitor Deadline Cloud juga menyediakan akses untuk mengeksplorasi penggunaan dan membuat anggaran. Monitor Deadline Cloud tersedia sebagai browser web dan aplikasi desktop.
- **AWS SDK dan AWS CLI** — Gunakan AWS Command Line Interface (AWS CLI) untuk memanggil API operasi Deadline Cloud dari baris perintah pada sistem lokal Anda. Untuk informasi selengkapnya, lihat [Menyiapkan stasiun kerja pengembang](#).

## Layanan terkait

Deadline Cloud bekerja dengan yang berikut: layanan AWS

- **Amazon CloudWatch** — Dengan CloudWatch, Anda dapat memantau proyek dan AWS sumber daya terkait. Untuk informasi selengkapnya, lihat [Pemantauan CloudWatch dengan](#).
- **Amazon EC2** — Ini layanan AWS menyediakan server virtual yang menjalankan aplikasi Anda di cloud. Anda dapat mengonfigurasi proyek untuk menggunakan EC2 instans Amazon untuk beban kerja Anda. Untuk informasi selengkapnya, lihat [EC2instans Amazon](#).
- **EC2Auto Scaling Amazon** — Dengan Auto Scaling, Anda dapat secara otomatis menambah atau mengurangi jumlah instans saat permintaan instans Anda berubah. Auto Scaling membantu memastikan bahwa Anda menjalankan jumlah instans yang diinginkan, meskipun instans gagal. Jika Anda mengaktifkan Auto Scaling dengan Deadline Cloud, instance yang diluncurkan oleh Auto Scaling secara otomatis terdaftar dengan beban kerja. Demikian juga, instance yang dihentikan

oleh Auto Scaling secara otomatis tidak terdaftar dari beban kerja. Untuk informasi selengkapnya, lihat [Panduan Pengguna Amazon EC2 Auto Scaling](#).

- AWS PrivateLink— AWS PrivateLink menyediakan konektivitas pribadi antara virtual private cloud (VPCs), layanan AWS, dan jaringan lokal Anda, tanpa mengekspos lalu lintas Anda ke internet publik. AWS PrivateLink membuatnya mudah untuk menghubungkan layanan di berbagai akun dan VPCs. Untuk informasi selengkapnya, lihat [AWS PrivateLink](#).
- Amazon S3 - Amazon S3 adalah layanan penyimpanan objek. Deadline Cloud menggunakan bucket Amazon S3 untuk menyimpan lampiran pekerjaan. Untuk informasi selengkapnya, lihat [Panduan Pengguna Amazon S3](#).
- IAM Pusat IAM Identitas - Pusat Identitas adalah layanan AWS tempat Anda dapat memberi pengguna akses masuk tunggal ke semua akun dan aplikasi yang ditetapkan dari satu tempat. Anda juga dapat mengelola akses multi-akun dan izin pengguna secara terpusat ke semua akun Anda. AWS Organizations Untuk informasi lebih lanjut, lihat [AWS IAM Identity Center FAQs](#).

## Bagaimana Deadline Cloud bekerja

Dengan Deadline Cloud, Anda dapat membuat dan mengelola proyek dan pekerjaan rendering langsung dari pipeline dan workstation pembuatan konten digital. DCC

Anda mengirimkan lowongan ke Deadline Cloud menggunakan pengirim pekerjaan AWS SDK, AWS Command Line Interface (AWS CLI), atau Deadline Cloud. Deadline Cloud mendukung Open Job Description (OpenJD) untuk spesifikasi template pekerjaan. Untuk informasi selengkapnya, lihat [Open Job Description](#) di GitHub situs web.

Deadline Cloud menyediakan pengirim pekerjaan. Pengirim pekerjaan adalah DCC plugin untuk mengirimkan pekerjaan render dari DCC antarmuka pihak ketiga, seperti atau. Maya Nuke Dengan submitter, artis dapat mengirimkan pekerjaan rendering dari antarmuka pihak ketiga ke Deadline Cloud di mana sumber daya proyek dikelola dan pekerjaan dipantau, semuanya di satu lokasi.

Dengan Deadline Cloud farm, Anda dapat membuat antrian dan armada, mengelola pengguna, dan mengelola penggunaan dan biaya sumber daya proyek. Sebuah peternakan terdiri dari antrian dan armada. Antrian adalah tempat pekerjaan yang diajukan berada dan dijadwalkan akan diberikan. Armada adalah sekelompok node pekerja yang menjalankan tugas untuk menyelesaikan pekerjaan. Antrian harus dikaitkan dengan armada sehingga pekerjaan dapat dibuat. Sebuah armada tunggal dapat mendukung beberapa antrian dan antrian dapat didukung oleh beberapa armada.

Pekerjaan terdiri dari langkah-langkah, dan setiap langkah terdiri dari tugas-tugas tertentu. Dengan monitor Deadline Cloud, Anda dapat mengakses status, log, dan metrik pemecahan masalah lainnya untuk pekerjaan, langkah, dan tugas.

## Izin di Deadline Cloud

Deadline Cloud mendukung hal-hal berikut:

- Mengelola akses ke API operasinya menggunakan AWS Identity and Access Management (IAM)
- Mengelola akses pengguna tenaga kerja menggunakan integrasi dengan AWS IAM Identity Center

Sebelum ada yang dapat mengerjakan proyek, mereka harus memiliki akses ke proyek itu dan pertanian terkait. Deadline Cloud terintegrasi dengan IAM Identity Center untuk mengelola otentikasi dan otorisasi tenaga kerja. Pengguna dapat ditambahkan langsung ke Pusat IAM Identitas, atau izin dapat dihubungkan ke penyedia identitas Anda yang ada (IDP) seperti Okta atau Active Directory Administrator TI dapat memberikan izin akses kepada pengguna dan grup pada tingkat yang berbeda. Setiap level berikutnya mencakup izin untuk level sebelumnya. Daftar berikut menjelaskan empat tingkat akses dari tingkat terendah ke tingkat tertinggi:

- Penampil — Izin untuk melihat sumber daya di peternakan, antrian, armada, dan pekerjaan yang dapat mereka akses. Penampil tidak dapat mengirimkan atau membuat perubahan pada pekerjaan.
- Kontributor — Sama seperti pemirsa, tetapi dengan izin untuk mengirimkan pekerjaan ke antrian atau peternakan.
- Manajer — Sama seperti kontributor, tetapi dengan izin untuk mengedit pekerjaan dalam antrian yang dapat mereka akses, dan memberikan izin pada sumber daya yang dapat mereka akses.
- Pemilik — Sama seperti manajer, tetapi dapat melihat dan membuat anggaran dan melihat penggunaan.

### Note

Izin ini tidak memberi pengguna akses ke AWS Management Console atau izin untuk mengubah infrastruktur Deadline Cloud.

Pengguna harus memiliki akses ke peternakan sebelum mereka dapat mengakses antrian dan armada terkait. Akses pengguna ditetapkan ke antrian dan armada secara terpisah di dalam peternakan.

Anda dapat menambahkan pengguna sebagai individu atau sebagai bagian dari grup. Menambahkan grup ke peternakan, armada, atau antrian dapat mempermudah pengelolaan izin akses untuk sekelompok besar orang. Misalnya, jika Anda memiliki tim yang mengerjakan proyek tertentu, Anda dapat menambahkan setiap anggota tim ke grup. Kemudian, Anda dapat memberikan izin akses ke seluruh grup untuk pertanian, armada, atau antrian yang sesuai.

## Dukungan perangkat lunak dengan Deadline Cloud

Deadline Cloud bekerja dengan aplikasi perangkat lunak apa pun yang dapat dijalankan dari antarmuka baris perintah dan dikendalikan dengan menggunakan nilai parameter. Deadline Cloud mendukung OpenJD spesifikasi untuk menggambarkan pekerjaan sebagai pekerjaan dengan langkah-langkah skrip perangkat lunak yang diparameterisasi (seperti melintasi rentang bingkai) ke dalam tugas. Kumpulkan instruksi OpenJD pekerjaan menjadi bundel pekerjaan dengan alat dan fitur Deadline Cloud untuk membuat, menjalankan, dan melisensikan langkah-langkah dari aplikasi perangkat lunak pihak ketiga.

Pekerjaan membutuhkan lisensi untuk dirender. Deadline Cloud menawarkan usage-based-licensing (UBL) untuk pilihan lisensi aplikasi perangkat lunak yang ditagih per jam per menit berdasarkan penggunaan. Dengan Deadline Cloud, Anda juga dapat menggunakan lisensi perangkat lunak Anda sendiri jika Anda mau. Jika suatu pekerjaan tidak dapat mengakses lisensi, itu tidak akan dirender dan menghasilkan kesalahan yang ditampilkan di log tugas di monitor Deadline Cloud.

# Memulai dengan Deadline Cloud

Untuk membuat peternakan di AWS Deadline Cloud, Anda dapat menggunakan [konsol Deadline Cloud](#) atau AWS Command Line Interface (AWS CLI). Gunakan konsol untuk pengalaman terpandu menciptakan pertanian, termasuk antrian dan armada. Gunakan AWS CLI untuk bekerja secara langsung dengan layanan, atau untuk mengembangkan alat Anda sendiri yang bekerja dengan Deadline Cloud.

Untuk membuat farm dan menggunakan monitor Deadline Cloud, siapkan akun Anda untuk Deadline Cloud. Anda hanya perlu menyiapkan infrastruktur monitor Deadline Cloud sekali per akun. Dari peternakan Anda, Anda dapat mengelola proyek Anda, termasuk akses pengguna ke pertanian Anda dan sumber dayanya.

Untuk membuat farm tanpa menyiapkan infrastruktur monitor Deadline Cloud, siapkan workstation pengembang untuk Deadline Cloud.

Untuk membuat peternakan dengan sumber daya minimal untuk menerima pekerjaan, pilih Mulai cepat di halaman beranda konsol. [Siapkan monitor Cloud Deadline](#) memandu Anda melalui langkah-langkah itu. Peternakan ini dimulai dengan antrian dan armada yang secara otomatis terkait. Pendekatan ini adalah cara mudah untuk membuat peternakan gaya kotak pasir untuk bereksperimen.

## Topik

- [Siapkan Anda Akun AWS](#)
- [Siapkan monitor Cloud Deadline](#)
- [Mengatur Deadline Pengirim Cloud](#)
- [Gunakan peternakan](#)

## Siapkan Anda Akun AWS

Siapkan Anda Akun AWS untuk menggunakan AWS Batas Waktu Cloud.

Jika Anda tidak memiliki Akun AWS, selesaikan langkah-langkah berikut untuk membuatnya.

Untuk mendaftar untuk Akun AWS

1. Buka <https://portal.aws.amazon.com/billing/pendaftaran>.

## 2. Ikuti petunjuk online.

Bagian dari prosedur pendaftaran melibatkan tindakan menerima panggilan telepon dan memasukkan kode verifikasi di keypad telepon.

Ketika Anda mendaftar untuk Akun AWS, sebuah Pengguna root akun AWS diciptakan. Pengguna root memiliki akses ke semua layanan AWS dan sumber daya di akun. Sebagai praktik keamanan terbaik, tetapkan akses administratif ke pengguna, dan gunakan hanya pengguna root untuk melakukan [tugas yang memerlukan akses pengguna root](#).

Ketika Anda pertama kali membuat Akun AWS, Anda mulai dengan satu identitas masuk yang memiliki akses lengkap ke semua layanan AWS dan sumber daya di akun. Identitas ini disebut Akun AWS pengguna root dan diakses dengan masuk dengan alamat email dan kata sandi yang Anda gunakan untuk membuat akun.

### Important

Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Lindungi kredensial pengguna root Anda dan gunakan kredensial tersebut untuk melakukan tugas yang hanya dapat dilakukan pengguna root. Untuk daftar lengkap tugas yang mengharuskan Anda masuk sebagai pengguna root, lihat [Tugas yang memerlukan kredensi pengguna root](#) di IAMPanduan Pengguna.

## Siapkan monitor Cloud Deadline

Untuk memulai, Anda harus membuat infrastruktur monitor Deadline Cloud dan menentukan pertanian Anda. Anda juga dapat melakukan langkah-langkah opsional tambahan termasuk menambahkan grup dan pengguna, memilih peran layanan, dan menambahkan tag ke sumber daya Anda.

### Langkah 1: Siapkan monitor Anda

Monitor Deadline Cloud menggunakan AWS IAM Identity Center untuk mengotorisasi pengguna. Instance IAM Identity Center yang Anda gunakan untuk Deadline Cloud harus sama Wilayah AWS sebagai monitor. Jika konsol Anda menggunakan Wilayah yang berbeda saat membuat monitor, Anda akan mendapatkan peringatan untuk mengubah ke Wilayah Pusat IAM Identitas.

Infrastruktur monitor Anda terdiri dari komponen-komponen berikut:

- Nama tampilan monitor: Nama tampilan Monitor adalah bagaimana Anda dapat mengidentifikasi monitor Anda - misalnya AnyCompany monitor. Nama monitor Anda juga menentukan monitor AndaURL.

 Important

Anda tidak dapat mengubah nama tampilan monitor setelah Anda selesai menyiapkan.

- Monitor URL: Anda dapat mengakses monitor Anda dengan menggunakan Monitor URL. URL ini didasarkan pada nama tampilan Monitor - misalnya <https://anycompanymonitor.awsapps.com>.

 Important

Anda tidak dapat mengubah Monitor URL setelah Anda selesai menyiapkan.

- Wilayah AWS: Wilayah AWS adalah lokasi fisik untuk koleksi AWS pusat data. Ketika Anda mengatur monitor Anda, Region default ke lokasi terdekat dengan Anda. Kami merekomendasikan untuk mengubah Wilayah sehingga letaknya paling dekat dengan pengguna Anda. Ini mengurangi lag dan meningkatkan kecepatan transfer data. AWS IAM Identity Center harus diaktifkan dalam hal yang sama Wilayah AWS sebagai Deadline Cloud.

 Important

Anda tidak dapat mengubah Region setelah selesai menyiapkan Deadline Cloud.

Selesaikan tugas di bagian ini untuk mengonfigurasi infrastruktur monitor Anda.

Untuk mengonfigurasi infrastruktur monitor Anda

1. Masuk ke AWS Management Console untuk memulai persiapan Welcome to Deadline Cloud, lalu pilih Berikutnya.
2. Masukkan nama tampilan Monitor — misalnya **AnyCompany Monitor**.
3. (Opsional) Untuk mengubah nama Monitor, pilih Edit URL.
4. (Opsional) Untuk mengubah Wilayah AWS jadi paling dekat dengan pengguna Anda, pilih Ubah Wilayah.

- a. Pilih Wilayah yang paling dekat dengan pengguna Anda.
  - b. Pilih Terapkan Wilayah.
- (Opsional) Untuk menambahkan grup dan pengguna, pilih [\(Opsional\) Tambahkan grup dan pengguna](#).
  - (Opsional) Untuk lebih menyesuaikan pengaturan monitor Anda, pilih [Pengaturan tambahan](#).
5. Jika Anda siap [Langkah 2: Tentukan detail pertanian](#), pilih Berikutnya.

## (Opsional) Tambahkan grup dan pengguna

Sebelum menyelesaikan pengaturan monitor Deadline Cloud, Anda dapat menambahkan pengguna monitor dan menambahkannya ke grup.

Setelah penyiapan selesai, Anda dapat membuat pengguna dan grup baru, dan mengelola pengguna seperti menetapkan grup, izin, dan aplikasi, atau menghapus pengguna dari monitor Anda.

## Pengaturan tambahan

Deadline Cloud setup mencakup pengaturan tambahan. Dengan pengaturan ini, Anda dapat melihat semua perubahan yang dibuat oleh pengaturan Deadline Cloud ke Akun AWS, konfigurasi peran pengguna monitor Anda, dan ubah jenis kunci enkripsi Anda.

### AWS IAM Identity Center

AWS IAM Identity Center adalah layanan masuk tunggal berbasis cloud untuk mengelola pengguna dan grup. IAM Identity Center juga dapat diintegrasikan dengan penyedia single sign-on (SSO) perusahaan Anda sehingga pengguna dapat masuk dengan akun perusahaan mereka.

Deadline Cloud mengaktifkan IAM Identity Center secara default, dan diperlukan untuk mengatur dan menggunakan Deadline Cloud. Instance IAM Identity Center yang Anda gunakan untuk Deadline Cloud harus sama Wilayah AWS sebagai monitor. Untuk informasi lebih lanjut, lihat [Apa itu AWS IAM Identity Center](#).

### Konfigurasi peran akses layanan

Sesi AWS Layanan dapat mengambil peran layanan untuk melakukan tindakan atas nama Anda. Deadline Cloud memerlukan peran pengguna monitor agar dapat memberi pengguna akses ke sumber daya di monitor Anda.

Anda dapat melampirkan AWS Identity and Access Management (IAM) kebijakan terkelola untuk memantau peran pengguna. Kebijakan tersebut memberi pengguna izin untuk melakukan tindakan tertentu, seperti membuat pekerjaan di aplikasi Deadline Cloud tertentu. Karena aplikasi bergantung pada kondisi tertentu dalam kebijakan terkelola, jika Anda tidak menggunakan kebijakan terkelola, aplikasi mungkin tidak berfungsi seperti yang diharapkan.

Anda dapat mengubah peran pengguna monitor setelah Anda menyelesaikan penyiapan, kapan saja. Untuk informasi selengkapnya tentang peran pengguna, lihat [IAMPeran](#).

Tab berikut berisi instruksi untuk dua kasus penggunaan yang berbeda. Untuk membuat dan menggunakan peran layanan baru, pilih tab Peran layanan baru. Untuk menggunakan peran layanan yang ada, pilih tab Peran layanan yang ada.

### New service role

Untuk membuat dan menggunakan peran layanan baru

1. Pilih Buat dan gunakan peran layanan baru.
2. (Opsional) Masukkan nama peran pengguna Layanan.
3. Pilih Lihat detail izin untuk informasi selengkapnya tentang peran tersebut.

### Existing service role

Untuk menggunakan peran layanan yang ada

1. Pilih Gunakan peran layanan yang ada.
2. Buka daftar dropdown untuk memilih peran layanan yang ada.
3. (Opsional) Pilih Lihat di IAM konsol untuk informasi selengkapnya tentang peran tersebut.

## Langkah 2: Tentukan detail pertanian

Kembali ke konsol Deadline Cloud, selesaikan langkah-langkah berikut untuk menentukan detail pertanian.

1. Di detail Pertanian, tambahkan Nama untuk pertanian.
2. Untuk Deskripsi, masukkan deskripsi pertanian. Deskripsi yang jelas dapat membantu Anda mengidentifikasi tujuan pertanian Anda dengan cepat.

3. (Opsional) Secara default, data Anda dienkripsi dengan kunci yang AWS memiliki dan mengelola untuk keamanan Anda. Anda dapat memilih Sesuaikan pengaturan enkripsi (lanjutan) untuk menggunakan kunci yang ada atau untuk membuat kunci baru yang Anda kelola.

Jika Anda memilih untuk menyesuaikan pengaturan enkripsi menggunakan kotak centang, masukkan AWS KMS ARN, atau buat yang baru AWS KMS dengan memilih Create new KMS key.

4. (Opsional) Pilih Tambahkan tag baru untuk menambahkan satu atau beberapa tag ke peternakan Anda.
5. Pilih salah satu opsi berikut:
  - Pilih Lewati untuk Meninjau dan Buat untuk [meninjau dan membuat peternakan Anda](#).
  - Pilih Berikutnya untuk melanjutkan ke langkah-langkah tambahan dan opsional.

## (Opsional) Langkah 3: Tentukan detail antrian

Antrian bertanggung jawab untuk melacak kemajuan dan penjadwalan pekerjaan untuk pekerjaan Anda.

1. Mulai dari detail Antrian, berikan Nama untuk antrian.
2. Untuk Deskripsi, masukkan deskripsi antrian. Deskripsi yang jelas dapat membantu Anda mengidentifikasi tujuan antrian dengan cepat.
3. Untuk lampiran Job, Anda dapat membuat bucket Amazon S3 baru atau memilih bucket Amazon S3 yang sudah ada. Jika Anda tidak memiliki bucket Amazon S3 yang ada, Anda harus membuatnya.
  - a. Untuk membuat bucket Amazon S3 baru, pilih Buat bucket pekerjaan baru. Anda dapat menentukan nama bucket pekerjaan di bidang awalan Root. Kami merekomendasikan memanggil ember **deadlinecloud-job-attachments-[MONITORNAME]**.

Anda hanya dapat menggunakan huruf kecil dan tanda hubung. Tidak ada spasi atau karakter khusus.
  - b. Untuk mencari dan memilih bucket Amazon S3 yang ada, pilih Pilih dari bucket Amazon S3 yang ada. Kemudian, cari bucket yang ada dengan memilih Browse S3. Saat daftar bucket Amazon S3 Anda yang tersedia ditampilkan, pilih bucket Amazon S3 yang ingin Anda gunakan untuk antrean Anda.

4. Jika Anda menggunakan armada yang dikelola pelanggan, pilih Aktifkan asosiasi dengan armada yang dikelola pelanggan.
  - Untuk armada yang dikelola pelanggan, tambahkan pengguna yang dikonfigurasi antrian, lalu atur kredensialnya dan/atau Windows. POSIX Atau, Anda dapat melewati fungsionalitas run-as dengan memilih kotak centang.
5. Antrian Anda memerlukan izin untuk mengakses Amazon S3 atas nama Anda. Kami menyarankan Anda membuat peran layanan baru untuk setiap antrian.
  - a. Untuk peran baru, selesaikan langkah-langkah berikut.
    - i. Pilih Buat dan gunakan peran layanan baru.
    - ii. Masukkan nama Peran untuk peran antrian Anda atau gunakan nama peran yang disediakan.
    - iii. (Opsional) Tambahkan peran antrian Deskripsi.
    - iv. Anda dapat melihat IAM izin untuk peran antrian dengan memilih Lihat detail izin.
  - b. Atau, Anda dapat memilih peran layanan yang ada.
6. (Opsional) Tambahkan variabel lingkungan untuk lingkungan antrian menggunakan nama dan pasangan nilai.
7. (Opsional) Tambahkan tag untuk antrian menggunakan pasangan kunci dan nilai.

Setelah Anda memasukkan semua detail antrian, pilih Berikutnya.

## (Opsional) Langkah 4: Tentukan detail armada

Armada mengalokasikan pekerja untuk melaksanakan tugas rendering Anda. Jika Anda membutuhkan armada untuk tugas rendering Anda, centang kotak untuk Buat armada.

1. Rincian armada
  - a. Berikan Nama dan Deskripsi opsional untuk armada Anda.
  - b. Pilih cara sumber daya komputasi Anda harus menskalakan. Opsi yang dikelola Layanan memungkinkan Deadline Cloud untuk menskalakan sumber daya komputasi Anda secara otomatis. Opsi yang dikelola Pelanggan membuat Anda mengendalikan penskalaan komputasi Anda sendiri.

2. Di bagian opsi Instans, pilih Spot atau Sesuai Permintaan. Instans Amazon EC2 On-Demand menyediakan ketersediaan yang lebih cepat dan instans Amazon EC2 Spot lebih baik untuk upaya penghematan biaya.
3. Untuk Penskalaan otomatis jumlah instans dalam armada Anda, pilih jumlah Instans Minimum dan Jumlah instans Maksimum.

Kami sangat menyarankan untuk selalu menetapkan jumlah minimum instans 0 untuk menghindari biaya tambahan.

4. Armada Anda memerlukan izin untuk CloudWatch menulis atas nama Anda. Kami menyarankan Anda membuat peran layanan baru untuk setiap armada.
  - a. Untuk peran baru, selesaikan langkah-langkah berikut.
    - i. Pilih Buat dan gunakan peran layanan baru.
    - ii. Masukkan nama Peran untuk peran armada Anda atau gunakan nama peran yang disediakan.
    - iii. (Opsional) Tambahkan peran armada Deskripsi.
    - iv. Untuk melihat IAM izin untuk peran armada, pilih Lihat detail izin.
  - b. Atau, Anda dapat menggunakan peran layanan yang ada.
5. (Opsional) Tambahkan tag untuk armada menggunakan pasangan kunci dan nilai.

Setelah Anda memasukkan semua detail armada, pilih Berikutnya.

## (Opsional) Langkah 5: Konfigurasi kemampuan pekerja

Tentukan kemampuan untuk instance pekerja Anda.

1. Pilih sistem operasi untuk pekerja di armada Anda. Untuk tutorial ini, tinggalkan default, Linux.
2. Tinjau pengaturan CPU arsitektur untuk kesadaran.
3. Perbarui jumlah minimum dan maksimum vCPUs untuk kemampuan perangkat keras Anda.
4. Perbarui jumlah memori minimum dan maksimum (GiB) untuk kemampuan perangkat keras Anda.
5. Anda dapat memfilter jenis instance dengan mengizinkan atau mengecualikan jenis instance pekerja. Di kedua opsi pemfilteran, Anda dapat memfilter hingga 10 jenis EC2 instans Amazon.
6. Di bawah Kemampuan tambahan (Opsional), Anda dapat menentukan EBS volume root berdasarkan Ukuran (GiB), IOPS, dan Throughput (MIB/s).

7. Setelah semua kemampuan pekerja ditetapkan, pilih Berikutnya untuk menentukan tingkat akses grup Anda.

## (Opsional) Langkah 6: Tentukan tingkat akses

Jika Anda memiliki grup yang terhubung ke monitor Anda, Anda dapat menentukan tingkat aksesnya. Izin untuk menggunakan fitur Deadline Cloud dikelola oleh tingkat akses. Anda dapat menetapkan tingkat akses yang berbeda ke grup pengguna.

1. Gunakan menu tingkat akses pertanian Deadline Cloud untuk memilih tingkat izin grup.
2. Pilih Berikutnya untuk melanjutkan dan meninjau semua detail pertanian yang dimasukkan.

## Langkah 7: Tinjau dan buat

Tinjau semua informasi yang dimasukkan untuk membuat peternakan Anda. Saat Anda siap, pilih Buat peternakan.

Kemajuan pembuatan peternakan Anda ditampilkan di halaman Peternakan. Pesan sukses ditampilkan saat peternakan Anda siap digunakan.

## Mengatur Deadline Pengirim Cloud

Proses ini untuk administrator dan artis yang ingin menginstal, mengatur, dan meluncurkan AWS Batas waktu pengirim Cloud. Submitter Deadline Cloud adalah plugin pembuatan konten digital (DCC). Artis menggunakannya untuk mengirimkan pekerjaan dari DCC antarmuka pihak ketiga yang mereka kenal.

### Note

Proses ini harus diselesaikan di semua workstation yang akan digunakan seniman untuk mengirimkan render.

Setiap workstation harus memiliki instalasi sebelum DCC menginstal submitter yang sesuai. Misalnya, jika Anda ingin mengunduh pengirim Deadline Cloud untuk Blender, Anda harus menginstal Blender di workstation Anda.

## Topik

- [Langkah 1: Instal pengirim Cloud Deadline](#)
- [Langkah 2: Instal dan atur monitor Deadline Cloud](#)
- [Langkah 3: Luncurkan submitter Deadline Cloud](#)

## Langkah 1: Instal pengirim Cloud Deadline

Bagian berikut memandu Anda melalui langkah-langkah untuk menginstal submitter Deadline Cloud.

### Unduh penginstal pengirim

Sebelum Anda dapat menginstal submitter Deadline Cloud, Anda harus mengunduh penginstal pengirim. Saat ini, penginstal submitter Deadline Cloud hanya mendukung Windows and Linux.

1. Masuk ke AWS Management Console dan buka [konsol](#) Deadline Cloud.
2. Dari panel navigasi samping, pilih Unduhan.
3. Temukan bagian Deadline Cloud submitter installer.
4. Pilih penginstal untuk sistem operasi komputer Anda, lalu pilih Unduh.

### (Opsional) Verifikasi keaslian perangkat lunak yang diunduh

Untuk memverifikasi bahwa perangkat lunak yang Anda unduh asli, gunakan prosedur berikut untuk salah satu Windows atau Linux. Anda mungkin ingin melakukan ini untuk memastikan tidak ada yang merusak file selama atau setelah proses pengunduhan.

Anda dapat menggunakan petunjuk ini untuk memverifikasi penginstal terlebih dahulu, dan kemudian memverifikasi monitor Deadline Cloud setelah Anda mengunduhnya. [Langkah 2: Instal dan atur monitor Deadline Cloud](#)

### Windows

Untuk memverifikasi keaslian file yang Anda unduh, selesaikan langkah-langkah berikut.

1. Dalam perintah berikut, ganti *file* dengan file yang ingin Anda verifikasi. Misalnya, **C:\PATH\TO\MY\DeadlineCloudSubmitter-windows-x64-installer.exe** . Juga, ganti *signtool-sdk-version* dengan versi SignTool SDKdiinstal. Misalnya, **10.0.22000.0**.

```
"C:\Program Files (x86)\Windows Kits\10\bin\signtool-sdk-  
version\x86\signtool.exe" verify /vfile
```

2. Misalnya, Anda dapat memverifikasi file installer submitter Deadline Cloud dengan menjalankan perintah berikut:

```
"C:\Program Files (x86)\Windows Kits\10\bin  
\10.0.22000.0\x86\signtool.exe" verify /v DeadlineCloudSubmitter-  
windows-x64-installer.exe
```

## Linux

Untuk memverifikasi keaslian file yang Anda unduh, gunakan alat baris gpg perintah.

1. Impor OpenPGP kunci dengan menjalankan perintah berikut:

```
gpg --import --armor <<EOF  
-----BEGIN PGP PUBLIC KEY BLOCK-----  
  
mQINBGX6GQsBEADduUtJgqSXI+q7606fsFwEYKmbnlyL0xKv1q32EZuyv0otZo5L  
le4m5Gg52AzrvPvDiUTLooAlvYeozaYyirIGsK08Ydz0Ftdjroiuh/mw9JSJDJRI  
rnRn5yKet1JFzckjopA3pjsTBP6lW/mb1bDBDEwwwtH0x91V7A03FJ9T7Uzu/qSh  
q0/UYdkafro3cPASvkqgDt2tCvURfBcUCAjZVFcLZcVD5iwXacxvKsxxS/e7kuVV  
I1+VGT8Hj8XzWYhjCZx0LZk/fvpYPMYEEujN0fYUp6RtMIXve0C9awwMCy5nBG2J  
eE2015DsCpTaBd4Fdr3LWcSs8JFA/YfP9auL3Ncz0ozPoVJt+fw8CB1VIX00J715  
hvHDjcC+5v0wxqALMG6+f/SX7CT8FXK+L3i0J5gBYUNXqHSxUdv8kt76/KVmQa1B  
Ak1+MPKpMq+1hw++S3G/1XqwWADNQBRRw7dSZHymQVXvPp1nsgc3hV7K10M+6s6g  
1g4mvFY41f6DhptwZLWyQXU8rBQpojvQfiSmDFrFPWF15BexesuVnkGIo1Qok1Kx  
AVUSdJPVEJCteyy7td4FPhBaSqT5vW3+ANbr9b/uoRYWJvn17dN0cc9HuRh/Ai+I  
nkfECo2WUDLZ0fEKGjGyFX+todWvJXjvc5kmE9Ty5vJp+M9Vvb8jd6t+mwARAQAB  
tCxBV1MgRGVhZGxpbmUgQ2xvdWQgPGF3cy1kZWFKbGluZUBhbWF6b24uY29tPokC  
VwQTAQgAQRyhbLhAwIwpqQeWoHH6pfbNP0a3bzzvBQJl+hkLAXsvBAUJA8JnAAUL  
CQgHAgIiAgYVCgkICwIDFgIBAh4HAheAAAoJEPbNP0a3bzzvKswQAjXzKSAY8sY8  
F6Eas2oYwIDDdDurs8FiEnFghjUE06MTt9AykF/jw+CQg2UzFtEy0bHBymghmXE  
3buVeom96tgM3ZDfZu+sxi5pGX6oAQnZ6riztN+VpkpQmLgwtMGpSML13KLwnv2k  
WK8mrR/fPMkfaewB7A6RIUYiW33GAL4KfMIs8/vIwIJw99NxHpZQVoU6dFpuDtE  
10uxGcCqGJ7mAmo6H/YawSNp2Ns80gyqIKYo7o3LJ+WRroIR1Qyctq8gnR9JvYXX  
42ASqLq5+0XKo4qh81blXKYqtc176BbbSNFjWnzIQgKDgNiHFZCdc0VgqDhw015r  
NICbqqwwNLj/Fr2kecYx180Ktp10j00w5I0yh3bf3MVGwnYRdjvA1v+/CO+55N4g  
z0kf50Lcdu5RtqV10XBCifn28pecqPaSdYcssYSR15DLiFktGbNzTgCZZwITTKQc  
af8PPdTGtnnb6P+cdbW3bt9MVtN5/dgSHLThnS8MPEuNCtkTnpXshuVuBGgwBMdb  
qUC+HjqvhZzbwns8dr5WI+6HWNBFgGANN6ageY158vVp0UkuNP8wcWjRARciHXZx
```

```
ku6W2jPTHDWGNrBQ02Fx7fd2QYJheIPPASHHcfJ0+xgWCof45D0vAxAJ8gGg9Eq+
gFWhsx4NSHn2gh1gDZ410u/4exJ1lwPM
=uVaX
-----END PGP PUBLIC KEY BLOCK-----
EOF
```

2. Tentukan apakah akan mempercayai OpenPGP kuncinya. Beberapa faktor yang perlu dipertimbangkan ketika memutuskan apakah akan mempercayai kunci di atas termasuk yang berikut:
  - Koneksi internet yang Anda gunakan untuk mendapatkan GPG kunci dari situs web ini aman.
  - Perangkat tempat Anda mengakses situs web ini aman.
  - AWS telah mengambil langkah-langkah untuk mengamankan hosting kunci OpenPGP publik di situs web ini.
3. Jika Anda memutuskan untuk mempercayai OpenPGP kunci, edit kunci untuk mempercayai dengan gpg mirip dengan contoh berikut:

```
$ gpg --edit-key 0xB840C08C29A90796A071FAA5F6CD3CE6B76F3CEF

gpg (GnuPG) 2.0.22; Copyright (C) 2013 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

pub 4096R/4BF0B8D2 created: 2023-06-23 expires: 2025-06-22 usage: SCEA
trust: unknown validity: unknown
[ unknown] (1). AWS Deadline Cloud example@example.com

gpg> trust
pub 4096R/4BF0B8D2 created: 2023-06-23 expires: 2025-06-22 usage: SCEA
trust: unknown validity: unknown
[ unknown] (1). AWS Deadline Cloud aws-deadline@amazon.com

Please decide how far you trust this user to correctly verify other users'
keys
(by looking at passports, checking fingerprints from different sources,
etc.)

1 = I don't know or won't say
2 = I do NOT trust
```

```
3 = I trust marginally
4 = I trust fully
5 = I trust ultimately
m = back to the main menu

Your decision? 5
Do you really want to set this key to ultimate trust? (y/N) y

pub 4096R/4BF0B8D2 created: 2023-06-23 expires: 2025-06-22 usage: SCEA
trust: ultimate validity: unknown
[ unknown] (1). AWS Deadline Cloud aws-deadline@amazon.com
Please note that the shown key validity is not necessarily correct
unless you restart the program.

gpg> quit
```

#### 4. Verifikasi Deadline Penginstal pengirim Cloud

Untuk memverifikasi installer submitter Deadline Cloud, selesaikan langkah-langkah berikut:

- a. Kembali ke halaman Unduhan [konsol](#) Cloud Deadline dan unduh file tanda tangan untuk penginstal pengirim Deadline Cloud.
- b. Verifikasi tanda tangan penginstal submitter Deadline Cloud dengan menjalankan:

```
gpg --verify ./DeadlineCloudSubmitter-linux-x64-
installer.run.sig ./DeadlineCloudSubmitter-linux-x64-
installer.run
```

#### 5. Verifikasi Batas Waktu Monitor Cloud

##### Note

Anda dapat memverifikasi unduhan monitor Deadline Cloud menggunakan file tanda tangan atau metode khusus platform. Untuk metode khusus platform, lihat Linux (DEB) tab atau Linux (Applmage) tab berdasarkan jenis file yang Anda unduh.

Untuk memverifikasi aplikasi desktop monitor Deadline Cloud dengan file tanda tangan, selesaikan langkah-langkah berikut:

- a. Kembali ke halaman Unduhan [konsol](#) Cloud Deadline dan unduh file.sig yang sesuai, lalu jalankan

Untuk .deb:

```
gpg --verify ./deadline-cloud-  
monitor_<APP_VERSION>_amd64.deb.sig ./deadline-cloud-  
monitor_<APP_VERSION>_amd64.deb
```

Untuk. AppImage:

```
gpg --verify ./deadline-cloud-  
monitor_<APP_VERSION>_amd64.AppImage.sig ./deadline-cloud-  
monitor_<APP_VERSION>_amd64.AppImage
```

- b. Konfirmasikan bahwa output terlihat mirip dengan yang berikut:

```
gpg: Signature made Mon Apr 1 21:10:14 2024 UTC
```

```
gpg: using RSA key B840C08C29A90796A071FAA5F6CD3CE6B7
```

Jika output berisi frasa `Good signature from "AWS Deadline Cloud"`, itu berarti tanda tangan telah berhasil diverifikasi dan Anda dapat menjalankan skrip instalasi monitor Deadline Cloud.

## Linux (DEB)

Untuk memverifikasi paket yang menggunakan Linux .deb biner, pertama selesaikan langkah 1-3 di Linux tab.

`dpkg` adalah alat manajemen paket inti di sebagian besar debian berdasarkan Linux distribusi. Anda dapat memverifikasi file.deb dengan alat ini.

1. Dari halaman Unduhan [Konsol](#) Cloud Deadline, unduh file Deadline Cloud monitor .deb.
2. Ganti `<APP_VERSION>` dengan versi file.deb yang ingin Anda verifikasi.

```
dpkg-sig --verify deadline-cloud-monitor_<APP_VERSION>_amd64.deb
```

3. Outputnya akan mirip dengan:

```
Processing deadline-cloud-monitor_<APP_VERSION>_amd64.deb... GOODSIG  
_gpgbuilder B840C08C29A90796A071FAA5F6CD3C 171200
```

4. Untuk memverifikasi file.deb, konfirmasikan bahwa GOODSIG ada dalam output.

## Linux (AppImage)

Untuk memverifikasi paket yang menggunakan Linux . AppImage biner, langkah lengkap pertama 1-3 di Linux tab, lalu selesaikan langkah-langkah berikut.

1. Dari AppImageUpdate [halaman](#) di GitHub, unduh validate-x86\_64. AppImageberkas.
2. Setelah mengunduh file, untuk menambahkan izin eksekusi, jalankan perintah berikut.

```
chmod a+x ./validate-x86_64.AppImage
```

3. Untuk menambahkan izin eksekusi, jalankan perintah berikut.

```
chmod a+x ./deadline-cloud-monitor_<APP_VERSION>_amd64.AppImage
```

4. Untuk memverifikasi tanda tangan monitor Deadline Cloud, jalankan perintah berikut.

```
./validate-x86_64.AppImage ./deadline-cloud-monitor_<APP_VERSION>_amd64.AppImage
```

Jika output berisi frasa `Validation successful`, itu berarti tanda tangan telah berhasil diverifikasi dan Anda dapat menjalankan skrip instalasi monitor Deadline Cloud dengan aman.

## Instal pengirim Cloud Deadline

Anda dapat menginstal submitter Deadline Cloud dengan Windows atau Linux. Dengan installer, Anda dapat menginstal pengirim berikut:

- Maya 2024
- Nuklir 14.0 - 15.0
- Houdini 19.5
- Tombol 12
- Blender 3.6
- Mesin Tidak Nyata 5

Anda dapat menginstal pengirim lain yang tidak tercantum di sini. Kami menggunakan pustaka Deadline Cloud untuk membangun submitter. Beberapa pengirim termasuk C4D, After Effects, 3ds Max, dan Rhino. [Anda dapat menemukan kode sumber untuk pustaka dan pengirim ini di organisasi aws-deadline. GitHub](#)

## Windows

1. Di browser file, navigasikan ke folder tempat penginstal diunduh, lalu pilih `DeadlineCloudSubmitter-windows-x64-installer.exe`.
  - a. Jika Windows melindungi tampilan pop-up PC Anda, pilih Info lebih lanjut.
  - b. Pilih Run pula.
2. Setelah AWS Deadline Cloud Submitter Setup Wizard terbuka, pilih Berikutnya.
3. Pilih ruang lingkup instalasi dengan menyelesaikan salah satu langkah berikut:
  - Untuk menginstal hanya untuk pengguna saat ini, pilih Pengguna.
  - Untuk menginstal untuk semua pengguna, pilih Sistem.

Jika Anda memilih Sistem, Anda harus keluar dari penginstal dan menjalankannya kembali sebagai administrator dengan menyelesaikan langkah-langkah berikut:

- a. Klik kanan pada **DeadlineCloudSubmitter-windows-x64-installer.exe**, dan kemudian pilih Run as administrator.
  - b. Masukkan kredensi administrator Anda, lalu pilih Ya.
  - c. Pilih Sistem untuk ruang lingkup instalasi.
4. Setelah memilih ruang lingkup instalasi, pilih Berikutnya.
  5. Pilih Berikutnya lagi untuk menerima direktori instalasi.
  6. Pilih Integrated submitter untuk Nuke, atau pengirim mana pun yang ingin Anda instal.
  7. Pilih Berikutnya.
  8. Tinjau instalasi, dan pilih Berikutnya.
  9. Pilih Berikutnya lagi, lalu pilih Selesai.

## Linux

### Note

The Deadline Cloud terintegrasi Nuke installer untuk Linux dan monitor Deadline Cloud hanya dapat diinstal pada Linux distribusi dengan setidaknya GLIBC 2,31.

1. Buka jendela terminal.

2. Untuk melakukan instalasi sistem installer, masukkan perintah **sudo -i** dan tekan Enter untuk menjadi root.
3. Arahkan ke lokasi tempat Anda mengunduh penginstal.

Misalnya, **cd /home/USER/Downloads**.

4. Untuk membuat installer dapat dieksekusi, masukkan. **chmod +x DeadlineCloudSubmitter-linux-x64-installer.run**
5. Untuk menjalankan installer submitter Deadline Cloud, masukkan. **./DeadlineCloudSubmitter-linux-x64-installer.run**
6. Ketika installer terbuka, ikuti petunjuk di layar Anda untuk menyelesaikan Setup Wizard.

## Langkah 2: Instal dan atur monitor Deadline Cloud

Anda dapat menginstal aplikasi desktop monitor Deadline Cloud dengan Windows atau Linux.

### Windows

1. Jika Anda belum melakukannya, masuk ke AWS Management Console dan buka [konsol](#) Deadline Cloud.
2. Dari panel navigasi kiri, pilih Unduhan.
3. Di bagian Monitor Deadline Cloud, pilih file untuk sistem operasi komputer Anda.
4. Untuk mengunduh monitor Deadline Cloud, pilih Unduh.

### Linux

Untuk menginstal monitor Deadline Cloud AppImage di RPM distro

1. Unduh monitor AppImage Deadline Cloud terbaru.
2. Untuk membuat AppImage executable, masukkan. **chmod a+x deadline-cloud-monitor\_<APP\_VERSION>\_amd64.AppImage**
3. Untuk mengatur jalur SSL sertifikat yang benar, masukkan **sudo ln -sf /etc/ssl/certs/ca-bundle.crt /etc/ssl/certs/ca-certificates.crt**.

Untuk menginstal monitor Deadline Cloud AppImage di distro Debian

1. Unduh monitor AppImage Deadline Cloud terbaru.

2.

**Note**

Langkah ini untuk Ubuntu 22 dan yang lebih tinggi. Untuk versi Ubuntu lainnya, lewati langkah ini.

Untuk menginstal libfuse2, masukkan **sudo apt update**

**sudo apt install libfuse2.**

3. Untuk membuat AppImage executable, masukkan. **chmod a+x deadline-cloud-monitor\_<APP\_VERSION>\_amd64.AppImage**

Untuk menginstal Deadline Cloud monitor paket Debian pada distro Debian

1. Unduh paket Debian monitor Deadline Cloud terbaru.

2.

**Note**

Langkah ini untuk Ubuntu 22 dan yang lebih tinggi. Untuk versi Ubuntu lainnya, lewati langkah ini.

Untuk menginstal libssl1.1, masukkan **wget http://nz2.archive.ubuntu.com/ubuntu/pool/main/o/openssl/libssl1.<APP\_VERSION>.1f-1ubuntu2.22\_amd64.deb**

**sudo dpkg -i libssl1.<APP\_VERSION>.1f-1ubuntu2.22\_amd64.deb.**

3. Untuk menginstal paket Debian monitor Deadline Cloud, masukkan **sudo apt update**

**sudo apt install ./deadline-cloud-monitor\_<APP\_VERSION>\_amd64.deb.**

4. Jika instalasi gagal pada paket yang memiliki dependensi yang tidak terpenuhi, perbaiki paket yang rusak dan kemudian jalankan perintah berikut.

**sudo apt --fix-missing update**

**sudo apt update**

**sudo apt install -f**

Setelah Anda menyelesaikan unduhan, Anda dapat memverifikasi keaslian perangkat lunak yang diunduh. Lihat Verifikasi keaslian perangkat lunak yang diunduh di Langkah 1.

Setelah mengunduh monitor Deadline Cloud dan memverifikasi keasliannya, gunakan prosedur berikut untuk mengatur monitor Deadline Cloud.

Untuk mengatur monitor Cloud Deadline

1. Buka Monitor Cloud Deadline.
2. Saat diminta untuk membuat profil baru, selesaikan langkah-langkah berikut.
  - a. Masukkan monitor Anda URL ke URL input, yang terlihat seperti **https://MY-MONITOR.deadlinecloud.amazonaws.com/**
  - b. Masukkan nama Profil.
  - c. Pilih Buat Profil.

Profil Anda dibuat dan kredensial Anda sekarang dibagikan dengan perangkat lunak apa pun yang menggunakan nama profil yang Anda buat.

3. Setelah membuat profil monitor Deadline Cloud, Anda tidak dapat mengubah nama profil atau studioURL. Jika Anda perlu melakukan perubahan, lakukan hal berikut sebagai gantinya:
  - a. Hapus profil. Di panel navigasi kiri, pilih Monitor Deadline Cloud, > Settings, > Delete.
  - b. Buat profil baru dengan perubahan yang Anda inginkan.
4. Dari panel navigasi kiri, gunakan opsi >Deadline Cloud monitor untuk melakukan hal berikut:
  - Ubah profil monitor Deadline Cloud untuk masuk ke monitor lain.
  - Aktifkan Autologin sehingga Anda tidak perlu memasukkan monitor Anda URL pada monitor Deadline Cloud berikutnya.
5. Tutup jendela monitor Deadline Cloud. Ini terus berjalan di latar belakang dan menyinkronkan kredensialmu setiap 15 menit.
6. Untuk setiap aplikasi pembuatan konten digital (DCC) yang Anda rencanakan untuk digunakan untuk proyek rendering Anda, selesaikan langkah-langkah berikut:
  - a. Dari submitter Deadline Cloud Anda, buka konfigurasi workstation Deadline Cloud.
  - b. Dalam konfigurasi workstation, pilih profil yang Anda buat di monitor Deadline Cloud. Kredensial Deadline Cloud Anda sekarang dibagikan dengan ini DCC dan alat Anda harus berfungsi seperti yang diharapkan.

## Langkah 3: Luncurkan submitter Deadline Cloud

Bagian berikut memandu Anda melalui langkah-langkah untuk meluncurkan plugin pengirim Deadline Cloud di Blender, Nuke, Maya, Houdini, KeyShot, dan Unreal Engine.

Untuk meluncurkan submitter Deadline Cloud di Blender

### Note

Support untuk Blender disediakan dengan menggunakan Conda lingkungan untuk armada yang dikelola layanan. Untuk informasi selengkapnya, lihat [Default Conda lingkungan antrian](#).

