



用户指南

AWS 截止日期云



版本 latest

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

AWS 截止日期云: 用户指南

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

Amazon 的商标和商业外观不得用于任何非 Amazon 的商品或服务，也不得以任何可能引起客户混淆、贬低或诋毁 Amazon 的方式使用。所有非 Amazon 拥有的其他商标均为各自所有者的财产，这些所有者可能附属于 Amazon、与 Amazon 有关联或由 Amazon 赞助，也可能不是如此。

Table of Contents

什么是截止日期云？	1
截止日期云的特点	1
概念和术语	2
截止日期云入门	4
访问截止日期云	4
相关服务	4
截止日期云的工作原理	5
.....	5
截止日期云中的权限	5
截止日期云提供软件支持	6
开始使用	7
设置您的 AWS 账户	7
设置显示器	8
步骤 1：设置显示器	8
步骤 2：定义服务器场详细信息	11
步骤 3：定义队列详细信息	11
第 4 步：定义舰队详细信息	12
步骤 5：配置工作人员要求	13
步骤 6：定义访问级别	13
第 7 步：查看并创建	14
设置开发者工作站	14
步骤 1：创建农场	14
步骤 2：运行工作器代理	18
步骤 3：提交并运行作业	20
步骤 4：运行带附件的作业	27
步骤 5：添加服务托管舰队	36
步骤 6：清理农场资源	38
设置提交者	41
第 1 步：安装 Deadline Cloud 提交器	42
第 2 步：安装和设置 Deadline 云监视器	49
第 3 步：启动 Deadline Cloud 提交器	51
使用农场	55
使用显示器	56
共享 Deadline Cloud 监控	56

打开截止日期云监视器	57
查看队列和舰队详情	58
查看和管理作业、步骤和任务	59
查看职位详情	60
查看步骤	61
查看任务	61
查看 日志	62
下载已完成的输出	63
农场	65
创建农场	65
删除农场	65
编辑农场	65
队列	67
创建队列	67
创建队列环境	69
默认Conda队列环境	69
删除队列	70
编辑队列	71
关联队列和舰队	71
管理车队	72
服务管理车队	72
视觉特效平台	73
客户管理的车队	74
创建 CMF	74
工作主机设置	79
管理访问权限	84
为作业安装软件	86
配置 凭证	87
创建 AMI	89
创建舰队基础设施	91
Connect 连接到许可证端点	101
管理用户	105
管理监视器的用户和群组	105
管理农场、队列和队列的用户和群组	107
作业	109
提交作业	110

更多提交职位的选项	111
安排作业	113
确定机队兼容性	113
舰队扩展	115
会话	115
步骤依赖关系	117
作业状态	118
修改作业	121
正在处理作业	125
排查作业	126
为什么创建我的任务失败了？	126
为什么我的工作不兼容？	126
为什么我的工作准备就绪？	127
为什么我的工作失败了？	127
为什么我的步骤处于待处理状态？	127
存储	128
Job 附件	128
对任务附件 S3 存储桶进行加密	129
管理 S3 存储桶中的任务附件	130
虚拟文件系统	130
共享存储	132
截止日期云中的存储配置文件	132
管理预算和使用情况	134
成本假设	134
使用预算管理器	135
先决条件	135
访问预算经理	135
创建预算	136
查看预算	137
编辑预算	137
停用预算	138
使用使用情况浏览器	138
先决条件	138
打开使用情况浏览器	138
使用使用情况浏览器	138
成本管理	141

成本管理最佳实践	142
安全性	144
数据保护	144
静态加密	145
传输中加密	146
密钥管理	146
互连网络流量隐私	155
选择退出	155
Identity and Access Management	156
受众	157
使用身份进行身份验证	157
使用策略管理访问	160
截止日期云如何与 IAM 配合使用	161
基于身份的策略示例	167
AWS 托管策略	171
故障排除	174
合规性验证	175
弹性	176
基础设施安全性	177
配置和漏洞分析	177
防止跨服务混淆座席	178
AWS PrivateLink	179
注意事项	179
Deadline Cloud 端点	179
创建终端节点	180
安全最佳实操	181
数据保护	181
IAM 权限	182
以用户和群组的身份运行作业	182
联网	182
Job 数据	182
农场结构	183
Job 附件队列	183
自定义软件存储桶	185
工作人员主机	186
工作站	187

监控	188
使用登录 CloudTrail	189
截止日期云中的信息 CloudTrail	189
了解截止日期云日志文件条目	193
使用监控 CloudWatch	194
对 EventBridge 事件采取行动	195
舰队规模建议变更	195
配额	198
AWS CloudFormation 资源	199
截止日期云和 AWS CloudFormation 模板	199
了解更多关于 AWS CloudFormation	199
文档历史记录	200
AWS 词汇表	201
.....	ccii

什么是 AWS 截止日期云？

Deadline Cloud 可用于直接从数字内容创作管道和工作站在亚马逊弹性计算云 (Amazon EC2) 实例上创建和管理渲染项目和作业。AWS 服务

Deadline Cloud 提供控制台界面、本地应用程序、命令行工具和 API。借助 Deadline Cloud，您可以创建、管理和监控农场、队列、作业、用户组和存储。您还可以指定硬件要求，为特定工作负载创建环境，并将制作所需的内容创建工具集成到您的 Deadline Cloud 管道中。

Deadline Cloud 提供了一个统一的界面，可以在一个地方管理所有渲染项目。您可以管理用户、为他们分配项目以及授予工作角色权限。

主题

- [截止日期云的特点](#)
- [截止日期云的概念和术语](#)
- [截止日期云入门](#)
- [访问截止日期云](#)
- [相关服务](#)
- [截止日期云的工作原理](#)

截止日期云的特点

以下是 Deadline Cloud 可以帮助您运行和管理可视化计算工作负载的一些主要方式：

- 快速创建您的农场、队列和舰队。监控他们的状态，深入了解农场的运营和工作。
- 集中管理 Deadline Cloud 用户和群组，并分配权限。
- 使用管理项目用户和外部身份提供 AWS IAM Identity Center 者的登录安全。
- 使用 AWS Identity and Access Management (IAM) 策略和角色安全地管理对项目资源的访问权限。
- 使用标签来组织和快速查找项目资源。
- 管理项目资源使用情况和项目的预估成本。
- 提供广泛的计算管理选项，以支持在云端或面对面渲染。

截止日期云的概念和术语

为了帮助您开始使用 De AWS adline Cloud，本主题解释了其一些关键概念和术语。

预算经理

预算经理是 Deadline Cloud 监控器的一部分。使用预算管理器来创建和管理预算。您还可以使用它来限制活动以保持在预算范围内。

截止日期云端客户端库

客户端库包括用于管理 Deadline Cloud 的命令行界面和库。功能包括根据 Open Job Description 规范向 Deadline Cloud 提交工作捆绑包、下载作业附件输出以及使用命令行界面监控您的农场。

数字内容创作应用程序 (DCC)

数字内容创作应用程序 (DCC) 是您在其中创建数字内容的第三方产品。DCC 的示例有 MayaNuke、和 HoudiniDeadline Cloud 为特定 DCC 提供了作业提交者集成插件。

服务器农场

农场是您的项目资源所在的地方。它由队列和舰队组成。

实例集

队列是一组执行渲染的工作节点。工作节点处理作业。一个队列可以关联到多个队列，一个队列可以关联到多个队列。

作业

作业是渲染请求。用户提交作业。作业包含以步骤和任务形式概述的特定作业属性。

Job 附件

作业附件是 Deadline Cloud 的一项功能，可用于管理作业的输入和输出。在渲染过程中，Job 文件作为作业附件上传。这些文件可以是纹理、3D 模型、照明装备和其他类似物品。

作业属性

Job 属性是您在提交渲染作业时定义的设置。一些示例包括帧范围、输出路径、作业附件、可渲染摄像机等。属性因提交渲染的 DCC 而异。

作业模板

作业模板定义运行时环境以及作为 Deadline Cloud 作业的一部分运行的所有进程。

队列

队列是已提交作业所在的位置，并计划进行渲染。队列必须与队列关联才能成功渲染。一个队列可以与多个队列相关联。

队列舰队关联

当队列与队列关联时，就存在队列与队列的关联。使用关联将车队中的工作人员安排到该队列中的作业。您可以启动和停止关联以控制工作日程安排。

步骤

步骤是在作业中运行的一个特定过程。

截止日期云提交者

Deadline Cloud 提交者是一个数字内容创作 (DCC) 插件。艺术家使用它从他们熟悉的第三方 DCC 界面提交作业。

标签

标签是您可以分配给 AWS 资源的标签。每个标签都包含您所定义的一个键和可选值。

使用标签，您可以用不同的方式对 AWS 资源进行分类。例如，您可以为账户中的 Amazon EC2 实例定义一组标签，以跟踪每个实例的所有者和堆栈级别。

您还可以按用途、所有者或环境对 AWS 资源进行分类。当您有许多相同类型的资源时，这种方法很有用。您可以根据为其分配的标签快速识别特定资源。

任务

任务是渲染步骤的单个组成部分。

基于使用量的许可 (UBL)

基于使用量的许可 (UBL) 是一种按需许可模式，适用于部分第三方产品。此模式按使用量付费，您需要按使用的小时数和分钟数付费。

使用情况浏览器

使用情况浏览器是 Deadline Cloud 监控器的一项功能。它提供了对您的成本和使用量的近似估计。

工作线程

工作人员属于舰队，他们运行 Deadline Cloud 分配的任务来完成步骤和作业。工作人员将任务操作的日志存储在 Amazon CloudWatch 日志中。工作人员还可以使用作业附件功能将输入和输出同步到亚马逊简单存储服务 (Amazon S3) 存储桶。

截止日期云入门

使用 Deadline Cloud 快速创建具有默认设置和资源的渲染农场，例如亚马逊 EC2 实例配置和亚马逊简单存储服务 (Amazon S3) Simple Storage Service 存储桶。

您还可以在创建渲染农场时定义设置和资源。与使用默认设置和资源相比，此方法花费的时间更长，但可以提供更多的控制权。

熟悉 Deadline Cloud [概念和术语](#)后，请参阅[入门](#)，了解有关创建农场、添加用户的 step-by-step 说明以及有用信息的链接。

访问截止日期云

您可以通过以下任何一种方式访问 Deadline Cloud：

- [Deadline Cloud 控制台](#) - 在浏览器中访问控制台以创建场及其资源，并管理用户访问权限。有关更多信息，请参阅 [入门](#)。
- [Deadline Cloud 监视器](#) — 管理您的渲染作业，包括更新优先级和作业状态。监控您的农场并查看日志和作业状态。对于拥有所有者权限的用户，Deadline Cloud 监视器还提供浏览使用情况和创建预算的权限。Deadline Cloud 监视器既可用作 Web 浏览器，也可用作桌面应用程序。
- [AWS SDK 和 AWS CLI](#) — 使用 AWS Command Line Interface (AWS CLI) 从本地系统的命令行调用 Deadline Cloud API 操作。有关更多信息，请参阅[设置开发人员工作站](#)。

相关服务

Deadline Cloud 适用于以下内容 AWS 服务：

- [Amazon CloudWatch](#) — 借助 CloudWatch，您可以监控您的项目和相关 AWS 资源。有关更多信息，请参阅 [Amazon CloudWatch 用户指南](#)。
- [Amazon EC2](#) - AWS 服务 它提供了在云中运行应用程序的虚拟服务器。您可以将您的项目配置为使用 Amazon EC2 实例来处理您的工作负载。有关更多信息，请参阅 [Amazon EC2 实例](#)。
- [Amazon EC2 Auto Scaling](#) — 借助 Auto Scaling，您可以根据实例需求的变化自动增加或减少实例数量。Auto Scaling 有助于确保即使实例出现故障，您也能运行所需数量的实例。如果您使用 Deadline Cloud 启用 Auto Scaling，则由 Auto Scaling 启动的实例将自动注册到工作负载。同样，由 Auto Scaling 终止的实例会自动从工作负载中注销。有关更多信息，请参阅 [Amazon EC2 Auto Scaling 用户指南](#)。

- AWS PrivateLink— 在虚拟私有云 (VPC) 和您的本地网络之间 AWS PrivateLink 提供私有连接，而不会将您的流量暴露给公共互联网。AWS 服务 AWS PrivateLink 可以轻松地跨不同账户和 VPC 连接服务。有关更多信息，请参阅 [AWS PrivateLink](#)。
- 亚马逊 S3 — 亚马逊 S3 是一项对象存储服务。Deadline Cloud 使用 Amazon S3 存储桶来存储任务附件。
- IAM Identity Center — IAM Identity Center 可以让用户从一个地方单点登录访问其所有分配的账户和应用程序。AWS 服务 您还可以集中管理 AWS Organizations 中所有账户的多账户访问权限和用户权限。有关更多信息，请参阅 [AWS IAM Identity Center 常见问题](#)。

截止日期云的工作原理

借助 Deadline Cloud，您可以直接从数字内容创作 (DCC) 管道和 workstation 创建和管理渲染项目和作业。

您可以使用 AWS SDK、AWS Command Line Interface (AWS CLI) 或 Deadline Cloud 作业提交者向 Deadline Cloud 提交作业。Deadline Cloud 支持职位模板规范的 OpenJD 职位描述 (OpenJD)。如需了解更多信息，请参阅 GitHub 网站上的 [Open Job Description](#)。

Deadline Cloud 提供作业提交者。作业提交器是一个 DCC 插件，用于从第三方 DCC 接口（例如 Maya Nuke 借助提交者，艺术家可以将渲染作业从第三方界面提交到 Deadline Cloud，在那里可以管理项目资源并监控作业，所有这些都集中在一个位置。

借助 Deadline Cloud 农场，您可以创建队列和队列、管理用户以及管理项目资源使用情况和成本。农场由队列和舰队组成。队列是已提交作业所在的位置，并计划进行渲染。队列是一组工作节点，它们运行任务以完成作业。队列必须与舰队关联才能渲染作业。一个队列可以支持多个节点，一个节点可以由多个队列支持。

作业由步骤组成，每个步骤由特定的任务组成。借助 Deadline Cloud 监控器，您可以访问作业、步骤和任务的状态、日志和其他故障排除指标。

截止日期云中的权限

截止日期云支持以下内容：

- 使用 AWS Identity and Access Management (IAM) 管理对其 API 操作的访问权限
- 使用与集成管理员工用户的访问权限 AWS IAM Identity Center

在任何人都可以参与某个项目之前，他们必须能够访问该项目和相关的农场。Deadline Cloud 与 IAM 身份中心集成，用于管理员工身份验证和授权。用户可以直接添加到 IAM Identity Center，也可以将其连接到您现有的身份提供商 (IdP)，例如 Okta 或 Active Directory。IT 管理员可以向不同级别的用户和群组授予访问权限。每个后续级别都包含前一个级别的权限。以下列表描述了从最低级别到最高级别的四个访问级别：

- **Viewer** — 查看服务器场、队列、队列中的资源以及他们有权访问的作业的权限。查看者无法提交或更改作业。
- **贡献者** - 与查看者相同，但有权向队列或群提交作业。
- **经理** — 与贡献者相同，但有权编辑他们有权访问的队列中的作业，并授予他们有权访问的资源的权限。
- **所有者** - 与经理相同，但可以查看和创建预算并查看使用情况。

Note

这些权限不允许用户访问 AWS Management Console 或修改 Deadline Cloud 基础架构。

用户必须有权访问服务器场，然后才能访问相关的队列和队列。用户访问权限是分别分配给服务器场内的队列和队列的。

您可以将用户添加为个人或群组成员。将群组添加到群组、队列或队列可以更轻松地管理大型群体的访问权限。例如，如果您的团队正在处理特定项目，则可以将每个团队成员添加到一个小组中。然后，您可以向整个群组授予相应服务器场、队列或队列的访问权限。

截止日期云提供软件支持

Deadline Cloud 可与任何可从命令行界面运行并使用参数值进行控制的软件应用程序配合使用。Deadline Cloud 支持将工作描述为作业的 OpenJD 规范，其软件脚本步骤被参数化（例如跨帧范围）转换为任务。使用 Deadline Cloud 工具和功能将 OpenJD 作业说明汇编成作业捆绑包，以便通过第三方软件应用程序创建、运行和许可这些步骤。

工作需要获得许可才能完成。Deadline Cloud 为精选的软件应用程序许可证提供基于使用量的许可 (UBL)，这些许可证根据使用情况按小时计费，以分钟为增量计费。借助 Deadline Cloud，如果你愿意，你也可以使用自己的软件许可证。如果作业无法访问许可证，则不会呈现并生成错误，该错误会显示在 Deadline Cloud 监视器的任务日志中。

截止日期云入门

要在 Deadline Cloud 中 AWS 创建场，您可以使用 [Deadline Cloud 控制台](#) 或 AWS Command Line Interface (AWS CLI)。使用控制台获得创建农场的引导式体验，包括队列和队列。使用可以直接 AWS CLI 使用该服务，或者开发自己的可与 Deadline Cloud 配合使用的工具。

要创建场并使用 Deadline Cloud 监控器，请为 Deadline Cloud 设置您的帐户。您只需要为每个账户设置一次 Deadline Cloud 监控基础架构。在您的服务器场中，您可以管理您的项目，包括用户对您的农场及其资源的访问权限。

要在不设置 Deadline Cloud 监控基础架构的情况下创建场地，请为 Deadline Cloud 设置开发人员工作站。

要创建用于接受任务的资源最少的服务器场，请在控制台主页中选择 Quickstart。 [设置 Deadline Cloud 监控器](#) 引导你完成这些步骤。这些服务器场从一个队列和一个自动关联的队列开始。这种方法是创建沙盒式农场进行实验的便捷方法。

主题

- [设置您的 AWS 账户](#)
- [设置 Deadline Cloud 监控器](#)
- [为 Deadline Cloud 设置开发者工作站](#)
- [设置 Deadline Cloud 提交者](#)
- [使用农场](#)

设置您的 AWS 账户

将您的设置为使用 AWS De AWS 账户 adline Cloud。

如果您没有 AWS 账户，请完成以下步骤来创建一个。

要注册 AWS 账户

1. 打开 <https://portal.aws.amazon.com/billing/signup>。
2. 按照屏幕上的说明进行操作。

在注册时，将接到一通电话，要求使用电话键盘输入一个验证码。

当您注册时 AWS 账户，就会创建AWS 账户根用户一个。根用户有权访问该账户中的所有 AWS 服务和资源。作为安全最佳实践，请为用户分配管理访问权限，并且只使用根用户来执行[需要根用户访问权限的任务](#)。

首次创建时 AWS 账户，首先要有一个登录身份，该身份可以完全访问账户中的所有资源 AWS 服务和资源。此身份被称为 AWS 账户 root 用户，使用您创建帐户时使用的电子邮件地址和密码登录即可访问该身份。

Important

强烈建议您不要使用根用户执行日常任务。保护好根用户凭证，并使用这些凭证来执行仅根用户可以执行的任务。有关需要您以根用户身份登录的任务的完整列表，请参阅《IAM 用户指南》中的[需要根用户凭证的任务](#)。

设置 Deadline Cloud 监控器

首先，你需要创建你的 Deadline Cloud 监控基础架构并定义你的农场。您还可以执行其他可选步骤，包括添加群组 and 用户、选择服务角色以及向资源添加标签。

步骤 1：设置显示器

Deadline Cloud 监控器用于 AWS IAM Identity Center 对用户进行授权。您用于 Deadline Cloud 的 IAM 身份中心实例必须与监控器 AWS 区域相同。如果您在创建监控器时控制台使用其他区域，则系统会提醒您更改为 IAM 身份中心区域。

您的显示器的基础架构由以下组件组成：

- 显示器显示名称：显示器显示名称是识别显示器的方式，例如AnyCompany 显示器。显示器的名称还决定了您的监视器网址。

Important

完成设置后，您无法更改显示器的显示名称。

- 监视器 URL：您可以使用监视器 URL 访问您的显示器。网址基于显示器的显示名称，例如 <https://anycompanymonitor.awsapps.com>。

⚠ Important

完成设置后，您无法更改监视器 URL。

- **AWS 区域**：AWS 区域是 AWS 数据中心集合的物理位置。设置显示器时，区域默认为离您最近的位置。我们建议更改区域，使其位于最靠近您的用户的位置。这样可以减少延迟并提高数据传输速度。AWS IAM Identity Center 必须与 Deadline Cloud AWS 区域 一样启用。

⚠ Important

设置完截止日期云后，您无法更改您的区域。

完成本节中的任务，配置显示器的基础架构。

配置显示器的基础架构

1. 登录以启动“欢迎来到AWS Management Console到 Deadline Cloud”设置，然后选择“下一步”。
2. 输入显示器显示名称，例如**AnyCompany Monitor**。
3. (可选) 要更改监控器名称，请选择编辑 URL。
4. (可选) 要更改以AWS 区域使其离您的用户最近，请选择更改区域。
 - a. 选择离您的大多数用户最近的区域。
 - b. 选择应用区域。
 - (可选) 要添加群组 and 用户，请选择 [\(可选 \) 添加群组 and 用户](#)。
 - (可选) 要进一步自定义显示器设置，请选择 [其他设置](#)。
5. 如果您准备好了 [步骤 2：定义服务器场详细信息](#)，请选择“下一步”。

(可选) 添加群组 and 用户

在完成 Deadline Cloud 监控器设置之前，您可以添加监控用户并将其添加到群组中。

安装完成后，您可以创建新用户和群组并管理用户，例如为他们分配群组、权限和应用程序，或从显示器中删除用户。

其他设置

Deadline Cloud 设置包括其他设置。使用这些设置，您可以查看 Deadline Cloud 设置对您的所有更改 AWS 账户、配置您的监控用户角色以及更改加密密钥类型。

AWS IAM Identity Center

AWS IAM Identity Center 是一项基于云的单点登录服务，用于管理用户和群组。IAM Identity Center 还可以与企业单点登录 (SSO) 提供商集成，以使用户能够使用其公司账户登录。

Deadline Cloud 默认启用 IAM 身份中心，并且需要设置和使用 Deadline Cloud。您用于 Deadline Cloud 的 IAM 身份中心实例必须与监控器 AWS 区域相同。有关更多信息，请参阅[什么是 AWS IAM Identity Center](#)。

配置服务访问角色

AWS 服务可以扮演服务角色来代表您执行操作。Deadline Cloud 需要监视用户角色才能允许用户访问您的显示器中的资源。

您可以将 AWS Identity and Access Management (IAM) 托管策略附加到监控用户角色。这些策略允许用户执行某些操作，例如在特定的 Deadline Cloud 应用程序中创建作业。由于此应用程序依赖于托管策略中的特定条件，所以如果您不使用托管策略，则此应用程序可能无法按预期运行。

完成设置后，您可以随时更改监控用户角色。有关用户角色的更多信息，请参阅[IAM 角色](#)。

以下选项卡包含两种不同用例的说明。要创建和使用新的服务角色，请选择新服务角色选项卡。要使用现有的服务角色，请选择现有服务角色选项卡。

New service role

创建和使用新的服务角色

1. 选择创建和使用新服务角色
2. (可选) 输入服务用户角色名称。
3. 选择查看权限详细信息以了解有关该角色的更多信息。

Existing service role

使用现有服务角色

1. 选择使用现有服务角色。
2. 打开下拉列表，选择一个现有服务角色。
3. (可选) 选择在 IAM 控制台中查看以了解有关该角色的更多信息。

步骤 2：定义服务器场详细信息

返回 Deadline Cloud 控制台，完成以下步骤以定义服务器场的详细信息。

1. 在农场详细信息中，为农场添加名称。
2. 在描述中，输入服务器场描述。清晰的描述可以帮助您快速确定农场的用途。
3. (可选) 默认情况下，为了您的安全，您的数据使用 AWS 拥有和管理的密钥进行加密。您可以选择“自定义加密设置 (高级)”以使用现有密钥或创建由您管理的新密钥。

如果您选择使用复选框自定义加密设置，请输入 AWS KMS ARN，或者 AWS KMS 通过选择创建新 KMS 密钥来创建新的 ARN。

4. (可选) 选择 Add new tag，向服务器场添加一个或多个标签。
5. 请选择以下选项之一：
 - 选择“跳至查看”和“创建”以[查看和创建您的农场](#)。
 - 选择“下一步”继续执行其他可选步骤。

(可选) 步骤 3：定义队列详细信息

队列负责跟踪任务的进度并安排工作。

1. 从队列详细信息开始，提供队列的名称。
2. 在描述中，输入队列描述。清晰的描述可以帮助您快速确定队列的用途。
3. 对于 Job 附件，您可以创建新的 Amazon S3 存储桶，也可以选择现有的 Amazon S3 存储桶。如果您没有现有 Amazon S3 存储桶，则需要创建一个。
 - a. 要创建新的 Amazon S3 存储桶，请选择创建新的任务存储桶。您可以在根前缀字段中定义任务存储桶的名称。我们建议您致电存储桶 `deadlinecloud-job-attachments-[MONITORNAME]`。

只能使用小写字母和破折号。没有空格或特殊字符。

- b. 要搜索并选择现有的 Amazon S3 存储桶，请选择从现有 Amazon S3 存储桶中选择。然后，通过选择 Browse S3 搜索现有存储桶。显示可用的 Amazon S3 存储桶列表时，选择要用于队列的 Amazon S3 存储桶。
4. 如果您使用的是客户管理的车队，请选择启用与客户管理的车队的关联。
 - 对于客户管理的队列，请添加队列配置的用户，然后设置 POSIX 和/或 Windows 凭据。或者，您可以通过选中复选框来绕过运行方式功能。
5. 您的队列需要获得代表您访问 Amazon S3 的权限。我们建议您为每个队列创建一个新的服务角色。
 - a. 对于新角色，请完成以下步骤。
 - i. 选择创建和使用新服务角色
 - ii. 输入队列角色的角色名称或使用提供的角色名称。
 - iii. (可选) 添加队列角色描述。
 - iv. 您可以通过选择查看权限详细信息来查看队列角色的 IAM 权限。
 - b. 或者，您可以选择现有的服务角色。
6. (可选) 使用名称和值对为队列环境添加环境变量。
7. (可选) 使用键和值对为队列添加标签。

输入所有队列详细信息后，选择“下一步”。

(可选) 步骤 4：定义舰队详细信息

舰队会分配工作人员来执行您的渲染任务。如果您需要舰队来执行渲染任务，请选中创建舰队复选框。

1. 舰队详情
 - a. 为您的舰队提供名称和可选描述。
 - b. 选择计算资源的扩展方式。服务管理选项允许 Deadline Cloud 自动扩展您的计算资源。客户管理选项使您可以控制自己的计算扩展。
2. 在实例选项部分，选择竞价或按需。Amazon EC2 按需实例可提供更快的可用性，而 Amazon EC2 竞价型实例更适合节省成本。
3. 要自动缩放队列中的实例数量，请同时选择最小实例数和最大实例数。

我们强烈建议始终将最小实例数设置为 **0** 以免产生额外费用。

4. 您的车队需要获得许可才能 CloudWatch 代表您写信。我们建议您为每个舰队创建一个新的服务角色。
 - a. 对于新角色，请完成以下步骤。
 - i. 选择创建和使用新服务角色
 - ii. 为您的舰队角色输入角色名称或使用提供的角色名称。
 - iii. (可选) 添加舰队角色描述。
 - iv. 您可以通过选择查看权限详细信息来查看舰队角色的 IAM 权限。
 - b. 或者，您可以使用现有的服务角色。
5. (可选) 使用键和值对为队列添加标签。

输入所有舰队详细信息后，选择下一步。

(可选) 步骤 5：配置工作人员要求

定义工作器实例的要求。

1. 查看操作系统 (OS) 和 CPU 架构设置以了解意识。
2. 根据您的硬件要求更新最小和最大 vCPU 数量。
3. 根据您的硬件要求更新最小和最大内存数 (GiB)。
4. 您可以通过允许或排除工作器实例的类型来筛选实例类型。在这两个筛选选项中，您最多可以筛选 10 个 Amazon EC2 实例类型。
5. 在“其他要求 (可选)”下，您可以按大小 (GiB)、IOPS 和吞吐量 (MiB /s) 定义根 EBS 卷。
6. 设置完所有工作人员要求后，选择“下一步”来定义群组的访问级别。

(可选) 步骤 6：定义访问级别

如果您有群组连接到显示器，则可以定义其访问级别。使用 Deadline Cloud 功能的权限由访问级别管理。您可以为用户组分配不同的访问级别。

1. 使用 De adline Cloud 场访问级别菜单选择群组的权限级别。
2. 选择“下一步”继续并查看输入的所有农场详细信息。

第 7 步：查看并创建

查看为创建农场而输入的所有信息。准备就绪后，选择创建农场。

农场的创建进度显示在“农场”页面上。当您的服务器场准备就绪可供使用时，系统会显示一条成功消息。

为 Deadline Cloud 设置开发者工作站

在本教程中，您将使用 AWS CloudShell 创建一个简单的开发者群组并运行工作器代理。然后，您可以提交并运行带有参数和附件的简单作业，添加服务托管队列，并在完成后清理农场资源。

以下各节将向您介绍 Deadline Cloud 的不同功能，以及它们是如何运作和协同工作的。遵循这些步骤对于开发和测试新的工作负载和自定义项非常有用。

主题

- [步骤 1：创建截止日期云场](#)
- [第 2 步：在 Deadline Cloud 中以开发者模式运行工作器代理](#)
- [第 3 步：使用 Deadline Cloud 提交和运行作业](#)
- [第 4 步：在 Deadline Cloud 中运行带有作业附件的作业](#)
- [第 5 步：在 Deadline Cloud 中向开发者群添加服务托管队列](#)
- [第 6 步：清理 Deadline Cloud 中的农场资源](#)

步骤 1：创建截止日期云场

要在 De AWS adline Cloud 中创建开发者群和队列资源，请使用 AWS Command Line Interface (AWS CLI)，如以下过程所示。您还将创建一个 AWS Identity and Access Management (IAM) 角色和一个客户托管队列 (CMF)，并将队列与您的队列关联。然后，您可以配置 AWS CLI 并确认您的服务器场已按指定设置并正常运行。

您可以使用此服务器场来探索 Deadline Cloud 的功能，然后开发和测试新的工作负载、自定义项和管道集成。

创建农场

1. 安装并配置 AWS Command Line Interface (AWS CLI) (如果尚未安装)。有关信息，请参阅[安装或更新到最新版本的 AWS CLI](#)。

2. 为您的农场创建一个名称，然后将该农场名称添加到~/.bashrc。这将使其可用于其他终端会话。

```
echo "DEV_FARM_NAME=DeveloperFarm" >> ~/.bashrc
source ~/.bashrc
```

3. 创建服务器场资源，并将其场 ID 添加到~/.bashrc。

```
aws deadline create-farm \
  --display-name "$DEV_FARM_NAME"

echo "DEV_FARM_ID=$(aws deadline list-farms \
  --query \"farms[?displayName=='$DEV_FARM_NAME'].farmId \
  | [0]\" --output text)" >> ~/.bashrc
source ~/.bashrc
```

4. 创建队列资源，并将其队列 ID 添加到 ~/.bashrc。

```
aws deadline create-queue \
  --farm-id $DEV_FARM_ID \
  --display-name "$DEV_FARM_NAME Queue" \
  --job-run-as-user '{"posix": {"user": "job-user", "group": "job-group"},
  "runAs": "QUEUE_CONFIGURED_USER"}'

echo "DEV_QUEUE_ID=$(aws deadline list-queues \
  --farm-id $DEV_FARM_ID \
  --query \"queues[?displayName=='$DEV_FARM_NAME Queue'].queueId \
  | [0]\" --output text)" >> ~/.bashrc
source ~/.bashrc
```

5. 为舰队创建 IAM 角色。此角色为队列中的工作人员主机提供在队列中运行作业所需的安全证书。

```
aws iam create-role \
  --role-name "${DEV_FARM_NAME}FleetRole" \
  --assume-role-policy-document \
  '{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Principal": {
          "Service": "credentials.deadline.amazonaws.com"
        },
      },
    ],
  }
```

```

        "Action": "sts:AssumeRole"
    }
]
}'
aws iam put-role-policy \
  --role-name "${DEV_FARM_NAME}FleetRole" \
  --policy-name WorkerPermissions \
  --policy-document \
  '{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": [
          "deadline:AssumeFleetRoleForWorker",
          "deadline:UpdateWorker",
          "deadline>DeleteWorker",
          "deadline:UpdateWorkerSchedule",
          "deadline:BatchGetJobEntity",
          "deadline:AssumeQueueRoleForWorker"
        ],
        "Resource": "*",
        "Condition": {
          "StringEquals": {
            "aws:PrincipalAccount": "${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}"
        }
    }
}
]
}'

```

6. 创建客户托管车队 (CMF)，并将其队列 ID 添加到 ~/.bashrc

```

FLEET_ROLE_ARN="arn:aws:iam::$(aws sts get-caller-identity \
    --query "Account" --output text):role/${DEV_FARM_NAME}FleetRole"
aws deadline create-fleet \
    --farm-id $DEV_FARM_ID \
    --display-name "$DEV_FARM_NAME CMF" \
    --role-arn $FLEET_ROLE_ARN \
    --max-worker-count 5 \
    --configuration \
    '{
        "customerManaged": {
            "mode": "NO_SCALING",
            "workerCapabilities": {
                "vCpuCount": {"min": 1},
                "memoryMiB": {"min": 512},
                "osFamily": "linux",
                "cpuArchitectureType": "x86_64"
            }
        }
    }
}'

echo "DEV_CMF_ID=$(aws deadline list-fleets \
    --farm-id \ $DEV_FARM_ID \
    --query \"fleets[?displayName=='\ $DEV_FARM_NAME CMF'].fleetId \
    | [0]\" --output text)" >> ~/.bashrc
source ~/.bashrc

```

7. 确保您可以访问 Deadline Cloud。

```
pip install deadline
```

8. 将 CMF 与您的队列关联。

```
aws deadline create-queue-fleet-association \  
  --farm-id $DEV_FARM_ID \  
  --queue-id $DEV_QUEUE_ID \  
  --fleet-id $DEV_CMF_ID
```

9. 要将默认场设置为场 ID，将队列设置为之前创建的队列 ID，请使用以下命令。

```
deadline config set defaults.farm_id $DEV_FARM_ID  
deadline config set defaults.queue_id $DEV_QUEUE_ID
```

10. (可选) 要确认您的服务器场是否按照您的规格进行设置，请使用以下命令：

- 列出所有农场 — **deadline farm list**
- 列出默认服务器场中的所有队列 — **deadline queue list**
- 列出默认服务器场中的所有舰队 — **deadline fleet list**
- 获取默认农场 — **deadline farm get**
- 获取默认队列 — **deadline queue get**
- 获取所有与默认队列关联的舰队 — **deadline fleet get**

第 2 步：在 Deadline Cloud 中以开发者模式运行工作器代理

必须先在工作服务器主机上以开发者模式运行 De AWS adline Cloud 工作器代理，然后才能在开发者群中运行提交到队列的作业。

在本教程的其余部分中，您将使用两个 AWS CloudShell 选项卡在开发者群中执行 AWS CLI 操作。在第一个选项卡中，您可以提交作业。在第二个选项卡中，您可以运行工作器代理。

Note

如果您的 CloudShell 会话闲置时间超过 20 分钟，则会话将超时并停止工作器代理。要重新启动工作器代理，请按照以下过程中的说明进行操作。

在开发者模式下运行工作器代理

1. 安装并配置 AWS Command Line Interface (AWS CLI) (如果尚未安装)。有关信息，请参阅[安装或更新到最新版本的 AWS CLI](#)。

- 当您的农场在第一个 CloudShell 选项卡中仍处于打开状态时，打开第二个 CloudShell 选项卡，然后创建demoenv-logs和demoenv-persist目录。

```
mkdir ~/demoenv-logs
mkdir ~/demoenv-persist
```

- 从 PyPI 下载并安装 Deadline Cloud 工作器代理包：

Note

启Windows用，需要将代理文件安装到 Python 的全局站点包目录中。目前不支持 Python 虚拟环境。

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

- 要允许工作器代理为正在运行的作业创建临时目录，请创建一个目录：

```
sudo mkdir /sessions
sudo chmod 750 /sessions
sudo chown cloudshell-user /sessions
```

- 使用您添加到的变量DEV_FARM_ID在开发者模式下运行 Deadline Cloud 工作器代理~/ .bashrc。 DEV_CMF_ID

```
deadline-worker-agent \
  --farm-id $DEV_FARM_ID \
  --fleet-id $DEV_CMF_ID \
  --run-jobs-as-agent-user \
  --logs-dir ~/demoenv-logs \
  --persistence-dir ~/demoenv-persist
```

当工作代理初始化然后轮询 UpdateWorkerSchedule API 操作时，将显示以下输出：

```
INFO    Worker Agent starting
[2024-03-27 15:51:01,292][INFO    ] # Worker Agent starting
[2024-03-27 15:51:01,292][INFO    ] AgentInfo
Python Interpreter: /usr/bin/python3
Python Version: 3.9.16 (main, Sep  8 2023, 00:00:00) - [GCC 11.4.1 20230605 (Red Hat 11.4.1-2)]
```

```
Platform: linux
...
[2024-03-27 15:51:02,528][INFO    ] # API.Resp # [deadline:UpdateWorkerSchedule]
(200) params={'assignedSessions': {}, 'cancelSessionActions': {},
'updateIntervalSeconds': 15} ...
[2024-03-27 15:51:17,635][INFO    ] # API.Resp # [deadline:UpdateWorkerSchedule]
(200) params=(Duplicate removed, see previous response) ...
[2024-03-27 15:51:32,756][INFO    ] # API.Resp # [deadline:UpdateWorkerSchedule]
(200) params=(Duplicate removed, see previous response) ...
...
```

6. 选择您的第一个 CloudShell 选项卡，然后列出车队中的员工。

```
deadline worker list --fleet-id $DEV_CMF_ID
```

将显示如下输出：

```
Displaying 1 of 1 workers starting at 0

- workerId: worker-8c9af877c8734e89914047111f
  status: STARTED
  createdAt: 2023-12-13 20:43:06+00:00
```

在生产配置中，Deadline Cloud 工作器代理需要在主机上以管理用户的身份设置多个用户和配置目录。您可以覆盖这些设置，因为您在自己的开发场中运行作业，只有您可以访问这些开发场。

第 3 步：使用 Deadline Cloud 提交和运行作业

要使用 De AWS adline Cloud 运行作业，请按以下步骤操作。使用第一个 AWS CloudShell 选项卡向您的开发者群提交作业。使用第二个 CloudShell 选项卡查看工作器代理的输出。

主题

- [提交simple_job样本](#)
- [提交simple_job带有参数的](#)
- [创建带有文件 I/O 的 simple_file_job 任务捆绑包](#)

提交simple_job样本

创建服务器场并运行工作器代理后，您可以将simple_job示例提交到 Deadline Cloud。

将simple_job样本提交到 Deadline Cloud

1. 安装并配置 AWS Command Line Interface (AWS CLI) (如果尚未安装)。有关信息，请参阅[安装或更新到最新版本的 AWS CLI](#)。

2. 从中下载示例 GitHub。

```
cd ~
git clone https://github.com/aws-deadline/deadline-cloud-samples.git
```

3. 选择您的第一个 CloudShell 选项卡，然后导航到任务捆绑包示例目录。

```
cd ~/deadline-cloud-samples/job_bundles/
```

4. 提交simple_job样本。

```
deadline bundle submit simple_job
```

5. 选择第二个 CloudShell 选项卡可查看有关呼叫BatchGetJobEntities、获取会话和运行会话操作的日志输出。

```
...
[2024-03-27 16:00:21,846][INFO    ] # Session.Starting
# [session-053d77cef82648fe2] Starting new Session.
[queue-3ba4ff683ff54db09b851a2ed8327d7b/job-d34cc98a6e234b6f82577940ab4f76c6]
[2024-03-27 16:00:21,853][INFO    ] # API.Req # [deadline:BatchGetJobEntity]
resource={'farm-id': 'farm-3e24cfc9bbcd423e9c1b6754bc1',
'fleet-id': 'fleet-246ee60f46d44559b6cce010d05', 'worker-id':
'worker-75e0fce9c3c344a69bff57fcd83'} params={'identifiers': [{'jobDetails':
{'jobId': 'job-d34cc98a6e234b6f82577940ab4'}]}} request_url=https://
scheduling.deadline.us-west-2.amazonaws.com/2023-10-12/farms/
farm-3e24cfc9bbcd423e /fleets/fleet-246ee60f46d44559b1 /workers/worker-
75e0fce9c3c344a69b /batchGetJobEntity
[2024-03-27 16:00:22,013][INFO    ] # API.Resp # [deadline:BatchGetJobEntity](200)
params={'entities': [{'jobDetails': {'jobId': 'job-d34cc98a6e234b6f82577940ab6',
'jobRunAsUser': {'posix': {'user': 'job-user', 'group': 'job-group'},
'runAs': 'QUEUE_CONFIGURED_USER'}, 'logGroupName': '/aws/deadline/
farm-3e24cfc9bbcd423e9c1b6754bc1/queue-3ba4ff683ff54db09b851a2ed83', 'parameters':
'*REDACTED*', 'schemaVersion': 'jobtemplate-2023-09'}]}], 'errors': []}
request_id=a3f55914-6470-439e-89e5-313f0c6
[2024-03-27 16:00:22,013][INFO    ] # Session.Add #
[session-053d77cef82648fea9c69827182] Appended new SessionActions.
```

```
(ActionIds: ['sessionaction-053d77cef82648fea9c69827182-0'])
[queue-3ba4ff683ff54db09b851a2ed8b/job-d34cc98a6e234b6f82577940ab6]
[2024-03-27 16:00:22,014][WARNING ] # Session.User #
[session-053d77cef82648fea9c69827182] Running as the Worker Agent's
user. (User: cloudshell-user) [queue-3ba4ff683ff54db09b851a2ed8b/job-
d34cc98a6e234b6f82577940ac6]
[2024-03-27 16:00:22,015][WARNING ] # Session.AWSCreds #
[session-053d77cef82648fea9c69827182] AWS Credentials are not available: Queue has
no IAM Role. [queue-3ba4ff683ff54db09b851a2ed8b/job-d34cc98a6e234b6f82577940ab6]
[2024-03-27 16:00:22,026][INFO ] # Session.Logs #
[session-053d77cef82648fea9c69827182] Logs streamed to: AWS CloudWatch
Logs. (LogDestination: /aws/deadline/farm-3e24cfc9bbcd423e9c1b6754bc1/
queue-3ba4ff683ff54db09b851a2ed83/session-053d77cef82648fea9c69827181)
[queue-3ba4ff683ff54db09b851a2ed83/job-d34cc98a6e234b6f82577940ab4]
[2024-03-27 16:00:22,026][INFO ] # Session.Logs #
[session-053d77cef82648fea9c69827182] Logs streamed to: local
file. (LogDestination: /home/cloudshell-user/demoenv-logs/
queue-3ba4ff683ff54db09b851a2ed8b/session-053d77cef82648fea9c69827182.log)
[queue-3ba4ff683ff54db09b851a2ed83/job-d34cc98a6e234b6f82577940ab4]
...
```

Note

仅显示工作器代理的日志输出。运行作业的会话有一个单独的日志。

6. 选择第一个选项卡，然后检查工作器代理写入的日志文件。
 - a. 导航到工作器代理日志目录并查看其内容。

```
cd ~/demoenv-logs
ls
```

- b. 打印工作器代理创建的第一个日志文件。

```
cat worker-agent-bootstrap.log
```

此文件包含工作人员代理输出，说明它如何调用 Deadline Cloud API 在您的队列中创建工作人员资源，然后担任队列角色。

- c. 打印工作器代理加入队列时的日志文件输出。

```
cat worker-agent.log
```

此日志包含有关工作代理执行的所有操作的输出，但不包含有关其运行作业的队列的输出，但这些资源的 ID 除外。

- d. 在与队列资源 ID 同名的目录中打印每个会话的日志文件。

```
cat $DEV_QUEUE_ID/session-*.log
```

如果作业成功，则日志文件输出将类似于以下内容：

```
cat $DEV_QUEUE_ID/$(ls -t $DEV_QUEUE_ID | head -1)
2024-03-27 16:00:22,026 WARNING Session running with no AWS Credentials.
2024-03-27 16:00:22,404 INFO
2024-03-27 16:00:22,405 INFO =====
2024-03-27 16:00:22,405 INFO ----- Running Task
2024-03-27 16:00:22,405 INFO =====
2024-03-27 16:00:22,406 INFO -----
2024-03-27 16:00:22,406 INFO Phase: Setup
2024-03-27 16:00:22,406 INFO -----
2024-03-27 16:00:22,406 INFO Writing embedded files for Task to disk.
2024-03-27 16:00:22,406 INFO Mapping: Task.File.runScript -> /sessions/
session-053d77cef82648fea9c698271812a/embedded_files_gj55_/tmp2u9yqtsz
2024-03-27 16:00:22,406 INFO Wrote: runScript -> /sessions/
session-053d77cef82648fea9c698271812a/embedded_files_gj55_/tmp2u9yqtsz
2024-03-27 16:00:22,407 INFO -----
2024-03-27 16:00:22,407 INFO Phase: Running action
2024-03-27 16:00:22,407 INFO -----
2024-03-27 16:00:22,407 INFO Running command /sessions/
session-053d77cef82648fea9c698271812a/tmpzuzxpslm.sh
2024-03-27 16:00:22,414 INFO Command started as pid: 471
2024-03-27 16:00:22,415 INFO Output:
2024-03-27 16:00:22,420 INFO Welcome to AWS Deadline Cloud!
2024-03-27 16:00:22,571 INFO
2024-03-27 16:00:22,572 INFO =====
2024-03-27 16:00:22,572 INFO ----- Session Cleanup
2024-03-27 16:00:22,572 INFO =====
2024-03-27 16:00:22,572 INFO Deleting working directory: /sessions/
session-053d77cef82648fea9c698271812a
```

7. 打印有关作业的信息。

```
deadline job get
```

提交作业时，系统会将其保存为默认值，因此您无需输入作业 ID。

提交simple_job带有参数的

您可以提交带有参数的作业。在以下步骤中，您可以编辑simple_job模板以包含自定义消息，提交simple_job，然后打印会话日志文件以查看消息。

提交带参数的simple_job示例

1. 选择您的第一个 CloudShell 选项卡，然后导航到任务捆绑包示例目录。

```
cd ~/deadline-cloud-samples/job_bundles/
```

2. 打印simple_job模板的内容。

```
cat simple_job/template.yaml
```

带有Message参数的parameterDefinitions部分应如下所示：

```
parameterDefinitions:
- name: Message
  type: STRING
  default: Welcome to AWS Deadline Cloud!
```

3. 提交带有参数值的simple_job示例，然后等待任务完成运行。

```
deadline bundle submit simple_job \  
-p "Message=Greetings from the developer getting started guide."
```

4. 要查看自定义消息，请查看最新的会话日志文件。

```
cd ~/demoenv-logs  
cat $DEV_QUEUE_ID/$(ls -t $DEV_QUEUE_ID | head -1)
```

创建带有文件 I/O 的 simple_file_job 任务捆绑包

渲染作业需要读取场景定义，从中渲染图像，然后将该图像保存到输出文件中。您可以通过让作业计算输入的哈希值而不是渲染图像来模拟此操作。

创建带有文件 I/O 的 simple_file_job 任务捆绑包

1. 选择您的第一个 CloudShell 选项卡，然后导航到任务捆绑包示例目录。

```
cd ~/deadline-cloud-samples/job_bundles/
```

2. 用新名称制作一份副本 simple_file_job。simple_job

```
cp -r simple_job simple_file_job
```

3. 按如下方式编辑作业模板：

Note

我们建议您使用 nano 这些步骤。如果您更喜欢使用 Vim，则必须使用设置其粘贴模式：`set paste`。

- a. 在文本编辑器中打开模板。

```
nano simple_file_job/template.yaml
```

- b. 添加以下内容 type、objectType、和 dataFlow parameterDefinitions。

```
- name: InFile
  type: PATH
  objectType: FILE
  dataFlow: IN
- name: OutFile
  type: PATH
  objectType: FILE
  dataFlow: OUT
```

- c. 将以下 bash 脚本命令添加到文件末尾，该命令从输入文件读取并写入输出文件。

```
# hash the input file, and write that to the output
```

```
sha256sum "{{Param.InFile}}" > "{{Param.OutFile}}"
```

更新后的内容 `template.yaml` 应与以下内容完全匹配：

```
specificationVersion: 'jobtemplate-2023-09'  
name: Simple File Job Bundle Example  
parameterDefinitions:  
  - name: Message  
    type: STRING  
    default: Welcome to AWS Deadline Cloud!  
  - name: InFile  
    type: PATH  
    objectType: FILE  
    dataFlow: IN  
  - name: OutFile  
    type: PATH  
    objectType: FILE  
    dataFlow: OUT  
steps:  
  - name: WelcomeToDeadlineCloud  
    script:  
      actions:  
        onRun:  
          command: '{{Task.File.runScript}}'  
      embeddedFiles:  
        - name: runScript  
          type: TEXT  
          runnable: true  
          data: |  
            #!/usr/bin/env bash  
            echo "{{Param.Message}}"  
  
            # hash the input file, and write that to the output  
            sha256sum "{{Param.InFile}}" > "{{Param.OutFile}}"
```

Note

如果要调整中的间距 `template.yaml`，请确保使用空格而不是缩进。

d. 保存文件，然后退出文本编辑器。

4. 为输入和输出文件提供参数值以提交 `simple_file_job`。

```
deadline bundle submit simple_file_job \  
  -p "InFile=simple_job/template.yaml" \  
  -p "OutFile=hash.txt"
```