1. Buka Blender.
2. Pilih Edit, lalu Preferensi. Di bawah Jalur File pilih Direktori Skrip, lalu pilih Tambah. Tambahkan direktori skrip untuk folder python tempat pengirim Blender diinstal:

```
Windows :
  %USERPROFILE%\DeadlineCloudSubmitter\Submitters\Blender\python\
Linux :
  ~/DeadlineCloudSubmitter/Submitters/Blender/python/
```

3. Mulai ulang Blender.
4. Pilih Edit, lalu Preferensi. Selanjutnya, pilih Add-on, lalu cari Deadline Cloud for Blender. Pilih kotak centang untuk mengaktifkan add-on.
5. Buka Blender adegan dengan dependensi yang ada di dalam direktori root aset.
6. Di menu Render, pilih dialog Deadline Cloud.
  - a. Jika Anda belum diautentikasi di pengirim Deadline Cloud, Status Kredensial ditampilkan sebagai `_ . NEEDS LOGIN`
  - b. Pilih Login.
  - c. Jendela browser login ditampilkan. Masuk dengan kredensi pengguna Anda.
  - d. Pilih Izinkan. Anda sekarang masuk dan Status Kredensial ditampilkan sebagai `AUTHENTICATED`
7. Pilih Kirim.

## Untuk meluncurkan submitter Deadline Cloud di Foundry Nuke

### Note

Support untuk Nuke disediakan dengan menggunakan Conda lingkungan untuk armada yang dikelola layanan. Untuk informasi selengkapnya, lihat [Default Conda lingkungan antrian](#).

1. Buka Nuke.
2. Buka Nuke script dengan dependensi yang ada dalam direktori root aset.
3. Pilih AWS Deadline, lalu pilih Kirim ke Deadline Cloud untuk meluncurkan submitter.
  - a. Jika Anda belum diautentikasi di pengirim Deadline Cloud, Status Kredensial ditampilkan sebagai `_ . NEEDS LOGIN`
  - b. Pilih Login.
  - c. Di jendela browser login, masuk dengan kredensi pengguna Anda.
  - d. Pilih Izinkan. Anda sekarang masuk dan Status Kredensial ditampilkan sebagai `AUTHENTICATED`
4. Pilih Kirim.

## Untuk meluncurkan submitter Deadline Cloud di Maya

### Note

Support untuk Maya and Arnold for Maya(MtoA) disediakan dengan menggunakan Conda lingkungan untuk armada yang dikelola layanan. Untuk informasi selengkapnya, lihat [Default Conda lingkungan antrian](#).

1. Buka Maya.
2. Tetapkan proyek Anda, dan buka file yang ada di dalam direktori root aset.
3. Pilih Windows → Pengaturan/Preferensi → Pengelola Plugin.
4. Cari DeadlineCloudSubmitter.
5. Untuk memuat plugin pengirim Deadline Cloud, pilih Loaded.

- a. Jika Anda belum diautentikasi di pengirim Deadline Cloud, Status Kredensial ditampilkan sebagai `_`. NEEDS LOGIN
  - b. Pilih Login.
  - c. Jendela browser login ditampilkan. Masuk dengan kredensi pengguna Anda.
  - d. Pilih Izinkan. Anda sekarang masuk dan Status Kredensial ditampilkan sebagai `AUTHENTICATED`
6. (Opsional) Untuk memuat plugin pengirim Deadline Cloud setiap kali Anda membuka Maya, pilih Muat otomatis.
  7. Pilih rak Deadline Cloud, lalu pilih tombol hijau untuk meluncurkan submitter.

Untuk meluncurkan submitter Deadline Cloud di Houdini

 Note

Support untuk Houdini disediakan dengan menggunakan Conda lingkungan untuk armada yang dikelola layanan. Untuk informasi selengkapnya, lihat [Default Conda lingkungan antrian](#).

1. Buka Houdini.
2. Di Network Editor, pilih jaringan /out.
3. Tekan tab, dan masukkan **deadline**.
4. Pilih opsi Deadline Cloud, dan sambungkan ke jaringan yang ada.
5. Klik dua kali node Deadline Cloud.

Untuk meluncurkan submitter Deadline Cloud di KeyShot

1. Terbuka KeyShot.
2. Pilih Windows, > Konsol skrip, > Kirim ke AWS Deadline Cloud, dan Jalankan.

Untuk meluncurkan submitter Deadline Cloud di Unreal Engine

Ini mengasumsikan Anda telah mengunduh Deadline Cloud.

1. Buat atau buka folder yang Anda gunakan untuk Unreal Engine proyek.

2. Buka baris perintah dan jalankan perintah berikut:
  - **git clone https://github.com/aws-deadline/deadline-cloud-for-unreal-engine**
  - **cd deadline-cloud-for-unreal/test\_projects**
  - **git lfs fetch -all**
3. Untuk men-download plugin untuk Unreal Engine, buka Unreal Engine folder proyek, dan luncurkan `deadline-cloud-forunreal/test_projects/pull_ue_plugin.bat`.

Ini menempatkan file plugin di `C://LocalProjectsUnrealDeadlineCloudTest/Plugins/UnrealDeadlineCloudService`.

4. Untuk mengunduh submitter, buka `UnrealDeadlineCloudService` folder, dan jalankan **deadline-cloud-forunreal/ test\_projects/Plugins/ UnrealDeadlineCloudService/install\_unreal\_submitter.bat**
5. Untuk meluncurkan submitter dari Unreal Engine, selesaikan langkah-langkah berikut:
  - a. Pilih Edit, > Pengaturan proyek.
  - b. Di bilah pencarian, masukkan **movie render pipeline**.
  - c. Sesuaikan pengaturan Pipeline Render Film berikut:
    - i. Untuk Pelaksana Jarak Jauh Default, masukkan **MoviePipelineDeadlineCloudRemote Executor**.
    - ii. Untuk Pekerjaan Pelaksana Default, masukkan **MoviePipelineDeadlineCloudExecutorJob**
    - iii. Untuk Default Job Settings Classes, pilih tanda plus, lalu masukkan **DeadlineCloudRenderStepSetting**.

Dengan pengaturan ini, Anda dapat memilih plugin Deadline Cloud dari Unreal Engine.

## Gunakan peternakan

Jika Anda telah mengikuti semua instruksi memulai, Anda telah menyiapkan semua yang Anda butuhkan untuk mulai mengirimkan pekerjaan dari workstation lokal Anda ke peternakan Anda, dan kemudian memantau pekerjaan dan sumber daya tersebut. Untuk informasi lebih lanjut tentang mengirimkan semua jenis pekerjaan atau pemantauan, lihat topik terkait di bawah ini.

- [Lowongan](#)
- [Menggunakan monitor](#)

# Menggunakan monitor Deadline Cloud

Monitor AWS Deadline Cloud memberi Anda tampilan keseluruhan pekerjaan komputasi visual Anda. Anda dapat menggunakannya untuk memantau dan mengelola pekerjaan, melihat aktivitas pekerja di armada, melacak anggaran dan penggunaan, dan untuk mengunduh hasil pekerjaan.

Setiap antrian memiliki monitor pekerjaan yang menunjukkan status pekerjaan, langkah, dan tugas. Monitor menyediakan cara untuk mengelola pekerjaan langsung dari monitor. Anda dapat membuat perubahan prioritas, membatalkan pekerjaan, dan meminta pekerjaan.

Monitor Deadline Cloud memiliki tabel yang menunjukkan status ringkasan untuk suatu pekerjaan, atau Anda dapat memilih pekerjaan untuk melihat log tugas terperinci yang membantu memecahkan masalah dengan pekerjaan.

Anda dapat menggunakan monitor Deadline Cloud untuk mengunduh hasil ke lokasi di workstation Anda yang ditentukan saat pekerjaan dibuat.

Monitor Deadline Cloud juga membantu Anda memantau penggunaan dan mengelola biaya. Untuk informasi selengkapnya, lihat [Mengelola anggaran dan penggunaan untuk Deadline Cloud](#).

## Topik

- [Bagikan URL monitor Cloud Deadline](#)
- [Buka monitor Deadline Cloud](#)
- [Lihat detail antrian dan armada di Deadline Cloud](#)
- [Melihat dan mengelola pekerjaan, langkah, dan tugas di Deadline Cloud](#)
- [Lihat detail pekerjaan di Deadline Cloud](#)
- [Lihat langkah di Deadline Cloud](#)
- [Melihat tugas di Deadline Cloud](#)
- [Lihat log di Deadline Cloud](#)
- [Unduh hasil jadi di Deadline Cloud](#)

## Bagikan URL monitor Cloud Deadline

Saat menyiapkan layanan Deadline Cloud, secara default Anda membuat URL yang membuka monitor Deadline Cloud untuk akun Anda. Gunakan URL ini untuk membuka monitor di browser

Anda atau di desktop Anda. Bagikan URL dengan pengguna lain sehingga mereka dapat mengakses monitor Deadline Cloud.

Sebelum pengguna dapat membuka monitor Deadline Cloud, Anda harus memberikan akses kepada pengguna. Untuk memberikan akses, tambahkan pengguna ke daftar pengguna yang berwenang untuk monitor atau tambahkan mereka ke grup dengan akses ke monitor. Untuk informasi selengkapnya, lihat [Mengelola pengguna di Deadline Cloud](#).

Untuk berbagi URL monitor

1. Buka [konsol Deadline Cloud](#).
2. Dari Mulai, pilih Go to Deadline Cloud dashboard.
3. Di panel navigasi, pilih Dashboard (Dasbor).
4. Di bagian Ikhtisar akun, pilih Detail akun.
5. Salin dan kemudian kirim URL dengan aman ke siapa saja yang perlu mengakses monitor Deadline Cloud.

## Buka monitor Deadline Cloud

Anda dapat membuka monitor Deadline Cloud dengan salah satu cara berikut:

- Konsol — Masuk ke AWS Management Console dan buka konsol Deadline Cloud.
- Web — Buka URL monitor yang Anda buat saat menyiapkan Deadline Cloud.
- Monitor — Gunakan monitor Cloud Deadline desktop.

Saat Anda menggunakan konsol, Anda harus dapat masuk AWS menggunakan AWS Identity and Access Management identitas, lalu masuk ke monitor dengan AWS IAM Identity Center kredensi. Jika Anda hanya memiliki kredensial Pusat Identitas IAM, Anda harus masuk menggunakan URL monitor atau aplikasi desktop.

Untuk membuka monitor Deadline Cloud (web)

1. Menggunakan browser, buka URL monitor yang Anda buat saat menyiapkan Deadline Cloud.
2. Masuk dengan kredensi pengguna Anda.

## Untuk membuka monitor Deadline Cloud (konsol)

1. Buka [konsol Deadline Cloud](#).
2. Di panel navigasi, pilih Peternakan.
3. Pilih peternakan, lalu pilih Kelola pekerjaan untuk membuka halaman monitor Deadline Cloud.
4. Masuk dengan kredensi pengguna Anda.

## Untuk membuka monitor Deadline Cloud (desktop)

1. Buka [konsol Deadline Cloud](#).

-atau-

Buka monitor Deadline Cloud - web dari URL monitor.

2. • Pada konsol Deadline Cloud, lakukan hal berikut:
  1. Di monitor, pilih Buka dasbor Deadline Cloud, lalu pilih Unduhan dari menu sebelah kiri.
  2. Dari monitor Deadline Cloud, pilih versi monitor untuk desktop Anda.
  3. Pilih Unduh.
- Pada monitor Deadline Cloud - web, lakukan hal berikut:
  - Dari menu sebelah kiri, pilih Pengaturan Workstation. Jika item pengaturan Workstation tidak terlihat, gunakan panah untuk membuka menu kiri.
  - Pilih Unduh.
  - Dari Pilih OS, pilih sistem operasi Anda.
3. Unduh monitor Cloud Deadline - desktop.
4. Setelah Anda mengunduh dan menginstal monitor, buka di komputer Anda.
  - Jika ini adalah pertama kalinya Anda membuka monitor Deadline Cloud, Anda harus memberikan URL monitor dan membuat nama profil. Selanjutnya Anda masuk ke monitor dengan kredensi Deadline Cloud Anda.
  - Setelah Anda membuat profil, Anda membuka monitor dengan memilih profil. Anda mungkin perlu memasukkan kredensi Deadline Cloud Anda.

## Lihat detail antrian dan armada di Deadline Cloud

Anda dapat menggunakan monitor Deadline Cloud untuk melihat konfigurasi antrian dan armada di peternakan Anda. Anda juga dapat menggunakan monitor untuk melihat daftar pekerjaan dalam antrian atau pekerja dalam armada.

Anda harus memiliki VIEWING izin untuk melihat detail antrian dan armada. Jika detail tidak ditampilkan, hubungi administrator Anda untuk mendapatkan izin yang benar.

Untuk melihat detail antrian

1. [Buka monitor Deadline Cloud.](#)
2. Dari daftar peternakan, pilih peternakan yang berisi antrian yang Anda minati.
3. Dalam daftar antrian, pilih antrian untuk menampilkan detailnya. Untuk membandingkan konfigurasi dua antrian atau lebih, pilih lebih dari satu kotak centang.
4. Untuk melihat daftar pekerjaan dalam antrian, pilih nama antrian dari daftar antrian atau dari panel detail.

Jika monitor sudah terbuka, Anda dapat memilih antrian dari daftar Antrian di panel navigasi kiri.

Untuk melihat detail armada

1. [Buka monitor Deadline Cloud.](#)
2. Dari daftar peternakan, pilih peternakan yang berisi armada yang Anda minati.
3. Di sumber daya Pertanian, pilih Armada.
4. Dalam daftar armada, pilih armada untuk menampilkan detailnya. Untuk membandingkan konfigurasi dua armada atau lebih, pilih lebih dari satu kotak centang.
5. Untuk melihat daftar pekerja di armada, pilih nama armada dari daftar armada atau dari panel detail.

Jika monitor sudah terbuka, Anda dapat memilih armada dari daftar Armada di panel navigasi kiri.

# Melihat dan mengelola pekerjaan, langkah, dan tugas di Deadline Cloud

Saat Anda memilih antrian, bagian monitor pekerjaan pada monitor Deadline Cloud menunjukkan pekerjaan dalam antrian tersebut, langkah-langkah dalam pekerjaan, dan tugas di setiap langkah. Ketika Anda memilih pekerjaan, langkah, atau tugas, Anda dapat menggunakan menu Tindakan untuk mengelola masing-masing.

Untuk membuka monitor pekerjaan, ikuti langkah-langkah untuk melihat antrian [Lihat detail antrian dan armada di Deadline Cloud](#), lalu pilih pekerjaan, langkah, atau tugas yang akan dikerjakan.

Untuk pekerjaan, langkah, dan tugas, Anda dapat melakukan hal berikut:

- Ubah status menjadi Requeued, Succeeded, Failed, atau Canceled.
- Unduh output yang diproses dari pekerjaan, langkah, atau tugas.
- Salin ID pekerjaan, langkah, atau tugas.

Untuk pekerjaan yang dipilih, Anda dapat:

- Arsipkan pekerjaan.
- Ubah properti pekerjaan, seperti mengubah prioritas atau melihat dependensi langkah ke langkah.
- Lihat detail tambahan menggunakan parameter pekerjaan.

Untuk informasi lebih lanjut, lihat [Lihat detail pekerjaan di Deadline Cloud](#).

Untuk setiap langkah, Anda dapat:

- Lihat dependensi untuk langkah tersebut. Dependensi untuk langkah harus diselesaikan sebelum langkah berjalan.

Lihat rinciannya di [Lihat langkah di Deadline Cloud](#).

Untuk setiap tugas, Anda dapat:

- Lihat log untuk tugas tersebut.
- Lihat parameter tugas.

Untuk informasi selengkapnya, lihat [Melihat tugas di Deadline Cloud](#).

## Arsipkan pekerjaan

Untuk mengarsipkan pekerjaan, itu harus dalam keadaan terminal, FAILED, SUCCEEDED, SUSPENDED, atau CANCELED. ARCHIVED Negara adalah final. Setelah pekerjaan diarsipkan, pekerjaan tidak dapat diulang atau dimodifikasi.

Data pekerjaan tidak terpengaruh oleh pengarsipan pekerjaan. Data dihapus ketika batas waktu tidak aktif tercapai, atau ketika antrian yang berisi pekerjaan dihapus.

Hal-hal lain yang terjadi pada pekerjaan yang diarsipkan:

- Pekerjaan yang diarsipkan disembunyikan di monitor Deadline Cloud.
- Pekerjaan yang diarsipkan terlihat dalam status hanya-baca dari Deadline Cloud CLI selama 120 hari sebelum penghapusan.

## Meminta pekerjaan

Saat Anda meminta ulang pekerjaan, semua tugas tanpa dependensi langkah beralih ke. READY Status langkah-langkah dengan dependensi beralih ke READY atau PENDING saat dipulihkan.

- Semua pekerjaan, langkah, dan tugas beralih ke PENDING.
- Jika sebuah langkah tidak memiliki ketergantungan, itu beralih ke READY.

## Lihat detail pekerjaan di Deadline Cloud

Halaman monitor Job di monitor Deadline Cloud memberi Anda hal-hal berikut:

- Pandangan keseluruhan tentang kemajuan suatu pekerjaan.
- Pandangan tentang langkah-langkah dan tugas yang membentuk pekerjaan.

Pilih pekerjaan dari daftar untuk melihat daftar langkah untuk pekerjaan itu, lalu pilih langkah dari daftar langkah untuk melihat tugas untuk pekerjaan itu. Setelah memilih item, Anda dapat menggunakan menu Tindakan untuk item tersebut untuk melihat detail.

## Untuk melihat detail pekerjaan

1. Ikuti langkah-langkah untuk melihat antrian di [Lihat detail antrian dan armada di Deadline Cloud](#).
2. Di panel navigasi, pilih antrian tempat Anda mengirimkan pekerjaan.
3. Pilih pekerjaan menggunakan salah satu metode berikut:
  - a. Dari daftar Pekerjaan, pilih pekerjaan untuk melihat detailnya.
  - b. Dari bidang pencarian, masukkan teks apa pun yang terkait dengan pekerjaan, seperti nama pekerjaan atau pengguna yang membuat pekerjaan. Dari hasil yang ditampilkan, pilih pekerjaan yang ingin Anda lihat.

Rincian pekerjaan mencakup langkah-langkah dalam pekerjaan dan tugas di setiap langkah. Anda dapat menggunakan menu Tindakan untuk melakukan hal berikut:

- Ubah status pekerjaan.
- Melihat dan memodifikasi properti pekerjaan. Anda dapat melihat dependensi di antara langkah-langkah dalam pekerjaan, dan mengubah prioritas pekerjaan. Umumnya, pekerjaan dengan prioritas lebih tinggi selesai lebih cepat.
- Lihat parameter untuk pekerjaan yang ditetapkan saat pekerjaan dikirimkan.
- Unduh output dari suatu pekerjaan. Ketika Anda men-download output dari pekerjaan, itu berisi semua output yang dihasilkan oleh langkah-langkah dan tugas dalam pekerjaan.

## Lihat langkah di Deadline Cloud

Gunakan monitor AWS Deadline Cloud untuk melihat langkah-langkah dalam pekerjaan pemrosesan Anda. Di monitor Job, daftar Langkah menunjukkan daftar langkah yang membentuk pekerjaan yang dipilih. Saat Anda memilih langkah, daftar Tugas menunjukkan tugas di langkah tersebut.

### Untuk melihat langkah

1. Ikuti langkah-langkah [Lihat detail pekerjaan di Deadline Cloud](#) untuk melihat daftar pekerjaan.
2. Pilih sebuah tugas dari daftar Tugas.
3. Pilih langkah dari daftar Langkah.

Anda dapat menggunakan menu Tindakan untuk melakukan hal berikut:

- Ubah status langkah.
- Unduh output dari langkah tersebut. Saat Anda mengunduh output dari sebuah langkah, itu berisi semua output yang dihasilkan oleh tugas di langkah tersebut.
- Lihat dependensi dari sebuah langkah. Tabel dependensi menunjukkan daftar langkah yang harus diselesaikan sebelum langkah yang dipilih dimulai, dan daftar langkah yang menunggu langkah ini selesai.

## Melihat tugas di Deadline Cloud

Gunakan monitor AWS Deadline Cloud untuk melihat tugas dalam pekerjaan pemrosesan Anda. Di monitor Job, daftar Tugas menampilkan tugas yang membentuk langkah yang dipilih dalam daftar Langkah.

Untuk melihat tugas

1. Ikuti langkah-langkah [Lihat detail pekerjaan di Deadline Cloud](#) untuk melihat daftar pekerjaan.
2. Pilih sebuah tugas dari daftar Tugas.
3. Pilih langkah dari daftar Langkah.
4. Pilih tugas dari daftar Tugas.

Anda dapat menggunakan menu Tindakan untuk melakukan hal berikut:

- Ubah status tugas.
- Lihat log tugas. Untuk informasi selengkapnya, lihat [Lihat log di Deadline Cloud](#).
- Lihat parameter yang ditetapkan saat tugas dibuat.
- Unduh output tugas. Saat Anda mengunduh output tugas, itu hanya berisi output yang dihasilkan oleh tugas yang dipilih.

## Lihat log di Deadline Cloud

Log memberi Anda informasi terperinci tentang status dan pemrosesan tugas. Di monitor AWS Deadline Cloud, Anda dapat melihat dua jenis log berikut:

- Log sesi merinci garis waktu tindakan, termasuk:
  - Tindakan pengaturan, seperti sinkronisasi lampiran dan memuat lingkungan perangkat lunak

- Menjalankan tugas atau serangkaian tugas
- Tindakan penutupan, seperti mematikan lingkungan pada pekerja

Sesi mencakup pemrosesan setidaknya satu tugas, dan dapat mencakup banyak tugas. Log sesi juga menampilkan informasi tentang jenis instans, vCPU, dan memori Amazon Elastic Compute Cloud (Amazon EC2). Log sesi juga menyertakan tautan ke log untuk pekerja yang digunakan dalam sesi.

- Log pekerja memberikan detail untuk timeline tindakan yang diproses pekerja selama siklus hidupnya. Log pekerja dapat berisi informasi tentang beberapa sesi.

Anda dapat mengunduh log sesi dan pekerja sehingga Anda dapat memeriksanya secara offline.

Untuk melihat log sesi

1. Ikuti langkah-langkah [Lihat detail pekerjaan di Deadline Cloud](#) untuk melihat daftar pekerjaan.
2. Pilih sebuah tugas dari daftar Tugas.
3. Pilih langkah dari daftar Langkah.
4. Pilih tugas dari daftar Tugas.
5. Dari menu Tindakan, pilih Lihat log.

Bagian Garis Waktu menunjukkan ringkasan tindakan untuk tugas tersebut. Untuk melihat lebih banyak tugas yang dijalankan dalam sesi dan untuk melihat tindakan shutdown untuk sesi, pilih Lihat log untuk semua tugas.

Untuk melihat log pekerja dari tugas

1. Ikuti langkah-langkah [Lihat detail pekerjaan di Deadline Cloud](#) untuk melihat daftar pekerjaan.
2. Pilih sebuah tugas dari daftar Tugas.
3. Pilih langkah dari daftar Langkah.
4. Pilih tugas dari daftar Tugas.
5. Dari menu Tindakan, pilih Lihat log.
6. Pilih Info sesi.
7. Pilih Lihat log pekerja.

Untuk melihat log pekerja dari detail armada

1. Ikuti langkah-langkah [Lihat detail antrian dan armada di Deadline Cloud](#) untuk melihat armada.
2. Pilih ID Pekerja dari daftar Pekerja.
3. Dari menu Tindakan, pilih Lihat log pekerja.

## Unduh hasil jadi di Deadline Cloud

Setelah pekerjaan selesai, Anda dapat menggunakan monitor AWS Deadline Cloud untuk mengunduh hasilnya ke workstation Anda. File output disimpan dengan nama dan lokasi yang Anda tentukan saat Anda membuat pekerjaan.

File output disimpan tanpa batas waktu. Untuk mengurangi biaya penyimpanan, pertimbangkan untuk membuat konfigurasi Siklus Hidup S3 untuk bucket Amazon S3 antrian Anda. Untuk informasi selengkapnya, lihat [Mengelola siklus hidup penyimpanan Anda](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Untuk mengunduh hasil akhir dari pekerjaan, langkah, atau tugas

1. Ikuti langkah-langkah [Lihat detail pekerjaan di Deadline Cloud](#) untuk melihat daftar pekerjaan.
2. Pilih pekerjaan, langkah, atau tugas yang ingin Anda unduh hasilnya.
  - Jika Anda memilih pekerjaan, Anda dapat mengunduh semua output untuk semua tugas di semua langkah untuk pekerjaan itu.
  - Jika Anda memilih langkah, Anda dapat mengunduh semua output untuk semua tugas di langkah itu.
  - Jika Anda memilih tugas, Anda dapat mengunduh output untuk tugas individual tersebut.
3. Dari menu Tindakan, pilih Unduh output.
4. Output akan diunduh ke lokasi yang ditetapkan saat pekerjaan dikirimkan.

### Note

Mengunduh output menggunakan menu saat ini hanya didukung untuk Windows dan Linux. Jika Anda memiliki Mac dan Anda memilih item menu keluaran Unduh, sebuah jendela menunjukkan AWS CLI perintah yang dapat Anda gunakan untuk mengunduh output yang dirender.

# Batas waktu Cloud farm

Peternakan adalah wadah untuk antrian yang mengelola pekerjaan dan armada sumber daya komputasi yang melakukan tugas.

Topik

- [Buat peternakan](#)
- [Hapus peternakan](#)
- [Edit peternakan](#)

## Buat peternakan

1. Dari [konsol Cloud Deadline](#), pilih Buka Dasbor.
2. Di bagian Farms di dasbor Deadline Cloud, pilih Actions → Create farm.
  - Atau, di panel sebelah kiri pilih Farms dan sumber daya lainnya, lalu pilih Create Farm.
3. Tambahkan Nama untuk peternakan Anda.
4. Untuk Deskripsi, masukkan deskripsi pertanian. Deskripsi yang jelas dapat membantu Anda mengidentifikasi tujuan pertanian Anda dengan cepat.
5. (Opsional) Secara default, data Anda dienkripsi dengan kunci yang AWS memiliki dan mengelola keamanan Anda. Anda dapat memilih Sesuaikan pengaturan enkripsi (lanjutan) untuk menggunakan kunci yang ada atau untuk membuat kunci baru yang Anda kelola.

Jika Anda memilih untuk menyesuaikan pengaturan enkripsi menggunakan kotak centang, masukkan AWS KMS ARN, atau buat yang AWS KMS baru dengan memilih Buat kunci KMS baru.

6. (Opsional) Pilih Tambahkan tag baru untuk menambahkan satu atau beberapa tag ke peternakan Anda.
7. Pilih Buat pertanian. Setelah pembuatan, pertanian Anda ditampilkan.

## Hapus peternakan

1. Dari dasbor Deadline Cloud, pilih Farms dan sumber daya lainnya.

2. Dalam daftar peternakan, pilih peternakan atau peternakan yang ingin Anda hapus, lalu pilih Hapus.

## Edit peternakan

1. Dari dasbor Deadline Cloud, pilih Farms dan sumber daya lainnya.
2. Dalam daftar peternakan, pilih peternakan atau peternakan yang ingin Anda hapus lalu pilih Edit.
3. Di jendela edit yang ditampilkan, ubah nama atau deskripsi peternakan, lalu pilih Simpan perubahan.

# Batas waktu antrian Cloud

Antrian adalah sumber daya pertanian yang mengelola dan memproses pekerjaan.

Untuk bekerja dengan antrian, Anda harus sudah memiliki monitor dan pertanian.

Topik

- [Membuat antrean](#)
- [Buat lingkungan antrian](#)
- [Hapus antrian](#)
- [Mengedit antrian](#)
- [Kaitkan antrian dan armada](#)

## Membuat antrean

1. Dari dasbor [konsol Deadline Cloud](#), pilih farm yang ingin Anda buat antrean.
  - Atau, di panel sisi kiri pilih Peternakan dan sumber daya lainnya, lalu pilih peternakan yang ingin Anda buat antrean.
2. Di tab Antrian, pilih Buat antrian.
3. Masukkan nama untuk antrian Anda.
4. Untuk Deskripsi, masukkan deskripsi antrian. Deskripsi membantu Anda mengidentifikasi tujuan antrian Anda.
5. Untuk lampiran Job, Anda dapat membuat bucket Amazon S3 baru atau memilih bucket Amazon S3 yang sudah ada.
  - a. Untuk membuat bucket Amazon S3 baru
    - i. Pilih Buat keranjang pekerjaan baru.
    - ii. Masukkan nama untuk ember. Kami merekomendasikan penamaan emberdeadlinecloud-job-attachments-[MONITORNAME].
    - iii. Masukkan awalan Root untuk menentukan atau mengubah lokasi root antrian Anda.
  - b. Untuk memilih bucket Amazon S3 yang ada
    - i. Pilih Pilih bucket S3 yang ada > Jelajahi S3.

- ii. Pilih bucket S3 untuk antrian Anda dari daftar bucket yang tersedia.
6. (Opsional) Untuk mengaitkan antrian Anda dengan armada yang dikelola pelanggan, pilih Aktifkan asosiasi dengan armada yang dikelola pelanggan.
7. Jika Anda mengaktifkan asosiasi dengan armada yang dikelola pelanggan, Anda harus menyelesaikan langkah-langkah berikut.

 Important

Kami sangat menyarankan untuk menentukan pengguna dan grup untuk fungsionalitas run-as. Jika tidak, itu akan menurunkan postur keamanan peternakan Anda karena pekerjaan kemudian dapat melakukan semua yang dapat dilakukan agen pekerja. Untuk informasi selengkapnya tentang potensi risiko keamanan, lihat [Menjalankan lowongan sebagai pengguna dan grup](#).

- a. Untuk Jalankan sebagai pengguna:

Untuk memberikan kredensi untuk pekerjaan antrian, pilih Pengguna yang dikonfigurasi antrian.

Atau, untuk memilih keluar dari pengaturan kredensial Anda sendiri dan menjalankan pekerjaan sebagai pengguna agen pekerja, pilih Pengguna agen pekerja.

- b. (Opsional) Untuk Run as user credentials, masukkan nama pengguna dan nama grup untuk memberikan kredensi untuk pekerjaan antrian.

Jika Anda menggunakan Windows armada, Anda harus membuat AWS Secrets Manager rahasia yang berisi kata sandi untuk Run as user. Jika Anda tidak memiliki rahasia yang ada dengan kata sandi, pilih Buat rahasia untuk membuka konsol Secrets Manager untuk membuat rahasia.

8. Membutuhkan anggaran membantu mengelola biaya untuk antrian Anda. Pilih salah satu Jangan memerlukan anggaran atau Memerlukan anggaran.
9. Antrian Anda memerlukan izin untuk mengakses Amazon S3 atas nama Anda. Anda dapat membuat peran layanan baru atau menggunakan peran layanan yang ada. Jika Anda tidak memiliki peran layanan yang ada, buat dan gunakan peran layanan baru.
  - a. Untuk menggunakan peran layanan yang ada, pilih Pilih peran layanan, lalu pilih peran dari menu tarik-turun.

- b. Untuk membuat peran layanan baru, pilih Buat dan gunakan peran layanan baru, lalu masukkan nama peran dan deskripsi.
10. (Opsional) Untuk menambahkan variabel lingkungan untuk lingkungan antrian, pilih Tambahkan variabel lingkungan baru, lalu masukkan nama dan nilai untuk setiap variabel yang Anda tambahkan.
11. (Opsional) Pilih Tambahkan tag baru untuk menambahkan satu atau beberapa tag ke antrian Anda.
12. Untuk membuat default Conda lingkungan antrian, tetap pilih kotak centang. Untuk mempelajari lebih lanjut tentang lingkungan antrian, lihat [Membuat lingkungan antrian](#). Jika Anda membuat antrian untuk armada yang dikelola pelanggan, kosongkan kotak centang.
13. Pilih Buat antrean.

## Buat lingkungan antrian

Lingkungan antrian adalah seperangkat variabel lingkungan dan perintah yang mengatur pekerja armada. Anda dapat menggunakan lingkungan antrian untuk menyediakan aplikasi perangkat lunak, variabel lingkungan, dan sumber daya lainnya untuk pekerjaan dalam antrian.

Saat Anda membuat antrian, Anda memiliki opsi untuk membuat default Conda lingkungan antrian. Lingkungan ini menyediakan akses armada yang dikelola layanan ke paket untuk DCC aplikasi mitra dan penyaji. Untuk informasi selengkapnya, lihat [Default Conda lingkungan antrian](#).

Anda dapat menambahkan lingkungan antrian menggunakan konsol, atau dengan mengedit json atau YAML template secara langsung. Prosedur ini menjelaskan cara membuat lingkungan dengan konsol.

1. Untuk menambahkan lingkungan antrian ke antrian, navigasikan ke antrian dan pilih tab Lingkungan antrian.
2. Pilih Tindakan, lalu Buat baru dengan formulir.
3. Masukkan nama dan deskripsi untuk lingkungan antrian.
4. Pilih Tambahkan variabel lingkungan baru, lalu masukkan nama dan nilai untuk setiap variabel yang Anda tambahkan.
5. (Opsional) Masukkan prioritas untuk lingkungan antrian. Prioritas menunjukkan urutan bahwa lingkungan antrian ini akan berjalan pada pekerja. Lingkungan antrian prioritas yang lebih tinggi akan berjalan terlebih dahulu.

## 6. Pilih Buat lingkungan antrian.

### Default Conda lingkungan antrian

Saat membuat antrian yang terkait dengan armada yang dikelola layanan, Anda memiliki opsi untuk menambahkan lingkungan antrian default yang mendukung [Conda](#) untuk mengunduh dan menginstal paket di lingkungan virtual untuk pekerjaan Anda.

Jika Anda menggunakan armada yang dikelola pelanggan, Anda harus mengonfigurasi lingkungan antrian yang memiliki perilaku yang sama dengan konsol Conda lingkungan antrian. Sebagai contoh, lihat [conda\\_queue\\_env\\_console\\_equivalent.yaml](#) di repositori pada [deadline-cloud-samples](#) GitHub

Conda menyediakan paket dari saluran. Saluran adalah lokasi di mana paket disimpan. Deadline Cloud menyediakan saluran, `deadline-cloud`, yang menjadi tuan rumah Conda paket yang mendukung DCC aplikasi mitra dan penyaji. Pilih setiap tab di bawah ini untuk melihat paket yang tersedia Linux atau Windows.

#### Linux

- Blender
  - `blender=3.6`
  - `blender-openjd`
- Houdini
  - `houdini=19.5`
  - `houdini-openjd`
- Maya
  - `maya=2024`
  - `maya-mtoa=2024.5.3`

Kami sedang menyelidiki laporan render yang macet dengan MToA versi 2024.5.3. Jika Anda memiliki tugas yang terhenti tanpa kesalahan, [hubungi dukungan](#).

- `maya-openjd`
- Nuklir
  - `nuke=15`
  - `nuke-openjd`

## Windows

- KeyShot
  - keyshot=2024.1
  - keyshot-openjd

Saat Anda mengirimkan pekerjaan ke antrian dengan default Conda lingkungan, lingkungan menambahkan dua parameter ke pekerjaan. Parameter ini menentukan Conda paket dan saluran yang akan digunakan untuk mengkonfigurasi lingkungan pekerjaan sebelum tugas diproses.

Parameternya adalah:

- CondaPackages— daftar [spesifikasi kecocokan paket](#) yang dipisahkan ruang, seperti `blender=3.6` atau `numpy>1.22` Defaultnya kosong untuk melewati pembuatan lingkungan virtual.
- CondaChannels— daftar spasi terpisah dari [Conda saluran](#) seperti `deadline-cloud,conda-forge`, atau `s3://amzn-s3-demo-bucket/conda/channel`. Defaultnya adalah `deadline-cloud`, saluran yang tersedia untuk armada yang dikelola layanan yang menyediakan DCC aplikasi dan perender mitra.

Saat Anda menggunakan pengirim terintegrasi untuk mengirim pekerjaan ke Deadline Cloud dari AndaDCC, pengirim mengisi nilai CondaPackages parameter berdasarkan aplikasi dan pengirim. DCC Misalnya, jika Anda menggunakan Blender CondaPackage parameter diatur ke `blender=3.6.* blender-openjd=0.4.*`.

## Hapus antrian

### Warning

Anda tidak dapat memulihkan pekerjaan dalam antrian jika Anda menghapus antrian. Menghapus antrian juga menghapus pekerjaan dalam antrian itu.

1. Dari dasbor Deadline Cloud, pilih Farms dan sumber daya lainnya.
2. Dalam daftar peternakan, pilih peternakan yang berisi antrian yang akan dihapus.
3. Pilih antrian, lalu pilih Hapus.

4. Di jendela konfirmasi, pilih Hapus. Antrian Anda dan semua pekerjaan dalam antrian akan dihapus.

## Mengedit antrian

1. Dari dasbor Deadline Cloud, pilih Farms dan sumber daya lainnya.
2. Dalam daftar peternakan, pilih peternakan yang berisi antrian untuk diedit.
3. Pilih antrian, lalu pilih Edit.
4. Anda dapat mengedit nama, deskripsi, persyaratan anggaran, opsi Jalankan sebagai pengguna, dan peran layanan yang ditetapkan. Anda juga dapat mengaitkan armada yang ada dengan antrian Anda.
5. Pilih Simpan perubahan.

## Kaitkan antrian dan armada

1. Pilih Antrian yang akan Anda kaitkan dengan armada.
2. Untuk memilih armada yang akan dikaitkan dengan antrian Anda, pilih Armada asosiasi.
3. Pilih dropdown Pilih armada. Daftar tampilan armada yang tersedia.
4. Dari daftar armada yang tersedia, pilih kotak centang di sebelah armada atau armada yang ingin Anda kaitkan dengan antrian Anda.
5. Pilih Kaitkan. Status asosiasi armada sekarang harus Terkait.

# Batas waktu Armada Cloud

Bagian ini menjelaskan cara mengelola armada yang dikelola layanan dan armada yang dikelola pelanggan () untuk Deadline Cloud. CMF

Anda dapat mengatur dua jenis armada Deadline Cloud:

- Armada yang dikelola layanan adalah armada pekerja yang memiliki pengaturan default yang disediakan oleh layanan ini, Deadline Cloud. Pengaturan default ini dirancang agar efisien dan hemat biaya.
- Armada yang dikelola pelanggan (CMFs) adalah armada pekerja yang Anda kelola. A CMF dapat tinggal di dalam AWS infrastruktur, on-premise, atau di pusat data yang ditempatkan bersama. A CMF memberikan kontrol penuh dan tanggung jawab armada. Ini termasuk penyediaan, operasi, manajemen, dan penonaktifan pekerja di armada.

Topik

- [Armada yang dikelola layanan](#)
- [Mengelola Deadline Cloud armada yang dikelola pelanggan](#)

## Armada yang dikelola layanan

Armada yang dikelola layanan adalah armada pekerja yang memiliki pengaturan default yang disediakan oleh Deadline Cloud. Pengaturan default ini dirancang agar efisien dan hemat biaya.

Beberapa pengaturan default membatasi jumlah waktu yang dapat dijalankan oleh pekerja dan tugas. Seorang pekerja hanya dapat berlari selama tujuh hari dan tugas hanya dapat berjalan selama lima hari. Ketika batas tercapai, tugas atau pekerja berhenti. Jika ini terjadi, Anda mungkin kehilangan pekerjaan yang sedang dijalankan oleh pekerja atau tugas. Untuk menghindari hal ini, pantau pekerja dan tugas Anda untuk memastikan mereka tidak melebihi batas durasi maksimum. Untuk mempelajari lebih lanjut tentang memantau pekerja Anda, lihat [Menggunakan monitor Deadline Cloud](#).

Buat armada yang dikelola layanan

1. Dari [konsol Deadline Cloud](#), navigasikan ke peternakan tempat Anda ingin membuat armada.
2. Pilih tab Armada.

3. Pilih Buat armada.
4. Masukkan Nama untuk armada Anda.
5. (Opsional) Masukkan Deskripsi. Deskripsi yang jelas dapat membantu Anda mengidentifikasi tujuan armada Anda dengan cepat.
6. Pilih jenis armada yang dikelola layanan.
7. Pilih opsi pasar instans Spot atau On-Demand untuk armada Anda. Instans spot adalah kapasitas tanpa reservasi yang dapat Anda gunakan dengan harga diskon, tetapi dapat terganggu oleh permintaan sesuai permintaan. Instans sesuai permintaan dihargai oleh yang kedua, tetapi tidak memiliki komitmen jangka panjang, dan tidak akan terganggu. Secara default, armada menggunakan instance Spot.
8. (Opsional) Tetapkan jumlah maksimum instans untuk skala armada sehingga kapasitas tersedia untuk pekerjaan dalam antrian. Kami menyarankan Anda meninggalkan jumlah minimum instans di 0 untuk memastikan armada melepaskan semua instance ketika tidak ada pekerjaan yang diantrian.
9. Untuk akses layanan armada Anda, pilih peran yang ada atau buat peran baru. Peran layanan memberikan kredensi untuk instance di armada, memberi mereka izin untuk memproses pekerjaan, dan kepada pengguna di monitor sehingga mereka dapat membaca informasi log.
10. Pilih Berikutnya.
11. Pilih sistem operasi untuk pekerja Anda. Anda dapat meninggalkan default, Linux atau memilih Windows.
12. Masukkan v minimum dan maksimum CPU yang Anda butuhkan untuk armada Anda.
13. Masukkan memori minimum dan maksimum yang Anda butuhkan untuk armada Anda.
14. (Opsional) Anda dapat memilih untuk mengizinkan atau mengecualikan jenis instans tertentu dari armada Anda untuk memastikan hanya jenis instans yang digunakan untuk armada ini.
15. (Opsional) Anda dapat menentukan ukuran volume Amazon Elastic Block Store (AmazonEBS) gp3 yang akan dilampirkan ke pekerja di armada ini. Untuk informasi selengkapnya, lihat [panduan EBS pengguna](#).
16. Pilih Berikutnya.
17. (Opsional) Tentukan kemampuan pekerja khusus yang menentukan fitur armada ini yang dapat digabungkan dengan kemampuan host khusus yang ditentukan pada pengiriman pekerjaan. Salah satu contohnya adalah jenis lisensi tertentu jika Anda berencana untuk menghubungkan armada Anda ke server lisensi Anda sendiri.
18. Pilih Berikutnya.

19. (Opsional) Untuk mengaitkan armada Anda dengan antrian, pilih antrian dari dropdown. Jika antrian diatur dengan default Conda lingkungan antrian, armada Anda secara otomatis dilengkapi dengan paket yang mendukung DCC aplikasi mitra dan perender. Untuk daftar paket yang disediakan, lihat [Default Conda lingkungan antrian](#).
20. Pilih Berikutnya.
21. (Opsional) Untuk menambahkan tag ke armada Anda, pilih Tambahkan tag baru, lalu masukkan kunci dan nilai untuk tag tersebut.
22. Pilih Berikutnya.
23. Tinjau pengaturan armada Anda, lalu pilih Buat armada.

## Gunakan lisensi Anda sendiri

Anda dapat membawa server lisensi Anda sendiri untuk digunakan dengan armada yang dikelola layanan Deadline Cloud. Dengan petunjuk di bawah ini, Anda dapat menggunakan Amazon EC2 Systems Manager (SSM) untuk meneruskan port dari instance worker ke server lisensi atau instance proxy Anda. Untuk membawa lisensi Anda sendiri, Anda dapat mengonfigurasi server lisensi menggunakan lingkungan antrian di peternakan Anda. Untuk mengonfigurasi server lisensi Anda, Anda harus sudah menyiapkan pertanian dan antrian.

### Topik

- [Konfigurasi lingkungan antrian](#)
- [\(Opsional\) Pengaturan instance proxy lisensi](#)
- [CloudFormation pengaturan template](#)

## Konfigurasi lingkungan antrian

Anda dapat mengonfigurasi lingkungan antrian dalam antrian untuk mengakses server lisensi Anda. Pertama, pastikan bahwa Anda memiliki AWS contoh dikonfigurasi dengan akses server lisensi menggunakan salah satu metode berikut:

- Server lisensi — Instance meng-host server lisensi secara langsung.
- Proxy lisensi — Instance memiliki akses jaringan ke server lisensi, dan meneruskan port server lisensi ke server lisensi. Untuk detail tentang cara mengonfigurasi instance proxy lisensi, lihat [\(Opsional\) Pengaturan instance proxy lisensi](#).