5. 打印有关作业的信息。

```
deadline job get
```

- 您将看到如下输出：

```
parameters:  
  Message:  
    string: Welcome to AWS Deadline Cloud!  
  InFile:  
    path: /local/home/cloudshell-user/BundleFiles/JobBundle-Examples/simple_job/  
template.yaml  
  OutFile:  
    path: /local/home/cloudshell-user/BundleFiles/JobBundle-Examples/hash.txt
```

- 尽管您只提供了相对路径，但参数设置了完整路径。将当前工作目录与作为参数提供的任何路径 AWS CLI 连接起来，而这些路径的类型为该路径PATH。
- 在另一个终端窗口中运行的工作器代理接起并运行作业。此操作将创建hash.txt文件，您可以使用以下命令查看该文件。

```
cat hash.txt
```

此命令将打印类似于以下内容的输出。

```
eea2df5d34b54be5ac34c56a24a8c237b8487231a607eaf530a04d76b89c9cd3 /local/home/  
cloudshell-user/BundleFiles/JobBundle-Examples/simple_job/template.yaml
```

第 4 步：在 Deadline Cloud 中运行带有作业附件的作业

许多服务器场使用共享文件系统在提交作业的主机和运行作业的主机之间共享文件。例如，在前面的simple_file_job示例中，本地文件系统在终端窗口之间共享，AWS CloudShell 终端窗口在您提交作业的选项卡一和运行工作代理的选项卡二中运行。

当提交者工作站和工作主机位于同一个局域网中时，共享文件系统更具优势。如果您将数据存储在本地的访问数据的工作站附近，那么使用基于云的服务器场意味着您必须通过高延迟 VPN 共享文件系统或在云中同步文件系统。这两个选项都不容易设置或操作。

AWS Deadline Cloud 提供了一个带有作业附件的简单解决方案，类似于电子邮件附件。使用作业附件，您可以将数据附加到作业中。然后，Deadline Cloud 会处理在亚马逊简单存储服务 (Amazon S3) 存储桶中传输和存储任务数据的细节。

内容创建工作流程通常是迭代的，这意味着用户提交作业时会包含一小部分修改过的文件。由于 Amazon S3 存储桶将任务附件存储在内容可寻址的存储中，因此每个对象的名称都基于对象数据的哈希值，并且目录树的内容以任务所附的清单文件格式存储。

要运行带有作业附件的作业，请完成以下步骤。

主题

- [向队列中添加作业附件配置](#)
- [提交 simple_file_job 时附上工作附件](#)
- [了解任务附件如何存储在 Amazon S3 中](#)

向队列中添加作业附件配置

要在队列中启用作业附件，请向账户中的队列资源添加作业附件配置。

向队列中添加作业附件配置

1. 安装并配置 AWS Command Line Interface (AWS CLI) (如果尚未安装)。有关信息，请参阅[安装或更新到最新版本的 AWS CLI](#)。
2. 选择您的第一个 CloudShell 选项卡，然后输入以下命令之一，使用 Amazon S3 存储桶存储任务附件。
 - 如果您没有现有的私有 Amazon S3 存储桶，则可以创建和使用新的 S3 存储桶。

```
DEV_FARM_BUCKET=$(echo $DEV_FARM_NAME \  
  | tr '[:upper:]' '[:lower:]')-$(xxd -l 16 -p /dev/urandom)  
if [ "$AWS_REGION" == "us-east-1" ]; then LOCATION_CONSTRAINT=  
else LOCATION_CONSTRAINT="--create-bucket-configuration \  
  LocationConstraint=${AWS_REGION}"  
fi  
aws s3api create-bucket \  
  $LOCATION_CONSTRAINT \  
  $DEV_FARM_BUCKET
```

```
--acl private \  
--bucket ${DEV_FARM_BUCKET}
```

- 如果您已经拥有私有 Amazon S3 存储桶，则可以通过将其 *MY_BUCKET_NAME* 替换为存储桶的名称来使用它。

```
DEV_FARM_BUCKET=MY_BUCKET_NAME
```

3. 创建或选择 Amazon S3 存储桶后，将存储桶名称添加到 `~/.bashrc`，以使该存储桶可用于其他终端会话。

```
echo "DEV_FARM_BUCKET=$DEV_FARM_BUCKET" >> ~/.bashrc
```

4. 为队列创建 AWS Identity and Access Management (IAM) 角色。

```
aws iam create-role --role-name "${DEV_FARM_NAME}QueueRole" \  
  --assume-role-policy-document \  
    '{  
      "Version": "2012-10-17",  
      "Statement": [  
        {  
          "Effect": "Allow",  
          "Principal": {  
            "Service": "credentials.deadline.amazonaws.com"  
          },  
          "Action": "sts:AssumeRole"  
        }  
      ]  
    }'  
aws iam put-role-policy \  
  --role-name "${DEV_FARM_NAME}QueueRole" \  
  --policy-name S3BucketsAccess \  
  --policy-document \  
    '{  
      "Version": "2012-10-17",  
      "Statement": [  
        {  
          "Action": [  
            "s3:GetObject*",  
            "s3:GetBucket*",  
            "s3:List*",  
            "s3:DeleteObject*",  
            "s3:PutObject",
```

```

        "s3:PutObjectLegalHold",
        "s3:PutObjectRetention",
        "s3:PutObjectTagging",
        "s3:PutObjectVersionTagging",
        "s3:Abort*"
    ],
    "Resource": [
        "arn:aws:s3:::$DEV_FARM_BUCKET",
        "arn:aws:s3:::$DEV_FARM_BUCKET/*"
    ],
    "Effect": "Allow"
}
]
}'

```

- 更新您的队列以包含任务附件设置和 IAM 角色。

```

QUEUE_ROLE_ARN="arn:aws:iam::$(aws sts get-caller-identity \
    --query "Account" --output text):role/${DEV_FARM_NAME}QueueRole"
aws deadline update-queue \
    --farm-id $DEV_FARM_ID \
    --queue-id $DEV_QUEUE_ID \
    --role-arn $QUEUE_ROLE_ARN \
    --job-attachment-settings \
    '{
        "s3BucketName": "'$DEV_FARM_BUCKET'",
        "rootPrefix": "JobAttachments"
    }'

```

- 确认您已更新队列。

```
deadline queue get
```

输出如下所示：

```

...
jobAttachmentSettings:
  s3BucketName: DEV_FARM_BUCKET
  rootPrefix: JobAttachments
roleArn: arn:aws:iam::ACCOUNT_NUMBER:role/DeveloperFarmQueueRole
...

```

提交simple_file_job时附上工作附件

使用作业附件时，任务捆绑包必须为 Deadline Cloud 提供足够的信息，以确定作业的数据流，例如使用PATH参数。如果是simple_file_job，您编辑template.yaml文件是为了告诉 Deadline Cloud 数据流在输入文件和输出文件中。

将作业附件配置添加到队列后，您可以提交带有作业附件的 simple_file_job 示例。完成此操作后，您可以查看日志记录和作业输出，以确认simple_file_job带有作业附件的运行正常。

提交带有作业附件的 simple_file_job 任务捆绑包

1. 选择您的第一个 CloudShell 选项卡，然后打开该JobBundle-Samples目录。

2.

```
cd ~/AmazonDeadlineCloud-DocumentationAndSamples/JobBundle-Samples
```

3. 将 simple_file_job 提交到队列。当系统提示您确认上传时，请输入y。

```
deadline bundle submit simple_file_job \  
  -p InFile=simple_job/template.yaml \  
  -p OutFile=hash-jobattachments.txt
```

4. 要查看作业附件数据传输会话日志输出，请选择第二个 CloudShell 选项卡。

```
JOB_ID=$(deadline config get defaults.job_id)  
SESSION_ID=$(aws deadline list-sessions \  
  --farm-id $DEV_FARM_ID \  
  --queue-id $DEV_QUEUE_ID \  
  --job-id $JOB_ID \  
  --query "sessions[0].sessionId" \  
  --output text)  
cat ~/demoenv-logs/$DEV_QUEUE_ID/$SESSION_ID.log
```

5. 列出在会话中运行的会话操作。

```
aws deadline list-session-actions \  
  --farm-id $DEV_FARM_ID \  
  --queue-id $DEV_QUEUE_ID \  
  --job-id $JOB_ID \  
  --session-id $SESSION_ID
```

输出如下所示：

```
{
  "sessionactions": [
    {
      "sessionId": "sessionaction-123-0",
      "status": "SUCCEEDED",
      "startedAt": "<timestamp>",
      "endedAt": "<timestamp>",
      "progressPercent": 100.0,
      "definition": {
        "syncInputJobAttachments": {}
      }
    },
    {
      "sessionId": "sessionaction-123-1",
      "status": "SUCCEEDED",
      "startedAt": "<timestamp>",
      "endedAt": "<timestamp>",
      "progressPercent": 100.0,
      "definition": {
        "taskRun": {
          "taskId": "task-abc-0",
          "stepId": "step-def"
        }
      }
    }
  ]
}
```

第一个会话操作下载了输入作业附件，而第二个操作则像以前一样运行任务，然后上传了输出作业附件。

6. 列出输出目录。

```
ls *.txt
```

如图所示hash.txt的输出，但hash-jobattachments.txt不存在。

7. 下载最近作业的输出。

```
deadline job download-output
```

8. 查看已下载文件的输出。

```
cat hash-jobattachments.txt
```

输出如下所示：

```
eea2df5d34b54be5ac34c56a24a8c237b8487231a607eaf530a04d76b89c9cd3 /tmp/openjd/  
session-123/assetroot-abc/simple_job/template.yaml
```

了解任务附件如何存储在 Amazon S3 中

您可以使用 AWS Command Line Interface (AWS CLI) 上传或下载任务附件的数据，这些数据存储在 Amazon S3 存储桶中。了解 Deadline Cloud 如何在 Amazon S3 上存储作业附件将有助于您开发工作负载和管道集成。

检查 Deadline Cloud 作业附件如何存储在 Amazon S3 中

1. 选择您的第一个 CloudShell 选项卡，然后打开任务捆绑包示例目录。

```
cd ~/AmazonDeadlineCloud-DocumentationAndSamples/JobBundle-Samples
```

2. 检查作业属性。

```
deadline job get
```

输出如下所示：

```
parameters:  
  Message:  
    string: Welcome to Amazon Deadline Cloud!  
  InFile:  
    path: /home/cloudshell-user/AmazonDeadlineCloud-DocumentationAndSamples/  
JobBundle-Samples/simple_job/template.yaml  
  OutFile:  
    path: /home/cloudshell-user/AmazonDeadlineCloud-DocumentationAndSamples/  
JobBundle-Samples/hash-jobattachments.txt  
attachments:  
  manifests:  
    - rootPath: /home/cloudshell-user/AmazonDeadlineCloud-DocumentationAndSamples/  
JobBundle-Samples  
      rootPathFormat: posix
```

```

outputRelativeDirectories:
- .
inputManifestPath: farm-3040c59a5b9943d58052c29d907a645d/queue-
cde9977c9f4d4018a1d85f3e6c1a4e6e/Inputs/
f46af01ca8904cd8b514586671c79303/0d69cd94523ba617c731f29c019d16e8_input.xxh128
inputManifestHash: f95ef91b5dab1fc1341b75637fe987ee
fileSystem: COPIED

```

附件字段包含清单结构列表，这些清单结构描述了作业运行时使用的输入和输出数据路径。查看rootPath提交作业的计算机上的本地目录路径。要查看包含清单文件的 Amazon S3 对象后缀，请查看inputManifestFile。清单文件包含任务输入数据的目录树快照的元数据。

3. 漂亮地打印 Amazon S3 清单对象以查看任务的输入目录结构。

```

MANIFEST_SUFFIX=$(aws deadline get-job \
  --farm-id $DEV_FARM_ID \
  --queue-id $DEV_QUEUE_ID \
  --job-id $JOB_ID \
  --query "attachments.manifests[0].inputManifestPath" \
  --output text)
aws s3 cp s3://$DEV_FARM_BUCKET/JobAttachments/Manifests/$MANIFEST_SUFFIX - | jq .

```

输出如下所示：

```

{
  "hashAlg": "xxh128",
  "manifestVersion": "2023-03-03",
  "paths": [
    {
      "hash": "2ec297b04c59c4741ed97ac8fb83080c",
      "mtime": 1698186190000000,
      "path": "simple_job/template.yaml",
      "size": 445
    }
  ],
  "totalSize": 445
}

```

4. 构造用于保存输出任务附件清单的 Amazon S3 前缀，并在其下列出对象。

```

SESSION_ACTION=$(aws deadline list-session-actions \
  --farm-id $DEV_FARM_ID \

```

```

--queue-id $DEV_QUEUE_ID \
--job-id $JOB_ID \
--session-id $SESSION_ID \
--query "sessionActions[?definition.taskRun != null] | [0]"
STEP_ID=$(echo $SESSION_ACTION | jq -r .definition.taskRun.stepId)
TASK_ID=$(echo $SESSION_ACTION | jq -r .definition.taskRun.taskId)
TASK_OUTPUT_PREFIX=JobAttachments/Manifests/$DEV_FARM_ID/$DEV_QUEUE_ID/$JOB_ID/
$STEP_ID/$TASK_ID/
aws s3api list-objects-v2 --bucket $DEV_FARM_BUCKET --prefix $TASK_OUTPUT_PREFIX

```

输出任务附件不是直接从任务资源中引用的，而是根据服务器场资源 ID 放置在 Amazon S3 存储桶中。

5. 获取特定会话操作 ID 的最新清单对象密钥，然后漂亮地打印清单对象。

```

SESSION_ACTION_ID=$(echo $SESSION_ACTION | jq -r .sessionActionId)
MANIFEST_KEY=$(aws s3api list-objects-v2 \
  --bucket $DEV_FARM_BUCKET \
  --prefix $TASK_OUTPUT_PREFIX \
  --query "Contents[*].Key" --output text \
  | grep $SESSION_ACTION_ID \
  | sort | tail -1)
MANIFEST_OBJECT=$(aws s3 cp s3://$DEV_FARM_BUCKET/$MANIFEST_KEY -)
echo $MANIFEST_OBJECT | jq .

```

您将在输出hash-jobattachments.txt中看到该文件的属性，如下所示：

```

{
  "hashAlg": "xxh128",
  "manifestVersion": "2023-03-03",
  "paths": [
    {
      "hash": "f60b8e7d0fabf7214ba0b6822e82e08b",
      "mtime": 1698785252554950,
      "path": "hash-jobattachments.txt",
      "size": 182
    }
  ],
  "totalSize": 182
}

```

每次运行任务时，您的作业只会有一个清单对象，但一般而言，每个任务运行可能会有更多对象。

6. 在前缀下查看可寻址内容的 Amazon S3 存储输出。Data

```
FILE_HASH=$(echo $MANIFEST_OBJECT | jq -r .paths[0].hash)
FILE_PATH=$(echo $MANIFEST_OBJECT | jq -r .paths[0].path)
aws s3 cp s3://$DEV_FARM_BUCKET/JobAttachments/Data/$FILE_HASH -
```

输出如下所示：

```
eea2df5d34b54be5ac34c56a24a8c237b8487231a607eaf530a04d76b89c9cd3 /tmp/openjd/
session-123/assetroot-abc/simple_job/template.yaml
```

第 5 步：在 Deadline Cloud 中向开发者群添加服务托管队列

AWS CloudShell 无法提供足够的计算容量来测试更大的工作负载。它也未配置为处理在多个工作主机上分配任务的作业。

您可以将 A CloudShell uto Scaling 服务托管队列 (SMF) 添加到您的开发者群中，而不必使用。SMF 为较大的工作负载提供了足够的计算容量，并且可以处理需要在多个工作主机上分配作业任务的作业。除非您关闭 CMF 工作程序，否则调度程序将同时使用 SMF 和 CMF 工作器来运行作业。

将服务托管队列添加到您的开发者群中

1. 安装并配置 AWS Command Line Interface (AWS CLI) (如果尚未安装)。有关信息，请参阅[安装或更新到最新版本的 AWS CLI](#)。
2. 选择您的第一个 AWS CloudShell 选项卡，然后创建服务托管队列并将其队列 ID 添加到 .bashrc。此操作使其可用于其他终端会话。

```
FLEET_ROLE_ARN="arn:aws:iam::$(aws sts get-caller-identity \
  --query "Account" --output text):role/${DEV_FARM_NAME}FleetRole"
aws deadline create-fleet \
  --farm-id $DEV_FARM_ID \
  --display-name "$DEV_FARM_NAME SMF" \
  --role-arn $FLEET_ROLE_ARN \
  --max-worker-count 5 \
  --configuration \
  '{
```

```

    "serviceManagedEc2": {
      "instanceCapabilities": {
        "vCpuCount": {
          "min": 2,
          "max": 4
        },
        "memoryMiB": {
          "min": 512
        },
        "osFamily": "linux",
        "cpuArchitectureType": "x86_64"
      },
      "instanceMarketOptions": {
        "type": "spot"
      }
    }
  }
}'

```

```

echo "DEV_SMF_ID=$(aws deadline list-fleets \
  --farm-id $DEV_FARM_ID \
  --query "fleets[?displayName=='$DEV_FARM_NAME SMF'].fleetId \
  | [0]" --output text)" >> ~/.bashrc
source ~/.bashrc

```

3. 将 SMF 与您的队列关联。

```

aws deadline create-queue-fleet-association \
  --farm-id $DEV_FARM_ID \
  --queue-id $DEV_QUEUE_ID \
  --fleet-id $DEV_SMF_ID

```

4.

Note

除非您关闭 CMF 工作程序，否则调度程序将同时使用 SMF 和 CMF 工作器来运行作业。

提交 `simple_file_job` 到队列。当系统提示您确认上传时，请输入 `y`。

```

deadline bundle submit simple_file_job \
  -p InFile=simple_job/template.yaml \
  -p OutFile=hash-jobattachments.txt

```

5. 确认 SMF 工作正常。

```
deadline fleet get
```

- 工作人员可能需要几分钟才能开始。
- 您的客户托管车队和服务托管车队将是ACTIVE。queueFleetAssociationsStatus
- SMF autoScalingStatus 将从变GROWING为。STEADY

您的状态将类似于以下内容：

```
fleetId: fleet-2cc78e0dd3f04d1db427e7dc1d51ea44
farmId: farm-63ee8d77cdab4a578b685be8c5561c4a
displayName: DeveloperFarm SMF
description: ''
status: ACTIVE
autoScalingStatus: STEADY
targetWorkerCount: 0
workerCount: 0
minWorkerCount: 0
maxWorkerCount: 5
```

6. 查看您提交的作业的日志。此日志存储在 Amazon Logs 的 CloudWatch 日志中，而不是 CloudShell 文件系统中。

```
JOB_ID=$(deadline config get defaults.job_id)
SESSION_ID=$(aws deadline list-sessions \
  --farm-id $DEV_FARM_ID \
  --queue-id $DEV_QUEUE_ID \
  --job-id $JOB_ID \
  --query "sessions[0].sessionId" \
  --output text)
aws logs tail /aws/deadline/$DEV_FARM_ID/$DEV_QUEUE_ID \
  --log-stream-names $SESSION_ID
```

第 6 步：清理 Deadline Cloud 中的农场资源

要开发和测试新的工作负载和管道集成，您可以继续使用为本教程创建的 Deadline Cloud 开发者群组。如果您不再需要开发者群组，则可以删除其资源，包括场、队列、队列、AWS Identity and

Access Management (IAM) 角色和 Amazon Logs 中的 CloudWatch 日志。删除这些资源后，您需要重新开始本教程才能使用这些资源。有关更多信息，请参阅 [为 Deadline Cloud 设置开发者工作站](#)。

清理开发者农场资源

1. 安装并配置 AWS Command Line Interface (AWS CLI) (如果尚未安装)。有关信息，请参阅[安装或更新到最新版本的 AWS CLI](#)。
2. 选择第一个 CloudShell 选项卡，然后停止队列的所有队列队列关联。

```
FLEETS=$(aws deadline list-queue-fleet-associations \  
  --farm-id $DEV_FARM_ID \  
  --queue-id $DEV_QUEUE_ID \  
  --query "queueFleetAssociations[].fleetId" \  
  --output text)  
for FLEET_ID in $FLEETS; do  
  aws deadline update-queue-fleet-association \  
    --farm-id $DEV_FARM_ID \  
    --queue-id $DEV_QUEUE_ID \  
    --fleet-id $FLEET_ID \  
    --status STOP_SCHEDULING_AND_CANCEL_TASKS  
done
```

3. 列出队列队列关联。

```
aws deadline list-queue-fleet-associations \  
  --farm-id $DEV_FARM_ID \  
  --queue-id $DEV_QUEUE_ID
```

在输出报告之前，您可能需要重新运行该命令"status": "STOPPED"，然后才能继续下一步。此过程可能需要几分钟才能完成。

```
{  
  "queueFleetAssociations": [  
    {  
      "queueId": "queue-abcdefgh01234567890123456789012id",  
      "fleetId": "fleet-abcdefgh01234567890123456789012id",  
      "status": "STOPPED",  
      "createdAt": "2023-11-21T20:49:19+00:00",  
      "createdBy": "arn:aws:sts::123456789012:assumed-role/RoleToBeAssumed/  
MySessionName",  
      "updatedAt": "2023-11-21T20:49:38+00:00",
```

```

        "updatedBy": "arn:aws:sts::123456789012:assumed-role/RoleToBeAssumed/
MySessionName"
    },
    {
        "queueId": "queue-abcdefgh01234567890123456789012id",
        "fleetId": "fleet-abcdefgh01234567890123456789012id",
        "status": "STOPPED",
        "createdAt": "2023-11-21T20:32:06+00:00",
        "createdBy": "arn:aws:sts::123456789012:assumed-role/RoleToBeAssumed/
MySessionName",
        "updatedAt": "2023-11-21T20:49:39+00:00",
        "updatedBy": "arn:aws:sts::123456789012:assumed-role/RoleToBeAssumed/
MySessionName"
    }
]
}

```

4. 删除队列的所有队列关联。

```

for FLEET_ID in $FLEETS; do
    aws deadline delete-queue-fleet-association \
        --farm-id $DEV_FARM_ID \
        --queue-id $DEV_QUEUE_ID \
        --fleet-id $FLEET_ID
done

```

5. 删除与您的队列关联的所有舰队。

```

for FLEET_ID in $FLEETS; do
    aws deadline delete-fleet \
        --farm-id $DEV_FARM_ID \
        --fleet-id $FLEET_ID
done

```

6. 删除队列。

```

aws deadline delete-queue \
    --farm-id $DEV_FARM_ID \
    --queue-id $DEV_QUEUE_ID

```

7. 删除农场。

```

aws deadline delete-farm \

```

```
--farm-id $DEV_FARM_ID
```

8. 删除农场的其他 AWS 资源。

- a. 删除舰队 AWS Identity and Access Management (IAM) 角色。

```
aws iam delete-role-policy \  
  --role-name "${DEV_FARM_NAME}FleetRole" \  
  --policy-name WorkerPermissions  
aws iam delete-role \  
  --role-name "${DEV_FARM_NAME}FleetRole"
```

- b. 删除队列 IAM 角色。

```
aws iam delete-role-policy \  
  --role-name "${DEV_FARM_NAME}QueueRole" \  
  --policy-name S3BucketsAccess  
aws iam delete-role \  
  --role-name "${DEV_FARM_NAME}QueueRole"
```

- c. 删除 Amazon CloudWatch 日志组。每个队列和队列都有自己的日志组。

```
aws logs delete-log-group \  
  --log-group-name "/aws/deadline/$DEV_FARM_ID/$DEV_QUEUE_ID"  
aws logs delete-log-group \  
  --log-group-name "/aws/deadline/$DEV_FARM_ID/$DEV_CMF_ID"  
aws logs delete-log-group \  
  --log-group-name "/aws/deadline/$DEV_FARM_ID/$DEV_SMF_ID"
```

设置 Deadline Cloud 提交者

此过程适用于想要安装、设置和启动 Deadline Cloud 提交器的管理员和艺术家。AWS Deadline Cloud 提交者是一个数字内容创作 (DCC) 插件。艺术家使用它从他们熟悉的第三方 DCC 界面提交作业。

Note

此过程必须在美术师用于提交渲染图的所有工作站上完成。

主题

- [第 1 步：安装 Deadline Cloud 提交器](#)
- [第 2 步：安装和设置 Deadline 云监视器](#)
- [第 3 步：启动 Deadline Cloud 提交器](#)

第 1 步：安装 Deadline Cloud 提交器

以下各节将指导您完成安装 Deadline Cloud 提交器的步骤。

下载提交者安装程序

在安装 Deadline Cloud 提交器之前，必须先下载提交者安装程序。目前，Deadline Cloud 提交者安装程序仅支持 Windows 和 Linux。

1. 登录 AWS Management Console 并打开 [Deadline Cloud 控制台](#)。
2. 从侧面导航窗格中选择“下载”。
3. 找到 Deadline Cloud 提交者安装程序部分。
4. 为您的计算机操作系统选择安装程序，然后选择“下载”。

(可选) 验证已下载软件的真实性的真实性

要验证您下载的软件是否真实，请对 Windows 或使用以下步骤 Linux。

Note

您可以按照这些说明先验证安装程序，然后在下一节（步骤 2）中下载 Deadline Cloud 监视器后对其进行验证。

Windows

要验证您下载的文件的安全性，请完成以下步骤。

1. 在以下命令中，*file* 替换为要验证的文件。例如，**`C:\PATH\TO\MY\DeadlineCloudSubmitter-windows-x64-installer.exe`**。另外，请 *signtool-sdk-version* 替换为已安装的 SignTool SDK 版本。例如，**`10.0.22000.0`**。

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

2. 例如，您可以通过运行以下命令来验证 Deadline Cloud 提交者安装程序文件：

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

Linux

要验证下载文件的真实性，请使用 gpg 命令行工具。

1. 通过运行以下命令导入 Deadline Cloud 提交者安装程序的 OpenPGP 密钥：

```
gpg --import --armor <<EOF  
-----BEGIN PGP PUBLIC KEY BLOCK-----  
  
mQINBGX6GQsBEADduUtJgqSXI+q7606fsFwEYKmbnlyL0xKvlq32EZuyv0otZo5L  
le4m5Gg52AzrvPvDiUTLooAlvYeozaYyirIGsK08Ydz0Ftdjroiuh/mw9JSJDJRI  
rnRn5yKet1JFzckjopA3pjsTBP6lW/mb1bDBDEwwwtH0x9lV7A03FJ9T7Uzu/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  
tCxBV1MgRGVhZGxpbnUgQ2xvdWQgPGF3cy1kZWFKbGluZUBhbWF6b24uY29tPokC  
VwQTAQgAQRyhBLhAwIwpqQeWoHH6pfbNP0a3bzzvBQJl+hkLAXsvBAUJA8JnAAUL  
CQgHAgIiAgYVCgkICwIDFgIBAh4HAheAAAoJEPbNP0a3bzzvKswQAJXzKSAY8sY8  
F6Eas2oYwIDDDurs8FiEnFghjUE06MTt9AykF/jw+CQg2UzFtEy0bHBymhgmhXE  
3buVeom96tgM3ZDfZu+sxi5pGX6oAQnZ6riztN+VpkpQmLgwtMGpSML13KLwnv2k  
WK8mrR/fPMkfaewB7A6RIUYiW33GAL4KfMIs8/vIwIJw99NxHpZQVoU6dFpuDtE  
10uxGcCqGJ7mAmo6H/YawSNp2Ns80gyqIKYo7o3LJ+WRroIR1Qyctq8gnR9JvYXX  
42ASqLq5+0XKo4qh81blXKYqtc176BbbSNFjWnzIQgKDgNiHFZCdc0VgqDhw015r  
NICbqqwwNLj/Fr2kecYx180Ktp10j00w5I0yh3bf3MVGwnYRdjvA1v+/CO+55N4g  
z0kf50Lcdu5RtqV10XBCifn28pecqPaSdYcssYSR15DLiFktGbNzTgCZZwITTKQc  
af8PPdTGtnnb6P+cdbW3bt9MVtN5/dgSHLThnS8MPEuNCtkTnpXshuVuBGgwBMdb  
qUC+HjqvhZzbwns8dr5WI+6HWNBFgGANn6ageY158vVp0UkuNP8wcWjRARciHXZx  
ku6W2jPTHDWGNrBQ02Fx7fd2QYJheIPPASHcfJ0+xgWCoF45D0vAxAJ8gGg9Eq+
```

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

2. 确定是否信任OpenPGP密钥。在决定是否信任上述密钥时需要考虑的一些因素包括：
 - 您用于从本网站获取 GPG 密钥的互联网连接是安全的。
 - 您访问本网站时使用的设备是安全的。
 - AWS 已采取措施保护本网站上OpenPGP公钥的托管。
3. 如果您决定信任该OpenPGP密钥，请使用gpg类似于以下示例的方法编辑该密钥以使其可信：

```
$ 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. 验证安装程序

要验证安装程序，请完成以下步骤：

- a. 返回 [Deadline Cloud 控制台](#) 下载页面，下载 Deadline Cloud 提交者安装程序的签名文件。
- b. 运行以下命令验证 Deadline Cloud 提交者安装程序的签名：

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

5. 验证截止日期云监控

Note

您可以使用签名文件或特定于平台的方法来验证 Deadline Cloud 监控器的下载。有关平台特定的方法，请根据您下载的文件类型查看 Linux (Applmage) 选项卡或选项卡。Linux (DEB)

要使用签名文件验证 Deadline Cloud 监控桌面应用程序，请完成以下步骤：

- a. 返回 [Deadline Cloud 控制台](#) 下载页面并下载相应的 .sig 文件，然后运行

对于 .deb：

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

对于。Applmage:

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

- b. 确认输出类似于以下内容：

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

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

如果输出包含短语 Good signature from "AWS Deadline Cloud"，则表示签名已成功通过验证，您可以运行 Deadline Cloud 监视器安装脚本。

Linux (DEB)

要验证使用 Linux .deb 二进制文件的软件包，请先完成选项卡中的 Linux 步骤 1-3。

dpkg 是大多数 debian 基础 Linux 发行版中的核心软件包管理工具。您可以使用该工具验证 .deb 文件。

1. 从 [Deadline Cloud 控制台](#) 下载页面下载 Deadline Cloud monitor .deb 文件。
2. **<APP_VERSION>** 替换为要验证的 .deb 文件的版本。

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

3. 输出将类似于：

```
Processing deadline-cloud-monitor_1.1.1_amd64.deb... GOODSIG  
_gpgbuilder B840C08C29A90796A071FAA5F6CD3C 171200
```

4. 要验证 .deb 文件，请确认输出中 GOODSIG 是否存在。

Linux (AppImage)

验证使用 Linux 的软件包 AppImage 二进制，首先完成 Linux 选项卡中的步骤 1-3。

1. 从 [Deadline Cloud 控制台](#) 的“下载”页面下载 Deadline Cloud AppImage 文件。
2. **<APP_VERSION>** 替换为要验证的文件，请完成以下步骤：
 - a. 写下来自 .AppImage 文件到 .sig 文件中。

```
./deadline-cloud-monitor_<APP_VERSION>_amd64.AppImage  
--appimage-signature > ./deadline-cloud-  
monitor_<APP_VERSION>_amd64.AppImage.sig
```

- b. 使用生成的.sig 文件使用以下命令进行验证。

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

- c. (可选) 如果显示权限被拒绝错误, 请使用以下命令添加执行权限。

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

- d. 确认输出类似于以下内容:

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

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

如果输出包含短语 Good signature from "AWS Deadline Cloud", 则表示签名已成功通过验证, 您可以运行 Deadline Cloud 监视器安装脚本。

安装 Deadline Cloud 提交者

您可以使用 Windows 或 Linux 安装 Deadline Cloud 提交器。使用安装程序, 您可以安装以下提交者:

- 2024 年 Maya
- Nuke 14.0-15.0
- Houdini 19.5
- 按键射击 12
- 搅拌机 3.6
- 虚幻引擎 5

Windows

1. 在文件浏览器中, 导航到安装程序下载的文件夹, 然后选择 DeadlineCloudSubmitter-windows-x64-installer.exe。
 - a. 如果显示了 Windows 保护了你的电脑的弹出窗口, 请选择“更多信息”。

- b. 无论如何都要选择“运行”。
2. De AWS adline Cloud 提交者设置向导打开后，选择“下一步”。
3. 通过完成以下步骤之一来选择安装范围：
 - 要仅为当前用户安装，请选择用户。
 - 要为所有用户安装，请选择“系统”。

如果选择“系统”，则必须退出安装程序，然后通过完成以下步骤以管理员身份重新运行它：

- a. 右键单击 **DeadlineCloudSubmitter-windows-x64-installer.exe**，然后选择“以管理员身份运行”。
- b. 输入您的管理员凭据，然后选择“是”。
- c. 选择系统作为安装范围。
4. 选择安装范围后，选择“下一步”。
5. 再次选择“下一步”以接受安装目录。
6. 为其选择集成提交者 Nuke，或您要安装的任何提交者。
7. 选择下一步。
8. 查看安装情况，然后选择“下一步”。
9. 再次选择“下一步”，然后选择“完成”。

Linux

Note

Deadline Cloud 集成Nuke安装程序Linux和 Deadline Cloud 监视器只能安装在至少 GLIBC 2.31 的Linux发行版上。

1. 打开终端窗口。
2. 要对安装程序进行系统安装，请输入命令 **sudo -i** 并按 Enter 键成为 root 用户。
3. 导航到下载安装程序的位置。

例如，**cd /home/*USER*/Downloads**。

4. 要使安装程序可执行，请输入 **chmod +x DeadlineCloudSubmitter-linux-x64-installer.run**。
5. 要运行 Deadline Cloud 提交者安装程序，请输入 **./DeadlineCloudSubmitter-linux-x64-installer.run**。
6. 安装程序打开后，按照屏幕上的提示完成安装向导。

您可以安装此处未列出的其他提交者。我们使用 Deadline Cloud 库来构建提交者。您可以在 [aws-GitHub deadline](#) 组织中找到这些库和提交者的源代码。

第 2 步：安装和设置 Deadline 云监视器

您可以使用 Windows 或安装 Deadline Cloud 监控桌面应用程序 Linux。

Windows

1. 如果您尚未登录，请登录 AWS Management Console 并打开 [Deadline Cloud 控制台](#)。
2. 从左侧导航窗格中选择“下载”。
3. 在 Deadline Cloud 监视器部分中，选择适用于您计算机操作系统的文件。
4. 要下载 Deadline Cloud 监视器，请选择下载。

Linux

在 RPM 发行版 AppImage 上安装 Deadline 云监视器

1. 下载最新的 Deadline 云监视器 AppImage。
2. 要使该 AppImage 文件成为可执行文件，请输入 **chmod a+x deadline-cloud-monitor_<APP_VERSION>_amd64.AppImage**。
3. 要设置正确的 SSL 证书路径，请输入 **sudo ln -sf /etc/ssl/certs/ca-bundle.crt /etc/ssl/certs/ca-certificates.crt**。

在 Debian 发行版 AppImage 上安装 Deadline Cloud 监视器

1. 下载最新的 Deadline 云监视器 AppImage。

2.

 Note

此步骤适用于 Ubuntu 22 及更高版本。对于其他版本的 Ubuntu，请跳过此步骤。

要安装 libfuse2，请输入 **sudo apt update**

sudo apt install libfuse2.

3. 要使该 AppImage 文件成为可执行文件，请输入 **chmod a+x deadline-cloud-monitor_<APP_VERSION>_amd64.AppImage.**

要在 Debian 发行版上安装 Debian Cloud 监控 Debian 软件包

1. 下载最新的 Deadline 云监控 Debian 软件包。

2.

 Note

此步骤适用于 Ubuntu 22 及更高版本。对于其他版本的 Ubuntu，请跳过此步骤。

要安装 libssl1.1，请输入 **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. 要安装 Deadline Cloud monitor Debian 软件包 **sudo apt update**

sudo apt install ./deadline-cloud-monitor_<APP_VERSION>_amd64.deb.

4. 如果在依赖关系未得到满足的软件包上安装失败，请修复损坏的软件包，然后运行以下命令。

sudo apt --fix-missing update

sudo apt update

sudo apt install -f

完成下载后，您可以验证所下载软件的真实性和完整性。请参阅步骤 1 中的验证已下载软件的真实性和完整性。

下载 Deadline Cloud 监视器并验证真实性后，使用以下步骤设置 Deadline Cloud 监视器。

设置 Deadline Cloud 监视器

1. 打开截止日期云监视器。
2. 当系统提示您创建新的配置文件时，请完成以下步骤。
 - a. 在 URL 输入中输入您的监视器 URL，如下所示 **https://MY-MONITOR.deadlinecloud.amazonaws.com/**
 - b. 输入配置文件名称。
 - c. 选择“创建个人资料”。

您的个人资料已创建，您的凭据现在可以与任何使用您创建的配置文件名称的软件共享。

3. 创建 Deadline Cloud 监视器配置文件后，您无法更改配置文件名称或工作室网址。如果您需要进行更改，请改为执行以下操作：
 - a. 删除个人资料。在左侧导航窗格中，选择 Deadline Cloud 监控、设置、删除。
 - b. 使用您想要的更改创建新的个人资料。
4. 在左侧导航窗格中，使用 >Deadline Cloud 监控选项执行以下操作：
 - 更改 Deadline Cloud 监视器配置文件以登录到其他显示器。
 - 启用自动登录，这样您就不必在随后打开 Deadline Cloud 监视器时输入监视器网址。
5. 关闭截止日期云监控窗口。它继续在后台运行，每 15 分钟同步一次您的凭证。
6. 对于计划用于渲染项目的每个数字内容创作 (DCC) 应用程序，请完成以下步骤：
 - a. 从 Deadline Cloud 提交者处打开 Deadline Cloud 工作站配置。
 - b. 在工作站配置中，选择您在 Deadline Cloud 监视器中创建的配置文件。现在，您的 Deadline Cloud 凭据已与此 DCC 共享，您的工具应该可以按预期运行。

第 3 步：启动 Deadline Cloud 提交器

以下各节将指导您完成在 Blender、Nuke、和 Houdini 中启动 Deadline Cloud 提交者插件的步骤。Maya

要在中启动 Deadline Cloud 提交者 Blender

Note

Support Blender 是使用服务管理队列的Conda环境提供的。有关更多信息，请参阅 [默认Conda队列环境](#)。

1. 打开 Blender。
2. 打开一个存在于资产根目录中的依赖项的Blender场景。
3. 在“渲染”菜单中，选择“截止日期云”对话框。
 - a. 如果您尚未在 Deadline Cloud 提交者中进行身份验证，则凭据状态将显示 N EEDS_LOGIN。
 - b. 选择登录。
 - c. 将显示登录浏览器窗口。使用您的用户凭据登录。
 - d. 选择允许。您现在已登录，凭证状态将显示为已验证。
4. 选择提交。

要在中启动 Deadline Cloud 提交者 Foundry Nuke

Note

Support Nuke 是使用服务管理队列的Conda环境提供的。有关更多信息，请参阅 [默认Conda队列环境](#)。

1. 打开 Nuke。
2. 打开一个存在于资产根目录中的依赖项的Nuke脚本。
3. 选择 Thinkbox，然后选择“提交到 Deadline Cloud”以启动提交者。
 - a. 如果您尚未在 Deadline Cloud 提交者中进行身份验证，则凭证状态将显示为 N EEDS_LOGIN。
 - b. 选择登录。
 - c. 在登录浏览器窗口中，使用您的用户凭据登录。
 - d. 选择允许。您现在已登录，凭证状态将显示为已验证。

4. 选择提交。

要在中启动 Deadline Cloud 提交者 Maya

Note

使用服务管理队列的Conda环境Arnold for Maya(MtoA)为Maya和提供支持。有关更多信息，请参阅 [默认Conda队列环境](#)。

1. 打开 Maya。
2. 设置您的项目，然后打开资产根目录中存在的文件。
3. 选择 Windows → 设置/首选项 → 插件管理器。
4. 搜索DeadlineCloud提交者。
5. 要加载 Deadline Cloud 提交者插件，请选择已加载。
 - a. 如果您尚未在 Deadline Cloud 提交者中进行身份验证，则凭证状态将显示为 N EEDS_LOGIN。
 - b. 选择登录。
 - c. 将显示登录浏览器窗口。使用您的用户凭据登录。
 - d. 选择允许。您现在已登录，凭证状态显示为已验证。
6. （可选）要在每次打开时加载 Deadline Cloud 提交者插件Maya，请选择自动加载。
7. 选择 Deadline Cloud 功能区，然后选择绿色按钮启动提交者。

要在中启动 Deadline Cloud 提交者 Houdini

Note

Support Houdini 是使用服务管理队列的Conda环境提供的。有关更多信息，请参阅 [默认Conda队列环境](#)。

1. 打开 Houdini。
2. 在网络编辑器中，选择 /out 网络。
3. 按 Tab 键，然后输入 **deadline**。

4. 选择 Deadline Cloud 选项，然后将其连接到您的现有网络。
5. 双击“截止日期云”节点。

要在中启动 Deadline Cloud 提交者 KeyShot

这假设你已经下载了 Deadline Cloud 和 PySide 2.

1. 将 Deadline-cloud-for-keyshot/keyshot_Script/Submit 到 AWS Deadline Cloud.py 的文件复制或链接到 KeyShot

例如，在开启Windows时，脚本文件夹的位置将是 **C:/Users/*USER*/Documents/KeyShot 12/Scripts**。

2. 设置以下环境变量。

- a. 将环境变量设置 **DEADLINE_PYTHON** 为截止日期云和 PySide 2 所在的 Python 安装路径。

例如，在开启Windows时，如果使用 Python 3.10，则命令可能是 **set DEADLINE_PYTHON=C:/Users/*USER*/AppData/Local/Programs/Python/Python310/python**

- b. 将环境变量 **DEADLINE_KEYSHOT** 设置为 keyshot_submitter 文件夹的路径。

例如，在 on 上Windows，如果源代码位于您的桌面上，则命令可能是 **set DEADLINE_KEYSHOT=C:/Users/*USER*/Desktop/deadline-cloud-for-keyshot/src/deadline/keyshot_submitter**。

3. 设置环境变量后，启动KeyShot。
4. 要从中启动提交者 KeyShot，请选择“脚本控制台”Windows、“提交到 AWS Deadline Cloud”和“运行”。

要在中启动 Deadline Cloud 提交者 Unreal Engine

这假设你已经下载了 Deadline Cloud。

1. 创建或打开用于Unreal Engine项目的文件夹。
2. 打开命令行并运行以下命令：
 - `git clone https://github.com/aws-deadline/deadline-cloud-for-unreal-engine`
 - `cd deadline-cloud-for-unreal/test_projects`

- `git lfs fetch -all`
- 3. 要下载的插件Unreal Engine，请打开Unreal Engine项目文件夹，然后启动 `deadline-cloud-forunreal/test_projects/pull_ue_plugin.bat`。

这会将插件文件放在 `C: UnrealDeadline CloudTest ///PluginsLocalProjects/ UnrealDeadlineCloudService`中。

- 4. 要下载提交者，请打开该 `UnrealDeadlineCloudService` 文件夹，然后运行 **`deadline-cloud-forunreal/ test_projects/Plugins/UnrealDeadlineCloudService/ install_unreal_submitter.bat`**。
- 5. 要从中启动提交者Unreal Engine，请完成以下步骤：
 - a. 选择“编辑”，>“项目设置”。
 - b. 在搜索栏中输入 **movie render pipeline**。
 - c. 调整以下影片渲染管道设置：
 - i. 在“默认远程执行器”中，输入**MoviePipelineDeadlineCloudRemote Executor**。
 - ii. 对于默认 Executor Job，输入 **MoviePipelineDeadlineCloudExecutorJob**
 - iii. 对于“默认 Job 设置类别”，选择加号，然后输入**DeadlineCloudRenderStepSetting**。

通过这些设置，您可以从中选择 Deadline Cloud 插件Unreal Engine。

使用农场

如果您已按照所有入门说明进行操作，则已设置好开始将作业从本地工作站提交到农场，然后监控这些作业和资源所需的一切。有关提交各种作业或监控的更多信息，请参阅下面的相关主题。

- [作业](#)
- [使用显示器](#)

使用截止日期云监视器

De AWS adline Cloud 监控器为您提供可视化计算作业的总体视图。您可以使用它来监控和管理作业、查看员工在车队中的活动、跟踪预算和使用情况，以及下载作业结果。

每个队列都有一个作业监视器，可向您显示作业、步骤和任务的状态。监视器提供了直接从显示器管理作业的方法。您可以更改优先级、取消任务和重新排队作业。

Deadline Cloud 监视器有一个显示任务摘要状态的表格，或者您可以选择一个作业来查看详细的任务日志，以帮助解决作业问题。

您可以使用 Deadline Cloud 监视器将结果下载到工作站上创建任务时指定的位置。

Deadline Cloud 监控器还可以帮助您监控使用情况和管理成本。有关更多信息，请参阅 [管理截止日期云的预算和使用情况](#)。

主题

- [共享 Deadline Cloud 监控](#)
- [打开截止日期云监视器](#)
- [在截止日期云中查看队列和舰队详情](#)
- [在 Deadline Cloud 中查看和管理作业、步骤和任务](#)
- [在截止日期云中查看职位详情](#)
- [在截止日期云中查看步骤](#)
- [在截止日期云中查看任务](#)
- [在截止日期云中查看日志](#)
- [在截止日期云中下载已完成的输出](#)

共享 Deadline Cloud 监控

设置 Deadline Cloud 服务时，默认情况下，您需要创建一个网址，用于为您的账户打开 Deadline Cloud 监视器。使用此 URL 在浏览器或桌面上打开显示器。与其他用户共享 URL，以便他们可以访问 Deadline Cloud 监视器。

在用户打开 Deadline Cloud 监视器之前，您必须向该用户授予访问权限。要授予访问权限，请将该用户添加到监视器的授权用户列表中，或者将其添加到有权访问监控器的群组中。有关更多信息，请参阅 [在截止日期云中管理用户](#)。

共享监视器 URL

1. 打开[截止日期云控制台](#)。
2. 从“开始”中，选择“前往截止日期云控制面板”。
3. 在导航窗格上，选择 Dashboard。
4. 在账户概述部分，选择账户详情。
5. 复制 URL，然后安全地将其发送给需要访问 Deadline Cloud 监视器的任何人。

打开截止日期云监视器

您可以通过以下任何一种方式打开 Deadline Cloud 监视器：

- 控制台-登录 AWS Management Console 并打开 Deadline Cloud 控制台。
- Web — 转到您在设置 Deadline Cloud 时创建的监视器 URL。
- 监控-使用桌面 Deadline Cloud 监视器

使用控制台时，必须能够 AWS 使用 AWS Identity and Access Management 身份登录，然后使用 AWS IAM Identity Center 凭据登录显示器。如果您只有 IAM Identity Center 证书，则必须使用监控 URL 或桌面应用程序登录。

打开 Deadline Cloud 监视器 (Web)

1. 使用浏览器打开您在设置 Deadline Cloud 时创建的监视器 URL。
2. 使用您的用户凭据登录。

打开 Deadline Cloud 监视器 (控制台)

1. 打开[截止日期云控制台](#)。
2. 在导航窗格中，选择农场。
3. 选择一个场，然后选择“管理作业”以打开 Deadline Cloud 监控页面。
4. 使用您的用户凭据登录。

打开 Deadline Cloud 监视器 (桌面)

1. 打开[截止日期云控制台](#)。

–或者–

从监视器 URL 打开 Deadline Cloud 监视器-Web。

2.
 - 在 Deadline Cloud 控制台上，执行以下操作：
 1. 在监视器中，选择“前往 Deadline Cloud 控制面板”，然后从左侧菜单中选择“下载”。
 2. 从 Deadline Cloud 监视器中，为您的桌面选择显示器版本。
 3. 选择下载。
 - 在 Deadline Cloud 监视器-网页版上，执行以下操作：
 - 从左侧菜单中选择“工作站设置”。如果工作站设置项不可见，请使用箭头打开左侧菜单。
 - 选择下载。
 - 从选择操作系统中，选择您的操作系统。
3. 下载 Deadline 云监视器-桌面。
4. 下载并安装显示器后，在计算机上将其打开。
 - 如果这是您第一次打开 Deadline Cloud 监视器，则必须提供监视器 URL 并创建配置文件名称。接下来，使用您的 Deadline Cloud 凭据登录显示器。
 - 创建配置文件后，您可以通过选择配置文件来打开显示器。您可能需要输入您的 Deadline Cloud 凭据。

在截止日期云中查看队列和舰队详情

您可以使用 Deadline Cloud 监视器来查看服务器场中队列和队列的配置。您还可以使用监视器查看队列中的作业或队列中的工作人员的列表。

您必须拥有查看队列和舰队详细信息的VIEWING权限。如果未显示详细信息，请联系您的管理员以获取正确的权限。

查看队列详情

1. [打开截止日期云监视器](#)。
2. 从服务器场列表中，选择包含您感兴趣的队列的服务器场。
3. 在队列列表中，选择一个队列以显示其详细信息。要比较两个或多个队列的配置，请选中多个复选框。
4. 要查看队列中的作业列表，请从队列列表或详细信息面板中选择队列名称。

如果监视器已打开，则可以从左侧导航窗格的“队列”列表中选择队列。

查看机群详细信息

1. [打开截止日期云监视器](#)。
2. 从农场列表中，选择包含您感兴趣的舰队的农场。
3. 在农场资源中，选择舰队。
4. 在舰队列表中，选择一个舰队以显示其详细信息。要比较两个或多个舰队的配置，请选中多个复选框。
5. 要查看车队中的工作人员名单，请从舰队列表或详细信息面板中选择车队名称。

如果监视器已打开，则可以从左侧导航窗格的舰队列表中选择舰队。

在 Deadline Cloud 中查看和管理作业、步骤和任务

选择队列时，Deadline Cloud 监视器的作业监视器部分会显示该队列中的作业、作业中的步骤以及每个步骤中的任务。选择作业、步骤或任务时，可以使用“操作”菜单来管理每个任务、步骤或任务。

要打开作业监视器，请按照步骤查看队列[在截止日期云中查看队列和舰队详情](#)，然后选择要使用的作业、步骤或任务。

对于作业、步骤和任务，您可以执行以下操作：

- 将状态更改为“已重新排队”、“成功”、“失败”或“已取消”。
- 从作业、步骤或任务中下载已处理的输出。
- 复制作业、步骤或任务的 ID。

对于所选作业，您可以：

- 将作业存档。
- 修改作业属性，例如更改优先级或查看步骤间的依赖关系。
- 使用作业的参数查看更多详细信息。

有关更多信息，请参阅 [在截止日期云中查看职位详情](#)。

对于每个步骤，您可以：

- 查看该步骤的依赖关系。必须先完成步骤的依赖关系，然后才能运行该步骤。

有关更多信息，请参阅 [在截止日期云中查看步骤](#)。

对于每项任务，您可以：

- 查看任务的日志。
- 查看任务参数。

有关更多信息，请参阅 [在截止日期云中查看任务](#)。

在截止日期云中查看职位详情

Deadline Cloud 监控器中的 Job 监控页面为您提供以下内容：

- 工作进度的总体视图。
- 构成任务的步骤和任务的视图。

从列表中选择一项作业以查看该作业的步骤列表，然后从步骤列表中选择一项步骤来查看该作业的任务。选择项目后，您可以使用该项目的“操作”菜单来查看详细信息。

查看职位详情

1. 按照步骤在中查看队列[在截止日期云中查看队列和舰队详情](#)。
2. 在导航窗格中，选择您提交作业的队列。
3. 使用以下方法之一选择作业：
 - a. 从“作业”列表中，选择一个作业以查看其详细信息。
 - b. 在搜索字段中，输入与该作业关联的任何文本，例如作业名称或创建该作业的用户。从显示的结果中，选择要查看的作业。

作业的详细信息包括作业中的步骤和每个步骤中的任务。您可以使用“操作”菜单执行以下操作：

- 更改作业的状态。
- 查看和修改作业的属性。您可以查看作业中各步骤之间的依赖关系，并更改作业的优先级。通常，优先级较高的作业会更快地完成。

- 查看提交作业时为作业设置的参数。
- 下载任务的输出。下载作业的输出时，它包含作业中的步骤和任务生成的所有输出。

在截止日期云中查看步骤

使用 De AWS adline Cloud 监视器查看处理任务中的步骤。在 Job 监视器中，Steps 列表显示构成所选作业的步骤列表。选择步骤后，任务列表会显示该步骤中的任务。

查看步骤

1. 按照中的[在截止日期云中查看职位详情](#)步骤查看作业列表。
2. 从作业列表中选择作业。
3. 从“步骤”列表选择一个步骤。

您可以使用“操作”菜单执行以下操作：

- 更改步骤的状态。
- 下载该步骤的输出。下载步骤的输出时，它包含该步骤中任务生成的所有输出。
- 查看步骤的依赖关系。依赖关系表显示了在选定步骤开始之前必须完成的步骤列表以及等待此步骤完成的步骤列表。

在截止日期云中查看任务

使用 De AWS adline Cloud 监视器查看处理任务中的任务。在 Job 监视器中，任务列表显示构成步骤列表中所选步骤的任务。

查看任务

1. 按照中的[在截止日期云中查看职位详情](#)步骤查看作业列表。
2. 从作业列表中选择作业。
3. 从“步骤”列表选择一个步骤。
4. 从“任务”列表中选择一项任务。

您可以使用“操作”菜单执行以下操作：

- 更改任务的状态。
- 查看任务日志。有关更多信息，请参阅 [在截止日期云中查看日志](#)。
- 查看创建任务时设置的参数。
- 下载任务的输出。下载任务的输出时，它仅包含所选任务生成的输出。

在截止日期云中查看日志

日志为您提供有关任务状态和处理的详细信息。在 De AWS adline Cloud 监控器中，您可以看到以下两种类型的日志：

- 会话日志详细说明了操作的时间表，包括：
 - 设置操作，例如同步附件和加载软件环境
 - 运行一项或一组任务
 - 关闭操作，例如关闭工作人员的环境

一个会话包括对至少一个任务的处理，并且可以包括多个任务。会话日志还显示有关亚马逊弹性计算云 (Amazon EC2) 实例类型、vCPU 和内存的信息。会话日志还包括指向会话中使用的工作器日志的链接。

- 工作日志提供了工作人员在其生命周期中处理的操作的时间表的详细信息。工作日志可以包含有关多个会话的信息。

您可以下载会话和工作器日志，以便可以离线查看它们。

查看会话日志

1. 按照中的 [在截止日期云中查看职位详情](#) 步骤查看作业列表。
2. 从作业列表中选择作业。
3. 从“步骤”列表选择一个步骤。
4. 从“任务”列表选择一项任务。
5. 从“操作”菜单中选择“查看日志”。

“时间表”部分显示了该任务的操作摘要。要查看会话中运行的更多任务以及会话的关闭操作，请选择查看所有任务的日志。

查看任务中的工作人员日志

1. 按照中的[在截止日期云中查看职位详情](#)步骤查看作业列表。
2. 从作业列表中选择作业。
3. 从“步骤”列表选择一个步骤。
4. 从“任务”列表中选择一项任务。
5. 从“操作”菜单中选择“查看日志”。
6. 选择会话信息。
7. 选择“查看工作人员日志”。

从舰队详细信息中查看工作人员日志

1. 按照中的步骤[在截止日期云中查看队列和舰队详情](#)查看舰队。
2. 从“工作人员”列表中选择工作人员 ID。
3. 从“操作”菜单中选择“查看工作人员日志”。

在截止日期云中下载已完成的输出

作业完成后，您可以使用 De AWS adline Cloud 监视器将结果下载到您的工作站。输出文件以您在创建作业时指定的名称和位置进行存储。

输出文件无限期存储。要降低存储成本，可以考虑为队列的 Amazon S3 存储桶创建 S3 生命周期配置。有关更多信息，请参阅 Amazon 简单存储服务用户指南中的管理存储[生命周期](#)。

下载作业、步骤或任务的已完成输出

1. 按照中的[在截止日期云中查看职位详情](#)步骤查看作业列表。
2. 选择要为其下载输出的作业、步骤或任务。
 - 如果选择一个作业，则可以下载该作业所有步骤中所有任务的所有输出。
 - 如果选择某个步骤，则可以下载该步骤中所有任务的所有输出。
 - 如果您选择一项任务，则可以下载该单个任务的输出。
3. 从“操作”菜单中选择“下载输出”。
4. 输出将下载到提交作业时设置的位置。

 Note

目前仅支持Windows和使用菜单下载输出Linux。如果您有Mac并选择了“下载输出”菜单项，则会出现一个窗口，显示可用于下载渲染输出的 AWS CLI 命令。

截止日期云农场

服务器场是用于管理任务的队列和执行任务的计算资源队列的容器。

主题

- [创建农场](#)
- [删除农场](#)
- [编辑农场](#)

创建农场

1. 从 [Deadline Cloud 控制台](#) 中，选择前往控制面板。
2. 在 Deadline Cloud 控制面板的“农场”部分，选择操作 → 创建农场。
 - 或者，在左侧面板中选择“农场和其他资源”，然后选择“创建农场”。
3. 为您的农场添加一个名称。
4. 在描述中，输入服务器场描述。清晰的描述可以帮助您快速确定农场的用途。
5. （可选）默认情况下，为了您的安全，您的数据使用 AWS 拥有和管理的密钥进行加密。您可以选择“自定义加密设置（高级）”以使用现有密钥或创建由您管理的新密钥。

如果您选择使用复选框自定义加密设置，请输入 AWS KMS ARN，或者 AWS KMS 通过选择创建新 KMS 密钥来创建新的 ARN。

6. （可选）选择 Add new tag，向服务器场添加一个或多个标签。
7. 选择“创建农场”。创建后，将显示您的农场。

删除农场

1. 从 Deadline Cloud 控制面板中，选择农场和其他资源。
2. 在服务器场列表中，选择要删除的一个或多个场，然后选择删除。

编辑农场

1. 从 Deadline Cloud 控制面板中，选择农场和其他资源。

2. 在服务器场列表中，选择要删除的一个或多个场，然后选择编辑。
3. 在显示的编辑窗口中，更改服务器场名称或描述，然后选择保存更改。

截止日期云队列

队列是一种管理和处理作业的场资源。

要处理队列，您应该已经设置了监视器和群组。

主题

- [创建队列](#)
- [创建队列环境](#)
- [删除队列](#)
- [编辑队列](#)
- [关联队列和舰队](#)

创建队列

1. 从 [Deadline Cloud 控制台](#) 仪表板中，选择要为其创建队列的场。
 - 或者，在左侧面板中选择“农场和其他资源”，然后选择要为其创建队列的场。
2. 在“队列”选项卡中，选择“创建队列”。
3. 输入队列的名称。
4. 在描述中，输入队列描述。描述可帮助您确定队列的用途。
5. 对于 Job 附件，您可以创建新的 Amazon S3 存储桶，也可以选择现有的 Amazon S3 存储桶。
 - a. 创建新的 Amazon S3 存储桶
 - i. 选择创建新的任务存储桶。
 - ii. 输入存储桶的名称。我们建议为存储桶命名 `deadlinecloud-job-attachments-[MONITORNAME]`。
 - iii. 输入根前缀以定义或更改队列的根位置。
 - b. 选择现有的 Amazon S3 存储桶
 - i. 选择“选择现有 S3 存储桶” > “浏览 S3”。
 - ii. 从可用存储桶列表中为您的队列选择 S3 存储桶。
6. （可选）要将您的队列与客户管理的队列关联，请选择启用与客户管理的队列的关联。

7. 如果您启用与客户管理的车队的关联，则必须完成以下步骤。

Important

我们强烈建议为运行方式功能指定用户和群组。如果你不这样做，就会降低你农场的安全状况，因为这样工作就可以做工作人员代理所能做的一切。有关潜在安全风险的更多信息，请参阅以[用户和群组身份运行作业](#)。

a. 对于以用户身份运行：

要为队列的作业提供凭据，请选择队列配置的用户。

或者，要选择不设置自己的凭证并以工作代理用户身份运行作业，请选择工作代理用户。

b. (可选) 在用户运行身份凭证中，输入用户名和组名以提供队列作业的凭据。

如果您使用的是Windows舰队，则必须创建一个包含用户运行身份密码的 AWS Secrets Manager 密钥。按照以下说明创建密钥。将 *jobuser* 替换为的名称。jobRunAsUser

i. 以管理员身份打开PowerShell或命令提示符。

ii. 创建用户。