Untuk menambahkan izin yang diperlukan ke peran antrian

1. Dari [konsol Cloud Deadline](#), pilih Buka Dasbor.
2. Dari dasbor, pilih pertanian, lalu antrian yang ingin Anda konfigurasi.
3. Dari detail antrian > peran layanan, pilih peran.
4. Pilih Tambahkan izin, lalu pilih Buat kebijakan sebaris.
5. Pilih editor JSON kebijakan, lalu salin dan tempel teks berikut ke editor.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Action": [
        "ssm:StartSession"
      ],
      "Resource": [
        "arn:aws:ssm:region::document/AWS-StartPortForwardingSession",
        "arn:aws:ec2:region:account_id:instance/instance_id"
      ]
    }
  ]
}
```

6. Sebelum menyimpan kebijakan baru, ganti nilai berikut dalam teks kebijakan:
  - Ganti `region` dengan AWS Wilayah tempat peternakan Anda berada
  - Ganti `instance_id` dengan ID instance untuk server lisensi atau instance proxy yang Anda gunakan
  - Ganti `account_id` dengan AWS nomor akun yang berisi peternakan Anda
7. Pilih Berikutnya.
8. Untuk nama Policy, masukkan **LicenseForwarding**.
9. Pilih Buat kebijakan untuk menyimpan perubahan dan membuat kebijakan dengan izin yang diperlukan.

Untuk menambahkan lingkungan antrian baru ke antrian

1. Dari [konsol Deadline Cloud](#), pilih Go to Dashboard jika Anda belum melakukannya.
2. Dari dasbor, pilih pertanian, lalu antrian yang ingin Anda konfigurasi.
3. Pilih Lingkungan Antrian > Tindakan > Buat baru dengan YAML.
4. Salin dan tempel teks berikut ke editor YAML skrip.

Windows

```
specificationVersion: "environment-2023-09"
parameterDefinitions:
  - name: LicenseInstanceId
    type: STRING
    description: >
      The Instance ID of the license server/proxy instance
    default: ""
  - name: LicenseInstanceRegion
    type: STRING
    description: >
      The region containing this farm
    default: ""
  - name: LicensePorts
    type: STRING
    description: >
      Comma-separated list of ports to be forwarded to the license server/proxy
      instance.
      Example: "2700,2701,2702"
    default: ""
environment:
  name: BYOL License Forwarding
  variables:
    example_LICENSE: 2700@localhost
  script:
    actions:
      onEnter:
        command: bash
        args: [ "{{Env.File.Enter}}" ]
      onExit:
        command: bash
        args: [ "{{Env.File.Exit}}" ]
  embeddedFiles:
```

```

- name: Enter
  filename: enter.ps1
  type: TEXT
  runnable: True
  data: |
    $ZIP_NAME="SessionManagerPlugin.zip"
    Invoke-WebRequest -Uri "https://s3.amazonaws.com/session-manager-
downloads/plugin/latest/windows/$ZIP_NAME" -OutFile $ZIP_NAME
    Expand-Archive -Path $ZIP_NAME
    Expand-Archive -Path .\SessionManagerPlugin\package.zip
    conda activate
    python {{Env.File.StartSession}} {{Session.WorkingDirectory}}\package
\bin\session-manager-plugin.exe
- name: Exit
  filename: exit.ps1
  type: TEXT
  runnable: True
  data: |
    Write-Output "Killing SSM Manager Plugin PIDs: $env:BYOL_SSM_PIDS"
    "$env:BYOL_SSM_PIDS".Split(",") | ForEach {
      Write-Output "Killing $_"
      Stop-Process -Id $_ -Force
    }
- name: StartSession
  type: TEXT
  data: |
    import boto3
    import json
    import subprocess
    import sys

    instance_id = "{{Param.LicenseInstanceId}}"
    region = "{{Param.LicenseInstanceRegion}}"
    license_ports_list = "{{Param.LicensePorts}}".split(",")

    ssm_client = boto3.client("ssm", region_name=region)
    pids = []

    for port in license_ports_list:
      session_response = ssm_client.start_session(
        Target=instance_id,
        DocumentName="AWS-StartPortForwardingSession",
        Parameters={"portNumber": [port], "localPortNumber": [port]}
      )

```

```

cmd = [
    sys.argv[1],
    json.dumps(session_response),
    region,
    "StartSession",
    "",
    json.dumps({"Target": instance_id}),
    f"https://ssm.{region}.amazonaws.com"
]

process = subprocess.Popen(cmd, stdout=subprocess.DEVNULL,
stderr=subprocess.DEVNULL)
pids.append(process.pid)
print(f"SSM Port Forwarding Session started for port {port}")

print(f"openjd_env: BYOL_SSM_PIDS={','.join(str(pid) for pid in
pids)}")

```

## Linux

```

specificationVersion: "environment-2023-09"
parameterDefinitions:
  - name: LicenseInstanceId
    type: STRING
    description: >
      The Instance ID of the license server/proxy instance
    default: ""
  - name: LicenseInstanceRegion
    type: STRING
    description: >
      The region containing this farm
    default: ""
  - name: LicensePorts
    type: STRING
    description: >
      Comma-separated list of ports to be forwarded to the license server/proxy
      instance.
      Example: "2700,2701,2702"
    default: ""
environment:

```

```

name: BYOL License Forwarding
variables:
  example_LICENSE: 2700@localhost
script:
  actions:
    onEnter:
      command: bash
      args: [ "{{Env.File.Enter}}" ]
    onExit:
      command: bash
      args: [ "{{Env.File.Exit}}" ]
  embeddedFiles:
    - name: Enter
      type: TEXT
      runnable: True
      data: |
        curl https://s3.amazonaws.com/session-manager-downloads/plugin/
latest/linux_64bit/session-manager-plugin.rpm -Ls | rpm2cpio - | cpio -iv
--to-stdout ./usr/local/sessionmanagerplugin/bin/session-manager-plugin >
{{Session.WorkingDirectory}}/session-manager-plugin
        chmod +x {{Session.WorkingDirectory}}/session-manager-plugin
        conda activate
        python {{Env.File.StartSession}} {{Session.WorkingDirectory}}/session-
manager-plugin
    - name: Exit
      type: TEXT
      runnable: True
      data: |
        echo Killing SSM Manager Plugin PIDs: $BYOL_SSM_PIDS
        for pid in ${BYOL_SSM_PIDS//,/ }; do kill $pid; done
    - name: StartSession
      type: TEXT
      data: |
        import boto3
        import json
        import subprocess
        import sys

        instance_id = "{{Param.LicenseInstanceId}}"
        region = "{{Param.LicenseInstanceRegion}}"
        license_ports_list = "{{Param.LicensePorts}}".split(",")

        ssm_client = boto3.client("ssm", region_name=region)
        pids = []

```

```

for port in license_ports_list:
    session_response = ssm_client.start_session(
        Target=instance_id,
        DocumentName="AWS-StartPortForwardingSession",
        Parameters={"portNumber": [port], "localPortNumber": [port]}
    )

    cmd = [
        sys.argv[1],
        json.dumps(session_response),
        region,
        "StartSession",
        "",
        json.dumps({"Target": instance_id}),
        f"https://ssm.{region}.amazonaws.com"
    ]

    process = subprocess.Popen(cmd, stdout=subprocess.DEVNULL,
stderr=subprocess.DEVNULL)
    pids.append(process.pid)
    print(f"SSM Port Forwarding Session started for port {port}")

print(f"openjd_env: BYOL_SSM_PIDS={' ','.join(str(pid) for pid in
pids)}")

```

5. Sebelum menyimpan lingkungan antrian, buat perubahan berikut pada teks lingkungan sesuai kebutuhan:
  - Perbarui nilai default untuk parameter berikut untuk mencerminkan lingkungan Anda:
    - LicenseInstanceId - ID EC2 instans Amazon dari server lisensi atau instance proxy Anda
    - LicenseInstanceRegion— AWS Wilayah yang berisi peternakan Anda
    - LicensePorts— Daftar port yang dipisahkan koma untuk diteruskan ke server lisensi atau instance proxy (misalnya 2700,2701)
  - Tambahkan variabel lingkungan lisensi yang diperlukan ke bagian variabel. Variabel-variabel ini harus mengarahkan DCCs ke localhost pada port server lisensi. Misalnya, jika server lisensi Foundry Anda mendengarkan pada port 6101, Anda akan menambahkan variabel sebagai **foundry\_LICENSE: 6101@localhost**

6. (Opsional) Anda dapat membiarkan Prioritas disetel ke 0, atau Anda dapat mengubahnya untuk mengurutkan prioritas secara berbeda di antara beberapa lingkungan antrian.
7. Pilih Buat lingkungan antrian untuk menyelamatkan lingkungan baru.

Dengan pengaturan lingkungan antrian, pekerjaan yang dikirimkan ke antrian ini akan mengambil lisensi dari server lisensi yang dikonfigurasi.

## (Opsional) Pengaturan instance proxy lisensi

Sebagai alternatif untuk menggunakan server lisensi, Anda dapat menggunakan proxy lisensi. Untuk membuat proxy lisensi, buat instans Amazon Linux 2023 baru yang memiliki akses jaringan ke server lisensi. Jika perlu, Anda dapat mengonfigurasi akses ini menggunakan VPN koneksi. Untuk informasi selengkapnya, lihat [VPNkoneksi](#) di Panduan VPC Pengguna Amazon.

Untuk menyiapkan instance proxy lisensi untuk Deadline Cloud, ikuti langkah-langkah dalam prosedur ini. Lakukan langkah-langkah konfigurasi berikut pada instance baru ini untuk mengaktifkan penerusan lalu lintas lisensi ke server lisensi Anda

1. Untuk menginstal HAProxy paket, masukkan

```
sudo yum install haproxy
```

2. Perbarui bagian server lisensi mendengarkan dari file konfigurasi `/etc/haproxy/haproxy.cfg` dengan yang berikut:
  - a. Ganti `LicensePort1` dan `LicensePort2` dengan nomor port yang akan diteruskan ke server lisensi. Tambahkan atau hapus nilai yang dipisahkan koma untuk mengakomodasi jumlah port yang diperlukan.
  - b. Ganti `LicenseServerHost` dengan nama host atau alamat IP server lisensi.

```
lobal
  log          127.0.0.1 local2
  chroot      /var/lib/haproxy
  user        haproxy
  group       haproxy
  daemon

defaults
  timeout queue      1m
```

```
timeout connect      10s
timeout client       1m
timeout server       1m
timeout http-keep-alive 10s
timeout check        10s
```

```
listen license-server
  bind *:LicensePort1,*:LicensePort2
  server license-server LicenseServerHost
```

3. Untuk mengaktifkan dan memulai HAProxy layanan, jalankan perintah berikut:

```
sudo systemctl enable haproxy
sudo service haproxy start
```

Setelah menyelesaikan langkah-langkah, permintaan lisensi yang dikirim ke localhost dari lingkungan antrian penerusan harus diteruskan ke server lisensi yang ditentukan.

## CloudFormation pengaturan template

Anda dapat menggunakan CloudFormation template untuk mengonfigurasi seluruh peternakan untuk menggunakan lisensi Anda sendiri.

1. Ubah template yang disediakan di langkah berikutnya untuk menambahkan variabel lingkungan lisensi yang diperlukan ke bagian variabel di bawah BYOLQueueEnvironment.
2. Gunakan yang berikut AWS CloudFormation templat.

```
AWSTemplateFormatVersion: 2010-09-09
Description: "Create AWS Deadline Cloud resources for BYOL"

Parameters:
  LicenseInstanceId:
    Type: AWS::EC2::Instance::Id
    Description: Instance ID for the license server/proxy instance
  LicensePorts:
    Type: String
    Description: Comma-separated list of ports to forward to the license instance

Resources:
  JobAttachmentBucket:
```

```
Type: AWS::S3::Bucket
Properties:
  BucketName: !Sub byol-example-ja-bucket-${AWS::AccountId}-${AWS::Region}
  BucketEncryption:
    ServerSideEncryptionConfiguration:
      - ServerSideEncryptionByDefault:
          SSEAlgorithm: AES256

Farm:
  Type: AWS::Deadline::Farm
  Properties:
    DisplayName: BYOLFarm

QueuePolicy:
  Type: AWS::IAM::ManagedPolicy
  Properties:
    ManagedPolicyName: BYOLQueuePolicy
    PolicyDocument:
      Version: 2012-10-17
      Statement:
        - Effect: Allow
          Action:
            - s3:GetObject
            - s3:PutObject
            - s3:ListBucket
            - s3:GetBucketLocation
          Resource:
            - !Sub ${JobAttachmentBucket.Arn}
            - !Sub ${JobAttachmentBucket.Arn}/job-attachments/*
          Condition:
            StringEquals:
              aws:ResourceAccount: !Sub ${AWS::AccountId}
        - Effect: Allow
          Action: logs:GetLogEvents
          Resource: !Sub arn:aws:logs:${AWS::Region}:${AWS::AccountId}:log-
group:/aws/deadline/${Farm.FarmId}/*
        - Effect: Allow
          Action:
            - s3:ListBucket
            - s3:GetObject
          Resource:
            - "*"
          Condition:
            ArnLike:
```

```
s3:DataAccessPointArn:
  - arn:aws:s3:*:*:accesspoint/deadline-software-*
StringEquals:
  s3:AccessPointNetworkOrigin: VPC
```

**BYOLSSMPolicy:**

Type: AWS::IAM::ManagedPolicy

**Properties:**

ManagedPolicyName: BYOLSSMPolicy

**PolicyDocument:**

Version: 2012-10-17

**Statement:**

- Effect: Allow

**Action:**

- ssm:StartSession

**Resource:**

- !Sub arn:aws:ssm:\${AWS::Region}::document/AWS-

StartPortForwardingSession

- !Sub arn:aws:ec2:\${AWS::Region}:\${AWS::AccountId}:instance/  
\${LicenseInstanceId}

**WorkerPolicy:**

Type: AWS::IAM::ManagedPolicy

**Properties:**

ManagedPolicyName: BYOLWorkerPolicy

**PolicyDocument:**

Version: 2012-10-17

**Statement:**

- Effect: Allow

**Action:**

- logs:CreateLogStream

Resource: !Sub arn:aws:logs:\${AWS::Region}:\${AWS::AccountId}:log-  
group:/aws/deadline/\${Farm.FarmId}/\*

**Condition:**

ForAnyValue:StringEquals:

aws:CalledVia:

- deadline.amazonaws.com

- Effect: Allow

**Action:**

- logs:PutLogEvents

- logs:GetLogEvents

Resource: !Sub arn:aws:logs:\${AWS::Region}:\${AWS::AccountId}:log-  
group:/aws/deadline/\${Farm.FarmId}/\*

**QueueRole:**

Type: AWS::IAM::Role

**Properties:**

RoleName: BYOLQueueRole

ManagedPolicyArns:

- !Ref QueuePolicy
- !Ref BYOLSSMPolicy

AssumeRolePolicyDocument:

Version: 2012-10-17

Statement:

- Effect: Allow

Action:

- sts:AssumeRole

Principal:

Service:

- credentials.deadline.amazonaws.com
- deadline.amazonaws.com

Condition:

StringEquals:

aws:SourceAccount: !Sub \${AWS::AccountId}

ArnEquals:

aws:SourceArn: !Ref Farm

**WorkerRole:**

Type: AWS::IAM::Role

**Properties:**

RoleName: BYOLWorkerRole

ManagedPolicyArns:

- arn:aws:iam::aws:policy/AWSDeadlineCloud-FleetWorker
- !Ref WorkerPolicy

AssumeRolePolicyDocument:

Version: 2012-10-17

Statement:

- Effect: Allow

Action:

- sts:AssumeRole

Principal:

Service: credentials.deadline.amazonaws.com

**Queue:**

Type: AWS::Deadline::Queue

**Properties:**

```
DisplayName: BYOLQueue
FarmId: !GetAtt Farm.FarmId
RoleArn: !GetAtt QueueRole.Arn
JobRunAsUser:
  Posix:
    Group: ""
    User: ""
  RunAs: WORKER_AGENT_USER
JobAttachmentSettings:
  RootPrefix: job-attachments
  S3BucketName: !Ref JobAttachmentBucket
```

**Fleet:**

```
Type: AWS::Deadline::Fleet
Properties:
  DisplayName: BYOLFleet
  FarmId: !GetAtt Farm.FarmId
  MinWorkerCount: 1
  MaxWorkerCount: 2
  Configuration:
    ServiceManagedEc2:
      InstanceCapabilities:
        VCpuCount:
          Min: 4
          Max: 16
        MemoryMiB:
          Min: 4096
          Max: 16384
        OsFamily: LINUX
        CpuArchitectureType: x86_64
      InstanceMarketOptions:
        Type: on-demand
  RoleArn: !GetAtt WorkerRole.Arn
```

**QFA:**

```
Type: AWS::Deadline::QueueFleetAssociation
Properties:
  FarmId: !GetAtt Farm.FarmId
  FleetId: !GetAtt Fleet.FleetId
  QueueId: !GetAtt Queue.QueueId
```

**CondaQueueEnvironment:**

```
Type: AWS::Deadline::QueueEnvironment
```

**Properties:**

```
FarmId: !GetAtt Farm.FarmId
Priority: 5
QueueId: !GetAtt Queue.QueueId
TemplateType: YAML
Template: |
  specificationVersion: 'environment-2023-09'
  parameterDefinitions:
    - name: CondaPackages
      type: STRING
      description: >
        This is a space-separated list of Conda package match specifications to
        install for the job.
        E.g. "blender=3.6" for a job that renders frames in Blender 3.6.

        See https://docs.conda.io/projects/conda/en/latest/user-guide/concepts/pkg-specs.html#package-match-specifications
      default: ""
      userInterface:
        control: LINE_EDIT
        label: Conda Packages
    - name: CondaChannels
      type: STRING
      description: >
        This is a space-separated list of Conda channels from which to install
        packages. Deadline Cloud SMF packages are
        installed from the "deadline-cloud" channel that is configured by
        Deadline Cloud.

        Add "conda-forge" to get packages from the https://conda-forge.org/
        community, and "defaults" to get packages
        from Anaconda Inc (make sure your usage complies with https://www.anaconda.com/terms-of-use).
      default: "deadline-cloud"
      userInterface:
        control: LINE_EDIT
        label: Conda Channels
  environment:
    name: Conda
    script:
      actions:
        onEnter:
          command: "conda-queue-env-enter"
```

```

        args: ["${Session.WorkingDirectory} /.env", "--packages",
"{{Param.CondaPackages}}", "--channels", "${Param.CondaChannels}"]
        onExit:
            command: "conda-queue-env-exit"

BYOLQueueEnvironment:
  Type: AWS::Deadline::QueueEnvironment
  Properties:
    FarmId: !GetAtt Farm.FarmId
    Priority: 10
    QueueId: !GetAtt Queue.QueueId
    TemplateType: YAML
    Template: !Sub |
      specificationVersion: "environment-2023-09"
      parameterDefinitions:
        - name: LicenseInstanceId
          type: STRING
          description: >
            The Instance ID of the license server/proxy instance
          default: "${LicenseInstanceId}"
        - name: LicenseInstanceRegion
          type: STRING
          description: >
            The region containing this farm
          default: "${AWS::Region}"
        - name: LicensePorts
          type: STRING
          description: >
            Comma-separated list of ports to be forwarded to the license server/
proxy instance.
            Example: "2700,2701,2702"
          default: "${LicensePorts}"
    environment:
      name: BYOL License Forwarding
      variables:
        example_LICENSE: 2700@localhost
      script:
        actions:
          onEnter:
            command: bash
            args: [ "${Env.File.Enter}"]
          onExit:
            command: bash
            args: [ "${Env.File.Exit}"] ]

```

```
embeddedFiles:
  - name: Enter
    type: TEXT
    runnable: True
    data: |
      curl https://s3.amazonaws.com/session-manager-downloads/
plugin/latest/linux_64bit/session-manager-plugin.rpm -Ls | rpm2cpio - | cpio
-iv --to-stdout ./usr/local/sessionmanagerplugin/bin/session-manager-plugin >
{{Session.WorkingDirectory}}/session-manager-plugin
      chmod +x {{Session.WorkingDirectory}}/session-manager-plugin
      conda activate
      python {{Env.File.StartSession}} {{Session.WorkingDirectory}}/
session-manager-plugin
  - name: Exit
    type: TEXT
    runnable: True
    data: |
      echo Killing SSM Manager Plugin PIDs: $BYOL_SSM_PIDS
      for pid in ${!BYOL_SSM_PIDS//,/ }; do kill $pid; done
  - name: StartSession
    type: TEXT
    data: |
      import boto3
      import json
      import subprocess
      import sys

      instance_id = "{{Param.LicenseInstanceId}}"
      region = "{{Param.LicenseInstanceRegion}}"
      license_ports_list = "{{Param.LicensePorts}}".split(",")

      ssm_client = boto3.client("ssm", region_name=region)
      pids = []

      for port in license_ports_list:
          session_response = ssm_client.start_session(
              Target=instance_id,
              DocumentName="AWS-StartPortForwardingSession",
              Parameters={"portNumber": [port], "localPortNumber": [port]}
          )

          cmd = [
              sys.argv[1],
              json.dumps(session_response),
```

```
        region,
        "StartSession",
        "",
        json.dumps({"Target": instance_id}),
        f"https://ssm.{region}.amazonaws.com"
    ]

    process = subprocess.Popen(cmd, stdout=subprocess.DEVNULL,
stderr=subprocess.DEVNULL)
    pids.append(process.pid)
    print(f"SSM Port Forwarding Session started for port {port}")

    print(f"openjd_env: BYOL_SSM_PIDS={' '.join(str(pid) for pid in
pids)}")
```

3. Saat menerapkan CloudFormation template, berikan parameter berikut:
  - Memperbarui LicenseInstanceID dengan ID EC2 Instans Amazon dari server lisensi atau instans proxy Anda
  - Perbarui LicensePorts dengan daftar port yang dipisahkan koma untuk diteruskan ke server lisensi atau instance proxy (misalnya 2700,2701)
4. Terapkan template untuk menyiapkan peternakan Anda dengan membawa kemampuan lisensi Anda sendiri.

## VFX Reference Platform kesesuaian

Bagian VFX Reference Platform adalah platform target umum untuk VFX industri. Untuk menggunakan EC2 instance Amazon armada yang dikelola layanan standar yang menjalankan Amazon Linux 2023 dengan perangkat lunak yang mendukung VFX Reference Platform, Anda harus mengingat pertimbangan berikut saat menggunakan armada yang dikelola layanan.

Bagian VFX Reference Platform diperbarui setiap tahun. Pertimbangan untuk menggunakan AL2 023 termasuk armada yang dikelola layanan Deadline Cloud didasarkan pada tahun kalender (CY) 2022 hingga 2024 Platform Referensi. Untuk informasi selengkapnya, silakan lihat [VFX Reference Platform](#).

**Note**

Jika Anda membuat kustom Amazon Machine Image (AMI) untuk armada yang dikelola pelanggan, Anda dapat menambahkan persyaratan ini saat menyiapkan instans AmazonEC2.

Untuk menggunakan VFX Reference Platform perangkat lunak yang didukung pada EC2 instans Amazon AL2 023, pertimbangkan hal berikut:

- Versi glibc yang diinstal dengan AL2 023 kompatibel untuk penggunaan runtime, tetapi tidak untuk membangun perangkat lunak yang kompatibel dengan VFX Reference Platform CY2024 atau sebelumnya.
- Python 3.9 dan 3.11 dilengkapi dengan armada yang dikelola layanan sehingga kompatibel dengan VFX Reference Platform CY2022 dan CY2 024. Python 3.7 dan 3.10 tidak disediakan dalam armada yang dikelola layanan. Perangkat lunak yang membutuhkan mereka harus menyediakan instalasi Python dalam antrian atau lingkungan kerja.
- Beberapa komponen pustaka Boost yang disediakan dalam armada yang dikelola layanan adalah versi 1.75, yang tidak kompatibel dengan VFX Reference Platform. Jika aplikasi Anda menggunakan Boost, Anda harus menyediakan versi pustaka Anda sendiri untuk kompatibilitas.
- TBBPembaruan Intel 3 disediakan dalam armada yang dikelola layanan. Ini kompatibel dengan VFX Reference Platform CY2022, CY2 023, dan CY2 024.
- Pustaka lain dengan versi yang ditentukan oleh VFX Reference Platform tidak disediakan oleh armada yang dikelola layanan. Anda harus menyediakan perpustakaan dengan aplikasi apa pun yang digunakan pada armada yang dikelola layanan. Untuk daftar pustaka, lihat [platform referensi](#).

## Mengelola Deadline Cloud armada yang dikelola pelanggan

Bagian ini menjelaskan cara mengelola armada yang dikelola pelanggan (CMF) untuk Deadline Cloud.

CMF adalah armada pekerja yang Anda kelola. CMF dapat berada di dalam AWS infrastruktur, di lokasi, atau di pusat data yang terletak bersama. CMF memberikan kontrol penuh dan tanggung jawab armada. Ini termasuk penyediaan, operasi, manajemen, dan penonaktifan pekerja di armada.

### Topik

- [Buat armada yang dikelola pelanggan](#)
- [Pengaturan dan konfigurasi host pekerja](#)
- [Kelola akses ke rahasia pengguna Windows pekerjaan](#)
- [Instal dan konfigurasi perangkat lunak yang diperlukan untuk pekerjaan](#)
- [Mengkonfigurasi kredensial AWS](#)
- [Buat Amazon Machine Image](#)
- [Buat infrastruktur armada dengan grup Amazon EC2 Auto Scaling](#)
- [Connect armada yang dikelola pelanggan ke titik akhir lisensi](#)

## Buat armada yang dikelola pelanggan

Untuk membuat armada yang dikelola pelanggan (CMF), selesaikan langkah-langkah berikut.

### Deadline Cloud console

Untuk menggunakan konsol Deadline Cloud untuk membuat armada yang dikelola pelanggan

1. Buka [konsol](#) Deadline Cloud.
2. Pilih Peternakan. Daftar pajangan pertanian yang tersedia.
3. Pilih nama Peternakan tempat Anda ingin bekerja.
4. Pilih tab Armada.
5. Pilih Buat armada.
6. Masukkan Nama untuk armada Anda.
7. (Opsional) Masukkan Deskripsi untuk armada Anda.
8. Pilih Pelanggan yang dikelola untuk jenis Armada.
9. Pilih jenis Auto Scaling. Untuk informasi selengkapnya, lihat [Menggunakan EventBridge untuk menangani peristiwa Auto Scaling](#).
  - Tanpa penskalaan: Anda membuat armada di lokasi dan ingin memilih keluar dari Deadline Cloud Auto Scaling.
  - Rekomendasi penskalaan: Anda membuat armada Amazon Elastic Compute Cloud (Amazon EC2).
10. Pilih akses layanan armada Anda.

- a. Sebaiknya gunakan opsi Buat dan gunakan peran layanan baru untuk setiap armada untuk kontrol izin yang lebih terperinci. Opsi ini dipilih secara default.
  - b. Anda juga dapat menggunakan peran layanan yang ada dengan memilih Pilih peran layanan.
11. Tinjau pilihan Anda, lalu pilih Berikutnya.
  12. Pilih sistem operasi untuk armada Anda. Semua pekerja armada harus memiliki sistem operasi yang sama.
  13. Pilih arsitektur CPU host.
  14. Pilih kemampuan perangkat keras vCPU dan memori minimum dan maksimum untuk memenuhi tuntutan beban kerja armada Anda.
  15. (Opsional) Pilih panah untuk memperluas bagian Tambahkan kemampuan.
  16. (Opsional) Pilih kotak centang untuk Tambahkan kemampuan GPU - Opsional, lalu masukkan GPU dan memori minimum dan maksimum.
  17. Tinjau pilihan Anda, lalu pilih Berikutnya.
  18. (Opsional) Tentukan kemampuan pekerja khusus, lalu pilih Berikutnya.
  19. Menggunakan dropdown, pilih satu atau lebih antrian untuk dikaitkan dengan armada.

 Note

Kami merekomendasikan untuk mengaitkan armada hanya dengan antrian yang semuanya berada dalam batas kepercayaan yang sama. Ini memastikan batas keamanan yang kuat antara menjalankan pekerjaan pada pekerja yang sama.

20. Tinjau asosiasi antrian, lalu pilih Berikutnya.
21. (Opsional) Untuk lingkungan antrian Conda Default, kami akan membuat lingkungan untuk antrian Anda yang akan menginstal paket Conda yang diminta oleh pekerjaan.

 Note

Lingkungan antrian Conda digunakan untuk menginstal paket Conda yang diminta oleh pekerjaan. Biasanya, Anda harus menghapus centang pada lingkungan antrian Conda pada antrian yang terkait dengan CMF karena CMF tidak akan memiliki perintah Conda yang diperlukan diinstal secara default.

22. (Opsional) Tambahkan tag ke CMF Anda. Untuk informasi selengkapnya, lihat [Menandai AWS sumber daya Anda](#).
23. Tinjau konfigurasi armada Anda dan buat perubahan apa pun.
24. Pilih Buat armada.
25. Pilih tab Armada, lalu catat ID Armada.

## AWS CLI

Untuk menggunakan AWS CLI untuk membuat armada yang dikelola pelanggan

1. Buka terminal.
2. Buat `fleet-trust-policy.json` di editor baru.
  - a. Tambahkan kebijakan IAM berikut, ganti teks **MIRING** dengan ID AWS akun dan ID pertanian Deadline Cloud Anda.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "credentials.deadline.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "ACCOUNT_ID"
        },
        "ArnEquals": {
          "aws:SourceArn":
            "arn:aws:deadline:*:ACCOUNT_ID:farm/FARM_ID"
        }
      }
    }
  ]
}
```

- b. Simpan perubahan Anda.
3. Buat `fleet-policy.json`.

## a. Tambahkan kebijakan IAM berikut.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "deadline:AssumeFleetRoleForWorker",
        "deadline:UpdateWorker",
        "deadline>DeleteWorker",
        "deadline:UpdateWorkerSchedule",
        "deadline:BatchGetJobEntity",
        "deadline:AssumeQueueRoleForWorker"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:PrincipalAccount": "${aws:ResourceAccount}"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "logs>CreateLogStream"
      ],
      "Resource": "arn:aws:logs:*:*:*:/aws/deadline/*",
      "Condition": {
        "StringEquals": {
          "aws:PrincipalAccount": "${aws:ResourceAccount}"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "logs:PutLogEvents",
        "logs:GetLogEvents"
      ],
      "Resource": "arn:aws:logs:*:*:*:/aws/deadline/*",
      "Condition": {
        "StringEquals": {
```

```

    "aws:PrincipalAccount": "${aws:ResourceAccount}"
  }
}
]
}

```

b. Simpan perubahan Anda.

4. Tambahkan peran IAM untuk digunakan oleh pekerja di armada Anda.

```

aws iam create-role --role-name FleetWorkerRoleName --assume-role-policy-
document file://fleet-trust-policy.json
aws iam put-role-policy --role-name FleetWorkerRoleName --policy-name
FleetWorkerPolicy --policy-document file://fleet-policy.json

```

5. Buat `create-fleet-request.json`.

a. Tambahkan kebijakan IAM berikut, ganti teks *ITALICIZED* dengan nilai CMF Anda.

 Note

Anda dapat menemukan *ROLE\_ARN* di `create-cmf-fleet.json`  
Untuk *OS\_FAMILY*, Anda harus memilih salah satu dari `linux`, `macos` atau `windows`

```

{
  "farmId": "FARM_ID",
  "displayName": "FLEET_NAME",
  "description": "FLEET_DESCRIPTION",
  "roleArn": "ROLE_ARN",
  "minWorkerCount": 0,
  "maxWorkerCount": 10,
  "configuration": {
    "customerManaged": {
      "mode": "NO_SCALING",
      "workerCapabilities": {
        "vCpuCount": {
          "min": 1,
          "max": 4
        }
      }
    },
  },
}

```

```
        "memoryMiB": {
            "min": 1024,
            "max": 4096
        },
        "osFamily": "OS_FAMILY",
        "cpuArchitectureType": "x86_64",
    },
},
}
```

b. Simpan perubahan Anda.

6. Buat armada Anda.

```
aws deadline create-fleet --cli-input-json file://create-fleet-request.json
```

## Pengaturan dan konfigurasi host pekerja

Host pekerja mengacu pada mesin host yang menjalankan pekerja Deadline Cloud. Bagian ini menjelaskan cara mengatur host pekerja dan mengonfigurasinya untuk kebutuhan spesifik Anda. Setiap host pekerja menjalankan program yang disebut agen pekerja. Agen pekerja bertanggung jawab untuk:

- Mengelola siklus hidup pekerja.
- Sinkronisasi pekerjaan yang ditugaskan, kemajuan dan hasilnya.
- Memantau pekerjaan yang sedang berjalan.
- Meneruskan log ke tujuan yang dikonfigurasi.

Kami menyarankan Anda menggunakan agen pekerja Deadline Cloud yang disediakan. Agen pekerja adalah open source dan kami mendorong permintaan fitur, tetapi Anda juga dapat mengembangkan dan menyesuaikan agar sesuai dengan kebutuhan Anda.

Untuk menyelesaikan tugas di bagian berikut, Anda memerlukan yang berikut:

### Linux

- A Linux Instans Amazon Elastic Compute Cloud (AmazonEC2) berbasis Amazon. Kami merekomendasikan Amazon Linux 2023.

- `sudo` hak istimewa.
- Python 3.9 atau lebih tinggi.

## Windows

- A Windows Instans Amazon Elastic Compute Cloud (AmazonEC2) berbasis Amazon. Kami merekomendasikan Windows Server 2022.
- Akses administrator ke host pekerja
- Python 3.9 atau lebih tinggi diinstal untuk semua pengguna

## Membuat dan mengkonfigurasi lingkungan virtual Python

Anda dapat membuat lingkungan virtual Python di Linux jika Anda telah menginstal Python 3.9 atau lebih besar dan menemukannya di file Anda. PATH

### Note

Pada Windows, file agen harus diinstal ke direktori paket situs global Python. Lingkungan virtual Python saat ini tidak didukung.

Untuk membuat dan mengaktifkan lingkungan virtual Python

1. Buka AWS CLI.
2. Buat dan aktifkan lingkungan virtual Python.

```
python3 -m venv /opt/deadline/worker
source /opt/deadline/worker/bin/activate
pip install --upgrade pip
```

## Instal Deadline Agen pekerja Cloud

Setelah Anda mengatur Python Anda dan menciptakan lingkungan virtual di Linux, instal paket Python agen pekerja Deadline Cloud.

## Untuk menginstal paket Python agen pekerja

1. Buka terminal.
  - a. Pada Linux, buka terminal sebagai root pengguna (atau gunakan `sudo/su`)
  - b. Pada Windows, buka prompt perintah administrator atau PowerShell terminal.
2. Unduh dan instal paket agen pekerja Deadline Cloud dari PyPI:

```
python -m pip install deadline-cloud-worker-agent
```

## Konfigurasi agen pekerja Cloud Deadline

Anda dapat mengonfigurasi pengaturan agen pekerja Deadline Cloud dengan tiga cara. Kami menyarankan Anda menggunakan sistem operasi yang diatur `install-deadline-worker`.

Argumen baris perintah - Anda dapat menentukan argumen saat menjalankan agen pekerja Deadline Cloud dari baris perintah. Beberapa pengaturan konfigurasi tidak tersedia melalui argumen baris perintah. Untuk melihat semua argumen baris perintah yang tersedia, masukkan `deadline-worker-agent --help` untuk melihat semua argumen baris perintah yang tersedia.

Variabel lingkungan — Anda dapat mengonfigurasi agen pekerja Deadline Cloud dengan menyetel variabel lingkungan yang dimulai dengan `DEADLINE_WORKER_`. Misalnya, Anda dapat menggunakan `export DEADLINE_WORKER_VERBOSE=true` untuk mengatur output agen pekerja ke `verbose`. Untuk contoh dan informasi lebih lanjut, lihat `/etc/amazon/deadline/worker.toml.example` di Linux atau `C:\ProgramData\Amazon\Deadline\Config\worker.toml.example` pada Windows.

File konfigurasi - Ketika Anda menginstal agen pekerja, itu membuat file konfigurasi yang terletak di `/etc/amazon/deadline/worker.toml` on Linux atau `C:\ProgramData\Amazon\Deadline\Config\worker.toml` pada Windows. Agen pekerja memuat file konfigurasi ini saat dimulai. Anda dapat menggunakan contoh file konfigurasi (`/etc/amazon/deadline/worker.toml.example` on Linux atau `C:\ProgramData\Amazon\Deadline\Config\worker.toml.example` pada Windows) untuk menyesuaikan file konfigurasi agen pekerja default untuk kebutuhan spesifik Anda.

Terakhir, kami sarankan Anda mengaktifkan auto shutdown untuk agen pekerja setelah perangkat lunak Anda digunakan dan berfungsi seperti yang diharapkan. Hal ini memungkinkan armada pekerja untuk meningkatkan skala ketika diperlukan dan untuk menutup ketika pekerjaan rendering

selesai. Penskalaan otomatis membantu memastikan Anda hanya menggunakan sumber daya sesuai kebutuhan. Untuk mengaktifkan instance yang dimulai oleh grup penskalaan otomatis untuk dimatikan, Anda harus menambahkan `shutdown_on_stop=true` ke file `worker.toml` konfigurasi.

Untuk mengaktifkan auto shutdown

Sebagai **root** pengguna:

- Instal agen pekerja dengan parameter **--allow-shutdown**.

Linux

Masukkan:

```
/opt/deadline/worker/bin/install-deadline-worker \  
  --farm-id FARM_ID \  
  --fleet-id FLEET_ID \  
  --region REGION \  
  --allow-shutdown
```

Windows

Masukkan:

```
install-deadline-worker ^  
  --farm-id FARM_ID ^  
  --fleet-id FLEET_ID ^  
  --region REGION ^  
  --allow-shutdown
```

## Buat pengguna dan grup pekerjaan

Bagian ini menjelaskan hubungan pengguna dan grup yang diperlukan antara pengguna agen dan yang `jobRunAsUser` ditentukan pada antrian Anda.

Agan pekerja Deadline Cloud harus dijalankan sebagai pengguna khusus agen khusus di host. Anda harus mengonfigurasi `jobRunAsUser` properti antrian Deadline Cloud sehingga pekerja akan menjalankan pekerjaan antrian sebagai pengguna dan grup sistem operasi tertentu. Ini berarti Anda dapat mengontrol izin sistem file bersama yang dimiliki pekerjaan Anda. Ini juga menyediakan sebagai batas keamanan penting antara pekerjaan Anda dan pengguna agen pekerja.

## Linux pengguna pekerjaan dan grup

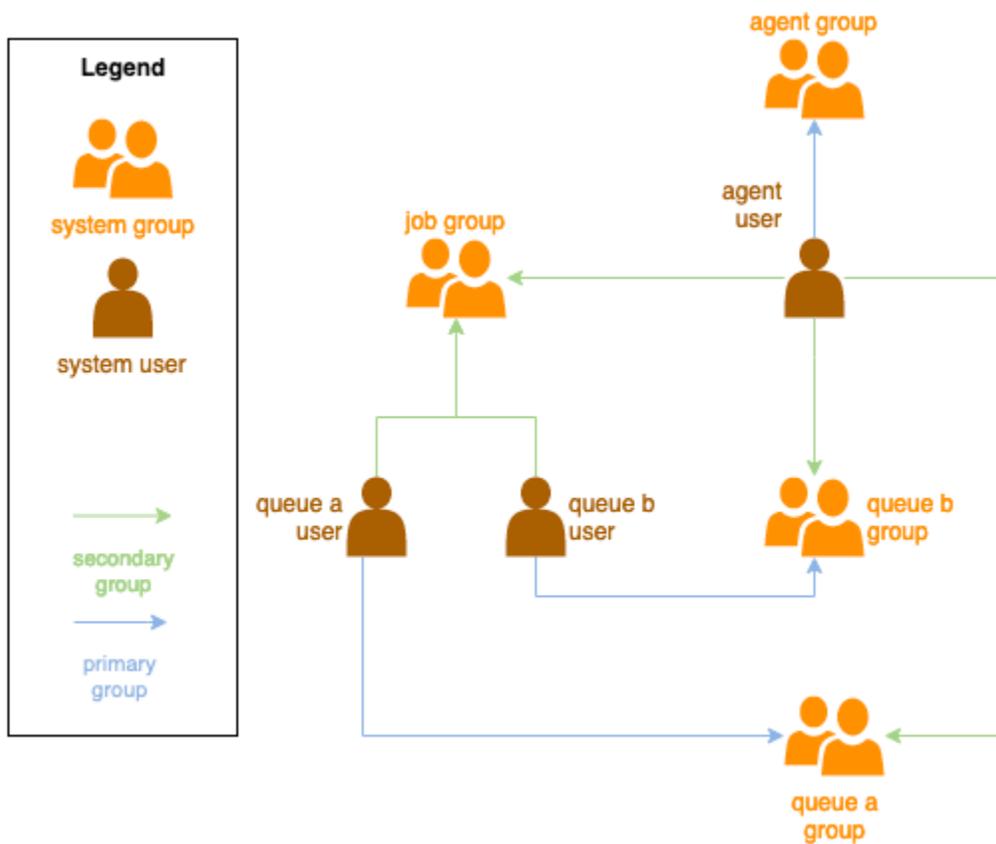
Untuk mengatur agen-pengguna Anda dan `jobRunAsUser`, pastikan Anda memenuhi persyaratan berikut:

- Ada kelompok untuk masing-masing `jobRunAsUser`, dan itu adalah kelompok utama untuk yang sesuai `jobRunAsUser`.
- Agen-pengguna termasuk dalam kelompok utama `jobRunAsUser` untuk antrian di mana pekerja memperoleh pekerjaan. Untuk praktik terbaik keamanan, kami merekomendasikan ini sebagai grup sekunder dari agen-pengguna. Grup bersama ini memungkinkan agen pekerja untuk membuat file tersedia untuk pekerjaan saat sedang berjalan.
- A `jobRunAsUser` bukan milik grup utama agen-pengguna. Untuk praktik terbaik keamanan:
  - File sensitif yang ditulis oleh agen pekerja dimiliki oleh kelompok utama agen.
  - Jika `jobRunAsUser` milik grup ini, dan file yang ditulis agen pekerja mungkin dapat diakses oleh pekerjaan yang dikirimkan ke antrian yang berjalan pada pekerja.
- Default AWS Wilayah harus sesuai dengan Wilayah pertanian milik pekerja. Untuk informasi selengkapnya, lihat [Pengaturan konfigurasi dan file kredensi](#).

Ini harus diterapkan pada:

- Agen-pengguna
- Semua `jobRunAsUser` akun antrian pada pekerja
- Agen-pengguna dapat menjalankan `sudo` perintah sebagai file. `jobRunAsUser`

Diagram berikut menggambarkan hubungan antara pengguna agen dan `jobRunAsUser` pengguna dan grup untuk antrian yang terkait dengan armada.



## Windows pengguna

Untuk menggunakan Windows pengguna sebagai `jobRunAsUser`, itu harus memenuhi persyaratan berikut:

- Semua `jobRunAsUser` pengguna antrian harus ada.
- Kata sandi mereka harus sesuai dengan nilai rahasia yang ditentukan dalam `JobRunAsUser` bidang antrian mereka. Untuk instruksi, lihat langkah 7 di [Membuat antrian](#).
- Agen-pengguna harus dapat masuk sebagai pengguna tersebut.

## Kelola akses ke rahasia pengguna Windows pekerjaan

Saat Anda mengonfigurasi antrian dengan a Windows `jobRunAsUser`, Anda harus menentukan AWS rahasia Secrets Manager. Nilai rahasia ini diharapkan menjadi objek yang dikodekan JSON dari bentuk:

```
{
```

```
"password": "JOB_USER_PASSWORD"
}
```

Agar Pekerja menjalankan pekerjaan sesuai antrian yang dikonfigurasi `jobRunAsUser`, peran IAM armada harus memiliki izin untuk mendapatkan nilai rahasia. Jika rahasia dienkripsi menggunakan kunci KMS yang dikelola pelanggan, maka peran IAM armada juga harus memiliki izin untuk mendekripsi menggunakan kunci KMS.

Sangat disarankan untuk mengikuti prinsip hak istimewa paling sedikit untuk rahasia ini. Ini berarti bahwa akses untuk mengambil nilai rahasia dari antrian `jobRunAsUser` → `windows` → `passwordArn` harus:

- diberikan untuk peran armada ketika asosiasi antrian-armada dibuat antara armada dan antrian
- dicabut dari peran armada ketika asosiasi antrian-armada dihapus antara armada dan antrian

Selanjutnya, AWS rahasia Secrets Manager yang berisi `jobRunAsUser` kata sandi harus dihapus ketika tidak lagi digunakan.

## Berikan akses ke rahasia kata sandi

Armada Cloud deadline memerlukan akses ke `jobRunAsUser` kata sandi yang disimpan dalam rahasia kata sandi antrian saat antrian dan armada terkait. Sebaiknya gunakan kebijakan sumber daya AWS Secrets Manager untuk memberikan akses ke peran armada. Jika Anda benar-benar mematuhi pedoman ini, lebih mudah untuk menentukan peran armada mana yang memiliki akses ke rahasia.

Untuk memberikan akses ke rahasia

1. Buka konsol AWS Secret Manager ke rahasia.
2. Di bagian “Izin sumber daya”, tambahkan pernyataan kebijakan formulir:

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    //...
    {
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "FLEET_ROLE_ARN"
      },
    },
  ],
}
```

```

    "Action" : "secretsmanager:GetSecretValue",
    "Resource" : "*"
  }
  //...
]
}

```

## Cabut akses ke rahasia kata sandi

Ketika armada tidak lagi memerlukan akses ke antrian, hapus akses ke rahasia kata sandi untuk antrian `jobRunAsUser`. Sebaiknya gunakan kebijakan sumber daya AWS Secrets Manager untuk memberikan akses ke peran armada. Jika Anda benar-benar mematuhi pedoman ini, lebih mudah untuk menentukan peran armada mana yang memiliki akses ke rahasia.

Untuk mencabut akses ke rahasia

1. Buka konsol AWS Secret Manager ke rahasia.
2. Di bagian Izin sumber daya, hapus pernyataan kebijakan formulir:

```

{
  "Version" : "2012-10-17",
  "Statement" : [
    //...
    {
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "FLEET_ROLE_ARN"
      },
      "Action" : "secretsmanager:GetSecretValue",
      "Resource" : "*"
    }
    //...
  ]
}

```

## Instal dan konfigurasi perangkat lunak yang diperlukan untuk pekerjaan

Setelah menyiapkan agen pekerja Deadline Cloud, Anda dapat menyiapkan host pekerja dengan perangkat lunak apa pun yang diperlukan untuk menjalankan pekerjaan.

Saat Anda mengirimkan pekerjaan ke antrian dengan yang terkait `jobRunAsUser`, pekerjaan berjalan sebagai pengguna tersebut. Ketika pekerjaan dikirimkan dengan perintah yang bukan jalur absolut, perintah itu harus ditemukan di PATH pengguna itu.

Di Linux, Anda dapat menentukan PATH untuk pengguna di salah satu dari berikut ini:

- mereka `~/.bashrc` atau `~/.bash_profile`
- file konfigurasi sistem seperti `/etc/profile.d/*` dan `/etc/profile`
- skrip startup shell: `/etc/bashrc`.

Di Windows, Anda dapat menentukan PATH untuk pengguna di salah satu dari berikut ini:

- variabel lingkungan khusus pengguna mereka
- variabel lingkungan seluruh sistem

## Instal adaptor alat pembuatan konten digital

Deadline Cloud menyediakan `OpenJobDescription` adaptor untuk menggunakan aplikasi pembuatan konten digital populer (DCC). Untuk menggunakan adaptor ini dalam armada yang dikelola pelanggan, Anda harus menginstal DCC perangkat lunak dan adaptor aplikasi. Kemudian, pastikan program yang dapat dieksekusi perangkat lunak tersedia di jalur pencarian sistem (misalnya, dalam variabel PATH lingkungan).

Untuk memasang DCC adaptor pada armada yang dikelola pelanggan

1. Buka terminal.
  - a. Di Linux, buka terminal sebagai `root` pengguna (atau gunakan `sudo/su`)
  - b. Di Windows, buka prompt perintah administrator atau PowerShell terminal.
2. Instal paket adaptor Deadline Cloud.

```
pip install deadline deadline-cloud-for-maya deadline-cloud-for-nuke deadline-cloud-for-blender
```

## Mengkonfigurasi kredensial AWS

Bagian ini menjelaskan cara mengkonfigurasi AWS kredensial.

Fase awal siklus hidup pekerja ini adalah bootstrap. Pada fase ini, perangkat lunak agen pekerja menciptakan pekerja di armada Anda, dan memperoleh AWS kredensial dari peran armada Anda untuk operasi lebih lanjut.

## AWS credentials for Amazon EC2

Untuk mengonfigurasi AWS kredensial untuk Amazon EC2

1. Buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Pilih Peran di panel navigasi, lalu Buat peran.
3. Pilih AWS layanan.
4. Pilih EC2 sebagai Layanan atau kasus penggunaan, lalu pilih Berikutnya.
5. Lampirkan kebijakan AWSDeadlineCloud-WorkerHost AWS terkelola.

## On-premise AWS credentials

Untuk mengonfigurasi AWS kredensial di lokasi

1. Buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Pilih Peran di panel navigasi, lalu Buat peran.
3. Pilih Akun AWS, lalu pilih Berikutnya.
4. Lampirkan kebijakan AWSDeadlineCloud-WorkerHost AWS terkelola.
5. Hasilkan akses AWS IAM dan kunci rahasia untuk pengguna IAM:
  - a. Untuk Peran IAM Di Mana Saja, lihat Peran [IAM](#) Di Mana Saja.
  - b. Untuk cara paling aman untuk menyiapkan kredensial di host, lihat [Memperoleh kredensial keamanan sementara dari AWS Identity and Access Management](#) Roles Anywhere.
  - c. Anda juga dapat menggunakan CLI sebagai otentikasi alternatif, untuk informasi selengkapnya lihat [Mengautentikasi dengan kredensi pengguna](#) IAM.
6. Simpan kunci ini dalam file AWS kredensial agen-pengguna di sistem file host pekerja.
  - a. Di Linux, ini terletak di `~/.aws/credentials`
  - b. Di Windows, ini terletak di `%USERPROFILE%\aws\credentials`

**Note**

Kredensial hanya boleh diakses oleh nama pengguna OS (`deadline-worker-agent`) yang menginstal agen pekerja.

```
# Replace keys below
[default]
aws_access_key_id=ACCESS_KEY_ID
aws_secret_access_key=SECRET_ACCESS_KEY
```

- Ubah `deadline-worker-agent` pemilik dan izin.

**Note**

Jika Anda mengubah nama pengguna (`deadline-worker-agent`) OS saat menginstal agen pekerja, gunakan nama itu sebagai gantinya.

## Buat Amazon Machine Image

Untuk membuat Amazon Machine Image (AMI) untuk digunakan di Amazon Elastic Compute Cloud (Amazon EC2) armada yang dikelola pelanggan (CMF), selesaikan tugas di bagian ini. Anda harus membuat instans Amazon EC2 sebelum melanjutkan. Untuk informasi selengkapnya, lihat [Meluncurkan instans Anda](#) di Panduan Pengguna Amazon EC2 untuk Instans Linux.