```
net user jobuser /add
```

iii. 设置密码。

```
net user jobuser *
```

iv. 为用户创建本地配置文件和主目录。运行以下命令并在出现提示时输入用户的密码。

```
runas /profile /user:jobuser "cmd.exe /C"
```

8. 要求预算有助于管理队列成本。选择“不需要预算”或“需要预算”。

9. 您的队列需要获得代表您访问 Amazon S3 的权限。您可以创建新的服务角色或使用现有的服务角色。如果您没有现有的服务角色，请创建并使用新的服务角色。

a. 要使用现有的服务角色，请选择选择服务角色，然后从下拉列表中选择一个角色。

b. 要创建新的服务角色，请选择创建并使用新的服务角色，然后输入角色名称和描述。

10. (可选) 要为队列环境添加环境变量，请选择添加新环境变量，然后为添加的每个变量输入名称和值。
11. (可选) 选择 Add new tag ，向队列中添加一个或多个标签。
12. 要创建默认Conda队列环境，请将该复选框保持选中状态。要了解有关队列环境的更多信息，请参阅[创建队列环境](#)。如果您要为客户管理的队列创建队列，请清除该复选框。
13. 选择创建队列。

创建队列环境

队列环境是一组用于设置车队工作人员的环境变量和命令。您可以使用队列环境为队列中的作业提供软件应用程序、环境变量和其他资源。

创建队列时，您可以选择创建默认Conda队列环境。此环境允许服务管理队列访问合作伙伴 DCC 应用程序和渲染器的软件包。有关更多信息，请参阅[默认Conda队列环境](#)。

您可以使用控制台添加队列环境，也可以直接编辑 json 或 YAML 模板来添加队列环境。此过程介绍如何使用控制台创建环境。

1. 要向队列添加队列环境，请导航到队列并选择队列环境选项卡。
2. 选择“操作”，然后选择“使用表单创建新内容”。
3. 输入队列环境的名称和描述。
4. 选择“添加新环境变量”，然后为添加的每个变量输入名称和值。
5. (可选) 输入队列环境的优先级。优先级表示此队列环境将在工作器上运行的顺序。优先级较高的队列环境将首先运行。
6. 选择“创建队列环境”。

默认Conda队列环境

创建与服务管理队列关联的队列时，您可以选择添加默认队列环境，该环境支持[Conda](#)在虚拟环境中为任务下载和安装软件包。

Conda提供来自频道的套餐。频道是存储包裹的位置。Deadline Cloud 提供了一个频道deadline-cloud，用于托管支持合作伙伴 DCC 应用程序和渲染器的软件包。程序包是：

- 搅拌机

- blender=3.6
- blender-openjd
- 胡迪尼
 - houdini=19.5
 - houdini-openjd
- Maya
 - maya=2024
 - maya-mtoa=2024.5.3
 - maya-openjd
- Nuke
 - nuke=15
 - nuke-openjd

当您使用默认Conda环境将作业提交到队列时，环境会向该作业添加两个参数。这些参数指定在处理任务之前用于配置作业环境的Conda包和通道。这些参数是：

- CondaPackages— 以空格分隔的[包裹匹配规格](#)列表，例如blender=3.6或numpy>1.22。默认值为空以跳过创建虚拟环境。
- CondaChannels— 空格分隔的[Conda频道](#)列表deadline-cloud，例如conda-forge、或s3://*DOC-EXAMPLE-BUCKET*/conda/channel。默认为服务托管队列可用的渠道，提供合作伙伴 DCC 应用程序和渲染器。deadline-cloud

当您使用集成提交者将作业从 DCC 发送到 Deadline Cloud 时，提交者会根据 DCC 应用程序和提交者填充CondaPackages参数的值。例如，如果您使用的是Blender，则该CondaPackage参数将设置为blender=3.6.* blender-openjd=0.4.*。

删除队列

Warning

如果删除队列，则无法恢复队列中的作业。删除队列也会删除该队列中的作业。

1. 从 Deadline Cloud 控制面板中，选择农场和其他资源。

2. 在服务器场列表中，选择包含要删除的队列的场。
3. 选择队列，然后选择“删除”。
4. 在确认窗口中，选择删除。您的队列和队列中的所有任务都已删除。

编辑队列

1. 从 Deadline Cloud 控制面板中，选择农场和其他资源。
2. 在服务器场列表中，选择包含要编辑的队列的场。
3. 选择队列，然后选择“编辑”。
4. 您可以编辑名称、描述、预算要求、以用户身份运行选项和分配的服务角色。您也可以将现有队列与队列关联。
5. 选择保存更改。

关联队列和舰队

1. 选择要与舰队关联的队列。
2. 要选择要与队列关联的舰队，请选择关联舰队。
3. 选择“选择舰队”下拉列表。将显示可用舰队列表。
4. 从可用舰队列表中，选中要与队列关联的一个或多个舰队旁边的复选框。
5. 选择关联。舰队关联状态现在应为“已关联”。

管理截止日期云舰队

本节介绍如何管理 Deadline Cloud 的服务管理车队 (SMF) 和客户管理的车队 (CMF)。

您可以设置两种类型的 Deadline Cloud 舰队：

- 服务管理车队是员工队伍，其默认设置由该服务 Deadline Cloud 提供。这些默认设置旨在提高效率 and 成本效益。
- 客户管理的车队 (CMF) 是您管理的员工队伍。CMF 可以驻留在 AWS 基础架构中、内部部署或位于同地的数据中心中。CMF 提供对舰队的全面控制 and 责任。这包括车队中的工作人员的配置、运营、管理和退役。

主题

- [管理 Deadline Cloud 服务管理车队](#)
- [管理截止日期云客户管理的车队](#)

管理 Deadline Cloud 服务管理车队

服务管理车队是员工队伍，其默认设置由 Deadline Cloud 提供。这些默认设置旨在提高效率 and 成本效益。

1. 要创建服务管理队列 (SMF)，请导航到要在其中创建队列的服务器场。
2. 选择“舰队”选项卡。
3. 选择 Create fleet (创建机群)。
4. 输入您的舰队的名称。
5. 输入 Description (描述)。清晰的描述可以帮助您快速确定车队的用途。
6. 选择服务管理的舰队类型。
7. 为您的队列选择 Spot 或按需实例市场选项。竞价型实例是非预留容量，您可以以折扣价使用，但可能会被按需请求中断。按需实例按秒定价，但没有长期承诺，也不会中断。默认情况下，队列使用竞价型实例。
8. 可选设置最大实例数以扩展队列，以便容量可用于队列中的任务。我们建议您将最小实例数保持不变，0 以确保队列在没有任务排队时释放所有实例。
9. 要获得舰队的服务访问权限，请选择现有角色或创建新角色。服务角色为队列中的实例提供证书，授予它们处理任务的权限，并向监控器中的用户提供证书，以便他们可以读取日志信息。

10. 选择下一步。
11. 输入队列所需的最小 vCPU 和最大 vCPU。
12. 输入您的舰队所需的最小和最大内存。
13. 可选您可以选择允许或排除队列中的特定实例类型，以确保该队列仅使用这些实例类型。
14. 可选您可以指定将连接到该队列中工作人员的亚马逊 Elastic Block Store (Amazon EBS) GP3 卷的大小。有关更多信息，请参阅 [EBS 用户指南](#)。
15. 选择下一步。
16. 可选定义自定义工作人员要求，这些要求定义此队列的功能，这些功能可以与提交作业时指定的自定义主机要求相结合。例如，如果您计划将车队连接到自己的许可证服务器，则使用特定的许可证类型。
17. 选择下一步。
18. 可选要将您的队列与队列关联，请从下拉列表中选择一个队列。如果队列设置为默认Conda队列环境，则系统会自动为你的队列提供支持合作伙伴 DCC 应用程序和渲染器的软件包。有关所提供软件包的列表，请参阅[默认Conda队列环境](#)。
19. 选择下一步。
20. 可选要向队列添加标签，请选择 Add new tag，然后输入该标签的密钥和值。
21. 选择下一步。
22. 查看您的舰队设置，然后选择创建舰队。创建完成后，将显示您的舰队。

VFX Reference Platform兼容性

VFX Reference Platform是视觉特效行业的常见目标平台。要将运行 Amazon Linux 2023 的标准服务托管队列 Amazon EC2 实例与支持的软件一起使用VFX Reference Platform，在使用服务托管队列时应记住以下注意事项。

每年更新VFX Reference Platform一次。使用 AL2023 (包括 Deadline Cloud 服务管理的舰队) 的这些注意事项基于日历年 (CY) 2022 至 2024 年参考平台。有关更多信息，请参阅 [VFX Reference Platform](#)。

Note

如果您要为客户管理的队列创建自定义 Amazon Machine Image (AMI)，则可以在准备 Amazon EC2 实例时添加这些要求。

要在 AL2023 Amazon EC2 实例上使用 VFX Reference Platform 支持的软件，请考虑以下事项：

- AL2023 中安装的 glibc 版本兼容运行时使用，但不适用于与 VFX Reference Platform CY2024 或更早版本兼容的构建软件。
- Python 3.9 和 3.11 随服务管理队列一起提供，使其与 VFX Reference Platform CY2022 和 CY2024 兼容。服务管理队列中未提供 Python 3.7 和 3.10。需要它们的软件必须在队列或作业环境中提供 Python 安装。
- 服务管理队列中提供的某些 Boost 库组件版本为 1.75，与不兼容。VFX Reference Platform 如果您的应用程序使用 Boost，则必须提供自己的库版本以实现兼容性。
- 英特尔 TBB 更新 3 在服务托管队列中提供。这与 VFX Reference Platform CY2022、CY2023 和 CY2024 兼容。
- 服务管理的队列 VFX Reference Platform 不提供具有指定版本的其他库。您必须向库提供服务托管队列上使用的任何应用程序。有关库的列表，请参阅[参考平台](#)。

管理截止日期云客户管理的车队

本节介绍如何管理 Deadline Cloud 的客户管理车队 (CMF)。

CMF 是您管理的员工队伍。CMF 可以驻留在 AWS 基础架构中、内部部署或位于同地的数据中心中。CMF 提供对舰队的全面控制 and 责任。这包括车队中的工作人员的配置、运营、管理和退役。

主题

- [创建由客户管理的车队](#)
- [工作主机设置和配置](#)
- [管理对 Windows 作业用户密钥的访问权限](#)
- [安装和配置作业所需的软件](#)
- [配置 AWS 凭证](#)
- [创建 Amazon Machine Image](#)
- [使用 Amazon EC2 Auto Scaling 组创建队列基础设施](#)
- [Connect 将客户管理的车队连接到许可证端点](#)

创建由客户管理的车队

要创建客户管理的队列 (CMF)，请完成以下步骤。

Deadline Cloud console

使用 Deadline Cloud 控制台创建客户管理的舰队

1. 打开截止日期云[控制台](#)。
2. 选择“农场”。将显示可用场列表。
3. 选择您要在其中工作的农场的名称。
4. 选择“舰队”选项卡。
5. 选择 Create fleet (创建机群)。
6. 输入您的舰队的名称。
7. (可选) 为您的舰队输入描述。
8. 为“舰队类型”选择“客户管理”。
9. 选择 Auto Scaling 类型。有关更多信息，请参阅[用于 EventBridge 处理 Auto Scaling 事件](#)。
 - 不扩展：你正在创建本地队列，想选择退出 Deadline Cloud Auto Scaling。
 - 扩展建议：您正在创建亚马逊弹性计算云 (Amazon EC2) 队列。
10. 选择您的车队的服务访问权限。
 - a. 我们建议为每个队列使用“创建并使用新的服务角色”选项，以实现更精细的权限控制。已默认选定此选项。
 - b. 您也可以通过选择“选择服务角色”来使用现有的服务角色。
11. 查看您的选择，然后选择“下一步”。
12. 为您的舰队选择操作系统。车队的所有工作人员都必须使用通用的操作系统。
13. 选择主机 CPU 架构。
14. 为该队列中的工作主机选择以下硬件要求。
 - a. 选择最低、最大 vCPU 和内存硬件要求，以满足队列的工作负载需求。
 - b. (可选) 选择 GPU 要求，然后输入最小和最大 GPU。
15. 查看您的选择，然后选择“下一步”。
16. (可选) 定义自定义工作人员要求。
17. 使用下拉列表选择一个或多个要与队列关联的队列。

Note

我们建议仅将队列与处于相同信任边界的队列相关联。这样可以确保在同一工作器上运行作业之间保持牢固的安全边界。

18. 查看队列关联，然后选择“下一步”。
19. (可选) 对于默认 Conda 队列环境，我们将为您的队列创建一个环境，该环境将安装任务请求的 Conda 软件包。

Note

Conda 队列环境用于安装作业请求的 Conda 软件包。通常，您应该取消选中与 CMF 关联的队列上的 Conda 队列环境，因为默认情况下 CMF 不会安装所需的 Conda 命令。

20. (可选) 向 CMF 添加标签。有关更多信息，请参阅为[AWS 资源添加标签](#)。
21. 查看您的机队配置并进行任何更改。
22. 选择 Create fleet (创建机群)。
23. 选择“舰队”选项卡，然后记下舰队 ID。

AWS CLI

使用创建由 AWS CLI 客户管理的车队

1. 打开 AWS CLI。
2. 编辑 `fleet-trust-policy.json`。
 - a. 添加以下 IAM 政策，将 `##### AWS ## ID # D` eadline Cloud Farm ID。

```
{
  "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. 保存您的更改。

3. 编辑 create-cmf-fleet.json。

a. 添加以下 IAM 策略。

```

{
  "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. 保存您的更改。

4. 添加 IAM 角色供队伍中的工作人员使用。

```

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. 编辑 create-fleet-request.json。

a. 添加以下 IAM 策略，用您的 CMF 值替换斜体文本。

Note

你可以在中找到 *ROLE_ARN#* create-cmf-fleet.json
对于 *OS_FAMILY*，必须选择或中的一个。linux macos 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. 保存您的更改。

6. 创建您的舰队。

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

工作主机设置和配置

工作主机是指运行 Deadline Cloud 工作线程的主机。本节介绍如何设置工作主机并根据您的特定需求对其进行配置。每台工作器主机都运行一个名为工作器代理的程序。工作人员代理负责：

- 管理工作员的生命周期。
- 同步分配的工作、其进度和结果。
- 监控正在运行的工作。
- 将日志转发到已配置的目的地。

我们建议您使用提供的 Deadline Cloud 工作者代理。worker 代理是开源的，我们鼓励您提出功能请求，但您也可以根据自己的需求进行开发和定制。

要完成以下各节中的任务，您需要具备以下条件：

Linux

- Linux基于亚马逊弹性计算云 (Amazon EC2) 的 Amazon EC2 实例。我们推荐亚马逊 Linux 2023。
- sudo特权。
- Python 3.9 或更高版本。

Windows

- Windows基于亚马逊弹性计算云 (Amazon EC2) 的 Amazon EC2 实例。我们推荐Windows Server 2022。
- 管理员对工作主机的访问权限
- 已为所有用户安装了 Python 3.9 或更高版本

创建和配置 Python 虚拟环境

Linux如果您已经安装了 Python 3.9 或更高版本并将其放置在您的 Python 虚拟环境中，则可以在上创建 Python 虚拟环境PATH。

创建和激活 Python 虚拟环境

1. 打开 AWS CLI.
2. 创建并激活 Python 虚拟环境。

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

安装 Deadline Cloud

设置 Python 并在上创建虚拟环境后，安装 Deadline Cloud 工作器代理 Python 软件包。Linux

安装工作器代理 Python 软件包

1. 打开终端。
 - a. 开启Linux，以root用户身份打开终端（或使用sudo/su）
 - b. 打开Windows，打开管理员命令提示符或 PowerShell终端。
2. 从 PyPI 下载并安装 Deadline Cloud 工作器代理包：

Note

启用Windows，必须将代理文件安装到 Python 的全局站点包目录中。目前不支持 Python 虚拟环境。

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

配置 Deadline 云端工作器代理

您可以通过三种方式配置 Deadline Cloud 工作器代理设置。我们建议您使用通过设置的操作系
统install-deadline-worker。

命令行参数 — 当从命令行运行 Deadline Cloud 工作器代理时，您可以指定参数。某些配置设置无法通
过命令行参数获得。要查看所有可用的命令行参数，请输入deadline-worker-agent --help以查
看所有可用的命令行参数。

环境变量 — 您可以通过设置以开头的环境变量来配置 Deadline Cloud 工作器代
理DEADLINE_WORKER_。例如，您可以使用export DEADLINE_WORKER_VERBOSE=true将
工作器代理的输出设置为详细。有关更多示例和信息，请参阅 /etc/amazon/deadline/
worker.toml.example on Linux 或 C:\ProgramData\Amazon\Deadline\Config
\worker.toml.example on Windows。

配置文件-安装工作器代理时，它会创建一个位于 /etc/amazon/deadline/worker.toml
on Linux 或 C:\ProgramData\Amazon\Deadline\Config\worker.toml on 的配置文件
Windows。工作器代理在启动时加载此配置文件。您可以使用示例配置文件（ /etc/amazon/

deadline/worker.toml.example 开启 Linux 或开 C:\ProgramData\Amazon\Deadline\Config\worker.toml.example 启 Windows) 根据您的特定需求定制默认工作器代理配置文件。

最后，我们建议您为工作器代理启用 auto shutdown。这允许工作人员队列在需要时扩大规模，并在渲染作业完成时关闭。自动缩放有助于确保您仅在需要时使用资源。

启用 auto 关机

作为 **root** 用户：

- 安装带有参数的工作器代理 **--allow-shutdown**。

Linux

输入：

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

Windows

输入：

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

创建作业用户和群组

本节介绍代理用户与队列中 jobRunAsUser 定义的用户之间所需的用户和群组关系。

Deadline Cloud 工作服务器代理应以主机上特定代理的专用用户身份运行。您应配置 Deadline Cloud 队列的 jobRunAsUser 属性，以便工作人员以特定的操作系统用户和组的身份运行队列作业。这意味着您可以控制作业拥有的共享文件系统权限。它还提供了作业和工作代理用户之间的重要安全边界。

Linux 工作用户和群组

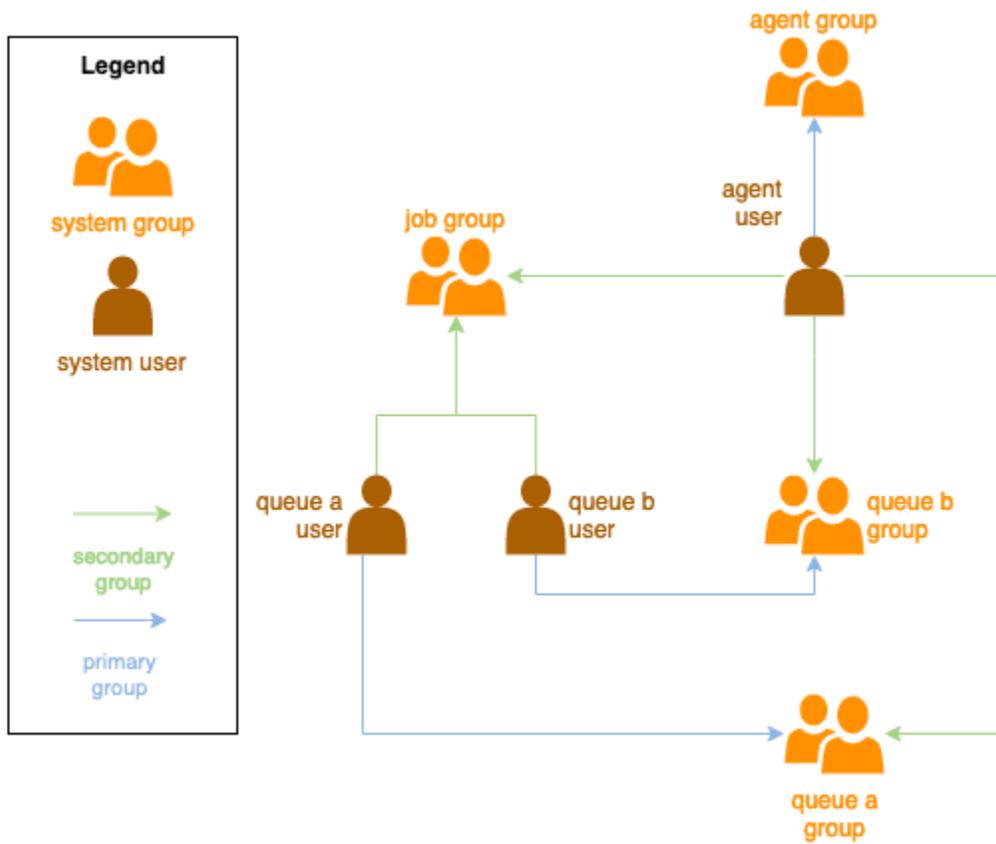
要设置您的代理用户和`jobRunAsUser`，请确保满足以下要求：

- 每个组都有一个组`jobRunAsUser`，它是其对应的主组`jobRunAsUser`。
- 代理用户属于该工作人员获得`jobRunAsUser`工作的队列的主组。为了获得最佳安全实践，我们建议将其作为代理用户的次要群组。此共享组允许工作器代理在作业运行时为其提供文件。
- A `jobRunAsUser` 不属于代理用户的主组。有关安全最佳实践，请执行以下操作：
 - 工作器代理写入的敏感文件归该代理的主组所有。
 - 如果`jobRunAsUser`属于该组，并且提交到工作器上运行的队列的作业可以访问工作器代理写入的文件。
- 默认 AWS 区域应与工作人员所属农场的区域相匹配。有关更多信息，请参阅[配置和凭据文件设置](#)。

这应该适用于：

- 代理用户
- 工作人员上的所有队列`jobRunAsUser`账户
- 代理用户可以`sudo`以`jobRunAsUser`

下图说明了代理用户与队列关联的`jobRunAsUser`用户和群组之间的关系。



Windows 用户

要使用Windows用户作为jobRunAsUser，它必须满足以下要求：

- 所有队列jobRunAsUser用户都必须存在。
- 他们的密码必须与队列JobRunAsUser字段中指定的密钥值相匹配。有关说明，请参阅中的步骤 7 [创建队列](#)。
- 代理用户必须能够以这些用户的身份登录。

管理对 Windows 作业用户密钥的访问权限

使用 Windows 配置队列时jobRunAsUser，必须指定 Secrets Manager AWS 密钥。这个秘密的值应该是 JSON 编码的对象，其形式为：

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

要使工作人员按照队列的配置运行作业 `jobRunAsUser`，队列的 IAM 角色必须具有获取密钥值的权限。如果使用客户管理的 KMS 密钥对密钥进行加密，则队列的 IAM 角色还必须具有使用 KMS 密钥进行解密的权限。

强烈建议对这些机密遵循最低权限原则。这意味着获取队列 `jobRunAsUser` windows → 的秘密值的访问权限 `passwordArn` 应为：

- 在舰队和队列之间创建队列队列关联时授予舰队角色
- 删除队列和队列之间的队列队列关联后，已从舰队角色中撤销

此外，当不再使用包含 `jobRunAsUser` 密码的 `Secrets Manager` 密钥时，应将其删除。

授予对密码密钥的访问权限

当队列和队列关联时，`jobRunAsUser` `line Cloud` 舰队需要访问存储在队列密码密钥中的密码。我们建议使用 `Secrets Manager` 资源策略来授予对舰队角色的访问权限。如果您严格遵守此准则，则可以更轻松地确定哪些舰队角色可以访问该机密。

授予访问密钥的权限

1. 打开 `AWS 密钥管理器控制台` 查看密钥。
2. 在“资源权限”部分，添加以下形式的政策声明：

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

撤消对密码密钥的访问权限

当队列不再需要访问队列时，请移除对队列密码密钥的访问权限 `jobRunAsUser`。我们建议使用 S AWS secrets Manager 资源策略来授予对舰队角色的访问权限。如果您严格遵守此准则，则可以更轻松确定哪些舰队角色可以访问该机密。

撤消对密钥的访问权限

1. 打开 AWS 密钥管理器控制台查看密钥。
2. 在资源权限部分中，删除以下形式的政策声明：

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

安装和配置作业所需的软件

设置 Deadline Cloud 工作器代理后，您可以为工作器主机准备运行作业所需的任何软件。

当您向关联的队列提交作业时 `jobRunAsUser`，该作业将以该用户的身份运行。所有命令都必须在该用户 PATH 中可用。

在 Linux 上，您可以在以下任一选项中 PATH 为用户指定：

- 他们的 `~/.bashrc` 或 `~/.bash_profile`
- 系统配置文件，例如 `/etc/profile.d/*` 和 `/etc/profile`
- shell 启动脚本：`/etc/bashrc`。

在 Windows 上，您可以在以下任一选项中 PATH 为用户指定：

- 他们特定于用户的环境变量
- 系统范围的环境变量

安装数字内容创作工具适配器

Deadline Cloud 为数字内容创作 (DCC) 应用程序提供第一方集成支持。要在客户管理的车队上使用这些集成，必须安装 DCC 软件和适配器。

在客户管理的车队上安装 DCC 适配器

1. 打开 a 终端。
 - a. 在 Linux 上，以 root 用户身份打开终端（或使用 sudo/su）
 - b. 在 Windows 上，打开管理员命令提示符或 PowerShell 终端。
2. 安装 Deadline Cloud 适配器包。

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

配置 AWS 凭证

本节介绍如何配置 AWS 凭证。

工作人员生命周期的这个初始阶段是自力更生。在此阶段，工作人员代理软件会在您的车队中创建一个工作人员，并从您的车队的角色中获取 AWS 凭证以进行进一步的操作。

AWS credentials for Amazon EC2

为 Amazon EC2 配置 AWS 证书

1. 通过 <https://console.aws.amazon.com/iam/> 打开 IAM 控制台。
2. 在导航窗格中选择角色，然后选择创建角色。
3. 选择 AWS 服务。
4. 选择 EC2 作为服务或用例，然后选择下一步。
5. 附加 AWSDeadlineCloud-WorkerHost AWS 托管策略。

On-premise AWS credentials

配置 AWS 本地凭证

1. 通过 <https://console.aws.amazon.com/iam/> 打开 IAM 控制台。
2. 在导航窗格中选择角色，然后选择创建角色。
3. 选择 AWS 账户，然后选择“下一步”。
4. 附加AWSDeadlineCloud-WorkerHost AWS 托管策略。
5. 为 AWS IAM 用户生成 IAM 访问权限和密钥：
 - a. 有关 IAM 角色无处不在，请参阅任何[地方的 IAM 角色](#)。
 - b. 有关在主机上设置证书的最安全方法，请参阅 [Anywhere 从 AWS Identity and Access Management 角色获取临时安全证书](#)。
 - c. 您也可以使用 CLI 作为替代身份验证，有关更多信息，请参阅使用 [IAM 用户证书进行身份验证](#)。
6. 将这些密钥存储在工作主机文件系统上的代理用户 AWS 凭证文件中。
 - a. 在 Linux 上，它位于 `~/.aws/credentials`
 - b. 在 Windows 上，它位于 `%USERPROFILE%\aws\credentials`

Note

只有安装工作器代理的操作系统用户名 (deadline-worker-agent) 才能访问凭证。

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

7. 更改deadline-worker-agent所有者和权限。

Note

如果您在安装工作器代理时更改了操作系统用户 (deadline-worker-agent) 名称，请改用该名称。

创建 Amazon Machine Image

要创建在亚马逊弹性计算云 Amazon Machine Image (Amazon EC2AMI) Elastic Compute Cloud 客户托管队列 (CMF) 中使用的 ()，请完成本节中的任务。在继续操作之前，您必须创建一个 Amazon EC2 实例。有关更多信息，请参阅《适用于 Linux [实例的 Amazon EC2 用户指南](#)》中的[启动您的实例](#)。

Important

AMI创建会创建 Amazon EC2 实例所连接卷的快照。实例上安装的所有软件都将保留，因此当您从中启动实例时，这些实例会被重复使用。AMI我们建议您采用修补策略，并在向您的车队申请之前，定期AMI使用更新的软件更新任何新内容。

准备 Amazon EC2 实例

在构建之前AMI，必须删除工作器状态。在工作器代理启动之间，工作器状态保持不变。如果此状态持续到上AMI，则从该状态启动的所有实例都将共享相同的状态。

我们还建议您删除所有现有的日志文件。在您准备 AMI 时，日志文件可以保留在 Amazon EC2 实例上。在诊断使用 AMI 的工作队列中可能存在的问题时，删除这些文件可以最大限度地减少混乱。

您还应该启用工作代理系统服务，以便在 Amazon EC2 启动时启动 Deadline Cloud 工作器代理。

最后，我们建议您启用工作器代理自动关机。这允许工作人员队列在需要时扩大规模，并在渲染作业完成时关闭。这种 auto Scaling 有助于确保您仅根据需要使用资源。

准备 Amazon EC2 实例

1. 打开 Amazon EC2 控制台。
2. 启动 Amazon EC2 实例。有关更多信息，请参阅[启动您的实例](#)。
3. 将主机设置为连接到您的身份提供商 (IdP)，然后挂载它需要的任何共享文件系统。
4. 然后 [安装 Deadline Cloud配置工作器代理](#)，按照教程进行和[创建作业用户和群组](#)。
5. 如果您正在准备一款AMI基于 Amazon Linux 2023 的版本来运行与视觉特效参考平台兼容的软件，则需要更新多项要求。有关信息，请参阅 [VFX Reference Platform兼容性](#)。
6. 打开终端。
 - a. 在 Linux 上，以root用户身份打开终端 (或使用sudo/su)
 - b. 在 Windows 上，打开管理员命令提示符或 PowerShell终端。

7. 确保工作器服务未运行且配置为在启动时启动：

- a. 在 Linux 上，运行

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

- b. 在 Windows 上，运行

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

8. 删除工作人员状态。

- a. 在 Linux 上，运行

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

- b. 在 Windows 上，运行

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

9. 删除日志文件。

- a. 在 Linux 上，运行

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

- b. 在 Windows 上，运行

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

10. 在 Windows 上，建议运行“开始”菜单中的 Amazon EC2Launch 设置应用程序，以完成实例的最终主机准备和关闭。

Note

你必须选择“不使用 Sysprep 关闭”，切勿选择“使用 Sysprep 关闭”。使用 Sysprep 关闭将导致所有本地用户都无法使用。有关更多信息，请参阅 [《Windows 实例用户指南》的“创建自定义 AMI”主题的“开始之前”部分](#)。

构建 AMI

要构建 AMI

1. 打开 Amazon EC2 控制台。
2. 在导航窗格中选择实例，然后选择您的实例。
3. 选择实例状态，然后选择停止实例。
4. 实例停止后，选择操作。
5. 选择图像和模板，然后选择创建图像。
6. 输入图像名称。
7. (可选) 输入图片的描述。
8. 选择创建映像。

使用 Amazon EC2 Auto Scaling 组创建队列基础设施

本节介绍如何创建 Amazon EC2 Auto Scaling 队列。

使用下面的 AWS CloudFormation YAML 模板创建一个 Amazon EC2 Auto Scaling (Auto Scaling) 组、一个包含两个子网、一个实例配置文件和一个实例访问角色的亚马逊虚拟私有云 (Amazon VPC)。这些是在子网中使用 Auto Scaling 启动实例所必需的。

您应该查看并更新实例类型列表以满足您的渲染需求。

创建 Amazon EC2 Auto Scaling 队列

1. 打开 AWS CloudFormation 控制台，[网址为 https://console.aws.amazon.com/cloudformation](https://console.aws.amazon.com/cloudformation)。
2. 使用参数 Farm ID、Fleet ID、和创建 CloudFormation 模板 AMI ID。

```
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 workers in 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
```

3. 创建 IAM 角色后，您需要确认以下内容：

- 附加到您的工作人员的 Amazon EC2 实例的 IAM 角色的证书可用于在该工作程序上运行的所有进程，包括作业。工作人员应拥有最少的操作权限：`deadline:CreateWorker`和`deadline:AssumeFleetRoleForWorker`。
- 工作器代理获取队列角色的凭证，并对其进行配置以供运行作业使用。Amazon EC2 实例配置文件角色不应包含您的任务所需的权限。

使用 Deadline Cloud 规模推荐功能自动扩展您的 Amazon EC2 机群

Deadline Cloud 利用亚马逊 EC2 Auto Scaling (Auto Scaling) 组自动扩展亚马逊 EC2 客户管理的队列 (CMF)。您需要配置舰队模式并在您的账户中部署所需的基础架构，才能实现队列自动扩展。您部署的基础架构适用于所有舰队，因此您只需要设置一次即可。

基本工作流程是：您将舰队模式配置为 `auto scale`，然后，每当建议的舰队规模 EventBridge 发生变化时，Deadline Cloud 就会为该舰队发送一个事件（一个事件包含舰队 ID、建议的舰队规模和其他元数据）。您将有一 EventBridge 条规则来筛选相关事件，并使用 Lambda 来使用它们。Lambda 将与 Amazon EC2 Auto Scaling 集成 `AutoScalingGroup`，以自动扩展亚马逊 EC2 队列。

将舰队模式设置为 **EVENT_BASED_AUTO_SCALING**

将您的舰队模式配置为 `EVENT_BASED_AUTO_SCALING`。您可以使用控制台来执行此操作，也可以使用直接调 AWS CLI 用 `CreateFleet` 或 `UpdateFleet` API。配置模式后，每当建议的队列规模发生变化时，Deadline Cloud 就会开始发送 EventBridge 事件。

- `UpdateFleet` 命令示例：

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

- `CreateFleet` 命令示例：

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

以下是在上述 CLI 命令中 `configuration.json` 使用的示例 (`--configuration file://configuration.json`)。

- 要在队列上启用 Auto Scaling，应将模式设置为 `EVENT_BASED_AUTO_SCALING`。
- `workerCapabilities` 这些是您创建 CMF 时分配给它的默认值。如果您需要增加 CMF 的可用资源，可以更改这些值。

配置队列模式后，Deadline Cloud 开始为该队列发出舰队规模建议事件。

```
{
  "customerManaged": {
    "mode": "EVENT_BASED_AUTO_SCALING",
    "workerCapabilities": {
      "vCpuCount": {
        "min": 1,
        "max": 4
      },
      "memoryMiB": {
        "min": 1024,
        "max": 4096
      },
      "osFamily": "linux",
      "cpuArchitectureType": "x86_64",
    }
  }
}
```

使用 AWS CloudFormation 模板部署 Auto Scaling 堆栈

您可以设置 EventBridge 规则来筛选事件，设置用于使用事件和控制 Auto Scaling 的 Lambda，以及用于存储未处理事件的 SQS 队列。使用以下 AWS CloudFormation 模板部署堆栈中的所有内容。成功部署资源后，您可以提交任务，队列将自动扩展。

```
Resources:
  AutoScalingLambda:
    Type: 'AWS::Lambda::Function'
    Properties:
      Code:
        ZipFile: |-
          """
          This lambda is configured to handle "Fleet Size Recommendation Change"
```



```
        return {
            'statusCode': 200,
            'body': json.dumps(f'Successfully set desired_capacity for {asg_name}
to {desired_capacity}')
        }
    Handler: index.lambda_handler
    Role: !GetAtt
        - AutoScalingLambdaServiceRole
        - Arn
    Runtime: python3.11
    DependsOn:
        - AutoScalingLambdaServiceRoleDefaultPolicy
        - AutoScalingLambdaServiceRole
    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 将客户管理的车队连接到许可证端点

De AWS adline Cloud (Deadline Cloud) 基于使用量的许可证服务器为选定的第三方产品提供按需许可证。这使您能够按使用量付费。您只能在使用的时间内进行更改。

只要 Deadline Cloud 工作人员可以与许可证服务器通信，基于 Deadline Cloud 使用情况的许可证服务器就可以用于任何类型的舰队。这是在服务管理的车队中自动设置的。只有客户管理的车队才需要此设置。

要创建许可证服务器，您需要满足以下条件：

- 服务器场的 VPC 的安全组，允许第三方许可证的流量。
- 一个附带策略的 AWS Identity and Access Management (IAM) 角色，该策略允许访问 Deadline Cloud 许可证端点操作。

主题

- [步骤 1：创建安全组](#)
- [步骤 2：设置许可证端点](#)
- [步骤 3：将渲染应用程序连接到端点](#)

步骤 1：创建安全组

使用亚马逊 VPC 控制台 (<https://console.aws.amazon.com/vpc/>) 为您的服务器场的 VPC 创建安全组。将安全组配置为允许以下入站规则：

- Autodesk Maya 和 Arnold — 2701-2702、TCP、IPv4

- Autodesk 3ds Max — 2704、TCP、IPv4
- Foundry Nuke — 6101、TCP、IPv4
- SideFX Houdini、Mantra 和 Karma — 1715-1717、TCP、IPv4

每条入站规则的来源都是舰队的工作人员安全组。

有关创建安全组的更多信息，请参阅 Amazon Virtual Private Cloud 用户指南中的[创建安全组](#)。

步骤 2：设置许可证端点

许可证端点为第三方产品提供对许可证服务器的访问权限。许可证请求将发送到许可证端点。端点会将它们路由到相应的许可证服务器。许可证服务器跟踪使用限制和授权。您创建的每个许可证端点都需要付费。有关更多信息，请参阅 [Amazon VPC pricing](#)。

您可以从中创建 AWS Command Line Interface 具有相应权限的许可证终端节点。有关创建许可证端点所需的策略，请参阅[允许创建许可证端点的策略](#)。

您可以使用以下 AWS Command Line Interface 命令使用 AWS CloudShell (<https://console.aws.amazon.com/cloudshell/>) 或任何其他 AWS CLI 环境来配置许可证端点。

1. 创建许可证端点。将安全组 ID、子网 ID 和 VPC ID 替换为您之前创建的值。如果您使用多个子网，请用空格将它们隔开。

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

2. 使用以下命令确认终端节点已成功创建。记住 VPC 终端节点的 DNS 名称。

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

3. 查看可用的计量产品列表：

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

4. 使用以下命令将计量产品添加到许可证端点。

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

```
--license-endpoint-id LICENSE_ENDPOINT_ID \  
--product-id PRODUCT_ID
```

您可以使用以下 `remove-metered-product` 命令从许可证端点中删除产品：

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

您可以使用以下 `delete-license-endpoint` 命令删除许可证端点：

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

步骤 3：将渲染应用程序连接到端点

设置许可证端点后，应用程序使用该端点的方法与使用第三方许可证服务器的方式相同。通常，您可以通过将环境变量或其他系统设置（例如 Microsoft Windows 注册表项）设置为许可证服务器的端口和地址来配置应用程序的许可证服务器。

要获取许可证端点 DNS 名称，请使用以下 AWS CLI 命令。

```
aws deadline get-license-endpoint
```

或者，您可以使用亚马逊 VPC 控制台 (<https://console.aws.amazon.com/vpc/>) 来识别 Deadline Cloud API 在上一步中创建的 VPC 终端节点。

配置示例

Example — Autodesk Maya 和 Arnold

将环境变量设置 `ADSKFLEX_LICENSE_FILE` 为：

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

Note

对于 Windows 工作人员，使用分号 (;) 代替冒号 (:) 来分隔端点。

Example — Autodesk 3ds Max

将环境变量设置ADSKFLEX_LICENSE_FILE为：

```
2704@VPC_Endpoint_DNS_Name
```

Example — 铸造核弹

将环境变量设置foundry_LICENSE为6101@VPC_Endpoint_DNS_Name要测试许可是否正常运行，可以在终端中运行Nuke：

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

Example — SideFX Houdini、Mantra 和 Karma

运行以下命令：