**Important**

Membuat sebuah snapshot dari volume terlampir instans Amazon EC2. Perangkat lunak apa pun yang diinstal pada instance tetap ada sehingga instance, yang digunakan kembali saat Anda meluncurkan instance dari AMI Kami merekomendasikan untuk mengadopsi strategi penambalan dan secara teratur memperbarui perangkat lunak baru AMI dengan yang diperbarui sebelum mendaftar ke armada Anda.

## Siapkan instans Amazon EC2

Sebelum Anda membangun AMI, Anda harus menghapus status pekerja. Negara pekerja tetap ada di antara peluncuran agen pekerja. Jika status ini berlanjut ke AMI, maka semua instance yang diluncurkan darinya akan berbagi status yang sama.

Kami juga menyarankan Anda menghapus file log yang ada. File log dapat tetap berada di instans Amazon EC2 saat Anda menyiapkan AMI. Menghapus file-file ini meminimalkan kebingungan saat mendiagnosis kemungkinan masalah dalam armada pekerja yang menggunakan AMI.

Anda juga harus mengaktifkan layanan sistem agen pekerja sehingga agen pekerja Deadline Cloud diluncurkan saat Amazon EC2 dimulai.

Terakhir, kami sarankan Anda mengaktifkan auto shutdown agen pekerja. Hal ini memungkinkan armada pekerja untuk meningkatkan skala saat dibutuhkan dan dimatikan saat pekerjaan rendering selesai. Penskalaan otomatis ini membantu memastikan Anda hanya menggunakan sumber daya sesuai kebutuhan.

Untuk menyiapkan instans Amazon EC2

1. Buka konsol Amazon EC2.
2. Luncurkan instans Amazon EC2. Untuk informasi selengkapnya, lihat [Meluncurkan instance Anda](#).
3. Siapkan host untuk terhubung ke penyedia identitas Anda (iDP), lalu pasang sistem file bersama yang dibutuhkannya.
4. Ikuti tutorial untuk [Instal Deadline Agen pekerja Cloud](#), kemudian [Konfigurasi agen pekerja](#), dan [Buat pengguna dan grup pekerjaan](#).
5. Jika Anda sedang mempersiapkan AMI berbasis Amazon Linux 2023 untuk menjalankan perangkat lunak yang kompatibel dengan Platform Referensi VFX, Anda perlu memperbarui beberapa persyaratan. Untuk informasi, lihat [VFX Reference Platform kesesuaian](#).
6. Buka terminal.
  - a. Di Linux, buka terminal sebagai root pengguna (atau gunakan sudo/su)
  - b. Windows Aktif, buka prompt perintah administrator atau PowerShell terminal.
7. Pastikan layanan pekerja tidak berjalan dan dikonfigurasi untuk memulai saat boot:
  - a. Di Linux, jalankan

```
systemctl stop deadline-worker  
systemctl enable deadline-worker
```

- b. WindowsAktif, jalankan

```
sc.exe stop DeadlineWorker  
sc.exe config DeadlineWorker start= auto
```

8. Hapus status pekerja.

- a. Di Linux, jalankan

```
rm -rf /var/lib/deadline/*
```

- b. WindowsAktif, jalankan

```
del /Q /S %PROGRAMDATA%\Amazon\Deadline\Cache\*
```

9. Hapus file log.

- a. Di Linux, jalankan

```
rm -rf /var/log/amazon/deadline/*
```

- b. WindowsAktif, jalankan

```
del /Q /S %PROGRAMDATA%\Amazon\Deadline\Logs\*
```

10. PadaWindows, disarankan untuk menjalankan aplikasi Amazon EC2Launch Settings yang ditemukan di menu Start untuk menyelesaikan persiapan host akhir dan shutdown instance.

 Note

Anda HARUS memilih Shutdown tanpa Sysprep dan tidak pernah memilih Shutdown dengan Sysprep. Mematikan dengan Sysprep akan menyebabkan semua pengguna lokal menjadi tidak dapat digunakan. Untuk informasi selengkapnya, lihat [bagian Sebelum Anda Memulai topik Buat AMI kustom dari Panduan Pengguna untuk Instans Windows](#).

## Membangun AMI

Untuk membangun AMI

1. Buka konsol Amazon EC2.
2. Pilih Instans di panel navigasi, lalu pilih instans Anda.
3. Pilih status Instance, lalu Stop instance.
4. Setelah instance Dihentikan, pilih Tindakan.
5. Pilih Gambar dan template, lalu Buat gambar.
6. Masukkan nama Gambar.
7. (Opsional) Masukkan deskripsi untuk gambar Anda.
8. Pilih Buat citra.

## Buat infrastruktur armada dengan grup Amazon EC2 Auto Scaling

Bagian ini menjelaskan cara membuat armada EC2 Auto Scaling Amazon.

Gunakan AWS CloudFormation YAMLtemplate di bawah ini untuk membuat grup Amazon EC2 Auto Scaling (Auto Scaling), Amazon Virtual Private Cloud (VPCAmazon) dengan dua subnet, profil instans, dan peran akses instans. Ini diperlukan untuk meluncurkan instance menggunakan Auto Scaling di subnet.

Anda harus meninjau dan memperbarui daftar jenis instance agar sesuai dengan kebutuhan rendering Anda.

Untuk penjelasan lengkap tentang sumber daya dan parameter yang digunakan dalam CloudFormation YAML template, lihat [referensi tipe sumber daya Deadline Cloud](#) di AWS CloudFormation Panduan Pengguna.

Untuk membuat armada EC2 Auto Scaling Amazon

1. Gunakan contoh berikut untuk membuat CloudFormation template yang mendefinisikan FarmID, FleetID, dan AMIID parameters. Simpan template ke .YAML file di komputer lokal Anda.

```
AWSTemplateFormatVersion: 2010-09-09
Description: Amazon Deadline Cloud customer-managed fleet
Parameters:
```

```
FarmId:
  Type: String
  Description: Farm ID
FleetId:
  Type: String
  Description: Fleet ID
AMIId:
  Type: String
  Description: AMI ID for launching workers
Resources:
  deadlineVPC:
    Type: 'AWS::EC2::VPC'
    Properties:
      CidrBlock: 100.100.0.0/16
  deadlineWorkerSecurityGroup:
    Type: 'AWS::EC2::SecurityGroup'
    Properties:
      GroupDescription: !Join
        - ' '
        - - Security group created for Deadline Cloud workers in the fleet
          - !Ref FleetId
      GroupName: !Join
        - ''
        - - deadlineWorkerSecurityGroup-
          - !Ref FleetId
      SecurityGroupEgress:
        - CidrIp: 0.0.0.0/0
          IpProtocol: '-1'
      SecurityGroupIngress: []
      VpcId: !Ref deadlineVPC
  deadlineIGW:
    Type: 'AWS::EC2::InternetGateway'
    Properties: {}
  deadlineVPCGatewayAttachment:
    Type: 'AWS::EC2::VPCGatewayAttachment'
    Properties:
      VpcId: !Ref deadlineVPC
      InternetGatewayId: !Ref deadlineIGW
  deadlinePublicRouteTable:
    Type: 'AWS::EC2::RouteTable'
    Properties:
      VpcId: !Ref deadlineVPC
  deadlinePublicRoute:
    Type: 'AWS::EC2::Route'
```

```

Properties:
  RouteTableId: !Ref deadlinePublicRouteTable
  DestinationCidrBlock: 0.0.0.0/0
  GatewayId: !Ref deadlineIGW
DependsOn:
  - deadlineIGW
  - deadlineVPCGatewayAttachment
deadlinePublicSubnet0:
  Type: 'AWS::EC2::Subnet'
  Properties:
    VpcId: !Ref deadlineVPC
    CidrBlock: 100.100.16.0/22
    AvailabilityZone: !Join
      - ''
      - - !Ref 'AWS::Region'
        - a
deadlineSubnetRouteTableAssociation0:
  Type: 'AWS::EC2::SubnetRouteTableAssociation'
  Properties:
    RouteTableId: !Ref deadlinePublicRouteTable
    SubnetId: !Ref deadlinePublicSubnet0
deadlinePublicSubnet1:
  Type: 'AWS::EC2::Subnet'
  Properties:
    VpcId: !Ref deadlineVPC
    CidrBlock: 100.100.20.0/22
    AvailabilityZone: !Join
      - ''
      - - !Ref 'AWS::Region'
        - c
deadlineSubnetRouteTableAssociation1:
  Type: 'AWS::EC2::SubnetRouteTableAssociation'
  Properties:
    RouteTableId: !Ref deadlinePublicRouteTable
    SubnetId: !Ref deadlinePublicSubnet1
deadlineInstanceAccessAccessRole:
  Type: 'AWS::IAM::Role'
  Properties:
    RoleName: !Join
      - '-_'
      - - deadline
        - InstanceAccess
      - !Ref FleetId
  AssumeRolePolicyDocument:

```

```
Statement:
  - Effect: Allow
    Principal:
      Service: ec2.amazonaws.com
    Action:
      - 'sts:AssumeRole'
Path: /
ManagedPolicyArns:
  - 'arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy'
  - 'arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore'
  - 'arn:aws:iam::aws:policy/AWSDeadlineCloud-WorkerHost'
deadlineInstanceProfile:
  Type: 'AWS::IAM::InstanceProfile'
  Properties:
    Path: /
    Roles:
      - !Ref deadlineInstanceAccessAccessRole
deadlineLaunchTemplate:
  Type: 'AWS::EC2::LaunchTemplate'
  Properties:
    LaunchTemplateName: !Join
      - ''
      - - deadline-LT-
        - !Ref FleetId
    LaunchTemplateData:
      NetworkInterfaces:
        - DeviceIndex: 0
          AssociatePublicIpAddress: true
          Groups:
            - !Ref deadlineWorkerSecurityGroup
          DeleteOnTermination: true
      ImageId: !Ref AMIID
      InstanceInitiatedShutdownBehavior: terminate
      IamInstanceProfile:
        Arn: !GetAtt
          - deadlineInstanceProfile
          - Arn
      MetadataOptions:
        HttpTokens: required
        HttpEndpoint: enabled

deadlineAutoScalingGroup:
  Type: 'AWS::AutoScaling::AutoScalingGroup'
  Properties:
```

```
AutoScalingGroupName: !Join
- ''
- - deadline-ASG-autoscalable-
- !Ref FleetId
MinSize: 0
MaxSize: 10
VPCZoneIdentifier:
- !Ref deadlinePublicSubnet0
- !Ref deadlinePublicSubnet1
NewInstancesProtectedFromScaleIn: true
MixedInstancesPolicy:
  InstancesDistribution:
    OnDemandBaseCapacity: 0
    OnDemandPercentageAboveBaseCapacity: 0
    SpotAllocationStrategy: capacity-optimized
    OnDemandAllocationStrategy: lowest-price
  LaunchTemplate:
    LaunchTemplateSpecification:
      LaunchTemplateId: !Ref deadlineLaunchTemplate
      Version: !GetAtt
        - deadlineLaunchTemplate
        - LatestVersionNumber
    Overrides:
      - InstanceType: m5.large
      - InstanceType: m5d.large
      - InstanceType: m5a.large
      - InstanceType: m5ad.large
      - InstanceType: m5n.large
      - InstanceType: m5dn.large
      - InstanceType: m4.large
      - InstanceType: m3.large
      - InstanceType: r5.large
      - InstanceType: r5d.large
      - InstanceType: r5a.large
      - InstanceType: r5ad.large
      - InstanceType: r5n.large
      - InstanceType: r5dn.large
      - InstanceType: r4.large
  MetricsCollection:
    - Granularity: 1Minute
    Metrics:
      - GroupMinSize
      - GroupMaxSize
      - GroupDesiredCapacity
```

- GroupInServiceInstances
- GroupTotalInstances
- GroupInServiceCapacity
- GroupTotalCapacity

2. Buka AWS CloudFormation konsol di <https://console.aws.amazon.com/cloudformation>.

Gunakan AWS CloudFormation konsol untuk membuat tumpukan menggunakan instruksi untuk mengunggah file template yang Anda buat. Untuk informasi selengkapnya, lihat [Membuat tumpukan di AWS CloudFormation konsol](#) di AWS CloudFormation Panduan Pengguna.

#### Note

- Kredensi dari IAM peran yang dilampirkan ke EC2 instans Amazon pekerja Anda tersedia untuk semua proses yang berjalan pada pekerja tersebut, yang mencakup pekerjaan. Pekerja harus memiliki hak istimewa paling sedikit untuk beroperasi: `deadline:CreateWorker` dan `deadline:AssumeFleetRoleForWorker`.
- Agen pekerja memperoleh kredensi untuk peran antrian dan mengonfigurasinya untuk digunakan dengan menjalankan pekerjaan. Peran profil EC2 instans Amazon tidak boleh menyertakan izin yang diperlukan oleh pekerjaan Anda.

## Skala otomatis EC2 armada Amazon Anda dengan fitur rekomendasi skala Deadline Cloud

Deadline Cloud memanfaatkan grup Amazon Auto EC2 Scaling (Auto Scaling) untuk menskalakan armada () yang dikelola pelanggan EC2 Amazon secara otomatis. CMF Anda perlu mengonfigurasi mode armada serta menerapkan infrastruktur yang diperlukan di akun Anda untuk membuat skala otomatis armada Anda. Infrastruktur yang Anda gunakan akan berfungsi untuk semua armada, jadi Anda hanya perlu mengaturnya sekali.

Alur kerja dasarnya adalah: Anda mengonfigurasi mode armada Anda ke skala otomatis, lalu Deadline Cloud akan mengirimkan EventBridge acara untuk armada tersebut setiap kali ukuran armada yang direkomendasikan berubah (satu peristiwa berisi id armada, ukuran armada yang direkomendasikan, dan metadata lainnya). Anda akan memiliki EventBridge aturan untuk memfilter acara yang relevan dan meminta Lambda untuk mengkonsumsinya. Lambda akan berintegrasi dengan Amazon Auto EC2 AutoScalingGroup Scaling untuk skala armada Amazon EC2 secara otomatis.

## Setel mode armada ke **EVENT\_BASED\_AUTO\_SCALING**

Konfigurasi mode armada Anda ke **EVENT\_BASED\_AUTO\_SCALING**. Anda dapat menggunakan konsol untuk melakukan ini, atau menggunakan AWS CLI untuk langsung memanggil `CreateFleet` atau `UpdateFleetAPI`. Setelah mode dikonfigurasi, Deadline Cloud mulai mengirimkan `EventBridge` peristiwa setiap kali ukuran armada yang direkomendasikan berubah.

- Contoh `UpdateFleet` perintah:

```
aws deadline update-fleet \  
  --farm-id FARM_ID \  
  --fleet-id FLEET_ID \  
  --configuration file://configuration.json
```

- Contoh `CreateFleet` perintah:

```
aws deadline create-fleet \  
  --farm-id FARM_ID \  
  --display-name "Fleet name" \  
  --max-worker-count 10 \  
  --configuration file://configuration.json
```

Berikut ini adalah contoh yang `configuration.json` digunakan dalam CLI perintah di atas (`--configuration file://configuration.json`).

- Untuk mengaktifkan Auto Scaling pada armada Anda, Anda harus mengatur mode ke **EVENT\_BASED\_AUTO\_SCALING**
- `workerCapabilities` ini adalah nilai default yang ditetapkan untuk CMF saat Anda membuatnya. Anda dapat mengubah nilai-nilai ini jika Anda perlu meningkatkan sumber daya yang tersedia untuk Anda CMF.

Setelah Anda mengonfigurasi mode armada, Deadline Cloud mulai memancarkan acara rekomendasi ukuran armada untuk armada tersebut.

```
{  
  "customerManaged": {  
    "mode": "EVENT_BASED_AUTO_SCALING",  
    "workerCapabilities": {  
      "vCpuCount": {
```

```

        "min": 1,
        "max": 4
    },
    "memoryMiB": {
        "min": 1024,
        "max": 4096
    },
    "osFamily": "linux",
    "cpuArchitectureType": "x86_64",
}
}
}

```

Terapkan tumpukan Auto Scaling menggunakan AWS CloudFormation templat

Anda dapat mengatur EventBridge aturan untuk memfilter peristiwa, Lambda untuk mengkonsumsi peristiwa dan mengontrol Auto Scaling, dan SQS antrian untuk menyimpan peristiwa yang belum diproses. Gunakan yang berikut ini AWS CloudFormation template untuk menyebarkan semuanya dalam tumpukan. Setelah Anda berhasil menyebarkan sumber daya, Anda dapat mengirimkan pekerjaan dan armada akan meningkat secara otomatis.

Resources:

AutoScalingLambda:

Type: 'AWS::Lambda::Function'

Properties:

Code:

ZipFile: |-

"""

This lambda is configured to handle "Fleet Size Recommendation Change" messages. It will handle all such events, and requires that the ASG is named based on the fleet id. It will scale up/down the fleet based on the recommended fleet size in the message.

Example EventBridge message:

```

{
  "version": "0",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "Fleet Size Recommendation Change",
  "source": "aws.deadline",
  "account": "111122223333",
  "time": "2017-12-22T18:43:48Z",
  "region": "us-west-1",
  "resources": [],
}

```



```
AutoScalingEventRule:
  Type: 'AWS::Events::Rule'
  Properties:
    EventPattern:
      source:
        - aws.deadline
      detail-type:
        - Fleet Size Recommendation Change
    State: ENABLED
  Targets:
    - Arn: !GetAtt
      - AutoScalingLambda
      - Arn
    DeadLetterConfig:
      Arn: !GetAtt
      - UnprocessedAutoScalingEventQueue
      - Arn
    Id: Target0
    RetryPolicy:
      MaximumRetryAttempts: 15
AutoScalingEventRuleTargetPermission:
  Type: 'AWS::Lambda::Permission'
  Properties:
    Action: 'lambda:InvokeFunction'
    FunctionName: !GetAtt
      - AutoScalingLambda
      - Arn
    Principal: events.amazonaws.com
    SourceArn: !GetAtt
      - AutoScalingEventRule
      - Arn
AutoScalingLambdaServiceRole:
  Type: 'AWS::IAM::Role'
  Properties:
    AssumeRolePolicyDocument:
      Statement:
        - Action: 'sts:AssumeRole'
          Effect: Allow
          Principal:
            Service: lambda.amazonaws.com
      Version: 2012-10-17
    ManagedPolicyArns:
      - !Join
      - ''
```

```
- - 'arn:'
- !Ref 'AWS::Partition'
- ':iam::aws:policy/service-role/AWSLambdaBasicExecutionRole'
AutoScalingLambdaServiceRoleDefaultPolicy:
  Type: 'AWS::IAM::Policy'
  Properties:
    PolicyDocument:
      Statement:
        - Action: 'autoscaling:SetDesiredCapacity'
          Effect: Allow
          Resource: '*'
      Version: 2012-10-17
    PolicyName: AutoScalingLambdaServiceRoleDefaultPolicy
  Roles:
    - !Ref AutoScalingLambdaServiceRole
UnprocessedAutoScalingEventQueue:
  Type: 'AWS::SQS::Queue'
  Properties:
    QueueName: deadline-unprocessed-autoscaling-events
    UpdateReplacePolicy: Delete
    DeletionPolicy: Delete
UnprocessedAutoScalingEventQueuePolicy:
  Type: 'AWS::SQS::QueuePolicy'
  Properties:
    PolicyDocument:
      Statement:
        - Action: 'sqs:SendMessage'
          Condition:
            ArnEquals:
              'aws:SourceArn': !GetAtt
                - AutoScalingEventRule
                - Arn
          Effect: Allow
          Principal:
            Service: events.amazonaws.com
          Resource: !GetAtt
            - UnprocessedAutoScalingEventQueue
            - Arn
      Version: 2012-10-17
  Queues:
    - !Ref UnprocessedAutoScalingEventQueue
```

## Connect armada yang dikelola pelanggan ke titik akhir lisensi

Bagian AWS Deadline Server lisensi berbasis penggunaan Cloud menyediakan lisensi sesuai permintaan untuk produk pihak ketiga tertentu. Dengan lisensi berbasis penggunaan, Anda dapat membayar saat Anda pergi. Anda hanya dikenakan biaya untuk waktu yang Anda gunakan.

Server lisensi berbasis penggunaan Deadline Cloud dapat digunakan dengan jenis armada apa pun selama pekerja Deadline Cloud dapat berkomunikasi dengan server lisensi. Ini secara otomatis diatur dalam armada yang dikelola layanan. Pengaturan ini hanya diperlukan untuk armada yang dikelola pelanggan.

Untuk membuat server lisensi, Anda memerlukan yang berikut ini:

- Grup keamanan untuk peternakan Anda VPC yang memungkinkan lalu lintas untuk lisensi pihak ketiga.
- Sesi AWS Identity and Access Management (IAM) peran dengan kebijakan terlampir yang memungkinkan akses ke operasi titik akhir lisensi Deadline Cloud.

### Topik

- [Langkah 1: Buat grup keamanan](#)
- [Langkah 2: Siapkan titik akhir lisensi](#)
- [Langkah 3: Hubungkan aplikasi rendering ke titik akhir](#)

### Langkah 1: Buat grup keamanan

Gunakan [VPCKonsol Amazon](#) untuk membuat grup keamanan untuk peternakan Anda VPC. Konfigurasi grup keamanan untuk mengizinkan aturan masuk berikut:

- Autodesk Maya dan Arnold — 2701 - 2702,, TCP IPv4
- Autodesk 3ds Maks - 2704,, TCP IPv4
- Pengecoran Nuke — 6101,, TCP IPv4
- SideFX Houdini, Mantra, dan Karma — 1715 - 1717,, TCP IPv4

Sumber untuk setiap aturan masuk adalah kelompok keamanan pekerja armada.

Untuk informasi selengkapnya tentang membuat grup keamanan, lihat [Membuat grup keamanan](#) di panduan pengguna Amazon Virtual Private Cloud.

## Langkah 2: Siapkan titik akhir lisensi

Titik akhir lisensi menyediakan akses ke server lisensi untuk produk pihak ketiga. Permintaan lisensi dikirim ke titik akhir lisensi. Titik akhir merutekan mereka ke server lisensi yang sesuai. Server lisensi melacak batas penggunaan dan hak. Ada biaya untuk setiap titik akhir lisensi yang Anda buat. Untuk informasi selengkapnya, lihat [VPCharga Amazon](#).

Anda dapat membuat titik akhir lisensi Anda dari AWS Command Line Interface dengan izin yang sesuai. Untuk kebijakan yang diperlukan untuk membuat titik akhir lisensi, lihat [Kebijakan untuk mengizinkan pembuatan titik akhir lisensi](#).

Anda dapat menggunakan [AWS CloudShell](#) atau lainnya AWS CLI lingkungan untuk mengkonfigurasi titik akhir lisensi menggunakan berikut AWS Command Line Interface perintah.

1. Buat titik akhir lisensi. Ganti ID grup keamanan, ID subnet, dan VPC ID dengan nilai yang Anda buat sebelumnya. Jika Anda menggunakan beberapa subnet, pisahkan dengan spasi.

```
aws deadline create-license-endpoint \  
  --security-group-id SECURITY_GROUP_ID \  
  --subnet-ids SUBNET_ID1 SUBNET_ID2 \  
  --vpc-id VPC_ID
```

2. Konfirmasikan bahwa titik akhir berhasil dibuat dengan perintah berikut. Ingat DNS nama VPC titik akhir.

```
aws deadline get-license-endpoint \  
  --license-endpoint-id LICENSE_ENDPOINT_ID
```

3. Lihat daftar produk meteran yang tersedia:

```
aws deadline list-available-metered-products
```

4. Tambahkan produk terukur ke titik akhir lisensi dengan perintah berikut.

```
aws deadline put-metered-product \  
  --license-endpoint-id LICENSE_ENDPOINT_ID \  
  --product-id PRODUCT_ID
```

Anda dapat menghapus produk dari titik akhir lisensi dengan `remove-metered-product` perintah:

```
aws deadline remove-metered-product \  
  --license-endpoint-id LICENSE_ENDPOINT_ID \  
  --product-id PRODUCT_ID
```

Anda dapat menghapus titik akhir lisensi dengan `delete-license-endpoint` perintah:

```
aws deadline delete-license-endpoint \  
  --license-endpoint-id LICENSE_ENDPOINT_ID
```

### Langkah 3: Hubungkan aplikasi rendering ke titik akhir

Setelah titik akhir lisensi diatur, aplikasi menggunakannya sama seperti mereka menggunakan server lisensi pihak ketiga. Anda biasanya mengonfigurasi server lisensi untuk aplikasi dengan menetapkan variabel lingkungan atau pengaturan sistem lainnya, seperti kunci registri Microsoft Windows, ke port dan alamat server lisensi.

Untuk mendapatkan DNS nama titik akhir lisensi, gunakan yang berikut AWS CLI perintah.

```
aws deadline get-license-endpoint --license-endpoint-id LICENSE_ENDPOINT_ID
```

Atau Anda dapat menggunakan [Amazon VPC Console](#) untuk mengidentifikasi VPC titik akhir yang dibuat oleh Deadline Cloud API di langkah sebelumnya.

#### Contoh konfigurasi

##### Example Autodesk Maya dan Arnold

Atur variabel lingkungan `ADSKFLEX_LICENSE_FILE` ke:

```
2702@VPC_Endpoint_DNS_Name:2701@VPC_Endpoint_DNS_Name
```

#### Note

Untuk Windows pekerja, gunakan semi-colon (;) alih-alih titik dua (:) untuk memisahkan titik akhir.

##### Example — Autodesk 3ds Maks

Atur variabel lingkungan `ADSKFLEX_LICENSE_FILE` ke:

```
2704@VPC_Endpoint_DNS_Name
```

### Example — Pengecoran Nuke

Setel variabel lingkungan `foundry_LICENSE` ke `6101@VPC_Endpoint_DNS_Name` Untuk menguji bahwa lisensi berfungsi dengan baik, Anda dapat menjalankan Nuke di terminal:

```
~/nuke/Nuke14.0v5/Nuke14.0 -x
```

### Example — SideFX Houdini, Mantra, dan Karma

Jalankan perintah berikut:

```
/opt/hfs19.5.640/bin/hserver -S  
"http://VPC_Endpoint_DNS_Name:1715;http://VPC_Endpoint_DNS_Name:1716;http://  
VPC_Endpoint_DNS_Name:1717;"
```

Untuk menguji bahwa lisensi berfungsi dengan baik, Anda dapat merender adegan Houdini melalui perintah ini:

```
/opt/hfs19.5.640/bin/hython ~/forpentest.hip -c "hou.node('/out/mantra1').render()"
```

# Mengelola pengguna di Deadline Cloud

AWS Deadline Cloud digunakan AWS IAM Identity Center untuk mengelola pengguna dan grup. IAM Identity Center adalah layanan single sign-on berbasis cloud yang dapat diintegrasikan dengan penyedia single-sign on () perusahaan Anda. SSO Dengan integrasi, pengguna dapat masuk dengan akun perusahaan mereka.

Deadline Cloud mengaktifkan IAM Identity Center secara default, dan diperlukan untuk mengatur dan menggunakan Deadline Cloud. Untuk informasi selengkapnya, lihat [Mengelola sumber identitas Anda](#).

Pemilik organisasi untuk Anda AWS Organizations bertanggung jawab untuk mengelola pengguna dan grup yang memiliki akses ke monitor Deadline Cloud Anda. Anda dapat membuat dan mengelola pengguna dan grup ini menggunakan IAM Identity Center atau konsol Deadline Cloud. Untuk informasi lebih lanjut, lihat [Apa yang dimaksud dengan AWS Organizations](#).

Anda membuat dan menghapus pengguna dan grup yang dapat menggunakan monitor untuk mengelola farm, antrian, dan armada menggunakan konsol Deadline Cloud. Ketika Anda menambahkan pengguna ke Deadline Cloud, mereka harus mengatur ulang kata sandi mereka menggunakan Pusat IAM Identitas sebelum mereka mendapatkan akses.

## Topik

- [Kelola pengguna dan grup untuk monitor](#)
- [Kelola pengguna dan grup untuk peternakan, antrian, dan armada](#)

## Kelola pengguna dan grup untuk monitor

Pemilik Organizations dapat menggunakan konsol Deadline Cloud untuk mengelola pengguna dan grup yang memiliki akses ke monitor Deadline Cloud. Anda dapat memilih dari pengguna dan grup Pusat IAM Identitas yang ada, atau Anda dapat menambahkan pengguna dan grup baru dari konsol.

1. Masuk ke AWS Management Console dan buka [konsol](#) Deadline Cloud. Dari halaman utama, di bagian Memulai, pilih Atur Batas Waktu Cloud atau Buka dasbor.
2. Di panel navigasi kiri, pilih Manajemen pengguna. Secara default, tab Grup dipilih.

Bergantung pada tindakan yang akan diambil, pilih tab Grup atau tab Pengguna.

## Groups

Untuk membuat grup

1. Pilih Buat grup.
2. Masukkan nama grup. Nama harus unik di antara kelompok-kelompok di organisasi Pusat IAM Identitas Anda.

Untuk menghapus grup

1. Pilih grup yang akan dihapus.
2. Pilih Hapus.
3. Dalam dialog konfirmasi, pilih Hapus grup.

 Note

Anda menghapus grup dari Pusat IAM Identitas. Anggota grup tidak dapat lagi masuk ke Deadline Cloud atau mengakses sumber daya pertanian.

## Users

Untuk menambahkan pengguna

1. Pilih tab Pengguna.
2. Pilih Add Users (Tambahkan pengguna).
3. Masukkan nama, alamat email, dan nama pengguna untuk pengguna baru.
4. (Opsional) Pilih satu atau beberapa grup Pusat IAM Identitas untuk menambahkan pengguna baru.
5. Pilih Kirim undangan untuk mengirim email kepada pengguna baru dengan instruksi untuk bergabung dengan organisasi Pusat IAM Identitas Anda.

Untuk menghapus pengguna

1. Pilih pengguna yang akan Anda hapus.
2. Pilih Hapus.

3. Dalam dialog konfirmasi, pilih Hapus pengguna.

 Note

Anda menghapus pengguna dari Pusat IAM Identitas. Pengguna tidak dapat lagi masuk ke monitor Deadline Cloud atau mengakses sumber daya pertanian.

## Kelola pengguna dan grup untuk peternakan, antrian, dan armada

Sebagai bagian dari mengelola pengguna dan grup, Anda dapat memberikan izin akses pada tingkat yang berbeda. Setiap level berikutnya mencakup izin untuk level sebelumnya. Daftar berikut menjelaskan empat tingkat akses dari tingkat terendah ke tingkat tertinggi:

- Penampil — Izin untuk melihat sumber daya di peternakan, antrian, armada, dan pekerjaan yang dapat mereka akses. Penampil tidak dapat mengirimkan atau membuat perubahan pada pekerjaan.
  - Kontributor — Sama seperti pemirsa, tetapi dengan izin untuk mengirimkan pekerjaan ke antrian atau peternakan.
  - Manajer — Sama seperti kontributor, tetapi dengan izin untuk mengedit pekerjaan dalam antrian yang dapat mereka akses, dan memberikan izin pada sumber daya yang dapat mereka akses.
  - Pemilik — Sama seperti manajer, tetapi dapat melihat dan membuat anggaran dan melihat penggunaan.
1. Jika Anda belum melakukannya, masuk ke AWS Management Console dan buka [konsol](#) Deadline Cloud.
  2. Di panel navigasi kiri, pilih Peternakan dan sumber daya lainnya.
  3. Pilih peternakan untuk dikelola. Pilih nama pertanian untuk membuka halaman detail. Anda dapat mencari peternakan menggunakan bilah pencarian.
  4. Untuk mengelola antrian atau armada, pilih tab Antrian atau Armada, lalu pilih antrian atau armada yang akan dikelola.
  5. Pilih tab Manajemen akses. Secara default, tab Grup dipilih. Untuk mengelola pengguna, pilih Pengguna.

Bergantung pada tindakan yang akan diambil, pilih tab Grup atau tab Pengguna.

## Groups

Untuk menambahkan grup

1. Pilih sakelar Grup.
2. Pilih Tambah grup.
3. Dari dropdown, pilih grup yang akan ditambahkan.
4. Untuk tingkat akses grup, pilih salah satu opsi berikut:
  - Penampil
  - Kontributor
  - Manajer
  - Pemilik
5. Pilih Tambahkan.

Untuk menghapus grup

1. Pilih grup yang akan dihapus.
2. Pilih Hapus.
3. Dalam dialog konfirmasi, pilih Hapus grup.

## Users

Untuk menambahkan pengguna

1. Untuk menambahkan pengguna, pilih Tambah pengguna.
2. Dari dropdown, pilih pengguna yang akan ditambahkan.
3. Untuk tingkat akses pengguna, pilih salah satu opsi berikut:
  - Penampil
  - Kontributor
  - Manajer
  - Pemilik
4. Pilih Tambahkan.

## Untuk menghapus pengguna

1. Pilih pengguna yang akan dihapus.
2. Pilih Hapus.
3. Dalam dialog konfirmasi, pilih Hapus pengguna.

# Lowongan kerja Deadline Cloud

Pekerjaan adalah serangkaian instruksi yang digunakan AWS Deadline Cloud untuk menjadwalkan dan menjalankan pekerjaan pada pekerja yang tersedia. Saat Anda membuat pekerjaan, Anda memilih pertanian dan antrian untuk mengirim pekerjaan. Anda juga menyediakan YAML file JSON atau yang memberikan instruksi bagi pekerja untuk memproses. Deadline Cloud menerima template pekerjaan yang mengikuti spesifikasi Open Job Description (OpenJD) untuk mendeskripsikan lowongan. Untuk informasi selengkapnya, lihat [Dokumentasi Open Job Description](#) di GitHub situs web.

Pekerjaan terdiri dari:

- Langkah - Mendefinisikan skrip untuk dijalankan pada pekerja. Langkah-langkah dapat memiliki persyaratan seperti memori pekerja minimum atau langkah-langkah lain yang perlu diselesaikan terlebih dahulu. Setiap langkah memiliki satu atau lebih tugas.
- Tugas — Unit kerja yang dikirim ke pekerja untuk melakukan. Tugas adalah kombinasi dari skrip dan parameter langkah, seperti nomor bingkai, yang digunakan dalam skrip. Pekerjaan selesai ketika semua tugas selesai untuk semua langkah.
- Lingkungan — Siapkan dan hancurkan instruksi yang dibagikan oleh beberapa langkah atau tugas.

Anda dapat membuat pekerjaan dengan salah satu cara berikut:

- Gunakan submitter Deadline Cloud.
- Buat bundel pekerjaan dan gunakan [antarmuka baris perintah Deadline Cloud](#) (Deadline CloudCLI).
- Gunakan AWS SDK.
- Gunakan AWS Command Line Interface (AWS CLI).

Submitter adalah plugin untuk perangkat lunak pembuatan konten digital Anda (DCC) yang mengelola pembuatan pekerjaan di antarmuka ke perangkat lunak AndaDCC. Setelah Anda membuat pekerjaan, Anda menggunakan submitter untuk mengirimkannya ke Deadline Cloud untuk diproses. Di belakang layar, submitter membuat template pekerjaan OpenJD yang menjelaskan pekerjaan. Pada saat yang sama, ia mengunggah file aset Anda ke bucket Amazon Simple Storage Service (Amazon S3). Untuk mengurangi waktu yang diperlukan untuk mengirim file, hanya file yang telah berubah sejak terakhir kali Anda mengunggah file yang dikirim ke Amazon S3.

Untuk membuat skrip dan pipeline Anda sendiri untuk mengirimkan pekerjaan ke Deadline Cloud, Anda dapat menggunakan operasi Deadline CloudCLI, the AWS SDK, atau AWS CLI to call untuk membuat, mendapatkan, melihat, dan mencantumkan pekerjaan. Topik berikut menjelaskan cara menggunakan Deadline CloudCLI.

Deadline Cloud CLI diinstal bersama dengan pengirim Deadline Cloud. Untuk informasi selengkapnya, lihat [Mengatur Deadline Pengirim Cloud](#).

Topik

- [Mengirimkan pekerjaan dengan Deadline Cloud CLI](#)
- [Menjadwalkan pekerjaan di Deadline Cloud](#)
- [Status pekerjaan di Deadline Cloud CLI](#)
- [Memodifikasi pekerjaan di Deadline Cloud](#)
- [Bagaimana Deadline Cloud memproses pekerjaan](#)
- [Memecahkan masalah Deadline pekerjaan Cloud](#)

## Mengirimkan pekerjaan dengan Deadline Cloud CLI

Untuk mengirimkan pekerjaan menggunakan antarmuka baris perintah Deadline Cloud (Deadline CloudCLI), gunakan perintah `deadline bundle submit`

Pekerjaan diserahkan ke antrian. Jika Anda belum menyiapkan pertanian dan antrian, gunakan [konsol](#) Deadline Cloud untuk menyiapkan pertanian dan antrian dan untuk melihat ID pertanian dan antrian. Untuk informasi selengkapnya, lihat [Menentukan detail pertanian](#) dan [Menentukan detail antrian](#).

Untuk mengatur farm default dan antrian untuk Deadline CloudCLI, gunakan perintah berikut. Saat mengatur default, Anda dapat menggunakan CLI perintah Deadline Cloud tanpa menentukan farm atau antrian. Dalam contoh berikut, ganti *farmId* dan *queueId* dengan informasi Anda sendiri:

```
deadline config set defaults.farm_id farmId
deadline config set defaults.queue_id queueId
```

Untuk menentukan langkah dan tugas dalam pekerjaan, buat template pekerjaan OpenJD. Untuk informasi selengkapnya, lihat [Skema Templat \[Versi: 2023-09\]](#) di repositori spesifikasi Open Job Description. GitHub

Contoh berikut adalah template YAML pekerjaan. Ini mendefinisikan pekerjaan dengan dua langkah dan lima tugas per langkah.

```
name: Sample Job
specificationVersion: jobtemplate-2023-09
steps:
- name: Sample Step 1
  parameterSpace:
    taskParameterDefinitions:
      - name: var
        range: 1-5
        type: INT
    script:
      actions:
        onRun:
          args:
            - '1'
          command: /usr/bin/sleep
- name: Sample Step 2
  parameterSpace:
    taskParameterDefinitions:
      - name: var
        range: 1-5
        type: INT
    script:
      actions:
        onRun:
          args:
            - '1'
          command: /usr/bin/sleep
```

Untuk membuat pekerjaan, buat folder baru bernama `sample_job`, lalu simpan file template di folder baru sebagai `template.yaml`. Anda mengirimkan pekerjaan dengan CLI perintah Deadline Cloud berikut:

```
deadline bundle submit path/to/sample_job
```

Respons dari perintah berisi pengidentifikasi untuk pekerjaan itu. Ingat ID sehingga Anda dapat memeriksa status pekerjaan nanti.

```
Submitting to Queue: test-queue
```

```
Waiting for Job to be created...
Submitted job bundle:
  sample_job
Job creation completed successfully
jobId
```

Ada opsi tambahan yang dapat Anda gunakan saat mengirimkan pekerjaan. Untuk informasi selengkapnya, lihat [Opsinya untuk mengirimkan pekerjaan dengan Deadline Cloud CLI](#).

## Opsinya untuk mengirimkan pekerjaan dengan Deadline Cloud CLI

CLI Perintah `deadline bundle submit` Deadline Cloud menyediakan opsi yang dapat Anda gunakan untuk menentukan informasi tambahan untuk suatu pekerjaan. Contoh berikut menunjukkan kepada Anda cara:

- Tentukan parameter yang digunakan saat memproses template pekerjaan.
- Lampirkan file dan folder di lingkungan bersama ke pekerjaan.
- Tetapkan jumlah maksimum kegagalan tugas sebelum pekerjaan dibatalkan.
- Atur jumlah maksimum percobaan ulang untuk suatu tugas.

## Parameter Tugas

`parameters` Opsi menetapkan nilai parameter pekerjaan saat Anda membuat pekerjaan. Template pekerjaan mendefinisikan bidang, dan `parameters` opsi menetapkan nilai. Parameter dapat memiliki nilai default. Jika nilai ditentukan untuk parameter, nilai yang ditentukan mengesampingkan nilai default.

Template pekerjaan berikut mendefinisikan `TestParameter` bidang:

```
name: Sample Job With Job Parameter
parameterDefinitions:
- default: test
  name: TestParameter
  type: STRING
specificationVersion: jobtemplate-2023-09
steps:
- description: step description
  name: MyStep
  parameterSpace:
```

```
taskParameterDefinitions:
  - name: var
    range: 1-5
    type: INT
script:
  actions:
    onRun:
      args:
        - '1'
      command: /usr/bin/sleep
```

Perintah berikut menetapkan nilai TestParameter untuk "Hello AWS ":

```
deadline bundle submit sample_job --parameter "TestParameter=Hello AWS"
```

## Profil penyimpanan

Profil penyimpanan membantu berbagi file antara pekerja dengan sistem operasi yang berbeda. Buat profil penyimpanan menggunakan konsol Deadline Cloud. Kemudian, gunakan `storage-profile-id` parameter untuk menggunakan profil penyimpanan. Untuk informasi selengkapnya, lihat [Penyimpanan bersama di Deadline Cloud](#).

Untuk mengatur profil penyimpanan untuk pengiriman pekerjaan, menggunakan Deadline CloudCLI, gunakan perintah berikut untuk mengatur parameter `storage-profile-id` konfigurasi:

```
deadline config set settings.storage_profile_id storageProfileId
```

## Tugas gagal maksimum

`max-failed-tasks-count`Opsi menetapkan jumlah maksimum tugas yang dapat gagal sebelum seluruh pekerjaan gagal dan semua tugas yang tersisa ditandai CANCELED. Nilai default-nya adalah 100.

```
deadline bundle submit sample_job --max-failed-tasks-count 10
```

## Percobaan ulang tugas maksimum yang gagal

`max-retries-per-task`Opsi ini menetapkan jumlah maksimum kali tugas dicoba ulang sebelum gagal. Ketika tugas dicoba lagi, itu diletakkan di READY negara bagian. Nilai bawaannya adalah 5.

```
deadline bundle submit sample_job --max-retries-per-task 10
```

## Menjadwalkan pekerjaan di Deadline Cloud

Setelah pekerjaan dibuat, AWS Deadline Cloud menjadwalkannya untuk diproses pada satu atau lebih armada yang terkait dengan antrian. Armada yang memproses tugas tertentu dipilih berdasarkan kemampuan yang dikonfigurasi untuk armada dan persyaratan tuan rumah dari langkah tertentu.

Pekerjaan dijadwalkan dalam urutan prioritas upaya terbaik, tertinggi ke terendah. Ketika dua pekerjaan memiliki prioritas yang sama, pekerjaan tertua dijadwalkan terlebih dahulu.

Bagian berikut memberikan rincian proses penjadwalan pekerjaan.

### Tentukan kompatibilitas armada

Setelah pekerjaan dibuat, Deadline Cloud memeriksa persyaratan host untuk setiap langkah dalam pekerjaan terhadap kemampuan armada yang terkait dengan antrian pekerjaan yang diajukan. Jika armada memenuhi persyaratan tuan rumah, pekerjaan itu dimasukkan ke READY negara bagian.

Jika ada langkah dalam pekerjaan yang memiliki persyaratan yang tidak dapat dipenuhi oleh armada yang terkait dengan antrian, status langkah diatur keNOT\_COMPATIBLE. Selain itu, sisa langkah dalam pekerjaan dibatalkan.

Kemampuan untuk armada ditetapkan pada tingkat armada. Bahkan jika seorang pekerja dalam armada memenuhi persyaratan pekerjaan, itu tidak akan diberikan tugas dari pekerjaan jika armadanya tidak memenuhi persyaratan pekerjaan.

Template pekerjaan berikut memiliki langkah yang menentukan persyaratan host untuk langkah tersebut:

```
name: Sample Job With Host Requirements
specificationVersion: jobtemplate-2023-09
steps:
- name: Step 1
  script:
    actions:
      onRun:
        args:
```

```

- '1'
  command: /usr/bin/sleep
hostRequirements:
  amounts:
  # Capabilities starting with "amount." are amount capabilities. If they start with
"amount.worker.",
  # they are defined by the OpenJD specification. Other names are free for custom
usage.
- name: amount.worker.vcpu
  min: 4
  max: 8
  attributes:
- name: attr.worker.os.family
  anyOf:
- linux

```

Pekerjaan ini dapat dijadwalkan ke armada dengan kemampuan sebagai berikut:

```

{
  "vCpuCount": {"min": 4, "max": 8},
  "memoryMiB": {"min": 1024},
  "osFamily": "linux",
  "cpuArchitectureType": "x86_64"
}

```

Pekerjaan ini tidak dapat dijadwalkan ke armada dengan salah satu kemampuan berikut:

```

{
  "vCpuCount": {"min": 4},
  "memoryMiB": {"min": 1024},
  "osFamily": "linux",
  "cpuArchitectureType": "x86_64"
}

```

The vCpuCount has no maximum, so it exceeds the maximum vCPU host requirement.

```

{
  "vCpuCount": {"max": 8},
  "memoryMiB": {"min": 1024},
  "osFamily": "linux",
  "cpuArchitectureType": "x86_64"
}

```

The vCpuCount has no minimum, so it doesn't satisfy the minimum vCPU host requirement.

```
{  
  "vCpuCount": {"min": 4, "max": 8},  
  "memoryMiB": {"min": 1024},  
  "osFamily": "windows",  
  "cpuArchitectureType": "x86_64"  
}
```

The osFamily doesn't match.

## Penskalaan armada

Ketika pekerjaan ditugaskan ke armada yang dikelola layanan yang kompatibel, armada diskalakan secara otomatis. Jumlah pekerja di armada berfluktuasi berdasarkan jumlah tugas yang tersedia untuk dijalankan armada.

Ketika pekerjaan ditugaskan ke armada yang dikelola pelanggan, pekerja mungkin sudah ada atau dapat dibuat menggunakan penskalaan otomatis berbasis peristiwa. Untuk informasi selengkapnya, lihat [Menggunakan EventBridge untuk menangani peristiwa penskalaan otomatis](#) di Panduan Pengguna Penskalaan EC2 Otomatis Amazon.

## Sesi

Tugas dalam suatu pekerjaan dibagi menjadi satu atau lebih sesi. Pekerja menjalankan sesi untuk mengatur lingkungan, menjalankan tugas, dan kemudian meruntuhkan lingkungan. Setiap sesi terdiri dari satu atau lebih tindakan yang harus dilakukan seorang pekerja.

Saat pekerja menyelesaikan tindakan bagian, tindakan sesi tambahan dapat dikirim ke pekerja. Pekerja menggunakan kembali lingkungan yang ada dan lampiran pekerjaan dalam sesi untuk menyelesaikan tugas dengan lebih efisien.

Lampiran Job dibuat oleh pengirim yang Anda gunakan, sebagai bagian dari paket pekerjaan Deadline Cloud Anda. CLI Anda juga dapat membuat lampiran pekerjaan dengan menggunakan `--attachments` opsi untuk `create-job` AWS CLI perintah. Lingkungan didefinisikan di dua tempat: lingkungan antrian yang dilampirkan ke antrian tertentu, dan lingkungan langkah pekerjaan yang ditentukan dalam templat pekerjaan.

Ada empat jenis tindakan sesi:

- `syncInputJobAttachments`— Mengunduh lampiran pekerjaan input ke pekerja.
- `envEnter`— Melakukan `onEnter` tindakan untuk suatu lingkungan.

- `taskRun`— Melakukan `onRun` tindakan untuk suatu tugas.
- `envExit`— Melakukan `onExit` tindakan untuk suatu lingkungan.