```
/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;"
```

要测试许可是否正常运行，你可以通过以下命令渲染 Houdini 场景：

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

在截止日期云中管理用户

AWS Deadline Cloud 用于 AWS IAM Identity Center 管理用户和群组。IAM Identity Center 是一项基于云的单点登录服务，可以与您的企业单点登录 (SSO) 提供商集成。通过集成，用户可以使用其公司帐户登录。

Deadline Cloud 默认启用 IAM 身份中心，并且需要设置和使用 Deadline Cloud。有关更多信息，请参阅[管理您的身份源](#)。

您的组织所有者负责管理有权访问您 AWS Organizations 的 Deadline Cloud 监控器的用户和群组。您可以使用 IAM 身份中心或 Deadline Cloud 控制台创建和管理这些用户和群组。有关更多信息，请参阅[什么是 AWS Organizations](#)。

您可以使用 Deadline Cloud 控制台创建和移除可以使用监控器管理农场、队列和队列的用户和群组。当您用户添加到 Deadline Cloud 时，他们必须使用 IAM Identity Center 重置密码，然后才能获得访问权限。

主题

- [管理监视器的用户和群组](#)
- [管理农场、队列和队列的用户和群组](#)

管理监视器的用户和群组

Organizations 所有者可以使用 Deadline Cloud 控制台来管理有权访问 Deadline Cloud 监控器的用户和群组。您可以从现有的 IAM Identity Center 用户和群组中进行选择，也可以从控制台添加新的用户和群组。

1. 登录 AWS Management Console 并打开 [Deadline Cloud 控制台](#)。在主页的“入门”部分，选择“设置 Deadline Cloud”或“前往控制面板”。
2. 在左侧导航窗格中，选择用户管理。默认情况下，“群组”选项卡处于选中状态。

根据要采取的操作，选择“群组”选项卡或“用户”选项卡。

Monitor groups

创建组

1. 选择创建组。

2. 输入群组名称。该名称在您的 IAM Identity Center 组织中的群组中必须是唯一的。

移除群组

1. 选择要删除的群组。
2. 选择移除。
3. 在确认对话框中，选择移除群组。

Note

您正在从 IAM 身份中心移除该群组。群组成员无法再登录 Deadline Cloud 或访问农场资源。

Monitor users

添加用户

1. 选择用户选项卡。
2. 选择添加用户。
3. 输入新用户的姓名、电子邮件地址和用户名。
4. 如果需要，可以选择一个或多个 IAM Identity Center 群组来添加新用户。
5. 选择“发送邀请”，向新用户发送一封包含加入您的 IAM Identity Center 组织的说明的电子邮件。

删除用户

1. 选择要从显示器中移除的用户。
2. 选择移除。
3. 在确认对话框中，选择移除用户。

Note

您正在从 IAM 身份中心移除该用户。用户无法再登录 Deadline Cloud 监控器或访问服务器场资源。

管理农场、队列和队列的用户和群组

1. 如果您尚未登录，请登录 AWS Management Console 并打开 [Deadline Cloud 控制台](#)。
2. 在左侧导航窗格中，选择“农场和其他资源”。
3. 选择要管理的农场。选择服务器场名称以打开详细信息页面。您可以使用搜索栏搜索农场。
4. 要管理队列或队列，请选择队列或队列选项卡，然后选择要管理的队列或队列。
5. 选择“访问管理”选项卡。默认情况下，“群组”选项卡处于选中状态。要管理用户，请将切换开关移至“用户”。

根据要采取的操作，选择“群组”选项卡或“用户”选项卡。

有关访问级别的定义，请参阅[权限](#)。

Groups

添加组

1. 选择“群组”开关。
2. 选择 Add Group (添加组)。
3. 从下拉列表中选择要添加的群组。
4. 对于群组访问级别，请选择以下选项之一：
 - 查看者
 - 贡献者
 - 经理
 - 所有者
5. 选择添加。

移除群组

1. 选择要移除的群组。
2. 选择移除。
3. 在确认对话框中，选择 Remove。

Users

添加用户

1. 要添加用户，请选择添加用户。
2. 从下拉列表中，选择要添加到您的服务器场的用户。
3. 对于用户访问级别，请选择以下选项之一：
 - 查看者
 - 贡献者
 - 经理
 - 所有者
4. 选择添加。用户已添加到您的服务器场。

删除用户

1. 选择要删除的用户。
2. 在“移除确认”对话框中，选择“删除”。然后，该用户将从选定的服务器场中移除。

您还可以使用 IAM Identity Center 控制台为用户和群组添加或删除服务器场权限，[网址为 https://console.aws.amazon.com/singlesignon/](https://console.aws.amazon.com/singlesignon/)。

截止日期云端作业

作业是 Deadline Cloud 用来安排和运行可用工作人员的工作的一组指令。AWS 创建任务时，您可以选择要将任务发送到的场和队列。您还提供一个 JSON 或 YAML 文件，为工作人员提供处理说明。Deadline Cloud 接受遵循公开职位描述 (OpenJD) 规范的职位模板来描述职位。欲了解更多信息，请参阅 GitHub 网站上的 [Open Job Description 文档](#)。

一份工作包括：

- 步骤-定义要在工作人员上运行的脚本。步骤可以有诸如最低工作内存或其他需要先完成的步骤之类的要求。每个步骤都有一个或多个任务。
- 任务-指派给工作人员执行的工作单元。任务是步骤的脚本和脚本中使用的参数（例如帧号）的组合。当所有步骤的所有任务都完成时，作业即告完成。
- 环境-设置和拆除由多个步骤或任务共享的指令。

您可以通过以下任一方式创建作业：

- 使用 Deadline Cloud 提交者。
- 创建任务捆绑包并使用 [Deadline Cloud 命令行界面](#)（Deadline Cloud CLI）。
- 使用 AWS 软件开发工具包。
- 使用 AWS Command Line Interface (AWS CLI)。

提交者是您的数字内容创作 (DCC) 软件的插件，用于管理在 DCC 软件界面中创建作业。创建任务后，您可以使用提交者将其发送到 Deadline Cloud 进行处理。在幕后，提交者创建了一个 OpenJD 作业模板来描述该作业。同时，它会将您的资产文件上传到亚马逊简单存储服务 (Amazon S3) 存储桶。为了缩短发送文件所需的时间，只有自上次上传文件以来发生更改的文件才会发送到 Amazon S3。

要创建自己的脚本和管道以向 Deadline Cloud 提交作业，您可以使用 Deadline Cloud CLI、AWS SDK 或调用操作来创建、获取、查看和列出作业。AWS CLI 以下主题说明了如何使用 Deadline Cloud CLI。

Deadline Cloud CLI 与 Deadline Cloud 提交者一起安装。有关更多信息，请参阅 [设置 Deadline Cloud 提交者](#)。

主题

- [使用 Deadline Cloud CLI 提交作业](#)

- [在截止日期云中安排作业](#)
- [Deadline Cloud CLI 中的 Job 状态](#)
- [在截止日期云中修改作业](#)
- [截止日期云如何处理作业](#)
- [排除 Deadline C](#)

使用 Deadline Cloud CLI 提交作业

要使用 Deadline Cloud 命令行界面 (Deadline Cloud CLI) 提交作业，请使用该 `deadline bundle submit` 命令。

任务已提交到队列。如果您尚未设置场和队列，请使用 Deadline Cloud 控制台 (<https://console.aws.amazon.com/deadlinecloud/home>) 设置场和队列，并查看场和队列 ID。有关更多信息，请参阅[定义服务器场详细信息和定义队列详细信息](#)。

要为 Deadline Cloud CLI 设置默认场和队列，请使用以下命令。设置默认值时，无需指定场或队列即可使用 Deadline Cloud CLI 命令。在以下示例中，`queueId` 用您自己的信息替换 `farmId` 和：

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

要指定作业中的步骤和任务，请创建 OpenJD 作业模板。有关更多信息，请参阅 [Open Job Description 规范存储库中的模板架构 \[版本：2023-09\]](#)。GitHub

以下示例是 YAML 作业模板。它定义了一个包含两个步骤、每个步骤五个任务的作业。

```
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
```

要创建作业，请创建一个名为的新文件夹 `sample_job`，然后在新文件夹中将模板文件另存为 `template.yaml`。您可以使用以下 Deadline Cloud CLI 命令提交作业：

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

命令的响应包含任务的标识符。记住该 ID，以便稍后可以查看任务的状态。

```
Submitting to Queue: test-queue
Waiting for Job to be created...
Submitted job bundle:
  sample_job
Job creation completed successfully
jobId
```

提交工作时，您还可以使用其他选项。有关更多信息，请参阅 [更多使用 Deadline Cloud CLI 提交作业的选项](#)。

更多使用 Deadline Cloud CLI 提交作业的选项

De deadline bundle submit adline Cloud CLI 命令提供了可用于为任务指定其他信息的选项。下面的示例向您演示如何：

- 指定处理作业模板时使用的参数。
- 将共享环境中的文件和文件夹附加到作业。

- 设置取消任务之前的最大任务失败次数。
- 设置任务的最大重试次数。

任务参数

该parameters选项在您创建作业时设置作业参数的值。作业模板定义字段，parameters选项设置值。参数可以有默认值。如果为参数指定了值，则指定的值将覆盖默认值。

以下作业模板定义了该TestParameter字段：

```
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
```

以下命令将的值设置为“Hello AWS”：TestParameter

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

存储配置文件

存储配置文件有助于在使用不同操作系统的员工之间共享文件。使用 Deadline Cloud 控制台创建存储配置文件。然后，使用storage-profile-id参数使用存储配置文件。有关更多信息，请参阅[截止日期云中的共享存储](#)。

要使用 Deadline Cloud CLI 为提交作业设置存储配置文件，请使用以下命令设置 `storage-profile-id` 配置参数：

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

最大失败任务数

该 `max-failed-tasks-count` 选项设置了在整个作业失败并标记所有剩余任务之前可能失败的最大任务数 `CANCELED`。默认值是 100。

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

失败任务的最大重试次数

该 `max-retries-per-task` 选项设置任务失败前重试的最大次数。当任务被重试时，它会进入 `READY` 状态。默认值是 5。

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

在截止日期云中安排作业

任务创建后，De AWS adline Cloud 会安排在与队列关联的一个或多个队列上对其进行处理。处理特定任务的队列是根据为队列配置的功能和特定步骤的主机要求来选择的。

作业按尽力而为的优先顺序排列，从高到低。当两个作业具有相同优先级时，将首先安排最早的作业。

以下各节详细介绍了安排作业的过程。

确定舰队兼容性

创建任务后，Deadline Cloud 会根据与提交任务的队列关联的队列的能力来检查作业中每个步骤的主机要求。如果舰队符合东道主的要求，则该任务将进入该 `READY` 状态。

如果任务中的任何步骤具有与队列关联的队列无法满足的要求，则该步骤的状态将设置为 `NOT_COMPATIBLE`。此外，作业中的其余步骤也将被取消。

舰队的能力是在舰队级别设置的。即使车队中的工人符合工作要求，如果其车队不符合工作要求，也不会从工作中为其分配任务。

以下作业模板的步骤指定了该步骤的主机要求：

```
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
```

可以将此任务安排给具有以下功能的舰队：

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

无法将此任务安排给具有以下任何功能的舰队：

```
{
  "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.
```

舰队扩展

将任务分配给兼容的服务托管队列时，队列会自动缩放。车队中的工作人员数量会根据可供车队运行的任务数量而波动。

将任务分配给客户管理的车队时，工作人员可能已经存在，或者可以使用基于事件的 auto Scaling 创建工作人员。有关更多信息，请参阅 Amazon EC2 Auto Scaling 用户指南中的用于 EventBridge 处理自动扩展[事件](#)。

会话

作业中的任务分为一个或多个会话。工作人员运行会话来设置环境，运行任务，然后拆除环境。每个会话都由工作人员必须采取的一项或多项操作组成。

工作人员完成分区操作后，可以向该工作人员发送其他会话操作。工作人员在会话中重复使用现有环境和作业附件，以更高效地完成作业。

作业附件由您使用的提交者创建，作为 Deadline Cloud CLI 任务捆绑包的一部分。您也可以使用 `create-job` AWS CLI 命令的 `--attachments` 选项来创建作业附件。环境在两个位置定义：附加到特定队列的队列环境和在作业模板中定义的作业步骤环境。

有四种会话操作类型：

- `syncInputJobAttachments`— 将输入的作业附件下载给工作人员。
- `envEnter`— 对环境执行`onEnter`操作。
- `taskRun`— 执行任务的`onRun`操作。
- `envExit`— 对环境执行`onExit`操作。

以下作业模板具有步骤环境。它有一个`onEnter`用于设置步骤环境的`onRun`定义、一个定义要运行的任务的定义以及一个用于拆除步骤环境的`onExit`定义。为此作业创建的会话将包括一个`envEnter`操作、一个或多个`taskRun`操作，然后是一个`envExit`操作。

```
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 }}
```

步骤依赖关系

Deadline Cloud 支持定义步骤之间的依赖关系，以便一个步骤等到另一个步骤完成后再开始。您可以为一个步骤定义多个依赖关系。只有在所有依赖项都完成之后，才会安排具有依赖关系的步骤。

如果作业模板定义了循环依赖关系，则该作业将被拒绝，作业状态将设置为CREATE_FAILED。

以下作业模板创建了一个包含两个步骤的作业。StepB取决于StepA。StepB仅在成功StepA完成后运行。

作业创建后，StepA处于READY状态并StepB处于PENDING状态。StepA完成后，StepB移动到READY状态。如果StepA失败或已取消，则StepAStepB移至CANCELED状态。

您可以为多个步骤设置依赖关系。例如，如果同时StepC依赖StepA和StepB，StepC则要等到其他两个步骤完成后才会启动。

```
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!
```

Deadline Cloud CLI 中的 Job 状态

本主题介绍如何使用 De AWS adline Cloud 命令行界面 (Deadline Cloud CLI) 查看任务或步骤的状态。如果要使用 Deadline Cloud 监控器来查看任务或步骤的状态，请参阅[在 Deadline Cloud 中查看和管理作业、步骤和任务](#)。

您可以使用 Deadline Cloud CLI 命令查看任务的状态。deadline job get --job-id对命令的响应包括作业或步骤的状态以及处于每种处理状态的任务数。

首次提交工作时，状态为CREATE_IN_PROGRESS。如果作业通过了验证检查，则其状态将更改为CREATE_COMPLETE。否则，状态将更改为CREATE_FAILED。

作业可能无法通过验证检查的一些可能原因包括：

- 作业模板不符合 OpenJD 规范。
- 该作业包含的步骤太多。
- 该作业包含的任务总数过多。

要查看作业中最大步骤和任务数的配额，请使用 Service Quotas 控制台。有关更多信息，请参阅 [配额 Deadline Cloud](#)。

也可能存在内部服务错误，导致无法创建作业。如果发生这种情况，则任务的状态代码为 INTERNAL_ERROR，状态消息字段将提供更详细的解释。

使用以下 Deadline Cloud CLI 命令查看任务的详细信息。在以下示例中，*jobID* 用您自己的信息替换：

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

该 `deadline job get` 命令的响应如下：

```
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
```

作业或步骤中的每项任务都有一个状态。将任务状态组合在一起，以提供作业和步骤的总体状态。在响应 `taskRunStatusCounts` 字段中报告每种状态下的任务数。

作业或步骤的状态取决于其任务的状态。状态由具有这些状态的任务按顺序确定。步骤状态的确定方式与任务状态相同。

以下列表描述了状态：

NOT_COMPATIBLE

该任务与服务器场不兼容，因为没有舰队可以完成任务中的一项任务。

RUNNING

一个或多个工作人员正在运行作业中的任务。只要至少有一个正在运行的任务，该作业就会被标记 `RUNNING`。

ASSIGNED

将工作中的任务分配给一个或多个工作人员，作为他们的下一个操作。环境（如果有）已设置完毕。

STARTING

一个或多个工作人员正在为运行任务设置环境。

SCHEDULED

该作业的任务将安排在一个或多个工作人员身上，作为该工作人员的下一步操作。

READY

该作业的至少一项任务已准备就绪，可供处理。

INTERRUPTING

作业中至少有一个任务被中断。当你手动更新任务的状态时，可能会出现中断。它也可能是为了应对亚马逊弹性计算云 (Amazon EC2) 现货价格变动造成的中断。

FAILED

作业中的一个或多个任务未成功完成。

CANCELED

任务中的一个或多个任务已被取消。

SUSPENDED

作业中至少有一项任务已暂停。

PENDING

任务中的一项任务正在等待其他资源的可用性。

SUCCEEDED

作业中的所有任务均已成功处理。

在截止日期云中修改作业

您可以使用以下 AWS Command Line Interface (AWS CLI) `update` 命令修改作业的配置，或者设置作业、步骤或任务的目标状态：

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

在以下命令示例中，将每个 `update` 命令替换 *user input placeholder* 为您自己的信息。

您还可以使用 Deadline Cloud 监控器来修改作业的配置。有关更多信息，请参阅 [在 Deadline Cloud 中查看和管理作业、步骤和任务](#)。

Example — 重新排队作业

除非存在步骤依赖关系，否则作业中的所有任务都会切换到READY状态。具有依赖关系的步骤在恢复时切换到任一READY或PENDING。

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

Example — 取消作业

作业中所有没有状态SUCCEEDED或已标记FAILED的任务CANCELED。

```
aws deadline update-job \  

```

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

Example — 将任务标记为失败

作业中所有处于该状态的任务SUCCEEDED都保持不变。所有其他任务都已标记FAILED。

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

Example — 将工作标记为成功

作业中的所有任务都将变为SUCCEEDED状态。

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

Example — 暂停作业

作业中处于SUCCEEDED、CANCELED、或FAILED状态的任务不会改变。所有其他任务都已标记SUSPENDED。

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

Example — 更改作业的优先级

更新任务的优先级以更改其调度顺序。优先级较高的作业通常先安排。

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

```
--queue-id queueID \  
--job-id jobID \  
--priority 100
```

Example — 更改允许的失败任务数

更新在取消剩余任务之前该任务可以执行的最大失败任务数。

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

Example — 更改允许的任务重试次数

更新任务失败前任务的最大重试次数。已达到最大重试次数的任务在增加该值之前无法重新排队。

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

Example — 存档作业

将作业的生命周期状态更新为ARCHIVED。无法安排或修改已存档的作业。您只能存档处于FAILED、CANCELED、SUCCEEDED、或SUSPENDED状态的作业。

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

Example — 重新排队步骤

除非存在步骤依赖关系，否则步骤中的所有任务都会切换到READY状态。具有依赖关系的步骤中的任务会切换到READY或PENDING，任务将恢复。

```
aws deadline update-step \  
--farm-id farmID \  
--step-id stepID \  
--priority 100
```

```
--queue-id queueID \  
--job-id jobID \  
--step-id stepID \  
--target-task-run-status PENDING
```

Example — 取消步骤

步骤中所有没有状态SUCCEEDED或已标记FAILED的任务CANCELED。

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

Example — 将步骤标记为失败

步骤中所有状态为的任务SUCCEEDED均保持不变。所有其他任务都已标记FAILED。

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

Example — 将步骤标记为成功

步骤中的所有任务都已标记SUCCEEDED。

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

Example — 暂停步骤

处于SUCCEEDED、CANCELED、或FAILED状态的步骤中的任务不会更改。所有其他任务都已标记SUSPENDED。

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

Example — 更改任务的状态

当您使用 `aws deadline update-task` Cloud CLI 命令时，任务会切换到指定的状态。