Template pekerjaan berikut memiliki lingkungan langkah. Ini memiliki `onEnter` definisi untuk mengatur lingkungan langkah, `onRun` definisi yang mendefinisikan tugas yang akan dijalankan, dan `onExit` definisi untuk meruntuhkan lingkungan langkah. Sesi yang dibuat untuk pekerjaan ini akan mencakup `envEnter` tindakan, satu atau lebih `taskRun` tindakan, dan kemudian `envExit` tindakan.

```
name: Sample Job with Maya Environment
specificationVersion: jobtemplate-2023-09
steps:
- name: Maya Step
  stepEnvironments:
  - name: Maya
    description: Runs Maya in the background.
    script:
      embeddedFiles:
      - name: initData
        filename: init-data.yaml
        type: TEXT
        data: |
          scene_file: MyAwesomeSceneFile
          renderer: arnold
          camera: persp
    actions:
      onEnter:
        command: MayaAdaptor
        args:
        - daemon
        - start
        - --init-data
        - file://{{Env.File.initData}}
      onExit:
        command: MayaAdaptor
        args:
        - daemon
        - stop
  parameterSpace:
    taskParameterDefinitions:
    - name: Frame
      range: 1-5
      type: INT
```

```
script:
  embeddedFiles:
  - name: runData
    filename: run-data.yaml
    type: TEXT
    data: |
      frame: {{Task.Param.Frame}}
actions:
  onRun:
    command: MayaAdaptor
    args:
    - daemon
    - run
    - --run-data
    - file//{{ Task.File.runData }}
```

## Ketergantungan langkah

Deadline Cloud mendukung mendefinisikan dependensi antar langkah sehingga satu langkah menunggu hingga langkah lain selesai sebelum memulai. Anda dapat menentukan lebih dari satu ketergantungan untuk satu langkah. Langkah dengan ketergantungan tidak dijadwalkan sampai semua dependensinya selesai.

Jika template pekerjaan mendefinisikan dependensi melingkar, pekerjaan ditolak dan status pekerjaan disetel ke. `CREATE_FAILED`

Template pekerjaan berikut membuat pekerjaan dengan dua langkah. StepBtergantung padaStepA. StepBhanya berjalan setelah StepA selesai dengan sukses.

Setelah pekerjaan dibuat, StepA berada di `READY` negara bagian dan StepB berada di `PENDING` negara bagian. Setelah StepA selesai, StepB pindah ke `READY` negara bagian. Jika StepA gagal, atau StepA jika dibatalkan, StepB pindah ke `CANCELED` negara bagian.

Anda dapat mengatur ketergantungan pada beberapa langkah. Misalnya, jika StepC tergantung pada keduanya StepA danStepB, StepC tidak akan dimulai sampai dua langkah lainnya selesai.

```
name: Step-Step Dependency Test
specificationVersion: 'jobtemplate-2023-09'
steps:
- name: A
  script:
    actions:
```

```
    onRun:
      command: bash
      args: ['{{ Task.File.run }}']
  embeddedFiles:
  - name: run
    type: TEXT
    data: |
      #!/bin/env bash

      set -euo pipefail

      sleep 1
      echo Task A Done!
- name: B
  dependencies:
  - dependsOn: A # This means Step B depends on Step A
  script:
    actions:
      onRun:
        command: bash
        args: ['{{ Task.File.run }}']
  embeddedFiles:
  - name: run
    type: TEXT
    data: |
      #!/bin/env bash

      set -euo pipefail

      sleep 1
      echo Task B Done!
```

## Status pekerjaan di Deadline Cloud CLI

Topik ini menjelaskan cara menggunakan antarmuka baris perintah AWS Deadline Cloud (Deadline CloudCLI) untuk melihat status pekerjaan atau langkah. Jika Anda ingin menggunakan monitor Deadline Cloud untuk melihat status pekerjaan atau langkah, lihat [Melihat dan mengelola pekerjaan, langkah, dan tugas di Deadline Cloud](#).

Anda dapat melihat status pekerjaan menggunakan CLI perintah `deadline job get --job-id` Deadline Cloud. Respons terhadap perintah termasuk status pekerjaan atau langkah dan jumlah tugas di setiap status pemrosesan.

Ketika Anda pertama kali mengirimkan pekerjaan, statusnya adalah `CREATE_IN_PROGRESS`. Jika pekerjaan melewati pemeriksaan validasi, statusnya berubah menjadi `CREATE_COMPLETE`. Jika tidak, status berubah menjadi `CREATE_FAILED`.

Beberapa kemungkinan alasan bahwa pekerjaan dapat gagal dalam pemeriksaan validasi meliputi:

- Template pekerjaan tidak mengikuti spesifikasi OpenJD.
- Pekerjaan itu mengandung terlalu banyak langkah.
- Pekerjaan itu mengandung terlalu banyak tugas total.

Untuk melihat kuota untuk jumlah maksimum langkah dan tugas dalam suatu pekerjaan, gunakan konsol Service Quotas. Untuk informasi selengkapnya, lihat [Kuota untuk Deadline Cloud](#).

Mungkin juga ada kesalahan layanan internal yang mencegah pekerjaan dibuat. Jika ini terjadi, kode status pekerjaan adalah `INTERNAL_ERROR` dan bidang pesan status memberikan penjelasan yang lebih rinci.

Gunakan CLI perintah Deadline Cloud berikut untuk melihat detail pekerjaan. Dalam contoh berikut, ganti *JobId* dengan informasi Anda sendiri:

```
deadline job get --job-id jobId
```

Tanggapan dari `deadline job get` perintah tersebut adalah sebagai berikut:

```
jobId: jobId
name: Sample Job
lifecycleStatus: CREATE_COMPLETE
lifecycleStatusMessage: Job creation completed successfully
priority: 50
createdAt: 2024-03-26 18:11:19.065000+00:00
createdBy: Test User
startedAt: 2024-03-26 18:12:50.710000+00:00
taskRunStatus: STARTING
taskRunStatusCounts:
  PENDING: 0
  READY: 5
  RUNNING: 0
  ASSIGNED: 0
  STARTING: 0
```

```
SCHEDULED: 0
INTERRUPTING: 0
SUSPENDED: 0
CANCELED: 0
FAILED: 0
SUCCEEDED: 0
NOT_COMPATIBLE: 0
maxFailedTasksCount: 100
maxRetriesPerTask: 5
```

Setiap tugas dalam pekerjaan atau langkah memiliki status. Status tugas digabungkan untuk memberikan status keseluruhan untuk pekerjaan dan langkah. Jumlah tugas di setiap negara bagian dilaporkan di `taskRunStatusCounts` bidang respons.

Status pekerjaan atau langkah tergantung pada status tugasnya. Status ditentukan oleh tugas-tugas yang memiliki status ini, secara berurutan. Status langkah ditentukan sama dengan status pekerjaan.

Daftar berikut menjelaskan status:

#### NOT\_COMPATIBLE

Pekerjaan itu tidak kompatibel dengan pertanian karena tidak ada armada yang dapat menyelesaikan salah satu tugas dalam pekerjaan itu.

#### RUNNING

Satu atau lebih pekerja menjalankan tugas dari pekerjaan. Selama setidaknya ada satu tugas yang berjalan, pekerjaan itu ditandai `RUNNING`.

#### ASSIGNED

Satu atau lebih pekerja diberi tugas dalam pekerjaan sebagai tindakan mereka selanjutnya. Lingkungan, jika ada, sudah diatur.

#### STARTING

Satu atau lebih pekerja sedang menyiapkan lingkungan untuk menjalankan tugas.

#### SCHEDULED

Tugas untuk pekerjaan dijadwalkan pada satu atau lebih pekerja sebagai tindakan pekerja selanjutnya.

#### READY

Setidaknya satu tugas untuk pekerjaan itu siap diproses.

## INTERRUPTING

Setidaknya satu tugas dalam pekerjaan sedang terganggu. Gangguan dapat terjadi ketika Anda memperbarui status pekerjaan secara manual. Ini juga dapat terjadi sebagai respons terhadap gangguan karena perubahan harga Spot Amazon Elastic Compute Cloud EC2 (Amazon).

## FAILED

Satu atau lebih tugas dalam pekerjaan itu tidak berhasil diselesaikan.

## CANCELED

Satu atau lebih tugas dalam pekerjaan telah dibatalkan.

## SUSPENDED

Setidaknya satu tugas dalam pekerjaan telah ditangguhkan.

## PENDING

Tugas dalam pekerjaan sedang menunggu ketersediaan sumber daya lain.

## SUCCEEDED

Semua tugas dalam pekerjaan berhasil diproses.

## Memodifikasi pekerjaan di Deadline Cloud

Anda dapat menggunakan update perintah AWS Command Line Interface (AWS CLI) berikut untuk mengubah konfigurasi pekerjaan, atau untuk menetapkan status target pekerjaan, langkah, atau tugas:

- `aws deadline update-job`
- `aws deadline update-step`
- `aws deadline update-task`

Dalam contoh update perintah berikut, ganti masing-masing *placeholder masukan pengguna* dengan informasi Anda sendiri.

Anda juga dapat menggunakan monitor Deadline Cloud untuk memodifikasi konfigurasi pekerjaan. Untuk informasi selengkapnya, lihat [Melihat dan mengelola pekerjaan, langkah, dan tugas di Deadline Cloud](#).

### Example — Meminta pekerjaan

Semua tugas dalam pekerjaan beralih ke READY status, kecuali ada dependensi langkah. Langkah-langkah dengan dependensi beralih ke salah satu READY atau PENDING saat dipulihkan.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--target-task-run-status PENDING
```

### Example — Batalkan pekerjaan

Semua tugas dalam pekerjaan yang tidak memiliki status SUCCEEDED atau FAILED ditandai CANCELED.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--target-task-run-status CANCELED
```

### Example — Tandai pekerjaan gagal

Semua tugas dalam pekerjaan yang memiliki status SUCCEEDED dibiarkan tidak berubah. Semua tugas lainnya ditandai FAILED.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--target-task-run-status FAILED
```

### Example — Tandai pekerjaan yang sukses

Semua tugas dalam pekerjaan pindah ke SUCCEEDED negara bagian.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--target-task-run-status SUCCEEDED
```

```
--target-task-run-status SUCCEEDED
```

### Example — Menangguhkan pekerjaan

Tugas dalam pekerjaan diSUCCEEDED,CANCELED, atau FAILED negara bagian tidak berubah. Semua tugas lainnya ditandaiSUSPENDED.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--target-task-run-status SUSPENDED
```

### Example — Mengubah prioritas pekerjaan

Memperbarui prioritas pekerjaan untuk mengubah urutan yang dijadwalkan. Pekerjaan prioritas yang lebih tinggi umumnya dijadwalkan terlebih dahulu.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--priority 100
```

### Example — Ubah jumlah tugas gagal yang diizinkan

Memperbarui jumlah maksimum tugas yang gagal yang dapat dimiliki pekerjaan sebelum tugas yang tersisa dibatalkan.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--max-failed-tasks-count 200
```

### Example — Ubah jumlah percobaan ulang tugas yang diizinkan

Memperbarui jumlah maksimum percobaan ulang untuk tugas sebelum tugas gagal. Tugas yang telah mencapai jumlah percobaan ulang maksimum tidak dapat diulang sampai nilai ini meningkat.

```
aws deadline update-job \  

```

```
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--max-retries-per-task 10
```

### Example — Arsipkan pekerjaan

Memperbarui status siklus hidup pekerjaan ke ARCHIVED. Pekerjaan yang diarsipkan tidak dapat dijadwalkan atau diubah. Anda hanya dapat mengarsipkan pekerjaan yang ada di FAILED, CANCELED, SUCCEEDED, atau SUSPENDED.

```
aws deadline update-job \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--lifecycle-status ARCHIVED
```

### Example — Meminta satu langkah

Semua tugas di langkah beralih ke READY status, kecuali ada dependensi langkah. Tugas dalam langkah-langkah dengan dependensi beralih ke salah satu READY atau PENDING, dan tugas dipulihkan.

```
aws deadline update-step \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--target-task-run-status PENDING
```

### Example — Batalkan langkah

Semua tugas dalam langkah yang tidak memiliki status SUCCEEDED atau FAILED ditandai CANCELED.

```
aws deadline update-step \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--target-task-run-status CANCELED
```

## Example — Tandai langkah gagal

Semua tugas dalam langkah yang memiliki status SUCCEEDED dibiarkan tidak berubah. Semua tugas lainnya ditandai FAILED.

```
aws deadline update-step \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--target-task-run-status FAILED
```

## Example — Tandai langkah sukses

Semua tugas dalam langkah ditandai SUCCEEDED.

```
aws deadline update-step \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--target-task-run-status SUCCEEDED
```

## Example — Tangguhkan satu langkah

Tugas dalam langkah di SUCCEEDED, CANCELED, atau FAILED status tidak berubah. Semua tugas lainnya ditandai SUSPENDED.

```
aws deadline update-step \  
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--target-task-run-status SUSPENDED
```

## Example — Mengubah status tugas

Saat Anda menggunakan CLI perintah update-task Deadline Cloud, tugas beralih ke status yang ditentukan.

```
aws deadline update-task \  

```

```
--farm-id farmID \  
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--task-id taskID \  
--target-task-run-status SUCCEEDED | SUSPENDED | CANCELED | FAILED | PENDING
```

## Bagaimana Deadline Cloud memproses pekerjaan

Untuk memproses pekerjaan, AWS Deadline Cloud menggunakan template pekerjaan Open Job Description (OpenJD) untuk menentukan sumber daya yang dibutuhkan. Deadline Cloud memilih pekerja yang cocok untuk satu langkah dari armada yang terkait dengan antrian Anda. Pekerja yang dipilih memenuhi semua atribut kemampuan yang diperlukan untuk langkah tersebut.

Selanjutnya, Deadline Cloud mengirimkan instruksi kepada pekerja untuk mengatur sesi untuk langkah tersebut. Perangkat lunak yang diperlukan untuk langkah tersebut harus tersedia pada instance pekerja agar pekerjaan dapat dijalankan. Layanan ini dapat membuka sesi pada beberapa pekerja jika pengaturan penskalaan untuk armada memiliki kapasitas.

Anda dapat mengatur perangkat lunak dalam Amazon Machine Image (AMI), atau pekerja Anda dapat memuat perangkat lunak saat runtime dari repositori atau manajer paket. Anda dapat menggunakan lingkungan antrian, pekerjaan, atau langkah untuk menyebarkan perangkat lunak yang Anda inginkan.

Layanan Deadline Cloud menggunakan template OpenJD untuk menentukan langkah-langkah yang diperlukan untuk pekerjaan itu, dan tugas yang diperlukan untuk setiap langkah. Beberapa langkah memiliki ketergantungan pada langkah lain, jadi Deadline Cloud menentukan urutan untuk menyelesaikan langkah-langkah tersebut. Kemudian, Deadline Cloud mengirimkan tugas untuk setiap langkah ke pekerja untuk diproses. Ketika tugas selesai, layanan mengirimkan tugas lain dalam sesi yang sama, atau pekerja dapat memulai sesi baru.

Anda dapat melacak kemajuan pekerjaan di monitor Deadline Cloud, antarmuka baris perintah Deadline Cloud (Deadline CloudCLI) atau file. AWS CLI Untuk informasi selengkapnya tentang menggunakan monitor, lihat [Menggunakan monitor Deadline Cloud](#). Untuk informasi selengkapnya tentang penggunaan Deadline CloudCLI, lihat [Status pekerjaan di Deadline Cloud CLI](#).

Setelah semua tugas di setiap langkah selesai, pekerjaan selesai dan output siap diunduh ke workstation Anda. Bahkan jika pekerjaan tidak selesai, output dari setiap langkah dan tugas yang selesai tersedia untuk diunduh.

Deadline Cloud menghapus pekerjaan 120 hari setelah diserahkan. Ketika pekerjaan dihapus, semua langkah dan tugas yang terkait dengan pekerjaan juga dihapus. Jika Anda perlu menjalankan kembali pekerjaan, kirimkan template OpenJD untuk pekerjaan itu lagi.

## Memecahkan masalah Deadline pekerjaan Cloud

Untuk informasi tentang masalah umum dengan pekerjaan di AWS Deadline Cloud, lihat topik berikut.

### Topik

- [Mengapa membuat pekerjaan saya gagal?](#)
- [Mengapa pekerjaan saya tidak kompatibel?](#)
- [Mengapa pekerjaan saya terjebak dalam siap?](#)
- [Mengapa pekerjaan saya gagal?](#)
- [Mengapa langkah saya tertunda?](#)

## Mengapa membuat pekerjaan saya gagal?

Beberapa kemungkinan alasan bahwa pekerjaan dapat gagal dalam pemeriksaan validasi meliputi:

- Template pekerjaan tidak mengikuti spesifikasi OpenJD.
- Pekerjaan itu mengandung terlalu banyak langkah.
- Pekerjaan itu mengandung terlalu banyak tugas total.
- Ada kesalahan layanan internal yang mencegah pekerjaan dibuat.

Untuk melihat kuota untuk jumlah maksimum langkah dan tugas dalam suatu pekerjaan, gunakan konsol Service Quotas. Untuk informasi selengkapnya, lihat [Kuota untuk Deadline Cloud](#).

## Mengapa pekerjaan saya tidak kompatibel?

Alasan umum bahwa pekerjaan tidak kompatibel dengan antrian termasuk yang berikut:

- Tidak ada armada yang terkait dengan antrian tempat pekerjaan itu diserahkan. Buka monitor Deadline Cloud, dan periksa apakah antrian memiliki armada terkait. Untuk informasi selengkapnya tentang cara melihat antrian, lihat [Lihat detail antrian dan armada di Deadline Cloud](#)
- Pekerjaan tersebut memiliki persyaratan tuan rumah yang tidak dipenuhi oleh armada mana pun yang terkait dengan antrian. Untuk memeriksanya, bandingkan `hostRequirements` entri dalam

templat pekerjaan dengan konfigurasi armada di peternakan Anda. Pastikan salah satu armada memenuhi persyaratan tuan rumah. Untuk informasi selengkapnya tentang kompatibilitas armada, lihat [Tentukan kompatibilitas armada](#). Untuk melihat konfigurasi armada, lihat [Lihat detail antrian dan armada di Deadline Cloud](#).

## Mengapa pekerjaan saya terjebak dalam siap?

Kemungkinan alasan pekerjaan Anda tampak macet di READY negara bagian termasuk yang berikut:

- Jumlah pekerja maksimum untuk armada yang terkait dengan antrian diatur ke nol. Untuk memeriksa, lihat [Lihat detail antrian dan armada di Deadline Cloud](#).
- Ada pekerjaan prioritas yang lebih tinggi dalam antrian. Untuk memeriksa, lihat [Lihat detail antrian dan armada di Deadline Cloud](#).
- Untuk armada yang dikelola pelanggan, periksa konfigurasi penskalaan otomatis. Untuk informasi selengkapnya, lihat [Skala otomatis EC2 armada Amazon Anda dengan fitur rekomendasi skala Deadline Cloud](#).

## Mengapa pekerjaan saya gagal?

Pekerjaan bisa gagal karena berbagai alasan. Untuk mencari masalah, buka monitor Deadline Cloud dan pilih pekerjaan yang gagal. Pilih tugas yang gagal dan kemudian lihat log untuk tugas tersebut. Untuk petunjuk, silakan lihat [Lihat log di Deadline Cloud](#).

- Jika Anda melihat kesalahan lisensi atau jika Anda mendapatkan tanda air yang terjadi karena perangkat lunak tidak memiliki lisensi yang valid, pastikan pekerja dapat terhubung ke server lisensi yang diperlukan. Untuk informasi selengkapnya, lihat [Connect armada yang dikelola pelanggan ke titik akhir lisensi](#).

## Mengapa langkah saya tertunda?

Langkah-langkah mungkin tetap dalam PENDING keadaan ketika satu atau lebih dependensi mereka tidak lengkap. Anda dapat memeriksa status dependensi menggunakan monitor Deadline Cloud. Untuk instruksi, lihat [Lihat langkah di Deadline Cloud](#).

# Penyimpanan file untuk Deadline Cloud

Pekerja harus memiliki akses ke lokasi penyimpanan yang berisi file input yang diperlukan untuk memproses pekerjaan, dan ke lokasi yang menyimpan output. AWS Deadline Cloud menyediakan dua opsi untuk lokasi penyimpanan:

- Dengan lampiran pekerjaan, Deadline Cloud mentransfer file input dan output untuk pekerjaan Anda bolak-balik antara workstation dan pekerja Deadline Cloud. Untuk mengaktifkan transfer file, Deadline Cloud menggunakan bucket Amazon Simple Storage Service (Amazon S3) di bucket Akun AWS.

Saat Anda menggunakan lampiran pekerjaan dengan armada yang dikelola layanan, Anda dapat menyiapkan sistem file virtual (VFS) di jaringan pribadi virtual (). VPN Kemudian pekerja dapat memuat file hanya bila diperlukan.

- Dengan penyimpanan bersama, Anda menggunakan berbagi file dengan sistem operasi Anda untuk menyediakan akses ke file.

Saat Anda menggunakan penyimpanan bersama lintas platform, Anda dapat membuat profil penyimpanan sehingga pekerja dapat memetakan jalur ke file di antara dua sistem operasi yang berbeda.

## Topik

- [Lampiran Job di Deadline Cloud](#)
- [Penyimpanan bersama di Deadline Cloud](#)

## Lampiran Job di Deadline Cloud

Lampiran Job memungkinkan Anda untuk mentransfer file bolak-balik antara workstation Anda dan AWS Batas Waktu Cloud. Dengan lampiran pekerjaan, Anda tidak perlu menyiapkan bucket Amazon S3 secara manual untuk file Anda. Sebagai gantinya, saat membuat antrian dengan konsol Deadline Cloud, Anda memilih bucket untuk lampiran pekerjaan Anda.

Pertama kali Anda mengirimkan pekerjaan ke Deadline Cloud, semua file untuk pekerjaan tersebut ditransfer ke Deadline Cloud. Untuk pengiriman berikutnya, hanya file yang telah berubah ditransfer, menghemat waktu dan bandwidth.

Setelah pemrosesan selesai, Anda dapat mengunduh hasilnya dari halaman detail pekerjaan, atau dengan menggunakan CLI `deadline job download-output` perintah Deadline Cloud.

Anda dapat menggunakan bucket S3 yang sama untuk beberapa antrian. Tetapkan awalan root yang berbeda untuk setiap antrian untuk mengatur lampiran di bucket.

Saat Anda membuat antrian dengan konsol, Anda dapat memilih yang sudah ada AWS Identity and Access Management (IAM) peran atau Anda dapat meminta konsol membuat peran baru. Jika konsol membuat peran, konsol akan menetapkan izin untuk mengakses bucket yang ditentukan untuk antrian. Jika memilih peran yang ada, Anda harus memberikan izin peran untuk mengakses bucket S3.

## Enkripsi untuk bucket S3 lampiran pekerjaan

File lampiran Job secara otomatis dienkripsi di bucket S3 Anda secara default. Pendekatan ini membantu mengamankan informasi Anda dari akses yang tidak sah. Anda tidak perlu melakukan apa pun agar file Anda dienkripsi dengan kunci yang disediakan oleh Deadline Cloud. Untuk informasi selengkapnya, lihat [Amazon S3 sekarang secara otomatis mengenkripsi semua objek baru di Panduan Pengguna](#) Amazon S3.

Anda dapat menggunakan pelanggan Anda sendiri yang dikelola AWS Key Management Service kunci untuk mengenkripsi bucket S3 yang berisi lampiran pekerjaan Anda. Untuk melakukannya, Anda harus mengubah IAM peran antrian yang terkait dengan bucket untuk mengizinkan akses ke AWS KMS key.

Untuk membuka editor IAM kebijakan untuk peran antrian

1. Masuk ke AWS Management Console dan buka [konsol](#) Deadline Cloud. Dari halaman utama, di bagian Memulai, pilih Lihat peternakan.
2. Dari daftar peternakan, pilih peternakan yang berisi antrian untuk dimodifikasi.
3. Dari daftar antrian, pilih antrian yang akan dimodifikasi.
4. Di bagian Detail antrian, pilih peran Layanan untuk membuka IAM konsol untuk peran layanan.

Selanjutnya, selesaikan prosedur berikut.

Untuk memperbarui kebijakan peran dengan izin untuk AWS KMS

1. Dari daftar kebijakan Izin, pilih kebijakan untuk peran tersebut.
2. Di bagian Izin yang ditentukan di bagian kebijakan ini, pilih Edit.

3. Pilih Tambahkan pernyataan baru.
4. Salin dan tempel kebijakan berikut ke editor. Ubah *Wilayah*, *accountID*, dan *keyID* untuk nilai-nilai Anda sendiri.

```
{
  "Effect": "Allow",
  "Action": [
    "kms:Decrypt",
    "kms:DescribeKey",
    "kms:GenerateDataKey"
  ],
  "Resource": [
    "arn:aws:kms:Region:accountID:key/keyID"
  ]
}
```

5. Pilih Berikutnya.
6. Tinjau perubahan kebijakan, lalu setelah puas, pilih Simpan perubahan.

## Mengelola lampiran pekerjaan di bucket S3

Deadline Cloud menyimpan file lampiran pekerjaan yang diperlukan untuk pekerjaan Anda di bucket S3. File-file ini terakumulasi dari waktu ke waktu, yang menyebabkan peningkatan biaya Amazon S3. Untuk mengurangi biaya, Anda dapat menerapkan konfigurasi Siklus Hidup S3 ke bucket S3 Anda. Konfigurasi ini dapat secara otomatis menghapus file di bucket. Karena bucket S3 ada di akun Anda, Anda dapat memilih untuk mengubah atau menghapus konfigurasi Siklus Hidup S3 kapan saja. Untuk informasi selengkapnya, lihat [Contoh konfigurasi Siklus Hidup S3 di Panduan Pengguna Amazon S3](#).

Untuk solusi manajemen bucket S3 yang lebih granular, Anda dapat mengatur Akun AWS untuk kedaluwarsa objek dalam ember S3 berdasarkan terakhir kali mereka diakses. Untuk informasi selengkapnya, lihat [Objek Amazon S3 kedaluwarsa berdasarkan tanggal akses terakhir untuk mengurangi biaya](#) AWS Blog Arsitektur.

## Batas waktu Cloud sistem file virtual

Dukungan sistem file virtual untuk lampiran pekerjaan di AWS Deadline Cloud memungkinkan perangkat lunak klien pada pekerja untuk berkomunikasi langsung dengan Amazon Simple Storage Service. Pekerja dapat memuat file hanya bila diperlukan alih-alih mengunduh semua file sebelum

diproses. File disimpan secara lokal. Pendekatan ini menghindari pengunduhan aset yang digunakan lebih dari sekali beberapa kali. Semua file dihapus setelah pekerjaan selesai.

- Sistem file virtual memberikan peningkatan kinerja yang signifikan untuk profil pekerjaan tertentu. Secara umum, himpunan bagian yang lebih kecil dari total file dengan armada pekerja yang lebih besar menunjukkan manfaat paling besar. Sejumlah kecil file dengan lebih sedikit pekerja memiliki waktu pemrosesan yang kira-kira setara.
- Dukungan sistem file virtual hanya tersedia untuk Linux pekerja di armada yang dikelola layanan.
- Sistem file virtual Deadline Cloud mendukung operasi berikut, tetapi tidak POSIX sesuai:
  - `Filecreate,,delete,open,close,read,,write,append,truncate,,rename,move,copy,stat,,fsync` dan `falloc`
  - Direktori `createdelete,rename,,move,copy`, dan `stat`
- Sistem file virtual dirancang untuk mengurangi transfer data dan meningkatkan kinerja ketika tugas Anda hanya mengakses sebagian dari kumpulan data besar, dan tidak dioptimalkan untuk semua beban kerja. Anda harus menguji beban kerja Anda sebelum menjalankan pekerjaan produksi.

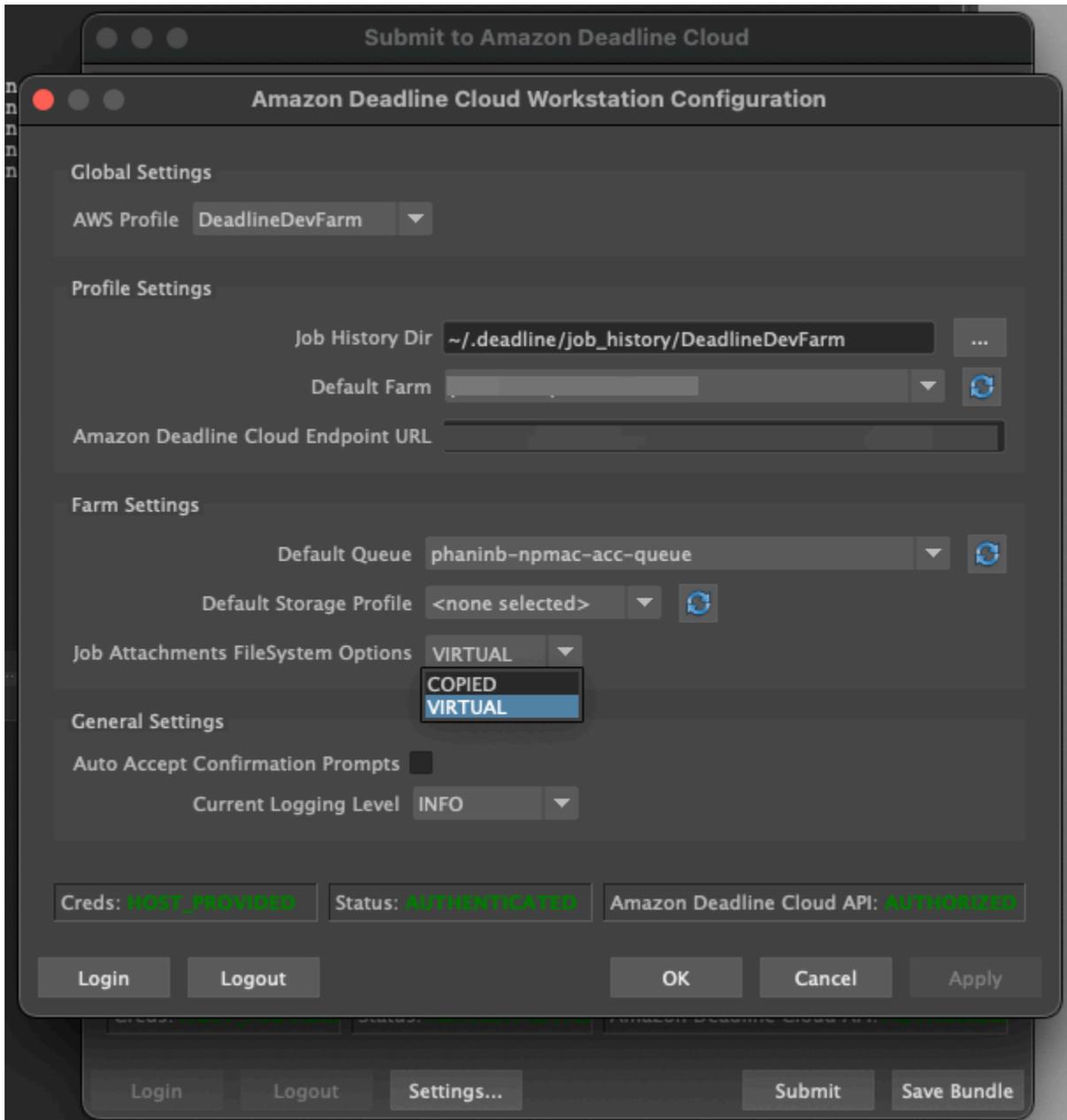
## Aktifkan VFS dukungan

Dukungan sistem file virtual (VFS) diaktifkan untuk setiap pekerjaan. Pekerjaan kembali ke kerangka kerja lampiran pekerjaan default dalam kasus ini:

- Profil instance pekerja tidak mendukung sistem file virtual.
- Masalah mencegah peluncuran proses sistem file virtual.
- Sistem file virtual tidak dapat dipasang.

Untuk mengaktifkan dukungan sistem file virtual menggunakan submitter

1. Saat mengirimkan pekerjaan, pilih tombol Pengaturan untuk membuka AWS Panel konfigurasi workstation Cloud batas waktu.
2. Dari tarik-turun opsi sistem file lampiran Job, pilih. VIRTUAL



3. Untuk menyimpan perubahan Anda, pilih OK.

Untuk mengaktifkan dukungan sistem file virtual menggunakan AWS CLI

- Gunakan perintah berikut saat Anda mengirimkan pekerjaan yang disimpan:

```
deadline bundle submit-job --job-attachments-file-system VIRTUAL
```

Untuk memverifikasi bahwa sistem file virtual berhasil diluncurkan untuk pekerjaan tertentu, tinjau log Anda di Amazon CloudWatch Logs. Cari pesan berikut:

```
Using mount_point mount_point  
Launching vfs with command command  
Launched vfs as pid PID number
```

Jika log berisi pesan berikut, dukungan sistem file virtual dinonaktifkan:

```
Virtual File System not found, falling back to COPIED for JobAttachmentsFileSystem.
```

## Memecahkan masalah dukungan sistem file virtual

Anda dapat melihat log untuk sistem file virtual Anda menggunakan monitor Deadline Cloud. Untuk petunjuk, silakan lihat [Lihat log di Deadline Cloud](#).

Log sistem file virtual juga dikirim ke grup CloudWatch Log yang terkait dengan antrian yang dibagikan dengan output agen pekerja.

## Penyimpanan bersama di Deadline Cloud

Untuk menggunakan penyimpanan bersama, pekerja menggunakan sistem berbagi file sistem operasi untuk akses ke ruang penyimpanan bersama untuk input dan output pekerjaan Anda.

Metode sebenarnya yang Anda gunakan untuk berbagi file tergantung pada sistem operasi Anda dan cara Anda menerapkan penyimpanan bersama di jaringan Anda. Anda bertanggung jawab atas cara Anda mengonfigurasi berbagi file dan memastikannya memenuhi kebutuhan Anda.

Jika Anda menggunakan solusi berbagi file lintas sistem, Anda dapat menggunakan profil penyimpanan untuk memetakan lokasi file di antara Linux and Windows sistem file.

## Profil penyimpanan di Deadline Cloud

Dengan profil penyimpanan, Anda dapat mengatur peternakan menggunakan penyimpanan bersama lintas platform. Profil penyimpanan memetakan jalur di seluruh sistem operasi untuk pekerjaan yang diproses pada pekerja dengan sistem operasi yang berbeda dari workstation tempat mereka dikirim.

Profil penyimpanan diperlukan saat Anda menggunakan armada yang dikelola pelanggan dengan campuran sistem operasi antara workstation dan pekerja. Profil penyimpanan tidak didukung pada armada yang dikelola layanan.

Setelah Anda membuat profil penyimpanan, Anda harus memberikan akses ke antrian dan armada yang menggunakan profil.

Untuk informasi selengkapnya, lihat [Profil penyimpanan dan pemetaan jalur](#) di AWS Panduan Pengembang Cloud Batas Waktu.

Untuk membuat profil penyimpanan

1. Buka [konsol Deadline Cloud](#).
2. Dari Mulai, pilih Go to Deadline Cloud dashboard.
3. Pilih peternakan, lalu pilih tab Profil penyimpanan.
4. Pilih Buat profil penyimpanan.
5. Pilih sistem operasi dari dropdown.
6. Berikan nama untuk profil. Nama yang jelas membantu Anda memilih profil penyimpanan yang akan digunakan saat mengirimkan pekerjaan.
7. Untuk nama Path, masukkan lokasi root data pekerjaan di workstation tempat Anda mengirimkan lowongan.
8. Pilih jenis Penyimpanan:
  - Lokal mengacu pada lokasi file yang tidak dibagi antara pekerja dan workstation. Mereka diunggah sebagai lampiran pekerjaan.
  - Shared mengacu pada penyimpanan yang dibagi antara pekerja dan workstation. File dalam penyimpanan bersama tidak diunggah sebagai lampiran pekerjaan.
9. Menyediakan jalur lokasi sistem File. Ini adalah direktori root untuk data pekerjaan Anda.
10. Pilih Buat.

Setelah Anda membuat profil penyimpanan, Anda harus mengubah antrian dan armada yang dikelola pelanggan untuk menggunakan profil baru. Untuk mengizinkan akses ke profil penyimpanan, gunakan prosedur berikut setelah Anda menyelesaikan prosedur sebelumnya.

Untuk memungkinkan antrian dan armada yang dikelola pelanggan menggunakan profil penyimpanan

1. Pilih tab Antrian atau Armada.
2. Pilih antrian atau armada untuk dimodifikasi.
3. Untuk mengubah antrian, pilih tab Profil penyimpanan yang diizinkan.

Untuk memodifikasi armada, pilih tab Profil penyimpanan.

4. Pilih Ubah profil penyimpanan.
5. Pilih profil penyimpanan untuk mengizinkan, dan lokasi sistem file dari profil itu.
6. Pilih Simpan perubahan.

# Mengelola anggaran dan penggunaan untuk Deadline Cloud

Bagian AWS Deadline Cloud budget manager dan usage explorer adalah alat manajemen biaya yang menyediakan perkiraan biaya penggunaan Deadline Cloud berdasarkan informasi yang tersedia tentang variabel biaya. Alat manajemen biaya tidak menjamin jumlah yang terutang untuk penggunaan Deadline Cloud Anda yang sebenarnya dan lainnya AWS layanan.

Untuk membantu Anda mengelola biaya untuk Deadline Cloud, Anda dapat menggunakan fitur berikut:

- **Manajer anggaran** — Dengan manajer anggaran Deadline Cloud, Anda dapat membuat dan mengedit anggaran untuk membantu mengelola biaya proyek.
- **Penjelajah penggunaan** — Dengan penjelajah penggunaan Deadline Cloud, Anda dapat melihat berapa banyak AWS sumber daya digunakan dan perkiraan biaya untuk sumber daya tersebut.

## Asumsi biaya

Perhitungan dasar yang digunakan oleh alat manajemen biaya Deadline Cloud adalah:

```
Cost per job =  
  (CMF run time x CMF compute rate) +  
  (SMF run time x SMF compute rate) +  
  (License run time x license rate)
```

- Run time adalah jumlah dari semua tugas dalam suatu pekerjaan, dari waktu mulai hingga waktu akhir.
- Tingkat komputasi ditentukan oleh [AWS Tenggat waktu harga Cloud](#) untuk armada yang dikelola layanan. Untuk armada yang dikelola pelanggan, tingkat komputasi diperkirakan \$1 per jam pekerja.
- Tarif lisensi ditentukan oleh harga lisensi basis Deadline Cloud dan hanya tersedia untuk armada yang dikelola layanan. Tingkatan tambahan tidak termasuk. Untuk informasi selengkapnya tentang harga lisensi, lihat [AWS Tenggat waktu harga Cloud](#).

Perkiraan biaya dari alat manajemen biaya Deadline Cloud dapat bervariasi dari biaya aktual Anda karena sejumlah alasan. Alasan umum meliputi:

- Sumber daya milik pelanggan dan harga mereka. Anda dapat memilih untuk membawa sumber daya Anda sendiri, baik dari AWS atau secara eksternal dari on-premise atau penyedia cloud lainnya. Biaya aktual dari sumber daya ini tidak dihitung.
- Biaya pekerja menganggur. Untuk armada dengan jumlah instans minimum lebih besar dari nol, pekerja idle tidak diperhitungkan dalam perhitungan.
- Kredit promosi, diskon, dan perjanjian harga khusus. Alat manajemen biaya tidak memperhitungkan kredit promosi, perjanjian harga pribadi, atau diskon lainnya. Anda mungkin memenuhi syarat untuk diskon lain yang bukan bagian dari perkiraan.
- Penyimpanan aset. Penyimpanan aset tidak termasuk dalam perkiraan biaya dan penggunaan.
- Perubahan harga. AWS menawarkan pay-as-you-go harga untuk sebagian besar layanan. Harga dapat berubah seiring waktu. Alat manajemen biaya menggunakan up-to-date harga terbanyak sekutu publik yang tersedia, tetapi mungkin ada penundaan setelah perubahan.
- Pajak. Alat manajemen biaya tidak termasuk pajak yang diterapkan untuk pembelian layanan kami.
- Pembulatan. Alat manajemen biaya melakukan pembulatan matematis data penetapan harga.
- Mata uang. Perkiraan biaya dibuat dalam dolar AS. Nilai tukar global bervariasi dari waktu ke waktu. Jika Anda menerjemahkan perkiraan ke basis mata uang yang berbeda pada pertukaran saat ini, perubahan nilai tukar mempengaruhi perkiraan.
- Lisensi luar. Jika Anda memilih untuk menggunakan lisensi yang telah dibeli sebelumnya (membawa lisensi Anda sendiri), alat manajemen biaya Deadline Cloud tidak dapat memperhitungkan biaya ini.

## Menggunakan manajer anggaran Deadline Cloud

Manajer anggaran Deadline Cloud membantu Anda mengontrol pengeluaran untuk sumber daya tertentu, seperti antrian, armada, atau pertanian. Anda dapat membuat jumlah dan batasan anggaran, dan menetapkan tindakan otomatis untuk membantu mengurangi atau menghentikan pengeluaran tambahan terhadap anggaran.

Bagian berikut memberi Anda langkah-langkah untuk menggunakan manajer anggaran Deadline Cloud.

### Topik

- [Prasyarat](#)
- [Akses pengelola anggaran](#)
- [Buat anggaran](#)

- [Lihat anggaran](#)
- [Edit anggaran](#)
- [Nonaktifkan anggaran](#)

## Prasyarat

Untuk menggunakan manajer anggaran Deadline Cloud, Anda harus memiliki tingkat OWNER akses. Untuk memberikan OWNER izin, ikuti langkah-langkah di [Mengelola pengguna di Deadline Cloud](#).

## Akses pengelola anggaran

Untuk mengakses manajer anggaran Deadline Cloud, gunakan prosedur berikut.

1. Masuk ke AWS Management Console dan buka [konsol](#) Deadline Cloud.
2. Pilih Lihat peternakan.
3. Temukan peternakan yang ingin Anda dapatkan informasinya, lalu pilih Kelola pekerjaan. Monitor Deadline Cloud terbuka di tab baru.
4. Di monitor Deadline Cloud, di panel navigasi kiri, pilih Anggaran.

Halaman ringkasan pengelola anggaran menampilkan daftar anggaran aktif dan tidak aktif:

- Anggaran aktif melacak sumber daya yang dipilih (antrian).
- Anggaran tidak aktif telah kedaluwarsa atau dibatalkan oleh pengguna, dan tidak lagi melacak biaya terhadap batas anggaran ini.

Setelah Anda memilih anggaran, halaman ringkasan anggaran berisi informasi dasar tentang anggaran. Informasi yang diberikan meliputi nama anggaran, status, sumber daya, persentase yang tersisa, jumlah yang tersisa, total anggaran, tanggal mulai, dan tanggal akhir.

## Buat anggaran

Untuk membuat anggaran, gunakan prosedur berikut.

1. Jika Anda belum melakukannya, masuk ke AWS Management Console, buka [konsol](#) Cloud Deadline, pilih pertanian, lalu pilih Kelola pekerjaan.
2. Dari halaman Manajer anggaran, pilih Buat anggaran.

3. Di bagian detail, masukkan nama Anggaran untuk anggaran.
4. (Opsional) Di bidang deskripsi, masukkan deskripsi singkat dan jelas untuk anggaran.
5. Dari Sumber Daya, pilih dropdown Antrian untuk menemukan dan pilih antrian yang ingin Anda buat anggaran.
6. Untuk Periode, tetapkan tanggal mulai dan berakhirnya anggaran dengan menyelesaikan langkah-langkah berikut:
  - a. Untuk Tanggal mulai, masukkan tanggal pertama pelacakan anggaran dalam format YYYY/MM/DD, atau pilih ikon kalender dan pilih tanggal.

Tanggal mulai default adalah tanggal pembuatan anggaran.
  - b. Untuk Tanggal akhir, masukkan tanggal terakhir pelacakan anggaran dalam format YYYY/MM/DD atau pilih ikon kalender dan pilih tanggal.

Tanggal akhir default adalah 120 hari dari tanggal mulai.
7. Untuk jumlah Anggaran, masukkan jumlah dolar dari anggaran.
8. (Opsional) Kami menyarankan Anda membuat peringatan batas. Di bagian Batasi tindakan, Anda dapat menerapkan tindakan otomatis yang terjadi ketika jumlah tertentu tetap ada dalam anggaran. Caranya, lakukan langkah-langkah berikut:
  - a. Pilih Tambahkan tindakan baru.
  - b. Untuk jumlah yang tersisa, masukkan jumlah dolar yang Anda inginkan untuk memulai tindakan.
  - c. Di dropdown Action, pilih tindakan yang Anda inginkan. Tindakan meliputi:
    - Berhenti setelah menyelesaikan pekerjaan saat ini — Semua pekerjaan yang sedang berjalan saat jumlah ambang terpenuhi terus berjalan (dan mengeluarkan biaya) hingga selesai.
    - Segera berhenti bekerja - Semua pekerjaan dibatalkan segera ketika jumlah ambang batas terpenuhi.
  - d. Untuk membuat peringatan batas tambahan, pilih Tambahkan tindakan baru dan ulangi dua langkah sebelumnya.
9. Pilih Buat anggaran. Halaman pengelola anggaran muncul. Anggaran yang baru dibuat ditampilkan di tab Anggaran aktif.

## Lihat anggaran

Setelah Anda membuat anggaran, Anda dapat melihat anggaran di halaman Manajer anggaran. Dari sana, Anda dapat melihat jumlah total anggaran dan keseluruhan biaya yang dialokasikan untuk anggaran tertentu.

Untuk melihat anggaran, gunakan prosedur berikut.

1. Jika Anda belum melakukannya, masuk ke AWS Management Console, buka [konsol](#) Cloud Deadline, pilih pertanian, lalu pilih Kelola pekerjaan.
2. Pilih Anggaran dari panel navigasi sisi kiri. Halaman Manajer Anggaran muncul.
3. Untuk melihat anggaran aktif, pilih tab Anggaran aktif, dan pilih nama anggaran yang ingin Anda lihat. Halaman detail anggaran muncul.
4. Untuk melihat detail anggaran untuk anggaran kedaluwarsa, pilih tab Anggaran tidak aktif. Kemudian, pilih nama anggaran yang ingin Anda lihat. Halaman detail anggaran muncul.

## Edit anggaran

Anda dapat mengedit anggaran aktif apa pun. Untuk mengedit anggaran aktif, gunakan prosedur berikut.

1. Jika Anda belum melakukannya, masuk ke AWS Management Console, buka [konsol](#) Cloud Deadline, pilih pertanian, lalu pilih Kelola pekerjaan.
2. Dari halaman Manajer Anggaran, di tab Anggaran aktif, pilih tombol di sebelah anggaran yang ingin Anda edit.
3. Dari menu tarik-turun Tindakan, pilih Edit anggaran.
4. Buat perubahan yang Anda inginkan, lalu pilih Perbarui anggaran.

## Nonaktifkan anggaran

Anda dapat menonaktifkan anggaran aktif apa pun. Menonaktifkan anggaran mengubah statusnya dari Aktif menjadi Tidak Aktif. Ketika anggaran dinonaktifkan, itu tidak lagi melacak sumber daya ke jumlah anggaran itu.

Untuk menonaktifkan anggaran, gunakan prosedur berikut.

1. Jika Anda belum melakukannya, masuk ke AWS Management Console, buka [konsol](#) Cloud Deadline, pilih pertanian, lalu pilih Kelola pekerjaan.
2. Dari halaman Manajer anggaran, di tab Anggaran Aktif, pilih tombol di sebelah anggaran yang ingin Anda nonaktifkan.
3. Dari menu tarik-turun Tindakan, pilih Nonaktifkan anggaran. Dalam beberapa saat, anggaran yang dipilih akan berubah dari Aktif menjadi Tidak Aktif dan akan berpindah dari tab Anggaran Aktif ke tab Anggaran Tidak Aktif.

## Menggunakan penjelajah penggunaan Deadline Cloud

Dengan penjelajah penggunaan Deadline Cloud, Anda dapat melihat metrik real-time pada aktivitas yang terjadi di setiap farm. Anda dapat melihat biaya pertanian dengan variabel yang berbeda, seperti antrian, pekerjaan, produk lisensi, atau jenis instance. Pilih berbagai kerangka waktu untuk melihat penggunaan selama periode waktu tertentu, dan lihat tren penggunaan selama waktu. Anda juga dapat melihat rincian rinci dari titik data yang dipilih, memungkinkan untuk melihat lebih dekat ke metrik. Penggunaan dapat ditunjukkan berdasarkan waktu (menit dan jam) atau dengan biaya (\$ USD).

Bagian berikut menunjukkan langkah-langkah untuk mengakses dan menggunakan penjelajah penggunaan Deadline Cloud.

### Topik

- [Prasyarat](#)
- [Buka penjelajah penggunaan](#)
- [Gunakan penjelajah penggunaan](#)

## Prasyarat

Untuk menggunakan penjelajah penggunaan Deadline Cloud, Anda harus memiliki salah satu MANAGER atau izin OWNER pertanian. Untuk informasi selengkapnya, lihat [Kelola pengguna dan grup untuk peternakan, antrian, dan armada](#).

## Buka penjelajah penggunaan

Untuk membuka penjelajah penggunaan Deadline Cloud, gunakan prosedur berikut.

1. Masuk ke AWS Management Console dan buka [konsol](#) Deadline Cloud.

2. Untuk melihat semua peternakan yang tersedia, pilih Lihat peternakan.
3. Temukan peternakan yang ingin Anda dapatkan informasinya, lalu pilih Kelola pekerjaan. Monitor Deadline Cloud terbuka di tab baru.
4. Di monitor Deadline Cloud, dari menu kiri, pilih Usage explorer.

## Gunakan penjelajah penggunaan

Dari halaman penjelajah penggunaan, Anda dapat memilih parameter tertentu di mana data dapat ditampilkan. Secara default, Anda melihat total penggunaan dalam waktu (jam dan menit) dalam 7 hari terakhir. Anda dapat mengubah parameter ini, dan informasi yang ditampilkan berubah secara dinamis sesuai dengan pengaturan parameter.

Anda dapat mengelompokkan hasil berdasarkan antrian, pekerjaan, penggunaan komputasi, jenis instans, atau produk lisensi. Jika Anda memilih produk lisensi, biaya dihitung untuk lisensi tertentu. Untuk semua kelompok lain waktu dihitung dengan menjumlahkan waktu yang dibutuhkan untuk setiap tugas untuk dijalankan.

Penjelajah penggunaan hanya mengembalikan 100 hasil berdasarkan kriteria filter yang Anda tetapkan. Hasilnya tercantum dalam urutan menurun berdasarkan tanggal yang dibuat stempel waktu. Jika ada lebih dari 100 hasil, Anda mendapatkan pesan kesalahan. Anda dapat memperbaiki kueri untuk mengurangi jumlah hasil:

- Pilih rentang waktu yang lebih kecil
- Pilih antrian yang lebih sedikit
- Pilih pengelompokan yang berbeda, seperti pengelompokan berdasarkan antrian, bukan pekerjaan

### Topik

- [Gunakan grafik visual untuk meninjau data](#)
- [Lihat rincian metrik](#)
- [Lihat perkiraan runtime antrian](#)

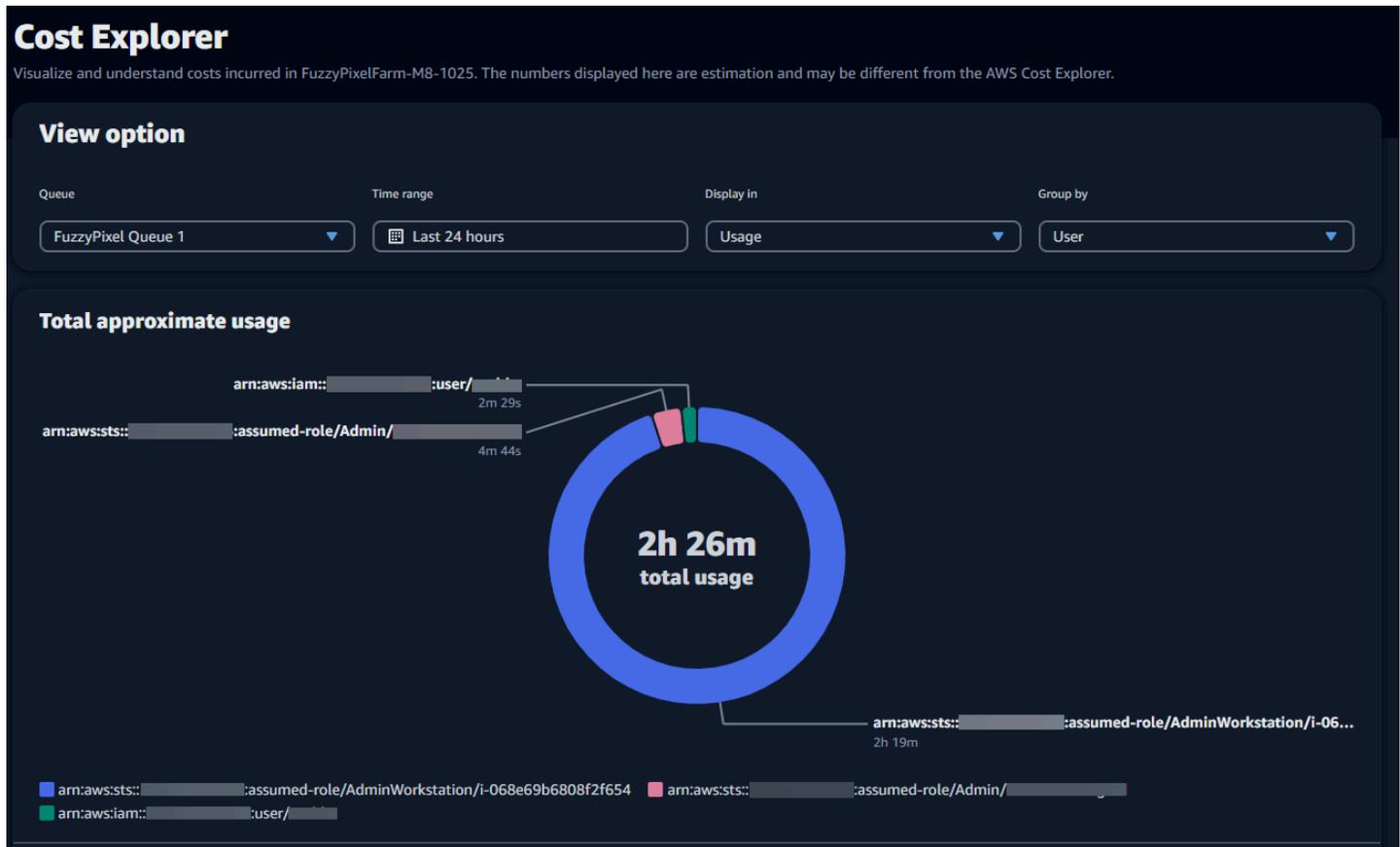
## Gunakan grafik visual untuk meninjau data

Anda dapat meninjau data dalam format visual untuk mengidentifikasi tren dan area potensial yang mungkin memerlukan lebih banyak analisis atau perhatian. Penjelajah penggunaan menawarkan

diagram lingkaran yang menampilkan penggunaan dan biaya keseluruhan dengan opsi untuk mengelompokkan total menjadi subtotal yang lebih kecil.

### Note

Bagan hanya menampilkan lima hasil teratas dengan hasil lain yang digabungkan dalam bagian “lainnya”. Anda dapat melihat semua hasil di bagian rincian di bawah grafik.



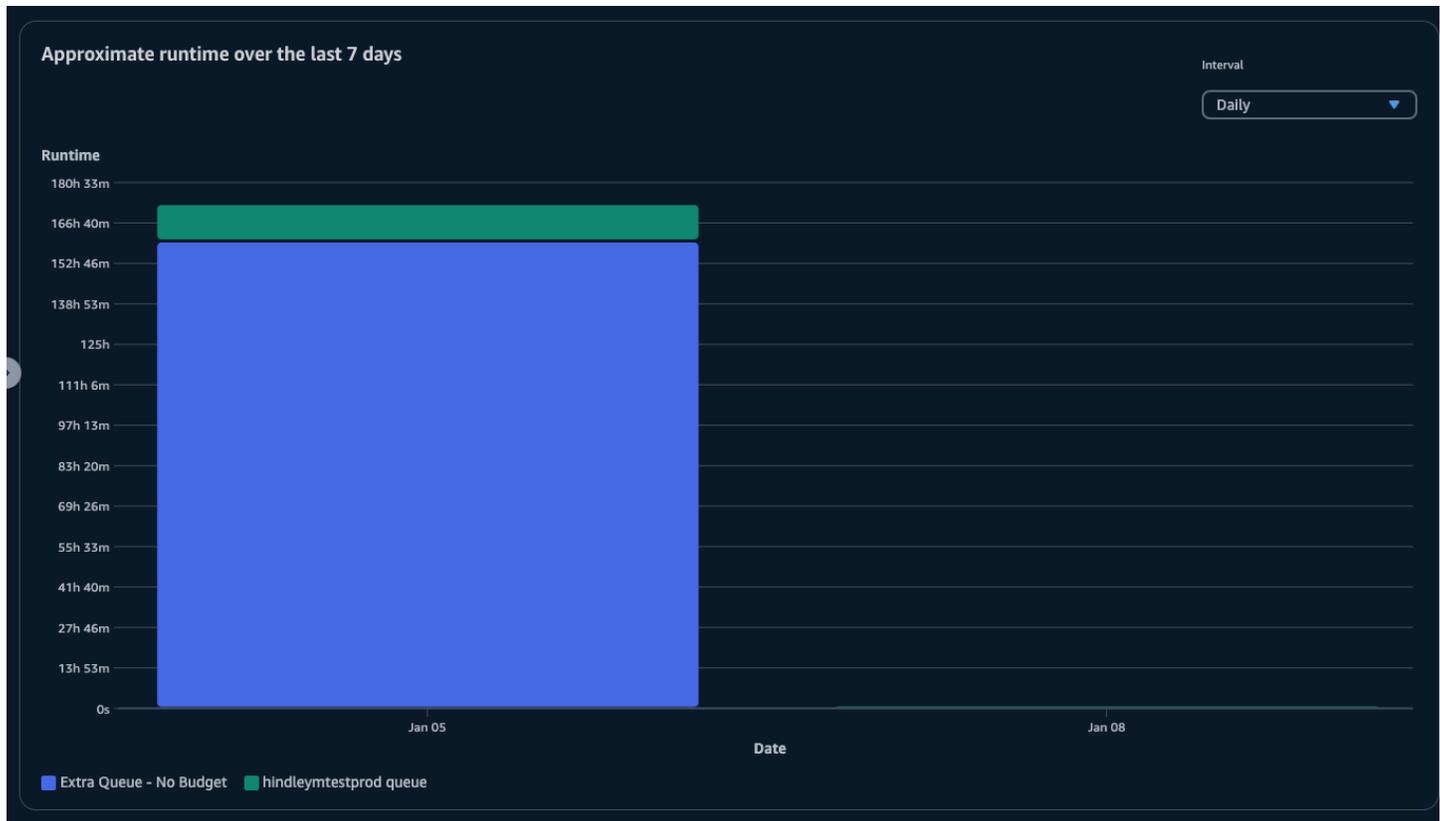
## Lihat rincian metrik

Di bawah diagram lingkaran, penjelajah penggunaan menawarkan rincian metrik tertentu yang lebih rinci, yang akan berubah saat parameter berubah. Secara default, lima hasil ditampilkan di penjelajah penggunaan. Anda dapat menggulir hasil menggunakan panah pagination di bagian breakdown.

Kerusakan diminimalkan secara default. Untuk memperluas dan menampilkan hasilnya, pilih panah Lihat semua kerusakan. Untuk mengunduh rincian, pilih Unduh data.

## Lihat perkiraan runtime antrian

Anda juga dapat melihat perkiraan runtime antrian Anda berdasarkan interval berbeda yang Anda tentukan. Opsi interval adalah per jam, harian, mingguan, dan bulanan. Setelah Anda memilih interval, grafik menampilkan perkiraan runtime antrian Anda.



## Manajemen biaya

AWS Deadline Cloud menyediakan anggaran dan penjelajah penggunaan untuk membantu Anda mengontrol dan memvisualisasikan biaya untuk pekerjaan Anda. Namun, Deadline Cloud menggunakan AWS layanan lain, seperti Amazon S3. Biaya untuk layanan tersebut tidak tercermin dalam anggaran Deadline Cloud atau penjelajah penggunaan dan dibebankan secara terpisah berdasarkan penggunaan. Bergantung pada cara Anda mengonfigurasi Deadline Cloud, Anda dapat menggunakan AWS layanan berikut, serta layanan lainnya:

Layanan	Halaman harga
CloudWatch Log Amazon	<a href="#">Harga Amazon CloudWatch Logs</a>

Layanan	Halaman harga
Amazon Elastic Compute Cloud	<a href="#">Harga Amazon Elastic Compute Cloud</a>
AWS Key Management Service	<a href="#">AWS Key Management Service harga</a>
AWS PrivateLink	<a href="#">AWS PrivateLink harga</a>
Amazon Simple Storage Service	<a href="#">Harga Amazon Simple Storage Service</a>
Amazon Virtual Private Cloud	<a href="#">Harga Amazon Virtual Private Cloud</a>

## Praktik terbaik manajemen biaya

Menggunakan praktik terbaik berikut dapat membantu Anda memahami dan mengontrol biaya saat menggunakan Deadline Cloud dan pengorbanan yang dapat Anda lakukan antara biaya dan efisiensi.

### Note

Biaya akhir menggunakan Deadline Cloud tergantung pada interaksi antara sejumlah AWS layanan, jumlah pekerjaan yang Anda proses, dan Wilayah AWS di mana Anda menjalankan pekerjaan Anda. Praktik terbaik berikut adalah pedoman dan mungkin tidak mengurangi biaya secara signifikan.

## Praktik terbaik untuk CloudWatch Log

Deadline Cloud mengirimkan log pekerja dan tugas ke CloudWatch Log. Anda dikenakan biaya untuk mengumpulkan, menyimpan, dan menganalisis log ini. Anda dapat mengurangi biaya dengan mencatat hanya jumlah minimum data yang diperlukan untuk memantau tugas Anda.

Saat Anda membuat antrian atau armada, Deadline Cloud membuat grup CloudWatch log Log dengan nama berikut:

- `aws/deadline/<FARM_ID>/<FLEET_ID>`
- `aws/deadline/<FARM_ID>/<QUEUE_ID>`

Secara default, log ini tidak pernah kedaluwarsa. Anda dapat menyesuaikan kebijakan penyimpanan grup log untuk menghapus log lama dan membantu mengurangi biaya penyimpanan. Anda juga dapat mengekspor log ke Amazon S3. Biaya penyimpanan Amazon S3 lebih rendah daripada biaya penyimpanan. CloudWatch Untuk informasi lebih lanjut, lihat [Mengekspor data log ke Amazon S3](#).

## Praktik terbaik untuk Amazon EC2

Anda dapat menggunakan instans Amazon EC2 untuk armada yang dikelola layanan dan yang dikelola pelanggan. Ada tiga pertimbangan:

- Untuk armada yang dikelola layanan, Anda dapat memilih untuk memiliki satu atau beberapa instance yang tersedia setiap saat dengan menetapkan jumlah pekerja minimum untuk armada. Ketika Anda menetapkan jumlah pekerja minimum di atas 0, armada selalu memiliki banyak pekerja yang berjalan. Ini dapat mengurangi jumlah waktu yang dibutuhkan Deadline Cloud untuk mulai memproses pekerjaan, namun Anda dikenakan biaya untuk waktu idle instans.
- Untuk armada yang dikelola layanan, tetapkan ukuran maksimum untuk armada. Ini membatasi jumlah instance yang dapat ditskalakan secara otomatis oleh armada. Armada tidak akan tumbuh melewati ukuran ini bahkan jika ada lebih banyak pekerjaan yang menunggu untuk diproses.
- Untuk armada yang dikelola layanan dan yang dikelola pelanggan, Anda dapat menentukan jenis instans Amazon EC2 di armada Anda. Menggunakan contoh yang lebih kecil harganya lebih murah per menit, tetapi mungkin membutuhkan waktu lebih lama untuk menyelesaikan pekerjaan. Sebaliknya, contoh yang lebih besar harganya lebih per menit, tetapi dapat mengurangi waktu untuk menyelesaikan pekerjaan. Memahami tuntutan yang ditempatkan pekerjaan Anda pada sebuah contoh dapat membantu mengurangi biaya Anda.
- Jika memungkinkan, pilih instans Amazon EC2 Spot untuk armada Anda. Instans spot tersedia dengan harga yang lebih murah, tetapi dapat terganggu oleh permintaan sesuai permintaan. Instans sesuai permintaan dibebankan oleh yang kedua dan tidak terganggu.

## Praktik terbaik untuk AWS KMS

Secara default, Deadline Cloud mengenkripsi data Anda dengan kunci yang AWS dimiliki. Anda tidak dikenakan biaya untuk kunci ini.

Anda dapat memilih untuk menggunakan kunci yang dikelola pelanggan untuk mengenkripsi data Anda. Ketika Anda menggunakan kunci Anda sendiri, Anda akan dikenakan biaya berdasarkan bagaimana kunci Anda digunakan. Jika Anda menggunakan kunci yang ada, ini akan menjadi biaya tambahan untuk penggunaan tambahan.

## Praktik terbaik untuk AWS PrivateLink

Anda dapat menggunakan AWS PrivateLink untuk membuat koneksi antara VPC dan Deadline Cloud menggunakan endpoint antarmuka. Saat membuat koneksi, Anda dapat memanggil semua tindakan Deadline Cloud API. Anda dikenakan biaya per jam untuk setiap titik akhir yang Anda buat. Jika Anda menggunakan PrivateLink, Anda harus membuat setidaknya tiga titik akhir, dan tergantung pada konfigurasi Anda, Anda mungkin memerlukan sebanyak lima.

## Praktik terbaik untuk Amazon S3

Deadline Cloud menggunakan Amazon S3 untuk menyimpan aset untuk diproses, lampiran pekerjaan, output, dan log. Untuk mengurangi biaya yang terkait dengan Amazon S3, kurangi jumlah data yang Anda simpan. Beberapa saran:

- Hanya menyimpan aset yang sedang digunakan atau yang akan segera digunakan.
- Gunakan [konfigurasi Siklus Hidup S3](#) untuk menghapus file yang tidak digunakan secara otomatis dari bucket S3.

## Praktik terbaik untuk Amazon VPC

Saat Anda menggunakan lisensi berbasis penggunaan untuk armada yang dikelola pelanggan, Anda membuat titik akhir lisensi Deadline Cloud, yang merupakan titik akhir Amazon VPC yang dibuat di akun Anda. Titik akhir ini dibebankan dengan tarif per jam. Untuk mengurangi biaya, hapus titik akhir saat Anda tidak menggunakan lisensi berbasis penggunaan.

# Keamanan di Deadline Cloud

Keamanan cloud di AWS adalah prioritas tertinggi. Sebagai AWS pelanggan, Anda mendapat manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama antara AWS dan kamu. [Model tanggung jawab bersama](#) menjelaskan hal ini sebagai keamanan cloud dan keamanan dalam cloud:

- Keamanan Cloud — AWS Bertanggung jawab untuk melindungi infrastruktur yang berjalan layanan AWS di AWS Cloud. AWS juga memberi Anda layanan yang dapat Anda gunakan dengan aman. Auditor pihak ketiga secara teratur menguji dan memverifikasi efektivitas keamanan kami sebagai bagian dari [AWS Program Kepatuhan](#) . Untuk mempelajari tentang program kepatuhan yang berlaku AWS Deadline Cloud, lihat [layanan AWS dalam Lingkup oleh Program Kepatuhan](#) .
- Keamanan di cloud — Tanggung jawab Anda ditentukan oleh layanan AWS yang Anda gunakan. Anda juga bertanggung jawab atas faktor lain, yang mencakup sensitivitas data Anda, persyaratan perusahaan Anda, serta undang-undang dan peraturan yang berlaku.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan Deadline Cloud. Topik berikut menunjukkan cara mengkonfigurasi Deadline Cloud untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga belajar cara menggunakan yang lain layanan AWS yang membantu Anda memantau dan mengamankan Deadline Cloud sumber daya.

## Topik

- [Perlindungan data di Deadline Cloud](#)
- [Identity and Access Management di Deadline Cloud](#)
- [Validasi kepatuhan untuk Deadline Cloud](#)
- [Ketahanan di Deadline Cloud](#)
- [Keamanan infrastruktur di Deadline Cloud](#)
- [Analisis konfigurasi dan kerentanan di Deadline Cloud](#)
- [Pencegahan confused deputy lintas layanan](#)
- [Akses AWS Deadline Cloud menggunakan titik akhir antarmuka \(AWS PrivateLink\)](#)
- [Praktik terbaik keamanan untuk Deadline Cloud](#)

## Perlindungan data di Deadline Cloud

Bagian AWS [model tanggung jawab bersama model](#) berlaku untuk perlindungan data di AWS Deadline Cloud. Seperti yang dijelaskan dalam model ini, AWS bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk menjaga kontrol atas konten Anda yang di-host di infrastruktur ini. Anda juga bertanggung jawab atas konfigurasi keamanan dan tugas manajemen untuk layanan AWS yang Anda gunakan. Untuk informasi selengkapnya tentang privasi data, lihat [Privasi Data FAQ](#). Untuk informasi tentang perlindungan data di Eropa, lihat [AWS Model Tanggung Jawab Bersama dan posting GDPR](#) blog di AWS Blog Keamanan.

Untuk tujuan perlindungan data, kami menyarankan Anda untuk melindungi Akun AWS kredensi 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 otentikasi multi-faktor (MFA) dengan setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan AWS sumber daya. Kami membutuhkan TLS 1.2 dan merekomendasikan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail. Untuk informasi tentang menggunakan CloudTrail jalur untuk menangkap AWS kegiatan, lihat [Bekerja dengan CloudTrail jalan setapak](#) di AWS CloudTrail Panduan Pengguna.
- Gunakan AWS solusi enkripsi, bersama dengan semua kontrol keamanan default di dalamnya layanan AWS.
- Gunakan layanan keamanan terkelola lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan FIPS 140-3 modul kriptografi yang divalidasi saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan FIPS titik akhir. Untuk informasi selengkapnya tentang FIPS titik akhir yang tersedia, lihat [Federal Information Processing Standard \(FIPS\) 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 ketika Anda bekerja dengan Deadline Cloud 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. Jika Anda memberikan URL ke server eksternal, kami sangat menyarankan agar Anda tidak menyertakan informasi kredensial dalam URL untuk memvalidasi permintaan Anda ke server tersebut.

Data yang dimasukkan ke dalam bidang nama di Deadline Cloud Template pekerjaan juga dapat dimasukkan dalam penagihan atau log diagnostik dan tidak boleh berisi informasi rahasia atau sensitif.

Topik

- [Enkripsi diam](#)
- [Enkripsi bergerak](#)
- [Manajemen kunci](#)
- [Privasi lalu lintas antar jaringan](#)
- [Menyisih](#)

## Enkripsi diam

AWS Deadline Cloud melindungi data sensitif dengan mengenkripsi saat istirahat menggunakan kunci enkripsi yang disimpan di [AWS Key Management Service \(AWS KMS\)](#). Enkripsi saat istirahat tersedia di semua Wilayah AWS di mana Deadline Cloud tersedia.

Menkripsi data berarti data sensitif yang disimpan pada disk tidak dapat dibaca oleh pengguna atau aplikasi tanpa kunci yang valid. Hanya pihak dengan kunci terkelola yang valid yang dapat mendekripsi data.

Untuk informasi tentang bagaimana Deadline Cloud menggunakan AWS KMS untuk mengenkripsi data saat istirahat, lihat [Manajemen kunci](#)

## Enkripsi bergerak

Untuk data dalam perjalanan, AWS Deadline Cloud menggunakan Transport Layer Security (TLS) 1.2 atau 1.3 untuk mengenkripsi data yang dikirim antara layanan dan pekerja. Kami membutuhkan TLS 1.2 dan merekomendasikan TLS 1.3. Selain itu, jika Anda menggunakan virtual private cloud (VPC), Anda dapat menggunakan AWS PrivateLink untuk membangun koneksi pribadi antara Anda VPC dan Deadline Cloud.

## Manajemen kunci

Saat membuat peternakan baru, Anda dapat memilih salah satu kunci berikut untuk mengenkripsi data pertanian Anda:

- **AWS KMS kunci yang dimiliki** — Jenis enkripsi default jika Anda tidak menentukan kunci saat membuat peternakan. KMS kunci dimiliki oleh AWS Deadline Cloud. Anda tidak dapat melihat, mengelola, atau menggunakan AWS kunci yang dimiliki. Namun, Anda tidak perlu mengambil tindakan apa pun untuk melindungi kunci yang mengenkripsi data Anda. Untuk informasi selengkapnya, silakan lihat [AWS kunci yang dimiliki](#) di AWS Key Management Service panduan pengembang.
- **KMS kunci terkelola pelanggan** — Anda menentukan kunci yang dikelola pelanggan saat membuat peternakan. Semua konten di dalam peternakan dienkripsi dengan kunci. KMS Kunci disimpan di akun Anda dan dibuat, dimiliki, dan dikelola oleh Anda dan AWS KMS dikenakan biaya. Anda memiliki kendali penuh atas KMS kunci. Anda dapat melakukan tugas-tugas seperti:
  - Menetapkan dan memelihara kebijakan utama
  - Menetapkan dan memelihara IAM kebijakan dan hibah
  - Mengaktifkan dan menonaktifkan kebijakan utama
  - Menambahkan tanda
  - Membuat alias kunci

Anda tidak dapat secara manual memutar kunci milik pelanggan yang digunakan dengan Deadline Cloud pertanian. Rotasi otomatis tombol didukung.

Untuk informasi selengkapnya, lihat [Kunci milik pelanggan](#) di AWS Key Management Service Panduan Pengembang.

Untuk membuat kunci terkelola pelanggan, ikuti langkah-langkah untuk [Membuat kunci terkelola pelanggan simetris](#) di AWS Key Management Service Panduan Pengembang.

## Bagaimana Deadline Cloud menggunakan AWS KMS hadiah

Deadline Cloud membutuhkan [hibah](#) untuk menggunakan kunci yang dikelola pelanggan Anda. Saat Anda membuat peternakan yang dienkripsi dengan kunci yang dikelola pelanggan, Deadline Cloud membuat hibah atas nama Anda dengan mengirimkan [CreateGrant](#) permintaan ke AWS KMS untuk mendapatkan akses ke KMS kunci yang Anda tentukan.

Deadline Cloud menggunakan beberapa hibah. Setiap hibah digunakan oleh bagian yang berbeda dari Deadline Cloud yang perlu mengenkripsi atau mendekripsi data Anda. Deadline Cloud juga menggunakan hibah untuk memungkinkan akses ke yang lain AWS layanan yang digunakan untuk menyimpan data atas nama Anda, seperti Amazon Simple Storage Service, Amazon Elastic Block Store, atau OpenSearch.

Hibah yang memungkinkan Deadline Cloud untuk mengelola mesin dalam armada yang dikelola layanan termasuk a Deadline Cloud nomor akun dan peran dalam `GrantPrincipal` bukan prinsipal layanan. Meskipun tidak khas, ini diperlukan untuk mengenkripsi EBS volume Amazon untuk pekerja dalam armada yang dikelola layanan menggunakan KMS kunci terkelola pelanggan yang ditentukan untuk pertanian.

## Kebijakan kunci yang dikelola pelanggan

Kebijakan utama mengontrol akses ke kunci yang dikelola pelanggan Anda. Setiap kunci harus memiliki persis satu kebijakan kunci yang berisi pernyataan yang menentukan siapa yang dapat menggunakan kunci dan bagaimana mereka dapat menggunakannya. Saat membuat kunci terkelola pelanggan, Anda dapat menentukan kebijakan kunci. Untuk informasi selengkapnya, lihat [Mengelola akses ke kunci terkelola pelanggan](#) di AWS Key Management Service Panduan Pengembang.

### IAMKebijakan minimal untuk CreateFarm

Untuk menggunakan kunci terkelola pelanggan Anda untuk membuat peternakan menggunakan konsol atau [CreateFarm](#) API operasi, berikut ini AWS KMS API operasi harus diizinkan:

- [kms:CreateGrant](#)— Menambahkan hibah ke kunci yang dikelola pelanggan. Memberikan akses konsol ke yang ditentukan AWS KMS kunci. Untuk informasi selengkapnya, lihat [Menggunakan hibah di](#) AWS Key Management Service panduan pengembang.
- [kms:Decrypt](#)— Memungkinkan Deadline Cloud untuk mendekripsi data di peternakan.
- [kms:DescribeKey](#)— Memberikan detail kunci yang dikelola pelanggan untuk memungkinkan Deadline Cloud untuk memvalidasi kuncinya.
- [kms:GenerateDataKey](#)— Memungkinkan Deadline Cloud untuk mengenkripsi data menggunakan kunci data yang unik.

Pernyataan kebijakan berikut memberikan izin yang diperlukan untuk operasi. `CreateFarm`

```
{
  "Version": "2012-10-17",
```

```

"Statement": [
  {
    "Sid": "DeadlineCreateGrants",
    "Effect": "Allow",
    "Action": [
      "kms:Decrypt",
      "kms:GenerateDataKey",
      "kms:CreateGrant",
      "kms:DescribeKey"
    ],
    "Resource": "arn:aws::kms:us-west-2:111122223333:key/1234567890abcdef0",
    "Condition": {
      "StringEquals": {
        "kms:ViaService": "deadline.us-west-2.amazonaws.com"
      }
    }
  }
]
}

```

### IAMKebijakan minimal untuk operasi hanya-baca

Untuk menggunakan kunci terkelola pelanggan Anda untuk hanya-baca Deadline Cloud operasi, seperti mendapatkan informasi tentang pertanian, antrian, dan armada. Berikut ini AWS KMS APloperasi harus diizinkan:

- [kms:Decrypt](#)— Memungkinkan Deadline Cloud untuk mendekripsi data di peternakan.
- [kms:DescribeKey](#)— Memberikan detail kunci yang dikelola pelanggan untuk memungkinkan Deadline Cloud untuk memvalidasi kuncinya.

Pernyataan kebijakan berikut memberikan izin yang diperlukan untuk operasi hanya-baca.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DeadlineReadOnly",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
        "kms:DescribeKey"
      ],
    }
  ],
}

```

```

    "Resource": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
    "Condition": {
      "StringEquals": {
        "kms:ViaService": "deadline.us-west-2.amazonaws.com"
      }
    }
  }
]
}

```

## IAMKebijakan minimal untuk operasi baca-tulis

Untuk menggunakan kunci terkelola pelanggan Anda untuk baca-tulis Deadline Cloud operasi, seperti membuat dan memperbarui peternakan, antrian, dan armada. Berikut ini AWS KMS API operasi harus diizinkan:

- [kms:Decrypt](#)— Memungkinkan Deadline Cloud untuk mendekripsi data di peternakan.
- [kms:DescribeKey](#)— Memberikan detail kunci yang dikelola pelanggan untuk memungkinkan Deadline Cloud untuk memvalidasi kuncinya.
- [kms:GenerateDataKey](#)— Memungkinkan Deadline Cloud untuk mengenkripsi data menggunakan kunci data yang unik.

Pernyataan kebijakan berikut memberikan izin yang diperlukan untuk operasi. CreateFarm

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DeadlineReadWrite",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
        "kms:DescribeKey",
        "kms:GenerateDataKey",
      ],
      "Resource": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "deadline.us-west-2.amazonaws.com"
        }
      }
    }
  ]
}

```

```

    }
  }
}

```

## Memantau kunci enkripsi Anda

Saat Anda menggunakan AWS KMS kunci yang dikelola pelanggan dengan Anda Deadline Cloud peternakan, Anda dapat menggunakan [AWS CloudTrail](#) atau [Amazon CloudWatch Log](#) untuk melacak permintaan itu Deadline Cloud mengirim ke AWS KMS.

CloudTrail acara untuk hibah

Contoh CloudTrail peristiwa berikut terjadi ketika hibah dibuat, biasanya ketika Anda memanggil `CreateFarm`, `CreateMonitor`, atau `CreateFleet` operasi.

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAIQDTESTANDEXAMPLE:SampleUser01",
    "arn": "arn:aws::sts::111122223333:assumed-role/Admin/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE3",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAIQDTESTANDEXAMPLE",
        "arn": "arn:aws::iam::111122223333:role/Admin",
        "accountId": "111122223333",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2024-04-23T02:05:26Z",
        "mfaAuthenticated": "false"
      }
    },
    "invokedBy": "deadline.amazonaws.com"
  },
  "eventTime": "2024-04-23T02:05:35Z",
  "eventSource": "kms.amazonaws.com",

```

```

"eventName": "CreateGrant",
"awsRegion": "us-west-2",
"sourceIPAddress": "deadline.amazonaws.com",
"userAgent": "deadline.amazonaws.com",
"requestParameters": {
  "operations": [
    "CreateGrant",
    "Decrypt",
    "DescribeKey",
    "Encrypt",
    "GenerateDataKey"
  ],
  "constraints": {
    "encryptionContextSubset": {
      "aws:deadline:farmId": "farm-abcdef12345678900987654321fedcba",
      "aws:deadline:accountId": "111122223333"
    }
  },
  "granteePrincipal": "deadline.amazonaws.com",
  "keyId": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
  "retiringPrincipal": "deadline.amazonaws.com"
},
"responseElements": {
  "grantId": "6bbe819394822a400fe5e3a75d0e9ef16c1733143fff0c1fc00dc7ac282a18a0",
  "keyId": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
},
"requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
"eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
"readOnly": false,
"resources": [
  {
    "accountId": "AWS Internal",
    "type": "AWS::KMS::Key",
    "ARN": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-EXAMPLE44444"
  }
],
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "111122223333",
"eventCategory": "Management"

```

}

## CloudTrail acara untuk dekripsi

Contoh CloudTrail peristiwa berikut terjadi ketika mendekripsi nilai menggunakan kunci yang dikelola KMS pelanggan.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAIQDTESTANDEXAMPLE:SampleUser01",
    "arn": "arn:aws::sts::111122223333:assumed-role/SampleRole/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAIQDTESTANDEXAMPLE",
        "arn": "arn:aws::iam::111122223333:role/SampleRole",
        "accountId": "111122223333",
        "userName": "SampleRole"
      },
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2024-04-23T18:46:51Z",
        "mfaAuthenticated": "false"
      }
    },
    "invokedBy": "deadline.amazonaws.com"
  },
  "eventTime": "2024-04-23T18:51:44Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "Decrypt",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "deadline.amazonaws.com",
  "userAgent": "deadline.amazonaws.com",
  "requestParameters": {
    "encryptionContext": {
      "aws:deadline:farmId": "farm-abcdef12345678900987654321fedcba",
      "aws:deadline:accountId": "111122223333",
      "aws-crypto-public-key": "AotL+SAMPLEVALUEiOMEXAMPLEEaaqNOTREALaGTESTONLY
+p/5H+EuKd4Q=="
    }
  }
}
```

```

    },
    "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
    "keyId": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111"
  },
  "responseElements": null,
  "requestID": "aaaaaaaa-bbbb-cccc-dddd-eeeeefffffff",
  "eventID": "ffffffff-eeee-dddd-cccc-bbbbbbaaaaaa",
  "readOnly": true,
  "resources": [
    {
      "accountId": "111122223333",
      "type": "AWS::KMS::Key",
      "ARN": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "111122223333",
  "eventCategory": "Management"
}

```

## CloudTrail acara untuk enkripsi

Contoh CloudTrail peristiwa berikut terjadi ketika mengenkripsi nilai menggunakan kunci yang dikelola KMS pelanggan.

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAIQDTESTANDEXAMPLE:SampleUser01",
    "arn": "arn:aws::sts::111122223333:assumed-role/SampleRole/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAIQDTESTANDEXAMPLE",
        "arn": "arn:aws::iam::111122223333:role/SampleRole",
        "accountId": "111122223333",
        "userName": "SampleRole"
      }
    }
  }
}

```

```

    },
    "webIdFederationData": {},
    "attributes": {
      "creationDate": "2024-04-23T18:46:51Z",
      "mfaAuthenticated": "false"
    }
  },
  "invokedBy": "deadline.amazonaws.com"
},
"eventTime": "2024-04-23T18:52:40Z",
"eventSource": "kms.amazonaws.com",
"eventName": "GenerateDataKey",
"awsRegion": "us-west-2",
"sourceIPAddress": "deadline.amazonaws.com",
"userAgent": "deadline.amazonaws.com",
"requestParameters": {
  "numberOfBytes": 32,
  "encryptionContext": {
    "aws:deadline:farmId": "farm-abcdef12345678900987654321fedcba",
    "aws:deadline:accountId": "111122223333",
    "aws-crypto-public-key": "AotL+SAMPLEVALUEi0MEXAMPLEEaaqNOTREALaGTESTONLY  

+p/5H+EuKd4Q=="
  }
},
"keyId": "arn:aws::kms:us-  

west-2:111122223333:key/abcdef12-3456-7890-0987-654321fedcba"
},
"responseElements": null,
"requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
"eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
"readOnly": true,
"resources": [
  {
    "accountId": "111122223333",
    "type": "AWS::KMS::Key",
    "ARN": "arn:aws::kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-  

EXAMPLE33333"
  }
],
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "111122223333",
"eventCategory": "Management"
}

```

## Menghapus kunci terkelola KMS pelanggan

Menghapus KMS kunci terkelola pelanggan di AWS Key Management Service (AWS KMS) bersifat destruktif dan berpotensi berbahaya. Ini secara permanen menghapus materi kunci dan semua metadata yang terkait dengan kunci. Setelah KMS kunci yang dikelola pelanggan dihapus, Anda tidak dapat lagi mendekripsi data yang dienkripsi oleh kunci tersebut. Ini berarti bahwa data menjadi tidak dapat dipulihkan.

Ini adalah sebabnya AWS KMS memberi pelanggan masa tunggu hingga 30 hari sebelum menghapus KMS kunci. Masa tunggu default adalah 30 hari.

### Tentang masa tunggu

Karena menghapus KMS kunci yang dikelola pelanggan merusak dan berpotensi berbahaya, kami mengharuskan Anda menetapkan masa tunggu 7—30 hari. Masa tunggu default adalah 30 hari.

Namun, masa tunggu sebenarnya mungkin hingga 24 jam lebih lama dari periode yang Anda jadwalkan. Untuk mendapatkan tanggal dan waktu aktual ketika kunci akan dihapus, gunakan [DescribeKey](#) operasi. Anda juga dapat melihat tanggal penghapusan kunci yang dijadwalkan di [AWS KMS konsol](#) di halaman detail kunci, di bagian Konfigurasi umum. Perhatikan zona waktu.

Selama masa tunggu, status dan status kunci yang dikelola pelanggan adalah Penghapusan tertunda.

- KMSKunci terkelola pelanggan yang menunggu penghapusan tidak dapat digunakan dalam operasi [kriptografi](#) apa pun.
- AWS KMS tidak [memutar kunci dukungan kunci](#) terkelola KMS pelanggan yang sedang menunggu penghapusan.

Untuk informasi selengkapnya tentang menghapus KMS kunci terkelola pelanggan, lihat [Menghapus kunci master pelanggan](#) di AWS Key Management Service Panduan Pengembang.

## Privasi lalu lintas antar jaringan

AWS Deadline Cloud mendukung Amazon Virtual Private Cloud (AmazonVPC) untuk mengamankan koneksi. Amazon VPC menyediakan fitur yang dapat Anda gunakan untuk meningkatkan dan memantau keamanan cloud pribadi virtual Anda (VPC).

Anda dapat menyiapkan armada (CMF) yang dikelola pelanggan dengan instans Amazon Elastic Compute Cloud (AmazonEC2) yang berjalan di dalam file. VPC Dengan menerapkan VPC titik akhir

Amazon untuk digunakan AWS PrivateLink, lalu lintas antara pekerja di Anda CMF dan Deadline Cloud titik akhir tetap di dalam AndaVPC. Selanjutnya, Anda dapat mengonfigurasi VPC untuk membatasi akses internet ke instans Anda.

Dalam armada yang dikelola layanan, pekerja tidak dapat dijangkau dari internet, tetapi mereka memiliki akses internet dan terhubung ke Deadline Cloud layanan melalui internet.

## Menyisih

AWS Deadline Cloud mengumpulkan informasi operasional tertentu untuk membantu kami mengembangkan dan meningkatkan Deadline Cloud. Data yang dikumpulkan mencakup hal-hal seperti AWS ID akun dan ID pengguna, sehingga kami dapat mengidentifikasi Anda dengan benar jika Anda memiliki masalah dengan Deadline Cloud. Kami juga mengumpulkan Deadline Cloud informasi spesifik, seperti Resource IDs (FarmId atau QueueID bila berlaku), nama produk (misalnya, JobAttachments WorkerAgent, dan lainnya) dan versi produk.

Anda dapat memilih untuk memilih keluar dari pengumpulan data ini menggunakan konfigurasi aplikasi. Setiap komputer berinteraksi dengan Deadline Cloud, baik workstation klien dan pekerja armada, perlu memilih keluar secara terpisah.

### Deadline Cloud monitor - desktop

Deadline Cloud monitor - desktop mengumpulkan informasi operasional, seperti ketika crash terjadi dan ketika aplikasi dibuka, untuk membantu kami mengetahui kapan Anda mengalami masalah dengan aplikasi. Untuk memilih keluar dari pengumpulan informasi operasional ini, buka halaman pengaturan dan hapus Aktifkan pengumpulan data untuk mengukur kinerja Deadline Cloud Monitor.

Setelah Anda memilih keluar, monitor desktop tidak lagi mengirimkan data operasional. Setiap data yang dikumpulkan sebelumnya disimpan dan masih dapat digunakan untuk meningkatkan layanan. Untuk informasi selengkapnya, lihat [Privasi Data FAQ](#).

### AWS Deadline Cloud CLI dan Tools

Bagian AWS Deadline Cloud CLI, pengirim, dan agen pekerja semuanya mengumpulkan informasi operasional seperti kapan crash terjadi dan kapan pekerjaan dikirimkan untuk membantu kami mengetahui kapan Anda mengalami masalah dengan aplikasi ini. Untuk memilih keluar dari pengumpulan informasi operasional ini, gunakan salah satu metode berikut:

- Di terminal, masukkan **deadline config set telemetry.opt\_out true**.

Ini akan memilih keluar dari CLI, pengirim, dan agen pekerja saat menjalankan sebagai pengguna saat ini.

- Saat memasang Deadline Cloud agen pekerja, tambahkan argumen baris **--telemetry-opt-out** perintah. Misalnya, **./install.sh --farm-id \$FARM\_ID --fleet-id \$FLEET\_ID --telemetry-opt-out**.
- Sebelum menjalankan agen pekerja, CLI, atau pengirim, tetapkan variabel lingkungan: **DEADLINE\_CLOUD\_TELEMETRY\_OPT\_OUT=true**

Setelah Anda memilih keluar, Deadline Cloud alat tidak lagi mengirim data operasional. Setiap data yang dikumpulkan sebelumnya disimpan dan masih dapat digunakan untuk meningkatkan layanan. Untuk informasi selengkapnya, lihat [Privasi Data FAQ](#).

## Identity and Access Management di Deadline Cloud

AWS Identity and Access Management (IAM) adalah layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya. IAM administrator mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya Deadline Cloud. IAM adalah sebuah layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

### Topik

- [Audiens](#)
- [Mengautentikasi dengan identitas](#)
- [Mengelola akses menggunakan kebijakan](#)
- [Bagaimana Deadline Cloud bekerja dengan IAM](#)
- [Contoh kebijakan berbasis identitas untuk Deadline Cloud](#)
- [AWS kebijakan terkelola untuk Deadline Cloud](#)
- [Pemecahan Masalah AWS Batas waktu Identitas dan akses Cloud](#)

## Audiens

Bagaimana Anda menggunakan AWS Identity and Access Management (IAM) berbeda, tergantung pada pekerjaan yang Anda lakukan di Deadline Cloud.

Pengguna layanan — Jika Anda menggunakan layanan Deadline Cloud untuk melakukan pekerjaan Anda, administrator Anda memberi Anda kredensi dan izin yang Anda butuhkan. Saat Anda menggunakan lebih banyak fitur Deadline Cloud 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 di Deadline Cloud, lihat [Pemecahan Masalah AWS Batas waktu Identitas dan akses Cloud](#).

Administrator layanan — Jika Anda bertanggung jawab atas sumber daya Deadline Cloud di perusahaan Anda, Anda mungkin memiliki akses penuh ke Deadline Cloud. Tugas Anda adalah menentukan fitur dan sumber daya Deadline Cloud mana yang harus diakses pengguna layanan Anda. Anda kemudian harus mengirimkan permintaan ke IAM administrator Anda untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep dasar IAM. Untuk mempelajari lebih lanjut tentang cara perusahaan Anda dapat menggunakan IAM Deadline Cloud, lihat [Bagaimana Deadline Cloud bekerja dengan IAM](#).

IAM administrator - Jika Anda seorang IAM administrator, Anda mungkin ingin mempelajari detail tentang cara menulis kebijakan untuk mengelola akses ke Deadline Cloud. Untuk melihat contoh Kebijakan berbasis identitas Cloud Deadline yang dapat Anda gunakan, lihat. IAM [Contoh kebijakan berbasis identitas untuk Deadline Cloud](#)

## Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredensial identitas Anda. Anda harus diautentikasi (masuk ke AWS) sebagai Pengguna root akun AWS, sebagai IAM pengguna, atau dengan mengambil IAM peran.

Anda dapat masuk ke AWS sebagai identitas federasi dengan menggunakan kredensi yang disediakan melalui sumber identitas. AWS IAM Identity Center Pengguna (Pusat IAM Identitas), autentikasi masuk tunggal perusahaan Anda, dan kredensi Google atau Facebook Anda adalah contoh identitas federasi. Saat Anda masuk sebagai identitas federasi, administrator Anda sebelumnya menyiapkan federasi identitas menggunakan IAM peran. Saat Anda mengakses AWS dengan menggunakan federasi, Anda secara tidak langsung mengambil peran.

Tergantung pada jenis pengguna Anda, Anda dapat masuk ke AWS Management Console atau AWS portal akses. Untuk informasi lebih lanjut tentang masuk AWS, lihat [Cara masuk ke Akun AWS](#) di AWS Sign-In Panduan Pengguna.

Jika Anda mengakses AWS secara terprogram, AWS menyediakan kit pengembangan perangkat lunak (SDK) dan antarmuka baris perintah (CLI) untuk menandatangani permintaan Anda secara

kriptografis dengan menggunakan kredensial Anda. Jika Anda tidak menggunakan AWS alat, Anda harus menandatangani permintaan sendiri. Untuk informasi selengkapnya tentang menggunakan metode yang disarankan untuk menandatangani permintaan sendiri, lihat [Menandatangani AWS API permintaan](#) di Panduan IAM Pengguna.

Apa pun metode autentikasi yang digunakan, Anda mungkin diminta untuk menyediakan informasi keamanan tambahan. Misalnya, AWS merekomendasikan agar Anda menggunakan otentikasi multi-faktor (MFA) untuk meningkatkan keamanan akun Anda. Untuk mempelajari selengkapnya, lihat [Autentikasi multi-faktor](#) di AWS IAM Identity Center Panduan Pengguna dan [Menggunakan otentikasi multi-faktor \(\) MFA di AWS](#) di Panduan Pengguna IAM.

## Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang memiliki akses lengkap ke semua layanan AWS dan sumber daya di akun. Identitas ini disebut Akun AWS pengguna root dan diakses dengan masuk dengan alamat email dan kata sandi yang Anda gunakan untuk membuat akun. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Lindungi kredensial pengguna root Anda dan gunakan kredensial tersebut untuk melakukan tugas yang hanya dapat dilakukan pengguna root. Untuk daftar lengkap tugas yang mengharuskan Anda masuk sebagai pengguna root, lihat [Tugas yang memerlukan kredensial pengguna root](#) di IAMPanduan Pengguna.

## Identitas gabungan

Sebagai praktik terbaik, mewajibkan pengguna manusia, termasuk pengguna yang memerlukan akses administrator, untuk menggunakan federasi dengan penyedia identitas untuk mengakses layanan AWS dengan menggunakan kredensial sementara.

Identitas federasi adalah pengguna dari direktori pengguna perusahaan Anda, penyedia identitas web, AWS Directory Service, direktori Pusat Identitas, atau pengguna mana pun yang mengakses layanan AWS dengan menggunakan kredensial yang disediakan melalui sumber identitas. Ketika akses identitas federasi Akun AWS, mereka mengambil peran, dan peran memberikan kredensial sementara.

Untuk manajemen akses terpusat, kami sarankan Anda menggunakan AWS IAM Identity Center. Anda dapat membuat pengguna dan grup di Pusat IAM Identitas, atau Anda dapat menghubungkan dan menyinkronkan ke sekumpulan pengguna dan grup di sumber identitas Anda sendiri untuk digunakan di semua Akun AWS dan aplikasi. Untuk informasi tentang Pusat IAM Identitas, lihat [Apa itu Pusat IAM Identitas?](#) di AWS IAM Identity Center Panduan Pengguna.

## Pengguna dan grup IAM

[IAMPengguna](#) adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus untuk satu orang atau aplikasi. Jika memungkinkan, sebaiknya mengandalkan kredensi sementara daripada membuat IAM pengguna yang memiliki kredensi jangka panjang seperti kata sandi dan kunci akses. Namun, jika Anda memiliki kasus penggunaan khusus yang memerlukan kredensi jangka panjang dengan IAM pengguna, kami sarankan Anda memutar kunci akses. Untuk informasi selengkapnya, lihat [Memutar kunci akses secara teratur untuk kasus penggunaan yang memerlukan kredensi jangka panjang](#) di IAMPanduan Pengguna.

[IAMGrup](#) adalah identitas yang menentukan kumpulan IAM pengguna. Anda tidak dapat masuk sebagai grup. Anda dapat menggunakan grup untuk menentukan izin bagi beberapa pengguna sekaligus. Grup mempermudah manajemen izin untuk sejumlah besar pengguna sekaligus. Misalnya, Anda dapat memiliki grup bernama IAMAdmins dan memberikan izin grup tersebut untuk mengelola sumber daya IAM.

Pengguna berbeda dari peran. Pengguna secara unik terkait dengan satu orang atau aplikasi, tetapi peran dimaksudkan untuk dapat digunakan oleh siapa pun yang membutuhkannya. Pengguna memiliki kredensial jangka panjang permanen, tetapi peran memberikan kredensial sementara. Untuk mempelajari lebih lanjut, lihat [Kapan membuat IAM pengguna \(bukan peran\)](#) di Panduan IAM Pengguna.

## IAMperan

[IAMPeran](#) adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus. Ini mirip dengan IAM pengguna, tetapi tidak terkait dengan orang tertentu. Anda dapat mengambil IAM peran sementara dalam AWS Management Console dengan [beralih peran](#). Anda dapat mengambil peran dengan memanggil AWS CLI atau AWS API operasi atau dengan menggunakan kustom URL. Untuk informasi selengkapnya tentang metode penggunaan peran, lihat [Menggunakan IAM peran](#) di Panduan IAM Pengguna.

IAMperan dengan kredensi sementara berguna dalam situasi berikut:

- Akses pengguna terfederasi – Untuk menetapkan izin ke identitas terfederasi, Anda membuat peran dan menentukan izin untuk peran tersebut. Ketika identitas terfederasi mengautentikasi, identitas tersebut terhubung dengan peran dan diberi izin yang ditentukan oleh peran. Untuk informasi tentang peran untuk federasi, lihat [Membuat peran untuk Penyedia Identitas pihak ketiga](#) di Panduan IAM Pengguna. Jika Anda menggunakan Pusat IAM Identitas, Anda mengonfigurasi set izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah diautentikasi, Pusat IAM

Identitas menghubungkan izin yang disetel ke peran. IAM Untuk informasi tentang set izin, lihat [Set izin](#) di AWS IAM Identity Center Panduan Pengguna.

- Izin IAM pengguna sementara — IAM Pengguna atau peran dapat mengambil IAM peran untuk sementara mengambil izin yang berbeda untuk tugas tertentu.
- Akses lintas akun — Anda dapat menggunakan IAM peran untuk memungkinkan seseorang (prinsipal tepercaya) di akun lain mengakses sumber daya di akun Anda. Peran adalah cara utama untuk memberikan akses lintas akun. Namun, dengan beberapa layanan AWS, Anda dapat melampirkan kebijakan langsung ke sumber daya (alih-alih menggunakan peran sebagai proxy). Untuk mempelajari perbedaan antara peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat [Akses sumber daya lintas akun di IAM](#) Panduan Pengguna. IAM
- Akses lintas layanan - Beberapa layanan AWS menggunakan fitur di lain layanan AWS. Misalnya, saat Anda melakukan panggilan dalam suatu layanan, biasanya layanan tersebut menjalankan aplikasi di Amazon EC2 atau menyimpan objek di Amazon S3. Sebuah layanan mungkin melakukannya menggunakan izin prinsipal yang memanggil, menggunakan peran layanan, atau peran terkait layanan.
- Teruskan sesi akses (FAS) — Saat Anda menggunakan IAM pengguna atau peran untuk melakukan tindakan AWS Anda dianggap sebagai kepala sekolah. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FAS menggunakan izin dari prinsipal yang memanggil layanan AWS, dikombinasikan dengan permintaan layanan AWS untuk membuat permintaan ke layanan hilir. FAS permintaan hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain layanan AWS atau sumber daya untuk diselesaikan. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan saat membuat FAS permintaan, lihat [Meneruskan sesi akses](#).
- Peran layanan — Peran layanan adalah [IAM peran](#) yang diasumsikan layanan untuk melakukan tindakan atas nama Anda. IAM Administrator dapat membuat, memodifikasi, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat [Membuat peran untuk mendelegasikan izin ke layanan AWS](#) di Panduan Pengguna IAM.
- Peran terkait layanan — Peran terkait layanan adalah jenis peran layanan yang ditautkan ke layanan AWS. Layanan dapat mengambil peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Akun AWS dan dimiliki oleh layanan. IAM Administrator dapat melihat, tetapi tidak mengedit izin untuk peran terkait layanan.
- Aplikasi yang berjalan di Amazon EC2 — Anda dapat menggunakan IAM peran untuk mengelola kredensial sementara untuk aplikasi yang berjalan pada EC2 instance dan membuat AWS CLI atau AWS API permintaan. Ini lebih baik untuk menyimpan kunci akses dalam EC2 instance.

Untuk menetapkan AWS peran ke sebuah EC2 instance dan membuatnya tersedia untuk semua aplikasinya, Anda membuat profil instance yang dilampirkan ke instance. Profil instance berisi peran dan memungkinkan program yang berjalan pada EC2 instance untuk mendapatkan kredensi sementara. Untuk informasi selengkapnya, lihat [Menggunakan IAM peran untuk memberikan izin ke aplikasi yang berjalan di EC2 instans Amazon](#) di IAMPanduan Pengguna.

Untuk mempelajari apakah akan menggunakan IAM peran atau IAM pengguna, lihat [Kapan membuat IAM peran \(bukan pengguna\)](#) di Panduan IAM Pengguna.

## Mengelola akses menggunakan kebijakan

Anda mengontrol akses di AWS dengan membuat kebijakan dan melampirkannya AWS identitas atau sumber daya. Kebijakan adalah objek di AWS bahwa, ketika dikaitkan dengan identitas atau sumber daya, mendefinisikan izin mereka. AWS mengevaluasi kebijakan ini ketika prinsipal (pengguna, pengguna root, atau sesi peran) membuat permintaan. Izin dalam kebijakan menentukan apakah permintaan diizinkan atau ditolak. Sebagian besar kebijakan disimpan di AWS sebagai JSON dokumen. Untuk informasi selengkapnya tentang struktur dan isi dokumen JSON kebijakan, lihat [Ringkasan JSON kebijakan](#) di Panduan IAM Pengguna.

Administrator dapat menggunakan AWS JSONkebijakan untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

Secara default, pengguna dan peran tidak memiliki izin. Untuk memberikan izin kepada pengguna untuk melakukan tindakan pada sumber daya yang mereka butuhkan, IAM administrator dapat membuat IAM kebijakan. Administrator kemudian dapat menambahkan IAM kebijakan ke peran, dan pengguna dapat mengambil peran.

IAMkebijakan menentukan izin untuk tindakan terlepas dari metode yang Anda gunakan untuk melakukan operasi. Misalnya, anggaplah Anda memiliki kebijakan yang mengizinkan tindakan `iam:GetRole`. Pengguna dengan kebijakan tersebut dapat memperoleh informasi peran dari AWS Management Console, AWS CLI, atau AWS API.

## Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan JSON izin yang dapat Anda lampirkan ke identitas, seperti pengguna, grup IAM pengguna, atau peran. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi

seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat [Membuat IAM kebijakan di Panduan](#) Pengguna. IAM

Kebijakan berbasis identitas dapat dikategorikan lebih lanjut sebagai kebijakan inline atau kebijakan yang dikelola. Kebijakan inline disematkan langsung ke satu pengguna, grup, atau peran. Kebijakan terkelola adalah kebijakan mandiri yang dapat Anda lampirkan ke beberapa pengguna, grup, dan peran di Akun AWS. Kebijakan terkelola meliputi AWS kebijakan terkelola dan kebijakan yang dikelola pelanggan. Untuk mempelajari cara memilih antara kebijakan terkelola atau kebijakan sebaris, lihat [Memilih antara kebijakan terkelola dan kebijakan sebaris](#) di IAMPanduan Pengguna.

## Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen JSON kebijakan yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan IAM peran dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus [menentukan prinsipal](#) dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau layanan AWS.

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan AWS kebijakan 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 [Ikhtisar daftar kontrol akses \(ACL\)](#) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

## Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang kurang umum. Jenis-jenis kebijakan ini dapat mengatur izin maksimum yang diberikan kepada Anda oleh jenis kebijakan yang lebih umum.

- **Batas izin** — Batas izin adalah fitur lanjutan tempat Anda menetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas (pengguna atau peran). IAM

IAM Anda dapat menetapkan batasan izin untuk suatu entitas. Izin yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas milik entitas dan batasan izinnya. Kebijakan berbasis sumber daya yang menentukan pengguna atau peran dalam bidang `Principal` tidak dibatasi oleh batasan izin. Penolakan eksplisit dalam salah satu kebijakan ini akan menggantikan pemberian izin. Untuk informasi selengkapnya tentang batas izin, lihat [Batas izin untuk IAM entitas](#) di IAM Panduan Pengguna.

- Kebijakan kontrol layanan (SCPs) — SCPs adalah JSON kebijakan yang menentukan izin maksimum untuk organisasi atau unit organisasi (OU) di AWS Organizations. AWS Organizations adalah layanan untuk mengelompokkan dan mengelola beberapa Akun AWS yang dimiliki bisnis Anda. Jika Anda mengaktifkan semua fitur dalam suatu organisasi, maka Anda dapat menerapkan kebijakan kontrol layanan (SCPs) ke salah satu atau semua akun Anda. SCP Membatasi izin untuk entitas di akun anggota, termasuk masing-masing Pengguna root akun AWS. Untuk informasi selengkapnya tentang Organizations dan SCPs, lihat [Kebijakan kontrol layanan](#) di AWS Organizations Panduan Pengguna.
- Kebijakan sesi – Kebijakan sesi adalah kebijakan lanjutan yang Anda berikan sebagai parameter ketika Anda membuat sesi sementara secara programatis untuk peran atau pengguna terfederasi. Izin sesi yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas pengguna atau peran dan kebijakan sesi. Izin juga bisa datang dari kebijakan berbasis sumber daya. Penolakan secara tegas dalam salah satu kebijakan ini membatalkan izin. Untuk informasi selengkapnya, lihat [Kebijakan sesi](#) di Panduan IAM Pengguna.

## Berbagai jenis kebijakan

Ketika beberapa jenis kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari caranya AWS menentukan apakah akan mengizinkan permintaan saat beberapa jenis kebijakan terlibat, lihat [Logika evaluasi kebijakan](#) di Panduan IAM Pengguna.

## Bagaimana Deadline Cloud bekerja dengan IAM

Sebelum Anda menggunakan IAM untuk mengelola akses ke Deadline Cloud, pelajari IAM fitur apa saja yang tersedia untuk digunakan dengan Deadline Cloud.

## IAMfitur yang dapat Anda gunakan dengan AWS Batas Waktu Cloud

IAMfitur	Dukungan Batas Waktu Cloud
<a href="#">Kebijakan berbasis identitas</a>	Ya
<a href="#">Kebijakan berbasis sumber daya</a>	Tidak
<a href="#">Tindakan kebijakan</a>	Ya
<a href="#">Sumber daya kebijakan</a>	Ya
<a href="#">kunci-kunci persyaratan kebijakan (spesifik layanan)</a>	Ya
<a href="#">ACLs</a>	Tidak
<a href="#">ABAC(tag dalam kebijakan)</a>	Ya
<a href="#">Kredensial sementara</a>	Ya
<a href="#">Teruskan sesi akses (FAS)</a>	Ya
<a href="#">Peran layanan</a>	Ya
<a href="#">Peran terkait layanan</a>	Tidak

Untuk mendapatkan tampilan tingkat tinggi tentang bagaimana Deadline Cloud dan lainnya layanan AWS bekerja dengan sebagian besar IAM fitur, lihat [AWS layanan yang bekerja dengan IAM](#) dalam Panduan IAM Pengguna.

## Kebijakan berbasis identitas untuk Deadline Cloud

Mendukung kebijakan berbasis identitas: Ya

Kebijakan berbasis identitas adalah dokumen kebijakan JSON izin yang dapat Anda lampirkan ke identitas, seperti pengguna, grup IAM pengguna, atau peran. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat [Membuat IAM kebijakan di Panduan](#) Pengguna. IAM

Dengan kebijakan IAM berbasis identitas, Anda dapat menentukan tindakan dan sumber daya yang diizinkan atau ditolak serta kondisi di mana tindakan diizinkan atau ditolak. Anda tidak dapat menentukan secara spesifik prinsipal dalam sebuah kebijakan berbasis identitas karena prinsipal berlaku bagi pengguna atau peran yang melekat kepadanya. Untuk mempelajari semua elemen yang dapat Anda gunakan dalam JSON kebijakan, lihat [referensi elemen IAM JSON kebijakan](#) di Panduan IAM Pengguna.

Contoh kebijakan berbasis identitas untuk Deadline Cloud

Untuk melihat contoh kebijakan berbasis identitas Deadline Cloud, lihat. [Contoh kebijakan berbasis identitas untuk Deadline Cloud](#)

## Kebijakan berbasis sumber daya dalam Deadline Cloud

Mendukung kebijakan berbasis sumber daya: Tidak

Kebijakan berbasis sumber daya adalah dokumen JSON kebijakan yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan IAM peran dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus [menentukan prinsipal](#) dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau layanan AWS.

Untuk mengaktifkan akses lintas akun, Anda dapat menentukan seluruh akun atau IAM entitas di akun lain sebagai prinsipal dalam kebijakan berbasis sumber daya. Menambahkan prinsipal akun silang ke kebijakan berbasis sumber daya hanya setengah dari membangun hubungan kepercayaan. Ketika kepala sekolah dan sumber daya berbeda Akun AWS, IAM administrator di akun tepercaya juga harus memberikan izin entitas utama (pengguna atau peran) untuk mengakses sumber daya. Mereka memberikan izin dengan melampirkan kebijakan berbasis identitas kepada entitas. Namun, jika kebijakan berbasis sumber daya memberikan akses ke prinsipal dalam akun yang sama, tidak diperlukan kebijakan berbasis identitas tambahan. Untuk informasi selengkapnya, lihat [Akses sumber daya lintas akun IAM di](#) Panduan IAM Pengguna.

## Tindakan kebijakan untuk Deadline Cloud

Mendukung tindakan kebijakan: Ya

Administrator dapat menggunakan AWS JSONkebijakan untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

ActionElemen JSON kebijakan menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam kebijakan. Tindakan kebijakan biasanya memiliki nama yang sama dengan yang terkait AWS APIoperasi. Ada beberapa pengecualian, seperti tindakan khusus izin yang tidak memiliki operasi yang cocok. API Ada juga beberapa operasi yang memerlukan beberapa tindakan dalam suatu kebijakan. Tindakan tambahan ini disebut tindakan dependen.

Menyertakan tindakan dalam kebijakan untuk memberikan izin untuk melakukan operasi terkait.

Untuk melihat daftar tindakan Deadline Cloud, lihat [Tindakan yang ditentukan oleh AWS Batas Waktu Cloud](#) dalam Referensi Otorisasi Layanan.

Tindakan kebijakan di Deadline Cloud menggunakan awalan berikut sebelum tindakan:

```
deadline
```

Untuk menetapkan secara spesifik beberapa tindakan dalam satu pernyataan, pisahkan tindakan tersebut dengan koma.

```
"Action": [  
  "deadline:action1",  
  "deadline:action2"  
]
```

Untuk melihat contoh kebijakan berbasis identitas Deadline Cloud, lihat. [Contoh kebijakan berbasis identitas untuk Deadline Cloud](#)

## Sumber daya kebijakan untuk Deadline Cloud

Mendukung sumber daya kebijakan: Ya

Administrator dapat menggunakan AWS JSONkebijakan untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

Elemen `Resource` JSON kebijakan menentukan objek atau objek yang tindakan tersebut berlaku. Pernyataan harus menyertakan elemen `Resource` atau `NotResource`. Sebagai praktik terbaik, tentukan sumber daya menggunakan [Amazon Resource Name \(ARN\)](#). Anda dapat melakukan ini untuk tindakan yang mendukung jenis sumber daya tertentu, yang dikenal sebagai izin tingkat sumber daya.

Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, misalnya operasi pencantuman, gunakan wildcard (\*) untuk menunjukkan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

```
"Resource": "*" 
```

Untuk melihat daftar jenis sumber daya Cloud Deadline dan jenisnya ARNs, lihat [Sumber daya yang ditentukan oleh AWS Batas Waktu Cloud](#) dalam Referensi Otorisasi Layanan. Untuk mempelajari tindakan mana yang dapat Anda tentukan ARN dari setiap sumber daya, lihat [Tindakan yang ditentukan oleh AWS Batas Waktu Cloud](#).

Untuk melihat contoh kebijakan berbasis identitas Deadline Cloud, lihat [Contoh kebijakan berbasis identitas untuk Deadline Cloud](#)

## Kunci kondisi kebijakan untuk Deadline Cloud

Mendukung kunci kondisi kebijakan khusus layanan: Ya

Administrator dapat menggunakan AWS JSON kebijakan untuk menentukan siapa yang memiliki akses ke apa. Yaitu, di mana utama dapat melakukan tindakan pada sumber daya, dan dalam kondisi apa.

Elemen `Condition` (atau blok `Condition`) akan memungkinkan Anda menentukan kondisi yang menjadi dasar suatu pernyataan berlaku. Elemen `Condition` bersifat opsional. Anda dapat membuat ekspresi bersyarat yang menggunakan [operator kondisi](#), misalnya sama dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta.

Jika Anda menentukan beberapa `Condition` elemen dalam pernyataan, atau beberapa kunci dalam satu `Condition` elemen, AWS mengevaluasi mereka menggunakan AND operasi logis. Jika Anda menentukan beberapa nilai untuk satu kunci kondisi, AWS mengevaluasi kondisi menggunakan OR operasi logis. Semua kondisi harus dipenuhi sebelum izin pernyataan diberikan.

Anda juga dapat menggunakan variabel placeholder saat menentukan kondisi. Misalnya, Anda dapat memberikan izin IAM pengguna untuk mengakses sumber daya hanya jika ditandai dengan nama IAM pengguna mereka. Untuk informasi selengkapnya, lihat [elemen IAM kebijakan: variabel dan tag](#) di Panduan IAM Pengguna.

AWS mendukung kunci kondisi global dan kunci kondisi khusus layanan. Untuk melihat semua AWS kunci kondisi global, lihat [AWS kunci konteks kondisi global](#) di Panduan IAM Pengguna.

Untuk melihat daftar kunci kondisi Deadline Cloud, lihat Kunci [kondisi untuk AWS Batas Waktu Cloud](#) dalam Referensi Otorisasi Layanan. Untuk mempelajari tindakan dan sumber daya yang dapat Anda gunakan kunci kondisi, lihat [Tindakan yang ditentukan oleh AWS Batas Waktu Cloud](#).

Untuk melihat contoh kebijakan berbasis identitas Deadline Cloud, lihat. [Contoh kebijakan berbasis identitas untuk Deadline Cloud](#)

## ACLs di Deadline Cloud

Mendukung ACLs: Tidak

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

## ABAC dengan Deadline Cloud

Mendukung ABAC (tag dalam kebijakan): Ya

Attribute-based access control (ABAC) adalah strategi otorisasi yang mendefinisikan izin berdasarkan atribut. Masuk AWS, atribut ini disebut tag. Anda dapat melampirkan tag ke IAM entitas (pengguna atau peran) dan ke banyak AWS sumber daya. Menandai entitas dan sumber daya adalah langkah pertama dari ABAC. Kemudian Anda merancang ABAC kebijakan untuk mengizinkan operasi ketika tag prinsipal cocok dengan tag pada sumber daya yang mereka coba akses.

ABAC membantu dalam lingkungan yang berkembang pesat dan membantu dengan situasi di mana manajemen kebijakan menjadi rumit.

Untuk mengendalikan akses berdasarkan tag, berikan informasi tentang tag di [elemen kondisi](#) dari kebijakan menggunakan kunci kondisi `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, atau `aws:TagKeys`.

Jika sebuah layanan mendukung ketiga kunci kondisi untuk setiap jenis sumber daya, nilainya adalah Ya untuk layanan tersebut. Jika suatu layanan mendukung ketiga kunci kondisi untuk hanya beberapa jenis sumber daya, nilainya adalah Parsial.

Untuk informasi lebih lanjut tentang ABAC, lihat [Apa itu ABAC?](#) dalam IAM User Guide. Untuk melihat tutorial dengan langkah-langkah penyiapan ABAC, lihat [Menggunakan kontrol akses berbasis atribut \(ABAC\)](#) di IAM Panduan Pengguna.

## Menggunakan kredensial sementara dengan Deadline Cloud

Mendukung kredensi sementara: Ya

Beberapa layanan AWS tidak berfungsi saat Anda masuk menggunakan kredensial sementara. Untuk informasi tambahan, termasuk yang layanan AWS bekerja dengan kredensi sementara, lihat [layanan AWS yang bekerja dengan IAM](#) dalam Panduan IAM Pengguna.

Anda menggunakan kredensi sementara jika Anda masuk ke AWS Management Console menggunakan metode apa pun kecuali nama pengguna dan kata sandi. Misalnya, ketika Anda mengakses AWS menggunakan tautan single sign-on (SSO) perusahaan Anda, proses itu secara otomatis membuat kredensi sementara. Anda juga akan secara otomatis membuat kredensial sementara ketika Anda masuk ke konsol sebagai seorang pengguna lalu beralih peran. Untuk informasi selengkapnya tentang beralih peran, lihat [Beralih ke peran \(konsol\)](#) di Panduan IAM Pengguna.

Anda dapat secara manual membuat kredensi sementara menggunakan AWS CLI atau AWS API. Anda kemudian dapat menggunakan kredensi sementara tersebut untuk mengakses AWS. AWS merekomendasikan agar Anda secara dinamis menghasilkan kredensi sementara alih-alih menggunakan kunci akses jangka panjang. Untuk informasi selengkapnya, lihat [Kredensi keamanan sementara](#) di IAM.

## Teruskan sesi akses untuk Deadline Cloud

Mendukung sesi akses maju (FAS): Ya

Saat Anda menggunakan IAM pengguna atau peran untuk melakukan tindakan AWS Anda dianggap sebagai kepala sekolah. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FAS menggunakan izin dari prinsipal yang memanggil layanan AWS, dikombinasikan dengan permintaan layanan AWS untuk membuat permintaan ke layanan hilir. FAS permintaan hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain layanan AWS.

atau sumber daya untuk diselesaikan. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan saat membuat FAS permintaan, lihat [Meneruskan sesi akses](#).

## Peran layanan untuk Deadline Cloud

Mendukung peran layanan: Ya

Peran layanan adalah [IAMperan](#) yang diasumsikan layanan untuk melakukan tindakan atas nama Anda. IAMAdministrator dapat membuat, memodifikasi, dan menghapus peran layanan dari dalamIAM. Untuk informasi selengkapnya, lihat [Membuat peran untuk mendelegasikan izin ke layanan AWS](#) di Panduan Pengguna IAM.

### Warning

Mengubah izin untuk peran layanan dapat merusak fungsionalitas Deadline Cloud. Edit peran layanan hanya jika Deadline Cloud memberikan panduan untuk melakukannya.

## Peran terkait layanan untuk Deadline Cloud

Mendukung peran terkait layanan: Tidak

Peran terkait layanan adalah jenis peran layanan yang ditautkan ke layanan AWS. Layanan dapat mengambil peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Akun AWS dan dimiliki oleh layanan. IAMAdministrator dapat melihat, tetapi tidak mengedit izin untuk peran terkait layanan.

Untuk detail tentang membuat atau mengelola peran terkait layanan, lihat [AWS layanan yang bekerja dengan IAM](#). Cari layanan dalam tabel yang memiliki Yes di kolom Peran terkait layanan. Pilih tautan Ya untuk melihat dokumentasi peran terkait layanan untuk layanan tersebut.

## Contoh kebijakan berbasis identitas untuk Deadline Cloud

Secara default, pengguna dan peran tidak memiliki izin untuk membuat atau memodifikasi sumber daya Deadline Cloud. Mereka juga tidak dapat melakukan tugas dengan menggunakan AWS Management Console, AWS Command Line Interface (AWS CLI), atau AWS API. Untuk memberikan izin kepada pengguna untuk melakukan tindakan pada sumber daya yang mereka butuhkan, IAM administrator dapat membuat IAM kebijakan. Administrator kemudian dapat menambahkan IAM kebijakan ke peran, dan pengguna dapat mengambil peran.

Untuk mempelajari cara membuat kebijakan IAM berbasis identitas menggunakan contoh dokumen kebijakan ini, lihat [Membuat JSON IAM kebijakan di Panduan Pengguna IAM](#).

Untuk detail tentang tindakan dan jenis sumber daya yang ditentukan oleh Deadline Cloud, termasuk format ARNs untuk setiap jenis sumber daya, lihat Kunci [tindakan, sumber daya, dan kondisi untuk AWS Batas Waktu Cloud](#) dalam Referensi Otorisasi Layanan.

## Topik

- [Praktik terbaik kebijakan](#)
- [Menggunakan konsol Deadline Cloud](#)
- [Kebijakan untuk mengirimkan pekerjaan ke antrian](#)
- [Kebijakan untuk mengizinkan pembuatan titik akhir lisensi](#)
- [Kebijakan untuk memungkinkan pemantauan antrian pertanian tertentu](#)

## Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus sumber daya Deadline Cloud di akun Anda. Tindakan ini dapat menimbulkan biaya untuk Akun AWS. Saat Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi berikut:

- Memulai dengan AWS kebijakan terkelola dan beralih ke izin hak istimewa terkecil — Untuk memulai pemberian izin kepada pengguna dan beban kerja Anda, gunakan AWS kebijakan terkelola yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di Akun AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan mendefinisikan AWS kebijakan yang dikelola pelanggan yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, silakan lihat [AWS kebijakan terkelola](#) atau [AWS kebijakan terkelola untuk fungsi pekerjaan](#) di Panduan IAM Pengguna.
- Menerapkan izin hak istimewa paling sedikit — Saat Anda menetapkan izin dengan IAM kebijakan, berikan hanya izin yang diperlukan untuk melakukan tugas. Anda melakukannya dengan mendefinisikan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, yang juga dikenal sebagai izin dengan hak akses paling rendah. Untuk informasi selengkapnya tentang penggunaan IAM untuk menerapkan izin, lihat [Kebijakan dan izin IAM di IAM](#) Panduan Pengguna.
- Gunakan ketentuan dalam IAM kebijakan untuk membatasi akses lebih lanjut — Anda dapat menambahkan kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya.

Misalnya, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakan SSL. Anda juga dapat menggunakan ketentuan untuk memberikan akses ke tindakan layanan jika digunakan melalui tindakan tertentu layanan AWS, seperti AWS CloudFormation. Untuk informasi selengkapnya, lihat [elemen IAM JSON kebijakan: Kondisi](#) dalam Panduan IAM Pengguna.

- Gunakan IAM Access Analyzer untuk memvalidasi IAM kebijakan Anda guna memastikan izin yang aman dan fungsional — IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan mematuhi bahasa IAM kebijakan ( ) JSON dan praktik terbaik. IAM Access Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat [Validasi kebijakan IAM Access Analyzer](#) di IAM Panduan Pengguna.
- Memerlukan otentikasi multi-faktor (MFA) - Jika Anda memiliki skenario yang mengharuskan IAM pengguna atau pengguna root di Akun AWS, nyalakan MFA untuk keamanan tambahan. Untuk meminta MFA kapan API operasi dipanggil, tambahkan MFA kondisi ke kebijakan Anda. Untuk informasi selengkapnya, lihat [Mengonfigurasi API akses MFA yang dilindungi](#) di IAM Panduan Pengguna.

Untuk informasi selengkapnya tentang praktik terbaik di IAM, lihat [Praktik terbaik keamanan IAM di](#) Panduan IAM Pengguna.

## Menggunakan konsol Deadline Cloud

Untuk mengakses AWS Konsol Cloud batas waktu, Anda harus memiliki set izin minimum. Izin ini harus memungkinkan Anda untuk membuat daftar dan melihat detail tentang sumber daya Batas Waktu Cloud di Akun AWS. Jika Anda membuat kebijakan berbasis identitas yang lebih ketat daripada izin minimum yang diperlukan, konsol tidak akan berfungsi sebagaimana dimaksud untuk entitas (pengguna atau peran) dengan kebijakan tersebut.

Anda tidak perlu mengizinkan izin konsol minimum untuk pengguna yang melakukan panggilan hanya ke AWS CLI atau AWS API. Sebagai gantinya, izinkan akses hanya ke tindakan yang cocok dengan API operasi yang mereka coba lakukan.

Untuk memastikan bahwa pengguna dan peran masih dapat menggunakan konsol Deadline Cloud, lampirkan juga Deadline Cloud atau *ConsoleAccessReadOnly* AWS kebijakan yang dikelola untuk entitas. Untuk informasi selengkapnya, lihat [Menambahkan izin ke pengguna](#) di Panduan IAM Pengguna.

## Kebijakan untuk mengirimkan pekerjaan ke antrian

Dalam contoh ini, Anda membuat kebijakan cakupan bawah yang memberikan izin untuk mengirimkan pekerjaan ke antrian tertentu di peternakan tertentu.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "SubmitJobsFarmAndQueue",
      "Effect": "Allow",
      "Action": "deadline:CreateJob",
      "Resource": "arn:aws:deadline:REGION:ACCOUNT_ID:farm/FARM_A/queue/QUEUE_B/
job/*"
    }
  ]
}
```

## Kebijakan untuk mengizinkan pembuatan titik akhir lisensi

Dalam contoh ini, Anda membuat kebijakan cakupan bawah yang memberikan izin yang diperlukan untuk membuat dan mengelola titik akhir lisensi. Gunakan kebijakan ini untuk membuat titik akhir lisensi untuk yang VPC terkait dengan peternakan Anda.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "SID": "CreateLicenseEndpoint",
    "Effect": "Allow",
    "Action": [
      "deadline:CreateLicenseEndpoint",
      "deadline>DeleteLicenseEndpoint",
      "deadline:GetLicenseEndpoint",
      "deadline:UpdateLicenseEndpoint",
      "deadline>ListLicenseEndpoints",
      "deadline:PutMeteredProduct",
      "deadline>DeleteMeteredProduct",
      "deadline>ListMeteredProducts",
      "deadline>ListAvailableMeteredProducts",
      "ec2:CreateVpcEndpoint",
      "ec2:DescribeVpcEndpoints",
      "ec2>DeleteVpcEndpoints"
    ]
  }
]
```

```
    ],  
    "Resource": "*"    
  }  
}
```

## Kebijakan untuk memungkinkan pemantauan antrian pertanian tertentu

Dalam contoh ini, Anda membuat kebijakan cakupan bawah yang memberikan izin untuk memantau pekerjaan dalam antrian tertentu untuk peternakan tertentu.

```
{  
  "Version": "2012-10-17",  
  "Statement": [{  
    "Sid": "MonitorJobsFarmAndQueue",  
    "Effect": "Allow",  
    "Action": [  
      "deadline:SearchJobs",  
      "deadline:ListJobs",  
      "deadline:GetJob",  
      "deadline:SearchSteps",  
      "deadline:ListSteps",  
      "deadline:ListStepConsumers",  
      "deadline:ListStepDependencies",  
      "deadline:GetStep",  
      "deadline:SearchTasks",  
      "deadline:ListTasks",  
      "deadline:GetTask",  
      "deadline:ListSessions",  
      "deadline:GetSession",  
      "deadline:ListSessionActions",  
      "deadline:GetSessionAction"  
    ],  
    "Resource": [  
      "arn:aws:deadline:REGION:123456789012:farm/FARM_A/queue/QUEUE_B",  
      "arn:aws:deadline:REGION:123456789012:farm/FARM_A/queue/QUEUE_B/*"  
    ]  
  }]  
}
```

## AWS kebijakan terkelola untuk Deadline Cloud

Sesi AWS kebijakan terkelola adalah kebijakan mandiri yang dibuat dan dikelola oleh AWS. AWS Kebijakan terkelola dirancang untuk memberikan izin bagi banyak kasus penggunaan umum sehingga Anda dapat mulai menetapkan izin kepada pengguna, grup, dan peran.

Perlu diingat bahwa AWS kebijakan terkelola mungkin tidak memberikan izin hak istimewa paling sedikit untuk kasus penggunaan spesifik Anda karena tersedia untuk semua AWS pelanggan untuk digunakan. Kami menyarankan Anda untuk mengurangi izin lebih lanjut dengan menentukan [kebijakan yang dikelola pelanggan](#) yang khusus untuk kasus penggunaan Anda.

Anda tidak dapat mengubah izin yang ditentukan dalam AWS kebijakan terkelola. Jika AWS memperbarui izin yang ditentukan dalam AWS kebijakan terkelola, pembaruan memengaruhi semua identitas utama (pengguna, grup, dan peran) yang dilampirkan kebijakan tersebut. AWS kemungkinan besar akan memperbarui AWS kebijakan terkelola saat baru layanan AWS diluncurkan atau API operasi baru tersedia untuk layanan yang ada.

Untuk informasi selengkapnya, silakan lihat [AWS kebijakan terkelola](#) dalam Panduan IAM Pengguna.

## AWS kebijakan terkelola: AWSDeadlineCloud-FleetWorker

Anda dapat melampirkan AWSDeadlineCloud-FleetWorker kebijakan ke AWS Identity and Access Management (IAM) identitas.

Kebijakan ini memberi pekerja di armada ini izin yang diperlukan untuk terhubung dan menerima tugas dari layanan.

Detail izin

Kebijakan ini mencakup izin berikut:

- `deadline`— Memungkinkan kepala sekolah untuk mengelola pekerja dalam armada.

Untuk JSON daftar detail kebijakan, lihat [AWSDeadlineCloud- FleetWorker](#) di panduan referensi Kebijakan AWS Terkelola.

## AWS kebijakan terkelola: AWSDeadlineCloud-WorkerHost

Anda dapat melampirkan AWSDeadlineCloud-WorkerHost kebijakan ke IAM identitas Anda.

Kebijakan ini memberikan izin yang diperlukan untuk awalnya terhubung ke layanan. Ini dapat digunakan sebagai profil instans Amazon Elastic Compute Cloud (AmazonEC2).

#### Detail izin

Kebijakan ini mencakup izin berikut:

- `deadline`— Memungkinkan kepala sekolah untuk membuat pekerja.

Untuk JSON daftar detail kebijakan, lihat [AWSDeadlineCloud- WorkerHost](#) di panduan referensi Kebijakan AWS Terkelola.

#### AWS kebijakan terkelola: AWSDeadlineCloud-UserAccessFarms

Anda dapat melampirkan `AWSDeadlineCloud-UserAccessFarms` kebijakan ke IAM identitas Anda.

Kebijakan ini memungkinkan pengguna untuk mengakses data pertanian berdasarkan peternakan tempat mereka menjadi anggota dan tingkat keanggotaan mereka.

#### Detail izin

Kebijakan ini mencakup izin berikut:

- `deadline`— Memungkinkan pengguna untuk mengakses data pertanian.
- `ec2`— Memungkinkan pengguna untuk melihat detail tentang jenis EC2 instans Amazon.
- `identitystore`— Memungkinkan pengguna untuk melihat nama pengguna dan grup.

Untuk JSON daftar detail kebijakan, lihat [AWSDeadlineCloud- UserAccessFarms](#) di panduan referensi Kebijakan AWS Terkelola.

#### AWS kebijakan terkelola: AWSDeadlineCloud-UserAccessFleets

Anda dapat melampirkan `AWSDeadlineCloud-UserAccessFleets` kebijakan ke IAM identitas Anda.

Kebijakan ini memungkinkan pengguna untuk mengakses data armada berdasarkan peternakan tempat mereka menjadi anggota dan tingkat keanggotaan mereka.

#### Detail izin

Kebijakan ini mencakup izin berikut:

- `deadline`— Memungkinkan pengguna untuk mengakses data pertanian.
- `ec2`— Memungkinkan pengguna untuk melihat detail tentang jenis EC2 instans Amazon.
- `identitystore`— Memungkinkan pengguna untuk melihat nama pengguna dan grup.

Untuk JSON daftar detail kebijakan, lihat [AWSDeadlineCloud- UserAccessFleets](#) di panduan referensi Kebijakan AWS Terkelola.

### AWS kebijakan terkelola: AWSDeadlineCloud-UserAccessJobs

Anda dapat melampirkan `AWSDeadlineCloud-UserAccessJobs` kebijakan ke IAM identitas Anda.

Kebijakan ini memungkinkan pengguna untuk mengakses data pekerjaan berdasarkan peternakan tempat mereka menjadi anggota dan tingkat keanggotaan mereka.

Detail izin

Kebijakan ini mencakup izin berikut:

- `deadline`— Memungkinkan pengguna untuk mengakses data pertanian.
- `ec2`— Memungkinkan pengguna untuk melihat detail tentang jenis EC2 instans Amazon.
- `identitystore`— Memungkinkan pengguna untuk melihat nama pengguna dan grup.

Untuk JSON daftar detail kebijakan, lihat [AWSDeadlineCloud- UserAccessJobs](#) di panduan referensi Kebijakan AWS Terkelola.

### AWS kebijakan terkelola: AWSDeadlineCloud-UserAccessQueues

Anda dapat melampirkan `AWSDeadlineCloud-UserAccessQueues` kebijakan ke IAM identitas Anda.

Kebijakan ini memungkinkan pengguna untuk mengakses data antrian berdasarkan peternakan tempat mereka menjadi anggota dan tingkat keanggotaan mereka.

Detail izin

Kebijakan ini mencakup izin berikut:

- `deadline`— Memungkinkan pengguna untuk mengakses data pertanian.

- `ec2`— Memungkinkan pengguna untuk melihat detail tentang jenis EC2 instans Amazon.
- `identitystore`— Memungkinkan pengguna untuk melihat nama pengguna dan grup.

Untuk JSON daftar detail kebijakan, lihat [AWSDeadlineCloud- UserAccessQueues](#) di panduan referensi Kebijakan AWS Terkelola.

## Tenggat waktu pembaruan Cloud ke AWS kebijakan terkelola

Lihat detail tentang pembaruan AWS kebijakan terkelola untuk Deadline Cloud sejak layanan ini mulai melacak perubahan ini. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan RSS feed di halaman riwayat Dokumen Cloud Batas Waktu.

Perubahan	Deskripsi	Tanggal
Deadline Cloud mulai melacak perubahan	Deadline Cloud mulai melacak perubahan AWS kebijakan terkelola.	April 2, 2024

## Pemecahan Masalah AWS Batas waktu Identitas dan akses Cloud

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan Deadline Cloud dan IAM.

### Topik

- [Saya tidak berwenang untuk melakukan tindakan di Deadline Cloud](#)
- [Saya tidak berwenang untuk melakukan iam: PassRole](#)
- [Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses sumber daya Deadline Cloud saya](#)

## Saya tidak berwenang untuk melakukan tindakan di Deadline Cloud

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 `mateojackson` IAM pengguna mencoba menggunakan konsol untuk melihat detail tentang `my-example-widget` sumber daya fiksi tetapi tidak memiliki izin `deadline:GetWidget` fiksi.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
deadline: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 `deadline:GetWidget`.

Jika Anda membutuhkan bantuan, hubungi AWS administrator. 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 ke Deadline Cloud.

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 IAM pengguna bernama `marymajor` mencoba menggunakan konsol untuk melakukan tindakan di Deadline Cloud. 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 membutuhkan bantuan, hubungi AWS administrator. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

## Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses sumber daya Deadline Cloud 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 mengetahui apakah Deadline Cloud mendukung fitur-fitur ini, lihat [Bagaimana Deadline Cloud bekerja dengan IAM](#).
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda di seluruh Akun AWS yang Anda miliki, lihat [Menyediakan akses ke IAM pengguna di pengguna lain Akun AWS yang Anda miliki](#) di Panduan IAM Pengguna.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda ke pihak ketiga Akun AWS, lihat [Menyediakan akses ke Akun AWS dimiliki oleh pihak ketiga](#) dalam Panduan IAM Pengguna.
- Untuk mempelajari cara menyediakan akses melalui federasi identitas, lihat [Menyediakan akses ke pengguna yang diautentikasi secara eksternal \(federasi identitas\) di 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](#) Panduan Pengguna. IAM

## Validasi kepatuhan untuk Deadline Cloud

Untuk mengetahui apakah layanan AWS berada dalam lingkup program kepatuhan tertentu, lihat [layanan AWS dalam Lingkup oleh Program Kepatuhan](#) dan pilih program kepatuhan yang Anda minati. Untuk informasi umum, lihat [AWS Program Kepatuhan](#) .

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. AWS menyediakan sumber daya berikut untuk membantu kepatuhan:

- [Panduan Memulai Cepat Keamanan dan Kepatuhan — Panduan](#) penerapan ini membahas pertimbangan arsitektur dan memberikan langkah-langkah untuk menerapkan lingkungan dasar AWS yang berfokus pada keamanan dan kepatuhan.
- [Arsitektur untuk HIPAA Keamanan dan Kepatuhan di Amazon Web Services](#) — Whitepaper ini menjelaskan bagaimana perusahaan dapat menggunakan AWS untuk membuat aplikasi HIPAA yang memenuhi syarat.

**Note**

Tidak semua layanan AWS HIPAA memenuhi syarat. Untuk informasi selengkapnya, lihat [Referensi Layanan yang HIPAA Memenuhi Syarat](#).

- [AWS Sumber Daya Kepatuhan](#) — Kumpulan buku kerja dan panduan ini mungkin berlaku untuk industri dan lokasi Anda.
- [AWS Panduan Kepatuhan Pelanggan](#) - Memahami model tanggung jawab bersama melalui lensa kepatuhan. Panduan merangkum praktik terbaik untuk mengamankan layanan AWS dan memetakan panduan untuk kontrol keamanan di berbagai kerangka kerja (termasuk Institut Standar dan Teknologi Nasional (NIST), Dewan Standar Keamanan Industri Kartu Pembayaran (PCI), dan Organisasi Internasional untuk Standardisasi (ISO)).
- [Mengevaluasi Sumber Daya dengan Aturan](#) di AWS Config Panduan Pengembang — The AWS Config Layanan menilai seberapa baik konfigurasi sumber daya Anda mematuhi praktik internal, pedoman industri, dan peraturan.
- [AWS Security Hub](#)— Ini layanan AWS memberikan pandangan komprehensif tentang keadaan keamanan Anda di dalam AWS. Security Hub menggunakan kontrol keamanan untuk mengevaluasi AWS sumber daya dan untuk memeriksa kepatuhan Anda terhadap standar industri keamanan dan praktik terbaik. Untuk daftar layanan dan kontrol yang didukung, lihat [Referensi kontrol Security Hub](#).
- [Amazon GuardDuty](#) - Ini layanan AWS Mendeteksi potensi ancaman terhadap Akun AWS, beban kerja, kontainer, dan data dengan memantau lingkungan Anda untuk aktivitas yang mencurigakan dan berbahaya. GuardDuty dapat membantu Anda mengatasi berbagai persyaratan kepatuhan, seperti PCIDSS, dengan memenuhi persyaratan deteksi intrusi yang diamanatkan oleh kerangka kerja kepatuhan tertentu.
- [AWS Audit Manager](#)— Ini layanan AWS membantu Anda terus mengaudit AWS penggunaan untuk menyederhanakan cara Anda mengelola risiko dan kepatuhan terhadap peraturan dan standar industri.

## Ketahanan di Deadline Cloud

Bagian AWS Infrastruktur global dibangun di sekitar Wilayah AWS dan Availability Zone. Wilayah 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 lebih lanjut tentang Wilayah AWS dan Availability Zone, lihat [AWS Infrastruktur Global](#).

AWS Deadline Cloud tidak mencadangkan data yang disimpan di bucket S3 lampiran pekerjaan Anda. [Anda dapat mengaktifkan pencadangan data lampiran pekerjaan Anda menggunakan mekanisme pencadangan Amazon S3 standar apa pun, seperti Pembuatan Versi S3 atau AWS Backup](#).

## Keamanan infrastruktur di Deadline Cloud

Sebagai layanan yang dikelola, AWS Deadline Cloud dilindungi oleh AWS keamanan jaringan global. Untuk informasi tentang AWS Layanan keamanan dan bagaimana AWS melindungi infrastruktur, lihat [AWS Keamanan Cloud](#). Untuk mendesain AWS lingkungan menggunakan praktik terbaik untuk keamanan infrastruktur, lihat [Perlindungan Infrastruktur](#) di Pilar Keamanan AWS Kerangka Kerja yang Diarsiteksikan dengan Baik.

Anda menggunakan AWS API panggilan yang diterbitkan untuk mengakses Deadline Cloud melalui jaringan. Klien harus mendukung hal-hal berikut:

- Keamanan Lapisan Transportasi (TLS). Kami membutuhkan TLS 1.2 dan merekomendasikan TLS 1.3.
- Suite cipher dengan kerahasiaan maju yang sempurna (PFS) seperti (Ephemeral Diffie-Hellman) atau DHE (Elliptic Curve Ephemeral Diffie-Hellman). ECDHE Sebagian besar sistem modern seperti Java 7 dan versi lebih baru mendukung mode-mode ini.

Selain itu, permintaan harus ditandatangani dengan menggunakan ID kunci akses dan kunci akses rahasia yang terkait dengan IAM prinsipal. Atau Anda dapat menggunakan [AWS Security Token Service](#) (AWS STS) untuk menghasilkan kredensial keamanan sementara untuk menandatangani permintaan.

Deadline Cloud tidak mendukung penggunaan AWS PrivateLink kebijakan titik akhir cloud pribadi virtual (VPC). Ia menggunakan AWS PrivateLink kebijakan default, yang memberikan akses penuh ke titik akhir. Untuk informasi selengkapnya, lihat [Kebijakan titik akhir default](#) di AWS PrivateLink panduan pengguna.

## Analisis konfigurasi dan kerentanan di Deadline Cloud

AWS menangani tugas-tugas keamanan dasar seperti sistem operasi tamu (OS) dan patching database, konfigurasi firewall, dan pemulihan bencana. Prosedur ini telah ditinjau dan disertifikasi oleh pihak ketiga yang sesuai. Untuk detail selengkapnya, lihat sumber daya berikut:

- [Model Tanggung Jawab Bersama](#)
- [Amazon Web Services: Gambaran Umum Proses Keamanan](#) (whitepaper)

AWS Deadline Cloud mengelola tugas pada armada yang dikelola layanan atau yang dikelola pelanggan:

- Untuk armada yang dikelola layanan, Deadline Cloud mengelola sistem operasi tamu.
- Untuk armada yang dikelola pelanggan, Anda bertanggung jawab untuk mengelola sistem operasi.

Untuk informasi tambahan tentang konfigurasi dan analisis kerentanan AWS Deadline Cloud, lihat

- [Praktik terbaik keamanan untuk Deadline Cloud](#)

## Pencegahan confused deputy lintas layanan

Masalah confused deputy adalah masalah keamanan saat entitas yang tidak memiliki izin untuk melakukan suatu tindakan dapat memaksa entitas yang lebih berhak untuk melakukan tindakan tersebut. Masuk AWS, peniruan lintas layanan dapat mengakibatkan masalah wakil yang membingungkan. Peniruan identitas lintas layanan dapat terjadi ketika satu layanan (layanan yang dipanggil) memanggil layanan lain (layanan yang dipanggil). Layanan pemanggilan dapat dimanipulasi menggunakan izinnya untuk bertindak pada sumber daya pelanggan lain dengan cara yang seharusnya tidak dilakukannya kecuali bila memiliki izin untuk mengakses. Untuk mencegah hal ini, AWS menyediakan alat yang membantu Anda melindungi data Anda untuk semua layanan dengan prinsipal layanan yang telah diberikan akses ke sumber daya di akun Anda.

Kami merekomendasikan menggunakan [aws:SourceArn](#) dan [aws:SourceAccount](#) kunci konteks kondisi global dalam kebijakan sumber daya untuk membatasi izin yang AWS Deadline Cloud memberikan layanan lain ke sumber daya. Gunakan `aws:SourceArn` jika Anda hanya ingin satu sumber daya dikaitkan dengan akses lintas layanan. Gunakan `aws:SourceAccount` jika Anda ingin mengizinkan sumber daya apa pun di akun tersebut dikaitkan dengan penggunaan lintas layanan.

Cara paling efektif untuk melindungi dari masalah wakil yang membingungkan adalah dengan menggunakan kunci konteks kondisi `aws:SourceArn` global dengan Nama Sumber Daya Amazon (ARN) lengkap dari sumber daya. Jika Anda tidak tahu sumber daya penuh ARN atau jika Anda menentukan beberapa sumber daya, gunakan kunci konteks kondisi `aws:SourceArn` global dengan karakter wildcard (\*) untuk bagian yang tidak diketahui dari file. ARN Misalnya, `arn:aws:deadline:*:123456789012:*`.

Jika `aws:SourceArn` nilainya tidak berisi ID akun, seperti bucket Amazon S3ARN, Anda harus menggunakan kedua kunci konteks kondisi global untuk membatasi izin.

Contoh berikut menunjukkan bagaimana Anda dapat menggunakan kunci konteks kondisi `aws:SourceAccount` global `aws:SourceArn` dan `Deadline Cloud` untuk mencegah masalah wakil yang membingungkan.

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "deadline.amazonaws.com"
    },
    "Action": "deadline:ActionName",
    "Resource": [
      "*"
    ],
    "Condition": {
      "ArnLike": {
        "aws:SourceArn": "arn:aws:deadline:*:123456789012:*"
      },
      "StringEquals": {
        "aws:SourceAccount": "123456789012"
      }
    }
  }
}
```

# Akses AWS Deadline Cloud menggunakan titik akhir antarmuka (AWS PrivateLink)

Anda dapat menggunakan AWS PrivateLink untuk membuat koneksi pribadi antara Anda VPC dan AWS Deadline Cloud. Anda dapat mengakses Deadline Cloud seolah-olah itu ada di AndaVPC, tanpa menggunakan gateway internet, NAT perangkat, VPN koneksi, atau AWS Direct Connect koneksi. Contoh di Anda VPC tidak memerlukan alamat IP publik untuk mengakses Deadline Cloud.

Anda membuat koneksi pribadi ini dengan membuat titik akhir antarmuka, didukung oleh AWS PrivateLink. Kami membuat antarmuka jaringan endpoint di setiap subnet yang Anda aktifkan untuk titik akhir antarmuka. Ini adalah antarmuka jaringan yang dikelola pemohon yang berfungsi sebagai titik masuk untuk lalu lintas yang ditakdirkan Deadline Cloud.

Untuk informasi selengkapnya, lihat [Akses layanan AWS lewat AWS PrivateLink](#) di AWS PrivateLink Panduan.

## Pertimbangan untuk Deadline Cloud

Sebelum Anda menyiapkan titik akhir antarmuka untuk Deadline Cloud, lihat [Mengakses AWS layanan menggunakan VPC titik akhir antarmuka](#) di AWS PrivateLink Panduan.

Deadline Cloud mendukung membuat panggilan ke semua API tindakannya melalui titik akhir antarmuka.

Secara default, akses penuh ke Deadline Cloud diizinkan melalui titik akhir antarmuka. Atau, Anda dapat mengaitkan grup keamanan dengan antarmuka jaringan titik akhir untuk mengontrol lalu lintas Deadline Cloud melalui titik akhir antarmuka.

Deadline Cloud tidak mendukung kebijakan VPC titik akhir. Untuk informasi selengkapnya, lihat [Mengontrol akses ke VPC titik akhir menggunakan kebijakan titik akhir](#) di AWS PrivateLink Panduan.

## Deadline Cloud titik akhir

Deadline Cloud menggunakan dua titik akhir untuk akses ke layanan menggunakan AWS PrivateLink.

Pekerja menggunakan `com.amazonaws.region.deadline.scheduling` titik akhir untuk mendapatkan tugas dari antrian, melaporkan kemajuan ke Deadline Cloud, dan untuk mengirim output tugas kembali. Jika Anda menggunakan armada yang dikelola pelanggan, titik akhir

penjadwalan adalah satu-satunya titik akhir yang perlu Anda buat kecuali Anda menggunakan operasi manajemen. Misalnya, jika pekerjaan menciptakan lebih banyak pekerjaan, Anda perlu mengaktifkan titik akhir manajemen untuk memanggil CreateJob operasi.

Bagian Deadline Cloud monitor menggunakan `com.amazonaws.region.deadline.management` untuk mengelola sumber daya di peternakan Anda, seperti membuat dan memodifikasi antrian dan armada atau mendapatkan daftar pekerjaan, langkah, dan tugas.

Deadline Cloud juga membutuhkan titik akhir untuk hal-hal berikut AWS titik akhir layanan:

- Deadline Cloud menggunakan AWS STS untuk mengautentikasi pekerja sehingga mereka dapat mengakses aset pekerjaan. Untuk informasi lebih lanjut tentang AWS STS, lihat [Kredensi keamanan sementara di dalam IAM](#) AWS Identity and Access Management Panduan Pengguna.
- Jika Anda menyiapkan armada yang dikelola pelanggan di subnet tanpa koneksi internet, Anda harus membuat VPC titik akhir untuk CloudWatch Amazon Logs agar pekerja dapat menulis log. Untuk informasi lebih lanjut, lihat [Memantau dengan CloudWatch](#).
- Jika Anda menggunakan lampiran pekerjaan, Anda harus membuat VPC endpoint untuk Amazon Simple Storage Service (Amazon S3) sehingga pekerja dapat mengakses lampiran. Untuk informasi selengkapnya, lihat Lampiran [Job di Deadline Cloud](#).

## Buat titik akhir untuk Deadline Cloud

Anda dapat membuat titik akhir antarmuka untuk Deadline Cloud menggunakan VPC konsol Amazon atau AWS Command Line Interface (AWS CLI). Untuk informasi selengkapnya, lihat [Membuat titik akhir antarmuka](#) di AWS PrivateLink Panduan.

Buat titik akhir manajemen dan penjadwalan untuk Deadline Cloud menggunakan nama layanan berikut. Ganti *region* dengan Wilayah AWS di mana Anda telah dikerahkan Deadline Cloud.

```
com.amazonaws.region.deadline.management
```

```
com.amazonaws.region.deadline.scheduling
```

Jika Anda mengaktifkan privat DNS untuk titik akhir antarmuka, Anda dapat membuat API permintaan Deadline Cloud menggunakan DNS nama Regional default. Misalnya, `worker.deadline.us-east-1.amazonaws.com` untuk operasi pekerja, atau `management.deadline.us-east-1.amazonaws.com` untuk semua operasi lainnya.

Anda juga harus membuat endpoint untuk AWS STS menggunakan nama layanan berikut:

```
com.amazonaws.region.sts
```

Jika armada yang dikelola pelanggan berada di subnet tanpa koneksi internet, Anda harus membuat titik akhir CloudWatch Log menggunakan nama layanan berikut:

```
com.amazonaws.region.logs
```

Jika Anda menggunakan lampiran pekerjaan untuk mentransfer file, Anda harus membuat titik akhir Amazon S3 menggunakan nama layanan berikut:

```
com.amazonaws.region.s3
```

## Praktik terbaik keamanan untuk Deadline Cloud

AWS Deadline Cloud (Deadline Cloud) menyediakan sejumlah fitur keamanan untuk dipertimbangkan saat Anda mengembangkan dan menerapkan kebijakan keamanan Anda sendiri. Praktik terbaik berikut adalah pedoman umum dan tidak mewakili solusi keamanan yang lengkap. Karena praktik terbaik ini mungkin tidak sesuai atau tidak memadai untuk lingkungan Anda, perlakukan itu sebagai pertimbangan yang bermanfaat, bukan sebagai resep.

### Note

Untuk informasi selengkapnya tentang pentingnya banyak topik keamanan, lihat [Model Tanggung Jawab Bersama](#).

## Perlindungan data

Untuk tujuan perlindungan data, kami menyarankan Anda untuk melindungi Akun AWS kredensi dan mengatur akun individu dengan 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 otentikasi multi-faktor (MFA) dengan setiap akun.

- Gunakan SSL/TLS untuk berkomunikasi dengan AWS sumber daya. Kami membutuhkan TLS 1.2 dan merekomendasikan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail.
- Gunakan AWS solusi 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 pribadi yang disimpan di Amazon Simple Storage Service (Amazon S3).
- Jika Anda memerlukan FIPS 140-2 modul kriptografi yang divalidasi saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan FIPS titik akhir. Untuk informasi selengkapnya tentang FIPS titik akhir yang tersedia, lihat [Federal Information Processing Standard \(FIPS\) 140-2](#).

Sebaiknya jangan pernah memasukkan informasi identitas yang sensitif, seperti nomor rekening pelanggan Anda, ke dalam bidang isian bebas seperti bidang Nama. Ini termasuk ketika Anda bekerja dengan AWS Deadline Cloud atau lainnya layanan AWS menggunakan konsol, API, AWS CLI, atau AWS SDKs. Data apa pun yang Anda masukkan ke Deadline Cloud atau layanan lain mungkin diambil untuk dimasukkan dalam log diagnostik. Ketika Anda memberikan URL ke server eksternal, jangan sertakan informasi kredensial dalam URL untuk memvalidasi permintaan Anda ke server tersebut.

## AWS Identity and Access Management izin

Kelola akses ke AWS sumber daya menggunakan pengguna, AWS Identity and Access Management (IAM) peran, dan dengan memberikan hak istimewa paling sedikit kepada pengguna. Menetapkan kebijakan dan prosedur manajemen kredensi untuk membuat, mendistribusikan, memutar, dan mencabut AWS akses kredensi. Untuk informasi selengkapnya, lihat [Praktik IAM Terbaik](#) di Panduan IAM Pengguna.

## Jalankan pekerjaan sebagai pengguna dan grup

Saat menggunakan fungsionalitas antrian di Deadline Cloud, ini adalah praktik terbaik untuk menentukan pengguna sistem operasi (OS) dan grup utamanya sehingga pengguna OS memiliki izin hak istimewa paling sedikit untuk pekerjaan antrian.

Saat Anda menentukan “Jalankan sebagai pengguna” (dan grup), proses apa pun untuk pekerjaan yang dikirimkan ke antrian akan dijalankan menggunakan pengguna OS tersebut dan akan mewarisi izin OS terkait pengguna tersebut.

Konfigurasi armada dan antrian bergabung untuk membangun postur keamanan. Di sisi antrian, “Job run as user” dan IAM peran dapat ditentukan untuk menggunakan OS dan AWS izin untuk pekerjaan antrian. Armada mendefinisikan infrastruktur (host pekerja, jaringan, penyimpanan bersama yang dipasang) yang, ketika dikaitkan dengan antrian tertentu, menjalankan pekerjaan dalam antrian. Data yang tersedia pada host pekerja perlu diakses oleh pekerjaan dari satu atau lebih antrian terkait. Menentukan pengguna atau grup membantu melindungi data dalam pekerjaan dari antrian lain, perangkat lunak lain yang diinstal, atau pengguna lain dengan akses ke host pekerja. Ketika antrian tanpa pengguna, itu berjalan sebagai pengguna agen yang dapat meniru (sudo) setiap pengguna antrian. Dengan cara ini, antrian tanpa pengguna dapat meningkatkan hak istimewa ke antrian lain.

## Jaringan

Untuk mencegah lalu lintas dicegat atau dialihkan, penting untuk mengamankan bagaimana dan di mana lalu lintas jaringan Anda diarahkan.

Kami menyarankan Anda mengamankan lingkungan jaringan Anda dengan cara berikut:

- Amankan tabel rute subnet Amazon Virtual Private Cloud (AmazonVPC) untuk mengontrol bagaimana lalu lintas lapisan IP dirutekan.
- Jika Anda menggunakan Amazon Route 53 (Route 53) sebagai DNS penyedia di penyiapan farm atau workstation Anda, amankan akses ke Route 53API.
- Jika Anda terhubung ke Deadline Cloud di luar AWS seperti dengan menggunakan workstation lokal atau pusat data lainnya, mengamankan infrastruktur jaringan lokal. Ini termasuk DNS server dan tabel rute pada router, switch, dan perangkat jaringan lainnya.

## Pekerjaan dan data pekerjaan

Tenggat waktu pekerjaan Cloud berjalan dalam sesi pada host pekerja. Setiap sesi menjalankan satu atau lebih proses pada host pekerja, yang umumnya mengharuskan Anda memasukkan data untuk menghasilkan output.

Untuk mengamankan data ini, Anda dapat mengonfigurasi pengguna sistem operasi dengan antrian. Agen pekerja menggunakan pengguna OS antrian untuk menjalankan sub-proses sesi. Sub-proses ini mewarisi izin pengguna OS antrian.

Kami menyarankan Anda mengikuti praktik terbaik untuk mengamankan akses ke data akses sub-proses ini. Untuk informasi selengkapnya, lihat [Model tanggung jawab bersama](#).

## Struktur pertanian

Anda dapat mengatur armada Deadline Cloud dan antrian banyak cara. Namun, ada implikasi keamanan dengan pengaturan tertentu.

Sebuah peternakan memiliki salah satu batas paling aman karena tidak dapat berbagi sumber daya Deadline Cloud dengan peternakan lain, termasuk armada, antrian, dan profil penyimpanan. Namun, Anda dapat berbagi eksternal AWS sumber daya dalam pertanian, yang membahayakan batas keamanan.

Anda juga dapat menetapkan batas keamanan antara antrian dalam peternakan yang sama menggunakan konfigurasi yang sesuai.

Ikuti praktik terbaik ini untuk membuat antrian aman di peternakan yang sama:

- Kaitkan armada hanya dengan antrian dalam batas keamanan yang sama. Perhatikan hal berikut:
  - Setelah pekerjaan berjalan di host pekerja, data mungkin tetap tertinggal, seperti di direktori sementara atau direktori home pengguna antrian.
  - Pengguna OS yang sama menjalankan semua pekerjaan pada host pekerja armada milik layanan, terlepas dari antrian mana Anda mengirimkan pekerjaan.
  - Pekerjaan mungkin membiarkan proses berjalan pada host pekerja, sehingga memungkinkan pekerjaan dari antrian lain untuk mengamati proses berjalan lainnya.
- Pastikan hanya antrian dalam batas keamanan yang sama yang berbagi bucket Amazon S3 untuk lampiran pekerjaan.
- Pastikan bahwa hanya antrian dalam batas keamanan yang sama berbagi pengguna OS.
- Amankan yang lain AWS sumber daya yang terintegrasi ke dalam pertanian hingga batas.

## Antrian lampiran pekerjaan

Lampiran Job dikaitkan dengan antrian, yang menggunakan bucket Amazon S3 Anda.

- Lampiran Job menulis dan membaca dari awalan root di bucket Amazon S3. Anda menentukan awalan root ini dalam `CreateQueue` API panggilan.
- Bucket memiliki kode yang sesuai `Queue Role`, yang menentukan peran yang memberi pengguna antrian akses ke awalan bucket dan root. Saat membuat antrian, Anda menentukan `Queue Role` Amazon Resource Name (ARN) di samping bucket lampiran pekerjaan dan awalan root.

- Panggilan resmi ke `AssumeQueueRoleForRead`, `AssumeQueueRoleForUser`, dan `AssumeQueueRoleForWorker` API operasi mengembalikan satu set kredensial keamanan sementara untuk `Queue Role`

Jika Anda membuat antrian dan menggunakan kembali bucket Amazon S3 dan awalan root, ada risiko informasi diungkapkan kepada pihak yang tidak berwenang. Misalnya, `QueueA` dan `QueueB` berbagi bucket dan awalan root yang sama. Dalam alur kerja yang aman, `ArtisTA` memiliki akses ke `QueueA` tetapi tidak `QueueB`. Namun, ketika beberapa antrian berbagi bucket, `ArtisTA` dapat mengakses data dalam data `QueueB` karena menggunakan bucket dan awalan root yang sama dengan `QueueA`.

Konsol mengatur antrian yang aman secara default. Pastikan antrian memiliki kombinasi yang berbeda antara bucket Amazon S3 dan awalan root kecuali mereka merupakan bagian dari batas keamanan umum.

Untuk mengisolasi antrian Anda, Anda harus mengonfigurasi `Queue Role` untuk hanya mengizinkan akses antrian ke bucket dan awalan root. Dalam contoh berikut, ganti masing-masing *placeholder* dengan informasi spesifik sumber daya Anda.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:ListBucket",
        "s3:GetBucketLocation"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::JOB_ATTACHMENTS_BUCKET_NAME",
        "arn:aws:s3:::JOB_ATTACHMENTS_BUCKET_NAME/JOB_ATTACHMENTS_ROOT_PREFIX/*"
      ],
      "Condition": {
        "StringEquals": { "aws:ResourceAccount": "ACCOUNT_ID" }
      }
    },
    {
      "Action": ["logs:GetLogEvents"],
      "Effect": "Allow",
```

```

    "Resource": "arn:aws:logs:REGION:ACCOUNT_ID:log-group:/aws/deadline/FARM_ID/*"
  }
]
}

```

Anda juga harus menetapkan kebijakan kepercayaan tentang peran tersebut. Dalam contoh berikut, ganti *placeholder* teks dengan informasi spesifik sumber daya Anda.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": ["sts:AssumeRole"],
      "Effect": "Allow",
      "Principal": { "Service": "deadline.amazonaws.com" },
      "Condition": {
        "StringEquals": { "aws:SourceAccount": "ACCOUNT_ID" },
        "ArnEquals": {
          "aws:SourceArn": "arn:aws:deadline:REGION:ACCOUNT_ID:farm/FARM_ID"
        }
      }
    },
    {
      "Action": ["sts:AssumeRole"],
      "Effect": "Allow",
      "Principal": { "Service": "credentials.deadline.amazonaws.com" },
      "Condition": {
        "StringEquals": { "aws:SourceAccount": "ACCOUNT_ID" },
        "ArnEquals": {
          "aws:SourceArn": "arn:aws:deadline:REGION:ACCOUNT_ID:farm/FARM_ID"
        }
      }
    }
  ]
}

```

## Bucket Amazon S3 perangkat lunak khusus

Anda dapat menambahkan pernyataan berikut ke perangkat lunak khusus Queue Role untuk mengakses perangkat lunak khusus di bucket Amazon S3 Anda. Dalam contoh berikut, ganti *SOFTWARE\_BUCKET\_NAME* dengan nama ember S3 Anda.

```
"Statement": [
  {
    "Action": [
      "s3:GetObject",
      "s3:ListBucket"
    ],
    "Effect": "Allow",
    "Resource": [
      "arn:aws:s3:::SOFTWARE_BUCKET_NAME",
      "arn:aws:s3:::SOFTWARE_BUCKET_NAME/*"
    ]
  }
]
```

Untuk informasi selengkapnya tentang praktik terbaik keamanan Amazon S3, lihat Praktik [terbaik keamanan untuk Amazon S3](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

## Tuan rumah pekerja

Host pekerja aman untuk membantu memastikan bahwa setiap pengguna hanya dapat melakukan operasi untuk peran yang ditetapkan.

Kami merekomendasikan praktik terbaik berikut untuk mengamankan host pekerja:

- Jangan gunakan `jobRunAsUser` nilai yang sama dengan beberapa antrian kecuali pekerjaan yang dikirimkan ke antrian tersebut berada dalam batas keamanan yang sama.
- Jangan atur antrian `jobRunAsUser` ke nama pengguna OS yang dijalankan oleh agen pekerja.
- Berikan izin OS dengan hak istimewa paling sedikit kepada pengguna antrian yang diperlukan untuk beban kerja antrian yang dimaksud. Pastikan bahwa mereka tidak memiliki izin menulis sistem file untuk bekerja file program agen atau perangkat lunak bersama lainnya.
- Pastikan hanya pengguna root yang aktif Linux dan akun Administrator pemilik di Windows memiliki dan dapat memodifikasi file program agen pekerja.
- Pada Linux host pekerja, pertimbangkan untuk mengonfigurasi umask penggantian `/etc/sudoers` yang memungkinkan pengguna agen pekerja meluncurkan proses sebagai pengguna antrian. Konfigurasi ini membantu memastikan pengguna lain tidak dapat mengakses file yang ditulis ke antrian.
- Berikan individu tepercaya akses paling tidak istimewa ke host pekerja.

- Batasi izin ke file konfigurasi DNS penggantian lokal (aktif `/etc/hosts` Linux dan `C:\Windows\system32\etc\hosts` pada Windows, dan untuk merutekan tabel pada workstation dan sistem operasi host pekerja.
- Batasi izin untuk DNS konfigurasi pada workstation dan sistem operasi host pekerja.
- Secara teratur menambal sistem operasi dan semua perangkat lunak yang diinstal. Pendekatan ini mencakup perangkat lunak yang khusus digunakan dengan Deadline Cloud seperti pengirim, adaptor, agen pekerja, OpenJD paket, dan lain-lain.
- Gunakan kata sandi yang kuat untuk Windows antrian `jobRunAsUser`.
- Putar kata sandi untuk antrian `jobRunAsUser` Anda secara teratur.
- Pastikan akses hak istimewa paling sedikit ke Windows kata sandi mengeluarkan dan menghapus rahasia yang tidak digunakan.
- Jangan berikan `jobRunAsUser` izin antrian perintah jadwal untuk dijalankan di masa mendatang:
  - Pada Linux, tolak akses akun ini ke `cron` dan `at`.
  - Pada Windows, tolak akses akun ini ke Windows penjadwal tugas.

#### Note

Untuk informasi selengkapnya tentang pentingnya menambal sistem operasi dan perangkat lunak yang diinstal secara teratur, lihat [Model Tanggung Jawab Bersama](#).

## Workstation

Sangat penting untuk mengamankan workstation dengan akses ke Deadline Cloud. Pendekatan ini membantu memastikan bahwa pekerjaan apa pun yang Anda kirimkan ke Deadline Cloud tidak dapat menjalankan beban kerja sewenang-wenang yang ditagih ke Akun AWS.

Kami merekomendasikan praktik terbaik berikut untuk mengamankan workstation artis. Untuk informasi selengkapnya, lihat [Model Tanggung Jawab Bersama](#).

- Amankan semua kredensial tetap yang menyediakan akses ke AWS, termasuk Deadline Cloud. Untuk informasi selengkapnya, lihat [Mengelola kunci akses untuk IAM pengguna](#) di Panduan IAM Pengguna.
- Hanya instal perangkat lunak tepercaya dan aman.

- Mengharuskan pengguna berfederasi dengan penyedia identitas untuk mengakses AWS dengan kredensi sementara.
- Gunakan izin aman pada file program pengirim Deadline Cloud untuk mencegah gangguan.
- Berikan individu tepercaya akses paling tidak istimewa ke workstation artis.
- Hanya gunakan pengirim dan adaptor yang Anda dapatkan melalui Deadline Cloud Monitor.
- Batasi izin `/etc/hosts` dan rute tabel pada workstation dan sistem operasi host pekerja.
- Batasi izin `/etc/resolv.conf` pada workstation dan sistem operasi host pekerja.
- Secara teratur menambal sistem operasi dan semua perangkat lunak yang diinstal. Pendekatan ini mencakup perangkat lunak yang khusus digunakan dengan Deadline Cloud seperti pengirim, adaptor, agen pekerja, OpenJD paket, dan lain-lain.

# AWS Batas Waktu Pemantauan Cloud

Pemantauan adalah bagian penting dalam menjaga keandalan, ketersediaan, dan kinerja AWS Deadline Cloud (Deadline Cloud) dan solusi Anda AWS . Kumpulkan data pemantauan dari semua bagian AWS solusi Anda sehingga Anda dapat lebih mudah men-debug kegagalan multi-titik jika terjadi. Sebelum Anda mulai memantau Deadline Cloud, Anda harus membuat rencana pemantauan yang mencakup jawaban atas pertanyaan-pertanyaan berikut:

- Apa tujuan pemantauan Anda?
- Sumber daya manakah yang akan Anda pantau?
- Seberapa seringkah Anda akan memantau sumber daya ini?
- Apa sajakah alat pemantauan yang akan Anda gunakan?
- Siapa yang akan melakukan tugas pemantauan?
- Siapa yang harus diberi tahu saat terjadi kesalahan?

AWS dan Deadline Cloud menyediakan alat yang dapat Anda gunakan untuk memantau sumber daya Anda dan menanggapi potensi insiden. Beberapa alat ini melakukan pemantauan untuk Anda, beberapa alat memerlukan intervensi manual. Anda harus mengotomatiskan tugas pemantauan sebanyak mungkin.

- Amazon CloudWatch memantau AWS sumber daya Anda dan aplikasi yang Anda jalankan AWS secara real time. Anda dapat mengumpulkan dan melacak metrik, membuat dasbor yang disesuaikan, dan mengatur alarm yang memberi tahu Anda atau mengambil tindakan saat metrik tertentu mencapai ambang batas yang ditentukan. Misalnya, Anda dapat CloudWatch melacak penggunaan CPU atau metrik lain dari instans Amazon EC2 Anda dan secara otomatis meluncurkan instans baru bila diperlukan. Untuk informasi selengkapnya, lihat [Panduan CloudWatch Pengguna Amazon](#).

Deadline Cloud memiliki tiga CloudWatch metrik.

- Amazon CloudWatch Logs memungkinkan Anda memantau, menyimpan, dan mengakses file log Anda dari instans Amazon EC2, CloudTrail, dan sumber lainnya. CloudWatch Log dapat memantau informasi dalam file log dan memberi tahu Anda ketika ambang batas tertentu terpenuhi. Anda juga dapat mengarsipkan data log dalam penyimpanan yang sangat durabel. Untuk informasi selengkapnya, lihat [Panduan Pengguna Amazon CloudWatch Logs](#).

- Amazon EventBridge dapat digunakan untuk mengotomatiskan AWS layanan Anda dan merespons secara otomatis peristiwa sistem, seperti masalah ketersediaan aplikasi atau perubahan sumber daya. Acara dari AWS layanan dikirimkan ke EventBridge dalam waktu dekat. Anda dapat menuliskan aturan sederhana untuk menunjukkan peristiwa mana yang sesuai kepentingan Anda, dan tindakan otomatis mana yang diambil ketika suatu peristiwa sesuai dengan suatu aturan. Untuk informasi selengkapnya, lihat [Panduan EventBridge Pengguna Amazon](#).
- AWS CloudTrail menangkap panggilan API dan peristiwa terkait yang dibuat oleh atau atas nama AWS akun Anda dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Anda dapat mengidentifikasi pengguna dan akun mana yang dipanggil AWS, alamat IP sumber dari mana panggilan dilakukan, dan kapan panggilan terjadi. Untuk informasi selengkapnya, silakan lihat [Panduan Pengguna AWS CloudTrail](#).

## Topik

- [Pencatatan panggilan dengan CloudTrail](#)
- [Pemantauan CloudWatch dengan](#)
- [Bertindak pada EventBridge acara](#)

## Pencatatan panggilan dengan CloudTrail

AWS Deadline Cloud terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau layanan AWS di Deadline Cloud. CloudTrail menangkap semua panggilan API untuk Deadline Cloud sebagai peristiwa. Panggilan yang diambil termasuk panggilan dari konsol Deadline Cloud dan panggilan kode ke operasi Deadline Cloud API.

Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara berkelanjutan ke bucket Amazon S3, termasuk acara untuk Deadline Cloud. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat ke Deadline Cloud, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Untuk mempelajari selengkapnya CloudTrail, lihat [Panduan AWS CloudTrail Pengguna](#).

## Informasi Batas waktu Cloud di CloudTrail

CloudTrail diaktifkan pada Akun AWS saat Anda membuat akun. Ketika aktivitas terjadi di Deadline Cloud, aktivitas tersebut dicatat dalam suatu CloudTrail peristiwa bersama dengan layanan AWS

peristiwa lain dalam riwayat Peristiwa. Anda dapat melihat, mencari, dan mengunduh acara terbaru di situs Anda Akun AWS. Untuk informasi selengkapnya, lihat [Melihat Acara dengan Riwayat CloudTrail Acara](#).

CloudTrail juga merekam peristiwa saat pengguna masuk ke monitor Deadline Cloud dan menerima AWS kredensi. Saat pengguna masuk, ada CloudTrail acara dengan sumber `signin.amazonaws.com` dan `nameUserAuthentication`. Ada peristiwa kedua ketika pengguna yang masuk diberi AWS kredensial dari sumber dan nama `sts.amazonaws.com AssumeRole` ID pengguna direkam dalam acara kedua di dalam nama sesi peran.

Untuk catatan acara yang sedang berlangsung di Anda Akun AWS, termasuk acara untuk Deadline Cloud, buat jejak. Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Secara default, saat Anda membuat jejak di konsol, jejak tersebut berlaku untuk semua Wilayah AWS. Jejak mencatat peristiwa dari semua Wilayah di AWS partisi dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi lainnya layanan AWS untuk menganalisis lebih lanjut dan menindaklanjuti data peristiwa yang dikumpulkan dalam CloudTrail log.

Untuk informasi selengkapnya, lihat berikut:

[Gambaran umum untuk membuat jejak](#)

[CloudTrail layanan dan integrasi yang didukung](#)

[Mengonfigurasi notifikasi Amazon SNS untuk CloudTrail](#)

[Menerima file CloudTrail log dari beberapa Wilayah](#)

[Menerima file CloudTrail log dari beberapa akun](#)

Deadline Cloud mendukung pencatatan tindakan berikut sebagai peristiwa dalam file CloudTrail log:

- [associate-member-to-farm](#)
- [associate-member-to-fleet](#)
- [associate-member-to-job](#)
- [associate-member-to-queue](#)
- [assume-fleet-role-for-baca](#)
- [assume-fleet-role-for-pekerja](#)

- [assume-queue-role-for-baca](#)
- [assume-queue-role-for-pengguna](#)
- [assume-queue-role-for-pekerja](#)
- [buat-anggaran](#)
- [buat-pertanian](#)
- [membuat-armada](#)
- [create-license-endpoint](#)
- [buat-monitor](#)
- [buat-antrian](#)
- [create-queue-environment](#)
- [create-queue-fleet-association](#)
- [create-storage-profile](#)
- [buat-pekerja](#)
- [hapus-anggaran](#)
- [hapus-pertanian](#)
- [hapus-armada](#)
- [delete-license-endpoint](#)
- [delete-metered-product](#)
- [hapus-monitor](#)
- [hapus-antrian](#)
- [delete-queue-environment](#)
- [delete-queue-fleet-association](#)
- [delete-storage-profile](#)
- [hapus-pekerja](#)
- [disassociate-member-from-farm](#)
- [disassociate-member-from-fleet](#)
- [disassociate-member-from-job](#)
- [disassociate-member-from-queue](#)
- [get-application-version](#)

- [dapatkan-anggaran](#)
- [dapatkan-pertanian](#)
- [get-feature-map](#)
- [dapatkan-armada](#)
- [get-license-endpoint](#)
- [dapatkan-monitor](#)
- [get-antrian](#)
- [get-queue-environment](#)
- [get-queue-fleet-association](#)
- [get-sessions-statistics-aggregation](#)
- [get-storage-profile](#)
- [get-storage-profile-for-antrian](#)
- [list-available-metered-products](#)
- [daftar-anggaran](#)
- [list-farm-members](#)
- [daftar-peternakan](#)
- [list-fleet-members](#)
- [daftar-armada](#)
- [list-job-members](#)
- [list-license-endpoints](#)
- [list-metered-products](#)
- [daftar-monitor](#)
- [list-queue-environments](#)
- [list-queue-fleet-associations](#)
- [list-queue-members](#)
- [daftar-antrian](#)
- [list-storage-profiles](#)
- [list-storage-profiles-for-antrian](#)
- [list-tags-for-resource](#)

- [put-metered-product](#)
- [start-sessions-statistics-aggregation](#)
- [tag-sumber daya](#)
- [untag-sumber daya](#)
- [pembaruan-anggaran](#)
- [pembaruan-pertanian](#)
- [pembaruan-armada](#)
- [pembaruan-monitor](#)
- [antrian pembaruan-](#)
- [update-queue-environment](#)
- [update-queue-fleet-association](#)
- [update-storage-profile](#)
- [pembaruan-pekerja](#)

Setiap entri peristiwa atau log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan berikut ini:

- Apakah permintaan itu dibuat dengan kredensial pengguna root atau AWS Identity and Access Management (IAM).
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara untuk satu peran atau pengguna terfederasi.
- Apakah permintaan tersebut dibuat oleh layanan lainnya.

Untuk informasi selengkapnya, lihat [elemen Identitas CloudTrail pengguna](#).

## Memahami Entri file log Deadline Cloud

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3 yang Anda tentukan. CloudTrail file log berisi satu atau lebih entri log. Peristiwa mewakili permintaan tunggal dari sumber manapun dan mencakup informasi tentang tindakan yang diminta, tanggal dan waktu tindakan, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, jadi file tersebut tidak muncul dalam urutan tertentu.

Contoh JSON ini menunjukkan log yang dihasilkan oleh panggilan ke **CreateFarm** API:

```
{
  "eventVersion": "0",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLE-PrincipalID:EXAMPLE-Session",
    "arn": "arn:aws:sts::111122223333:assumed-role/EXAMPLE-UserName/EXAMPLE-Session",
    "accountId": "111122223333",
    "accessKeyId": "EXAMPLE-accessKeyId",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "EXAMPLE-PrincipalID",
        "arn": "arn:aws:iam::111122223333:role/EXAMPLE-UserName",
        "accountId": "111122223333",
        "userName": "EXAMPLE-UserName"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-03-08T23:25:49Z"
      }
    }
  },
  "eventTime": "2021-03-08T23:25:49Z",
  "eventSource": "deadline.amazonaws.com",
  "eventName": "CreateFarm",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "192.0.2.0",
  "userAgent": "EXAMPLE-userAgent",
  "requestParameters": {
    "displayName": "example-farm",
    "kmsKeyArn": "arn:aws:kms:us-west-2:111122223333:key/111122223333",
    "X-Amz-Client-Token": "12abc12a-1234-1abc-123a-1a11bc1111a",
    "description": "example-description",
    "tags": {
      "purpose_1": "e2e"
      "purpose_2": "tag_test"
    }
  },
  "responseElements": {
    "farmId": "EXAMPLE-farmID"
  }
}
```

```
  },
  "requestID": "EXAMPLE-requestID",
  "eventID": "EXAMPLE-eventID",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "111122223333"
  "eventCategory": "Management",
}
```

Contoh menunjukkan AWS Wilayah, alamat IP, dan "requestParameters" lainnya seperti "displayName" dan "kmsKeyArn" yang dapat membantu Anda mengidentifikasi acara.

## Pemantauan CloudWatch dengan

Amazon CloudWatch (CloudWatch) mengumpulkan data mentah dan memprosesnya menjadi metrik yang dapat dibaca, mendekati waktu nyata. Anda dapat membuka CloudWatch konsol di <https://console.aws.amazon.com/cloudwatch/> untuk melihat dan memfilter metrik Deadline Cloud.

- Dalam armada yang dikelola pelanggan Deadline Cloud, CloudWatch mengirimkan dua metrik dan: `UnhealthyWorkerCount` `RecommendedFleetSize`
- Namespace untuk metrik ini adalah `AWS/DeadlineCloud`.
- Anda dapat menggunakan dimensi `farmID` dan `fleetID` untuk memfilter metrik.
- Kedua metrik menggunakan `unitcount`.

Statistik ini disimpan selama 15 bulan sehingga Anda dapat mengakses informasi historis untuk mendapatkan perspektif yang lebih baik tentang kinerja aplikasi atau layanan web Anda. Anda juga dapat mengatur alarm yang memperhatikan ambang batas tertentu dan mengirim notifikasi atau mengambil tindakan saat ambang batas tersebut terpenuhi. Untuk informasi selengkapnya, lihat [Panduan CloudWatch Pengguna Amazon](#).

Deadline Cloud memiliki dua jenis log — log tugas dan log pekerja. Log tugas adalah saat Anda menjalankan log eksekusi sebagai skrip atau saat DCC berjalan. Log tugas mungkin menampilkan peristiwa seperti pemuatan aset, rendering ubin, atau tekstur yang tidak ditemukan.

Log pekerja menunjukkan proses agen pekerja. Ini mungkin termasuk hal-hal seperti ketika agen pekerja memulai, mendaftarkan dirinya sendiri, melaporkan kemajuan, memuat konfigurasi, atau menyelesaikan tugas.

Untuk Deadline Cloud, pekerja mengunggah log ini ke CloudWatch Log. Secara default, log tidak pernah kedaluwarsa. Jika suatu pekerjaan menghasilkan volume data yang tinggi, Anda dapat dikenakan biaya tambahan. Untuk informasi selengkapnya, lihat [CloudWatch harga Amazon](#).

Anda dapat menyesuaikan kebijakan penyimpanan untuk setiap grup log. Retensi yang lebih pendek menghilangkan log lama dan dapat membantu mengurangi biaya penyimpanan. Untuk menyimpan log, Anda dapat mengarsipkannya ke Amazon Simple Storage Service sebelum menghapus log. Untuk informasi selengkapnya, lihat [Mengeksport data log ke Amazon S3 menggunakan konsol di panduan CloudWatch pengguna Amazon](#).

#### Note

CloudWatch pembacaan log dibatasi oleh AWS. Jika Anda berencana untuk bergabung dengan banyak artis, kami sarankan Anda menghubungi dukungan AWS pelanggan dan meminta kenaikan GetLogEvents kuota. CloudWatch Selain itu, kami sarankan Anda menutup portal tailing log saat Anda tidak men-debug.

Untuk informasi selengkapnya, lihat [Kuota CloudWatch log](#) di panduan CloudWatch pengguna Amazon.

## Bertindak pada EventBridge acara

Deadline Cloud mengirimkan acara EventBridge ke Amazon untuk memberi tahu Anda tentang perubahan pada status layanan. Anda dapat menggunakan EventBridge dan acara ini untuk menulis aturan yang mengambil tindakan, seperti memberi tahu Anda, ketika ada perubahan dalam armada Anda. Untuk informasi selengkapnya, lihat [Apa itu Amazon EventBridge](#)

## Perubahan rekomendasi ukuran armada

Saat mengonfigurasi armada untuk menggunakan penskalaan otomatis berbasis peristiwa, Deadline Cloud mengirimkan peristiwa yang dapat Anda gunakan untuk mengelola armada Anda. Masing-masing acara ini berisi informasi tentang ukuran saat ini dan ukuran armada yang diminta. Untuk contoh menggunakan EventBridge acara dan contoh fungsi Lambda untuk menangani acara, lihat [Skala otomatis EC2 armada Amazon Anda dengan fitur rekomendasi skala Deadline Cloud](#)

Acara perubahan rekomendasi ukuran armada dikirim ketika hal berikut terjadi:

- Ketika ukuran armada yang direkomendasikan berubah dan `oldFleetSize` berbeda dari `newFleetSize`.
- Ketika layanan mendeteksi bahwa ukuran armada sebenarnya tidak sesuai dengan ukuran armada yang direkomendasikan. Anda bisa mendapatkan ukuran armada yang sebenarnya dari `workerCount` dalam respons [GetFleet](#) operasi. Hal ini dapat terjadi ketika instans Amazon EC2 aktif gagal mendaftar sebagai pekerja Deadline Cloud.

Acara ini memiliki format sebagai berikut:

```
{
  "version": "0",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "Fleet Size Recommendation Change",
  "source": "aws.deadline",
  "account": "111122223333",
  "time": "2017-12-22T18:43:48Z",
  "region": "us-west-1",
  "resources": [],
  "detail": {
    "farmId": "farm-12345678900000000000000000000000",
    "fleetId": "fleet-12345678900000000000000000000000",
    "oldFleetSize": 1,
    "newFleetSize": 5,
  }
}
```

Bidang berikut menentukan pola acara:

```
"source": "aws.deadline"
```

Mengidentifikasi bahwa sumber acara ini adalah Deadline Cloud.

```
"detail-type": "Fleet Size Recommendation Change"
```

Mengidentifikasi tipe peristiwa.

```
"detail": { }
```

Memberikan informasi tentang perubahan yang direkomendasikan pada ukuran armada.

```
"farmId": "farm-12345678900000000000000000000000"
```

Pengidentifikasi peternakan yang berisi armada.

```
"fleetId": "fleet-1234567890000000000000000000000000"
```

Pengidentifikasi armada yang membutuhkan perubahan ukuran.

```
"oldFleetSize": 1
```

Ukuran armada saat ini.

```
"newFleetSize": 5
```

Ukuran baru armada yang direkomendasikan.

# Kuota untuk Deadline Cloud

AWS Deadline Cloud menyediakan sumber daya, seperti peternakan, armada, dan antrian, yang dapat Anda gunakan untuk memproses pekerjaan. Saat Anda membuat Akun AWS, kami menetapkan kuota default pada sumber daya ini untuk masing-masing Wilayah AWS.

Service Quotas adalah lokasi pusat di mana Anda dapat melihat dan mengelola kuota Anda. layanan AWS Anda juga dapat meminta peningkatan kuota untuk banyak sumber daya yang Anda gunakan.

Untuk melihat kuota Deadline Cloud, buka konsol [Service Quotas](#). Di panel navigasi, pilih layanan AWS dan pilih Deadline Cloud.

Untuk meminta penambahan kuota, lihat [Meminta penambahan kuota](#) di Panduan Pengguna Service Quotas. Jika kuota belum tersedia di Service Quotas, gunakan formulir peningkatan [kuota layanan](#).

# Membuat sumber daya Cloud AWS Deadline dengan AWS CloudFormation

AWS Deadline Cloud terintegrasi dengan AWS CloudFormation, layanan yang membantu Anda memodelkan dan mengatur AWS sumber daya Anda sehingga Anda dapat menghabiskan lebih sedikit waktu untuk membuat dan mengelola sumber daya dan infrastruktur Anda. Anda membuat templat yang menjelaskan semua AWS sumber daya yang Anda inginkan (seperti peternakan, antrian, dan armada), serta menyediakan serta mengonfigurasi sumber AWS CloudFormation daya tersebut untuk Anda.

Bila Anda menggunakan AWS CloudFormation, Anda dapat menggunakan kembali template Anda untuk mengatur sumber daya Deadline Cloud Anda secara konsisten dan berulang kali. Jelaskan sumber daya Anda sekali, lalu sediakan sumber daya yang sama berulang-ulang di beberapa Akun AWS dan Wilayah.

## Tenggat waktu Cloud dan template AWS CloudFormation

Untuk menyediakan dan mengonfigurasi sumber daya untuk Deadline Cloud dan layanan terkait, Anda harus memahami [AWS CloudFormation templat](#). Templat adalah file teks dengan format JSON atau YAML. Template ini menjelaskan sumber daya yang ingin Anda sediakan di AWS CloudFormation tumpukan Anda. Jika Anda tidak terbiasa dengan JSON atau YAMAL, Anda dapat menggunakan AWS CloudFormation Designer untuk membantu Anda memulai dengan template. AWS CloudFormation Untuk informasi selengkapnya, lihat [Apa itu AWS CloudFormation Designer?](#) di Panduan Pengguna AWS CloudFormation .

Deadline Cloud mendukung pembuatan peternakan, antrian, dan armada. AWS CloudFormation Untuk informasi selengkapnya, termasuk contoh template JSON dan YAMAL untuk farm, antrian, dan armada, lihat [AWS Deadline Cloud di Panduan Pengguna](#). AWS CloudFormation

## Pelajari lebih lanjut tentang AWS CloudFormation

Untuk mempelajari selengkapnya AWS CloudFormation, lihat sumber daya berikut:

- [AWS CloudFormation](#)
- [AWS CloudFormation Panduan Pengguna](#)
- [AWS CloudFormation Referensi API](#)

- [AWS CloudFormation Panduan Pengguna Antarmuka Baris Perintah](#)

# Riwayat dokumen untuk panduan pengguna Deadline Cloud

Tabel berikut menjelaskan perubahan penting dalam setiap rilis panduan pengguna AWS Deadline Cloud.

Perubahan	Deskripsi	Tanggal
<a href="#">Bawa lisensi Anda sendiri</a>	Menambahkan informasi tentang bagaimana Anda dapat menggunakan server lisensi atau instance proxy lisensi Anda sendiri dengan Deadline Cloud. Untuk informasi selengkapnya, lihat Armada yang <a href="#">dikelola layanan</a> .	Juli 26, 2024
<a href="#">Autodesk 3ds Maks UBL</a>	Menambahkan informasi tentang lisensi berbasis penggunaan Autodesk 3ds Max () UBL untuk Deadline Cloud. Untuk informasi selengkapnya, lihat <a href="#">Connect ke titik akhir lisensi</a> .	Juni 18, 2024
<a href="#">Fitur pemantauan dan manajemen biaya</a>	Anda dapat menggunakan EventBridge untuk mendukung pemantauan di Deadline Cloud. Untuk informasi selengkapnya, lihat <a href="#">Bertindak atas EventBridge acara</a> . Deadline Cloud menyediakan anggaran dan penjelajah penggunaan untuk membantu Anda mengontrol dan memvisualisasikan biaya untuk pekerjaan Anda. Pelajari tentang beberapa praktik	23 Mei 2024

terbaik untuk membantu mengelola biaya tersebut. Untuk informasi selengkapnya, lihat [Manajemen biaya](#).

### Rilis awal

Ini adalah rilis awal panduan pengguna Deadline Cloud. April 2, 2024

# AWS Glosarium

Untuk AWS terminologi terbaru, lihat [AWS glosarium di Referensi](#).Glosarium AWS

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.