```
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
```

截止日期云如何处理作业

为了处理作业，Deadline Cloud 使用空缺职位描述 (OpenJD) 作业模板来确定所需的资源。Deadline Cloud 从与您的队列关联的队列中为某一步骤选择合适的工作人员。所选工作人员符合该步骤所需的所有能力属性。

接下来，Deadline Cloud 向工作人员发送指令，要求他们为该步骤设置会话。该步骤所需的软件必须在工作器实例上可用，作业才能运行。如果队列的扩展设置有容量，则该服务可以在多个工作人员上打开会话。

您可以在 Amazon Machine Image (AMI) 中设置软件，或者您的工作人员可以在运行时从存储库或包管理器加载软件。您可以使用队列、作业或步骤环境来部署您喜欢的软件。

Deadline Cloud 服务使用 OpenJD 模板来确定作业所需的步骤以及每个步骤所需的任务。有些步骤依赖于其他步骤，因此 Deadline Cloud 决定了完成这些步骤的顺序。然后，Deadline Cloud 会将每个步骤的任务发送给工作人员进行处理。任务完成后，服务会在同一个会话中发送另一个任务，或者工作人员可以启动新的会话。

您可以在 Deadline Cloud 监控器、Deadline Cloud 命令行界面 (Deadline Cloud CLI) 或 AWS CLI。有关使用显示器的更多信息，请参阅[使用截止日期云监视器](#)。有关使用 Deadline Cloud CLI 的更多信息，请参阅[Deadline Cloud CLI 中的 Job 状态](#)。

每个步骤中的所有任务都完成后，作业就完成了，输出就可以下载到您的工作站了。即使任务没有完成，也可以下载每个步骤和已完成任务的输出。

Deadline Cloud 会在作业提交 120 天后将其删除。移除作业后，与该作业关联的所有步骤和任务也会被删除。如果您需要重新运行作业，请再次提交该作业的 OpenJD 模板。

排除 Deadline C

有关 De AWS adline Cloud 中作业的常见问题的信息，请参阅以下主题。

主题

- [为什么创建我的任务失败了？](#)
- [为什么我的工作不兼容？](#)
- [为什么我的工作准备就绪？](#)
- [为什么我的工作失败了？](#)
- [为什么我的步骤处于待处理状态？](#)

为什么创建我的任务失败了？

作业可能无法通过验证检查的一些可能原因包括：

- 作业模板不符合 OpenJD 规范。
- 该作业包含的步骤太多。
- 该作业包含的任务总数过多。
- 出现内部服务错误，导致无法创建作业。

要查看作业中最大步骤和任务数的配额，请使用 Service Quotas 控制台。有关更多信息，请参阅 [配额 Deadline Cloud](#)。

为什么我的工作不兼容？

作业与队列不兼容的常见原因包括以下几点：

- 没有队列与提交任务的队列相关联。打开 Deadline Cloud 监视器，检查队列中是否有关联的队列。有关如何查看队列的更多信息，请参阅[在截止日期云中查看队列和舰队详情](#)。

- 与队列关联的任何队列都无法满足该任务的主机要求。要进行检查，请将作业模板中的 `hostRequirements` 条目与农场中舰队的配置进行比较。确保其中一支舰队满足房东的要求。有关队列兼容性的更多信息，请参阅[确定机队兼容性](#)。要查看队列配置，请参阅[在截止日期云中查看队列和舰队详情](#)。

为什么我的工作准备就绪？

你的工作似乎陷入困境的可能原因包括以下几点：READY

- 与队列关联的队列的最大工作人员数设置为零。要进行检查，请参阅[在截止日期云中查看队列和舰队详情](#)。
- 队列中有更高优先级的作业。要进行检查，请参阅[在截止日期云中查看队列和舰队详情](#)。
- 对于客户管理的车队，请查看 `auto scaling` 配置。有关更多信息，请参阅[使用 Deadline Cloud 规模推荐功能自动扩展您的 Amazon EC2 机群](#)。

为什么我的工作失败了？

任务失败的原因有很多。要搜索问题，请打开 Deadline Cloud 监视器并选择失败的作业。选择失败的任务，然后查看该任务的日志。有关说明，请参阅[在截止日期云中查看日志](#)。

- 如果您看到许可证错误，或者由于软件没有有效的许可证而出现水印，请确保工作人员可以连接到所需的许可证服务器。有关更多信息，请参阅[Connect 将客户管理的车队连接到许可证端点](#)。

为什么我的步骤处于待处理状态？

当一个或多个依赖项未完成时，步骤可能会保持PENDING状态。你可以使用 Deadline Cloud 监视器检查依赖关系的状态。有关说明，请参阅[在截止日期云中查看步骤](#)。

截止日期云的文件存储

工作人员必须有权访问包含处理作业所需的输入文件的存储位置以及存储输出的位置。AWS Deadline Cloud 为存储位置提供了两个选项：

- 借助作业附件，Deadline Cloud 可以在工作站和 Deadline Cloud 工作人员之间来回传输作业的输入和输出文件。为了启用文件传输，Deadline Cloud 在您的存储区中使用亚马逊简单存储服务 (Amazon S3) 存储桶。AWS 账户

当您在服务托管队列中使用任务附件时，可以在虚拟专用网络 (VPN) 中设置虚拟文件系统 (VFS)。然后，工作人员只能在需要时加载文件。

- 使用共享存储，您可以使用与操作系统的文件共享来提供对文件的访问权限。

使用跨平台共享存储时，可以创建存储配置文件，以便工作人员可以在两个不同的操作系统之间映射文件路径。

主题

- [截止日期云中的 Job 附件](#)
- [截止日期云中的共享存储](#)

截止日期云中的 Job 附件

Job 附件使您能够在工作站和 De AWS adline Cloud 之间来回传输文件。使用任务附件，您无需为文件手动设置 Amazon S3 存储桶。相反，当您使用 Deadline Cloud 控制台创建队列时，您可以为任务附件选择存储桶。

首次向 Deadline Cloud 提交作业时，该作业的所有文件都将传输到 Deadline Cloud。对于后续提交，仅传输已更改的文件，从而节省时间和带宽。

处理完成后，您可以从任务详细信息页面下载结果，也可以使用 Deadline Cloud CLI `deadline job download-output` 命令下载结果。

您可以将相同的 S3 存储桶用于多个队列。为每个队列设置不同的根前缀以整理存储桶中的附件。

使用控制台创建队列时，您可以选择现有 AWS Identity and Access Management (IAM) 角色，也可以让控制台创建新角色。如果控制台创建了角色，则它会设置访问为队列指定的存储桶的权限。如果您选择现有角色，则必须向该角色授予访问 S3 存储桶的权限。

对任务附件 S3 存储桶进行加密

默认情况下，您的 S3 存储桶中会自动加密 Job 附件文件。这种方法有助于保护您的信息免遭未经授权的访问。您无需执行任何操作即可使用 Deadline Cloud 提供的密钥对文件进行加密。有关更多信息，请参阅 [Amazon S3 用户指南中的 Amazon S3 现在会自动加密所有新对象](#)。

您可以使用自己的客户托管 AWS Key Management Service 密钥对包含任务附件的 S3 存储桶进行加密。为此，您必须修改与存储桶关联的队列的 IAM 角色以允许访问 AWS KMS key。

打开队列角色的 IAM 策略编辑器

1. 登录 AWS Management Console 并打开 [Deadline Cloud 控制台](#)。在主页的“入门”部分，选择“查看农场”。
2. 从服务器场列表中，选择包含要修改的队列的场。
3. 从队列列表中选择要修改的队列。
4. 在队列详细信息部分，选择服务角色以打开该服务角色的 IAM 控制台。

接下来，完成以下步骤。

更新角色策略，使其具有以下权限 AWS KMS

1. 从权限策略列表中，为角色选择策略。
2. 在此策略中定义的权限部分中，选择编辑。
3. 选择添加新语句。
4. 将以下策略复制并粘贴到编辑器中。将 *Region*、*accountID*、和 *keyID* 更改为您自己的值。

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

5. 选择下一步。

6. 查看对政策的更改，然后在满意时选择“保存更改”。

管理 S3 存储桶中的任务附件

Deadline Cloud 将您的任务所需的任务附件存储在 S3 存储桶中。这些文件会随着时间的推移而累积，从而导致 Amazon S3 成本增加。为了降低成本，您可以将 S3 生命周期配置应用于 S3 存储桶。此配置可以自动删除存储桶中的文件。由于 S3 存储桶位于您的账户中，因此您可以随时选择修改或删除 S3 生命周期配置。有关更多信息，请参阅 [Amazon S3 用户指南中的 S3 生命周期配置示例](#)。

要获得更精细的 S3 存储桶管理解决方案，您可以将您的设置 AWS 账户 为根据上次访问时间在 S3 存储桶中使对象过期。有关更多信息，请参阅 AWS 架构博客 [上的基于上次访问日期使 Amazon S3 对象过期以降低成本](#)。

截止日期云虚拟文件系统

De AWS adline Cloud 中对作业附件的虚拟文件系统支持使工作人员上的客户端软件能够直接与 Amazon Simple Storage Service 通信。工作人员只能在需要时加载文件，而不是在处理之前下载所有文件。文件存储在本地。这种方法可以避免下载多次使用的资源。任务完成后，所有文件都将被删除。

- 虚拟文件系统为特定的作业配置文件提供了显著的性能提升。通常，文件总量中较小的子集和较大的工作人员队伍显示出最大的好处。工作线程较少的少量文件处理时间大致相同。
- 虚拟文件系统支持仅适用于服务管理队列中的 Linux 工作人员。
- Deadline Cloud 虚拟文件系统支持以下操作，但不兼容 POSIX：
 - 文件 `create`、`delete`、`open`、`close`、`read`、`write`、`append`、`truncate`、`rename`、`move`、`chown` 和 `falloc`
 - 目录 `create`、`delete`、`rename`、`move`、`copy`、和 `stat`
- 当您的任务仅访问大型数据集的一部分，并且未针对所有工作负载进行优化时，虚拟文件系统旨在减少数据传输并提高性能。在运行生产作业之前，您应该测试您的工作负载。

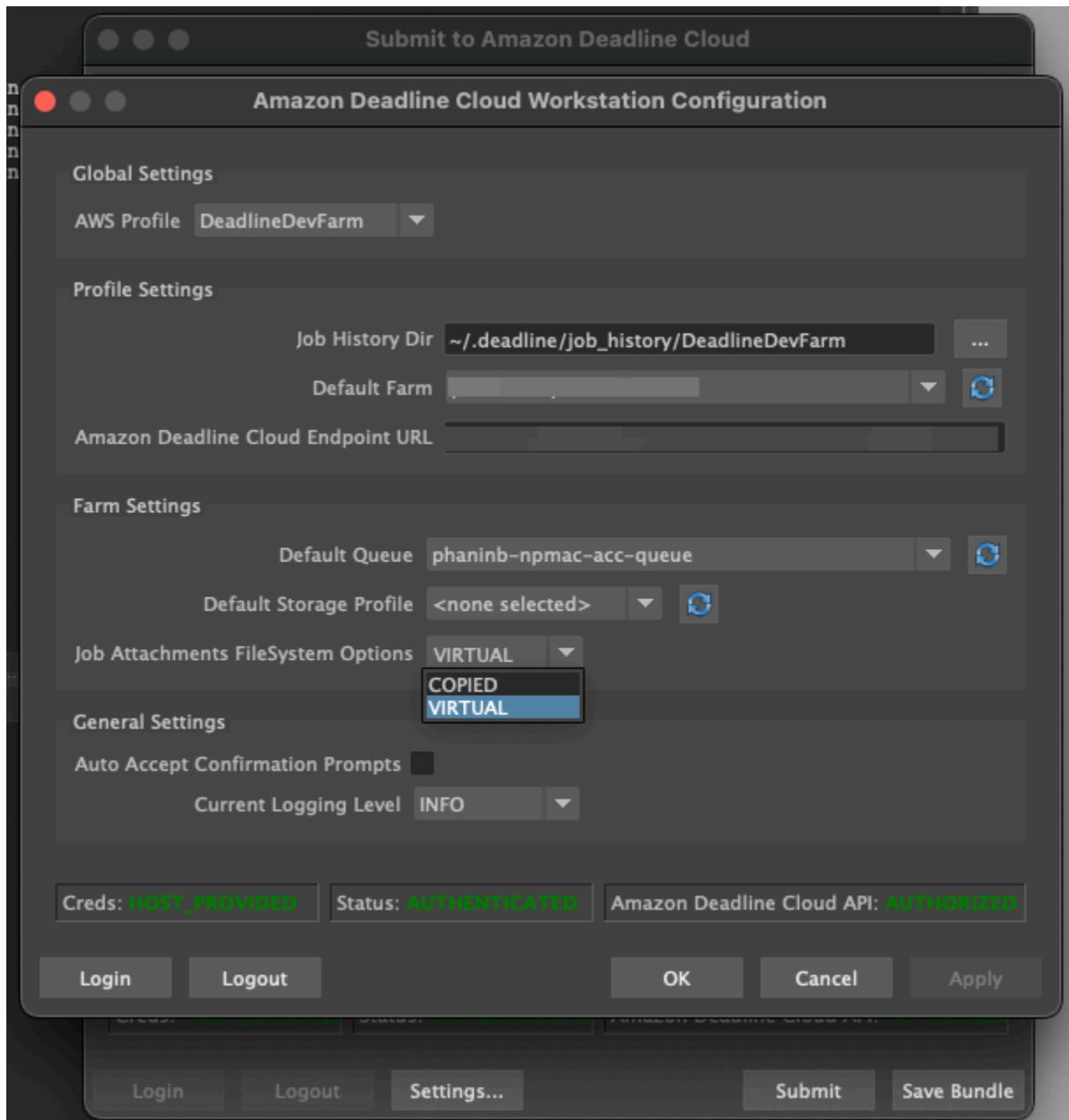
启用 VFS 支持

为每个作业启用了虚拟文件系统支持 (VFS)。在以下情况下，作业会回退到默认的作业附件框架：

- 工作器实例配置文件不支持虚拟文件系统。
- 问题导致无法启动虚拟文件系统进程。
- 无法装载虚拟文件系统。

使用提交者启用虚拟文件系统支持

1. 提交作业时，选择“设置”按钮以打开 De AWS adline Cloud 工作站配置面板。
2. 从 Job 附件文件系统选项下拉列表中，选择 VIRTUAL。



3. 要保存更改，请选择“确定”。

要启用虚拟文件系统支持，请使用 AWS CLI

- 提交保存的作业时，请使用以下命令：

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

要验证是否为特定任务成功启动了虚拟文件系统，请在 Amazon Logs 中查看您的 CloudWatch 日志。查找以下消息：

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

如果日志包含以下消息，则虚拟文件系统支持已禁用：

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

虚拟文件系统支持疑难解答

您可以使用 Deadline Cloud 监视器查看虚拟文件系统的日志。有关说明，请参阅[在截止日期云中查看日志](#)。

虚拟文件系统日志还会发送到与工作器代理输出共享的队列关联的 CloudWatch 日志组。

截止日期云中的共享存储

要使用共享存储，工作人员使用操作系统文件共享系统访问共享存储空间，用于作业的输入和输出。

您用于共享文件的实际方法取决于您的操作系统以及在网络上实现共享存储的方式。您负责如何配置文件共享并确保其满足您的需求。

如果您使用的是跨系统文件共享解决方案，则可以使用存储配置文件在Linux和文件系统之间映射Windows文件位置。

截止日期云中的存储配置文件

存储配置文件使您能够使用跨平台共享存储来设置服务器场。存储配置文件映射跨操作系统的路径，用于在与提交工作站不同的操作系统的工作系统上处理的作业。

当您使用客户管理的队列时，工作站和工作人员之间混合使用操作系统，则需要存储配置文件。服务托管队列不支持存储配置文件。

创建存储配置文件后，必须向使用该配置文件的队列和队列授予访问权限。

创建存储配置文件

1. 打开[截止日期云控制台](#)。
2. 从“开始”中，选择“前往截止日期云控制面板”。
3. 选择一个场，然后选择存储配置文件选项卡。
4. 选择创建存储配置文件。
5. 从下拉列表中选择一个操作系统。
6. 为配置文件提供一个名称。清晰的名称可帮助您选择提交作业时要使用的存储配置文件。
7. 在路径名中，输入您提交作业的工作站上作业数据的根位置。
8. 选择存储类型：
 - 本地是指工作人员和工作站之间不共享的文件位置。它们以作业附件的形式上传。
 - 共享是指工作人员和工作站之间共享的存储。共享存储空间中的文件不会作为作业附件上传。
9. 提供文件系统位置路径。这是您的作业数据的根目录。
10. 选择创建。

创建存储配置文件后，必须修改队列和客户管理的队列才能使用新的配置文件。要允许访问存储配置文件，请在完成前面的步骤后使用以下步骤。

允许队列和客户管理的队列使用存储配置文件

1. 选择“队列”或“队列”选项卡。
2. 选择要修改的队列或舰队。
3. 选择修改存储配置文件。
4. 选择要允许的存储配置文件以及该配置文件中的文件系统位置。
5. 选择保存更改。

管理截止日期云的预算和使用情况

De AWS adline Cloud 预算管理器和使用情况浏览器是成本管理工具，它们根据有关成本变量的可用信息提供使用 Deadline Cloud 的大致成本。成本管理工具不能保证您实际使用Deadline Cloud和其他 AWS 服务所欠的金额。

为了帮助您管理 Deadline Cloud 的成本，您可以使用以下功能：

- 预算经理 — 借助 Deadline Cloud 预算管理器，您可以创建和编辑预算以帮助管理项目成本。
- 使用情况浏览器-使用 Deadline Cloud 使用情况浏览器，您可以查看使用了多少 AWS 资源以及这些资源的估计成本。

成本假设

Deadline Cloud 成本管理工具使用的基本计算方法是：

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

- 运行时间是作业中所有任务的总和，从开始时间到结束时间。
- 计算费率由服务托管车队的 De [AWS adline Cloud 定价](#) 决定。对于客户管理的车队，计算费率估计为每工时 1 美元。
- 许可费率由 Deadline Cloud 的基本许可价格决定。不包括其他等级。有关许可证定价的更多信息，请参阅 De [AWS adline Cloud 定价](#)。

Deadline Cloud 成本管理工具估算的成本可能与您的实际成本有所不同，原因有很多。常见原因包括：

- 客户拥有的资源及其定价。您可以选择从本地 AWS 或其他云提供商那里自带资源，也可以从外部引入资源。未计算这些资源的实际成本。
- 闲置工人的成本。对于最小实例数大于零的车队，在计算中不考虑闲置员工。
- 促销积分、折扣和自定义定价协议。成本管理工具不考虑促销积分、私人定价协议或其他折扣。您可能会有资格获得不在估算范围内的其他折扣。

- 资产存储。成本和使用量估算中不包括资产存储。
- 价格的变化。AWS 为大多数服务提供 pay-as-you-go 定价。价格可能会随着时间的推移而变化。成本管理工具使用的 up-to-date 价格最多，但变更后可能会有延迟。
- 税收。成本管理工具不包括适用于我们购买服务的税款。
- 四舍五入。成本管理工具对定价数据进行数学四舍五入。
- 货币。费用估算以美元计算。全球汇率会随着时间的推移而变化。如果您根据当前汇率将估计值转换为不同的货币基础，则汇率的变化会影响估计值。
- 外部许可。如果您选择使用预先购买的许可证（自带许可证），Deadline Cloud 成本管理工具无法计算这笔费用。

使用截止日期云预算管理器

Deadline Cloud 预算管理器可帮助您控制给定资源（例如队列、舰队或农场）上的支出。您可以创建预算金额和限额，并设置自动操作以帮助减少或停止超出预算的额外支出。

以下各节为您提供使用 Deadline Cloud 预算管理器的步骤。

主题

- [先决条件](#)
- [访问预算经理](#)
- [创建预算](#)
- [查看预算](#)
- [编辑预算](#)
- [停用预算](#)

先决条件

要使用 Deadline Cloud 预算管理器，您必须具有OWNER访问级别。要授予OWNER权限，请按照中的步骤操作[在截止日期云中管理用户](#)。

访问预算经理

要访问 Deadline Cloud 预算管理器，请按以下步骤操作。

1. 登录 AWS Management Console 并打开 [Deadline Cloud 控制台](#)。

2. 选择“查看农场”。
3. 找到您要获取相关信息的农场，然后选择管理作业。Deadline Cloud 监控器将在新选项卡中打开。
4. 在 Deadline Cloud 监控器的左侧导航窗格中，选择预算。

预算经理摘要页面显示有效和无效预算的列表：

- 活动预算会根据所选资源（队列）进行跟踪。
- 无效预算要么已过期，要么已被用户取消，并且不再根据该预算的限制跟踪成本。

选择预算后，预算摘要页面将包含有关该预算的基本信息。提供的信息包括预算名称、状态、资源、剩余百分比、剩余金额、总预算、开始日期和结束日期。

创建预算

要创建预算，请按以下步骤操作。

1. 如果您尚未登录，请登录 AWS Management Console，打开 [Deadline Cloud 控制台](#)，选择一个农场，然后选择管理作业。
2. 在预算管理器页面中，选择创建预算。
3. 在详细信息部分，输入预算的预算名称。
4. （可选）在说明字段中，输入预算的清晰简短描述。
5. 从“资源”中选择“队列”下拉列表以查找并选择要为其创建预算的队列。
6. 对于期间，通过完成以下步骤来设置预算的开始和结束日期：

- a. 在“开始日期”中，以 YYYY/MM/DD 格式输入预算跟踪的起始日期，或者选择日历图标并选择日期。

默认起始日期是预算的创建日期。

- b. 在结束日期中，以 YYYY/MM/DD 格式输入预算跟踪的最后日期，或者选择日历图标并选择日期。

默认结束日期为自开始日期起 120 天。

7. 在预算金额中，输入预算的美元金额。
8. （可选）我们建议您创建限制提醒。在“限制操作”部分中，您可以实施在预算中仍有特定金额时发生的自动操作。为此，请完成以下步骤：

- a. 选择“添加新操作”。
 - b. 在剩余金额中，输入您要开始操作的美元金额。
 - c. 在“操作”下拉列表中，选择所需的操作。操作包括：
 - 完成当前工作后停止 — 当达到阈值金额时，当前正在运行的所有工作将继续运行（并产生成本），直到完成。
 - 立即停止工作 — 当达到阈值金额时，将立即取消所有工作。
 - d. 要创建其他限额提醒，请选择添加新操作并重复前两个步骤。
9. 选择创建预算。此时将显示预算经理页面。新创建的预算显示在“有效预算”选项卡中。

查看预算

创建预算后，您可以在预算管理器页面上查看预算。在这里，您可以查看预算的总金额和分配给特定预算的总成本。

要查看预算，请按以下步骤操作。

1. 如果您尚未登录，请登录 AWS Management Console，打开 [Deadline Cloud 控制台](#)，选择一个场，然后选择管理作业。
2. 从左侧导航窗格中选择预算。此时将出现“预算经理”页面。
3. 要查看有效预算，请选择有效预算选项卡，然后选择要查看的预算名称。此时将显示预算详情页面。
4. 要查看已到期预算的预算详细信息，请选择无效预算选项卡。然后，选择要查看的预算的名称。此时将显示预算详情页面。

编辑预算

您可以编辑任何有效的预算。要编辑有效预算，请按以下步骤操作。

1. 如果您尚未登录，请登录 AWS Management Console，打开 [Deadline Cloud 控制台](#)，选择一个场，然后选择管理作业。
2. 在预算管理器页面的有效预算选项卡中，选择要编辑的预算旁边的按钮。
3. 从右上角的操作下拉菜单中，选择编辑预算。
4. 根据需要进行更改，然后选择更新预算。

停用预算

您可以停用任何有效预算。停用预算会将其状态从“有效”更改为“无效”。停用预算后，它将不再根据该预算的金额跟踪资源。

要停用预算，请按以下步骤操作。

1. 如果您尚未登录，请登录 AWS Management Console，打开 [Deadline Cloud 控制台](#)，选择一个场，然后选择管理作业。
2. 在预算管理页面有效预算选项卡中，选择要停用的预算旁边的按钮。
3. 从右上角的操作下拉菜单中，选择停用预算。稍后，所选预算将从“有效”变为“无效”，并将从“有效预算”选项卡移至“无效预算”选项卡。

使用截止日期云使用情况浏览器

使用 Deadline Cloud 使用情况浏览器，您可以查看每个服务器场上发生的活动的实时指标。您可以通过不同的变量来查看服务器场的成本，例如队列、作业、许可产品或实例类型。选择不同的时间范围以查看特定时间段内的使用情况，并查看一段时间内的使用趋势。您还可以查看所选数据点的详细细分，从而可以更仔细地查看指标。使用情况可以按时间（分钟和小时）或成本（美元）显示。

以下各节向您展示了访问和使用 Deadline Cloud 使用情况浏览器的步骤。

主题

- [先决条件](#)
- [打开使用情况浏览器](#)
- [使用使用情况浏览器](#)

先决条件

要使用 Deadline Cloud 使用情况浏览器，您必须拥有MANAGER或OWNER场权限。有关更多信息，请参阅 [管理农场、队列和队列的用户和群组](#)。

打开使用情况浏览器

要打开 Deadline Cloud 使用情况浏览器，请按以下步骤操作。

1. 登录 AWS Management Console 并打开 [Deadline Cloud 控制台](#)。

2. 要查看所有可用的农场，请选择查看农场。
3. 找到您要获取相关信息的农场，然后选择管理作业。Deadline Cloud 监控器将在新选项卡中打开。
4. 在 Deadline Cloud 监视器中，从左侧菜单中选择“使用情况资源管理器”。

使用使用情况浏览器

在使用情况资源管理器页面中，您可以选择显示数据的特定参数。默认情况下，您会看到过去 7 天内按时间（小时和分钟）表示的总使用量。您可以更改这些参数，并且显示的信息会根据参数设置动态变化。

您可以根据队列、作业、计算使用情况、实例类型或许可产品对结果进行分组。如果您选择许可产品，则按特定许可证计算成本。对于所有其他组，时间是通过将每个任务的运行时间相加来计算的。

根据您的筛选条件，使用情况浏览器仅返回 100 个结果。结果按创建日期的时间戳降序列出。如果结果超过 100 个，则会收到一条错误消息。您可以优化查询以减少结果数量：

- 选择较小的时间范围
- 选择更少的队列
- 选择不同的分组，例如按队列而不是按作业分组

主题

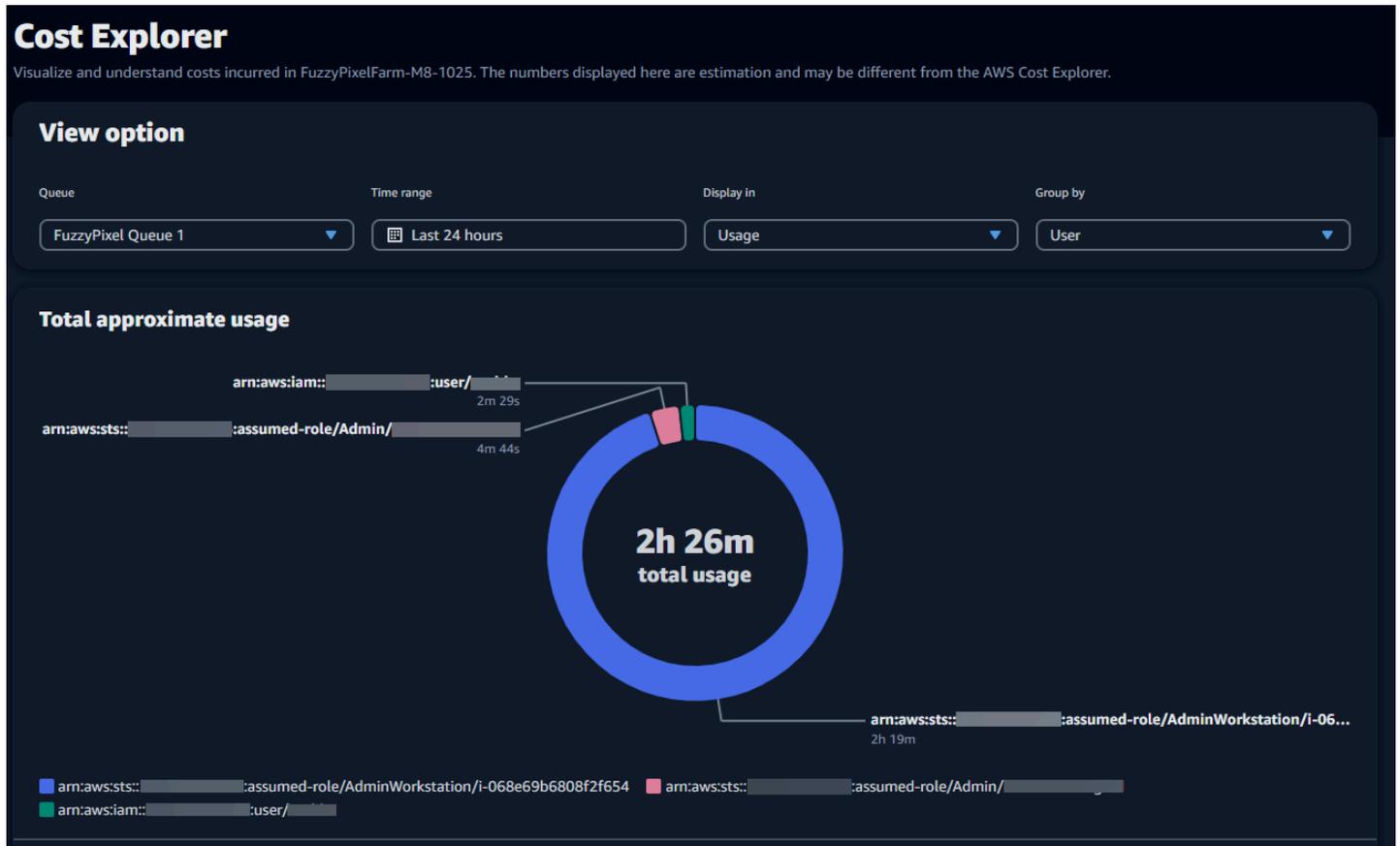
- [使用可视化图表查看数据](#)
- [查看指标明细](#)
- [查看队列的大致运行时间](#)

使用可视化图表查看数据

您可以以可视化格式查看数据，以确定趋势和可能需要更多分析或关注的潜在领域。使用情况浏览器提供了显示总体使用量和成本的饼图，并可以选择将总量分组为较小的小计。

Note

该图表仅显示前五个结果以及其他结果合并在一个“其他”部分中。您可以在图表下方的细分部分中查看所有结果。



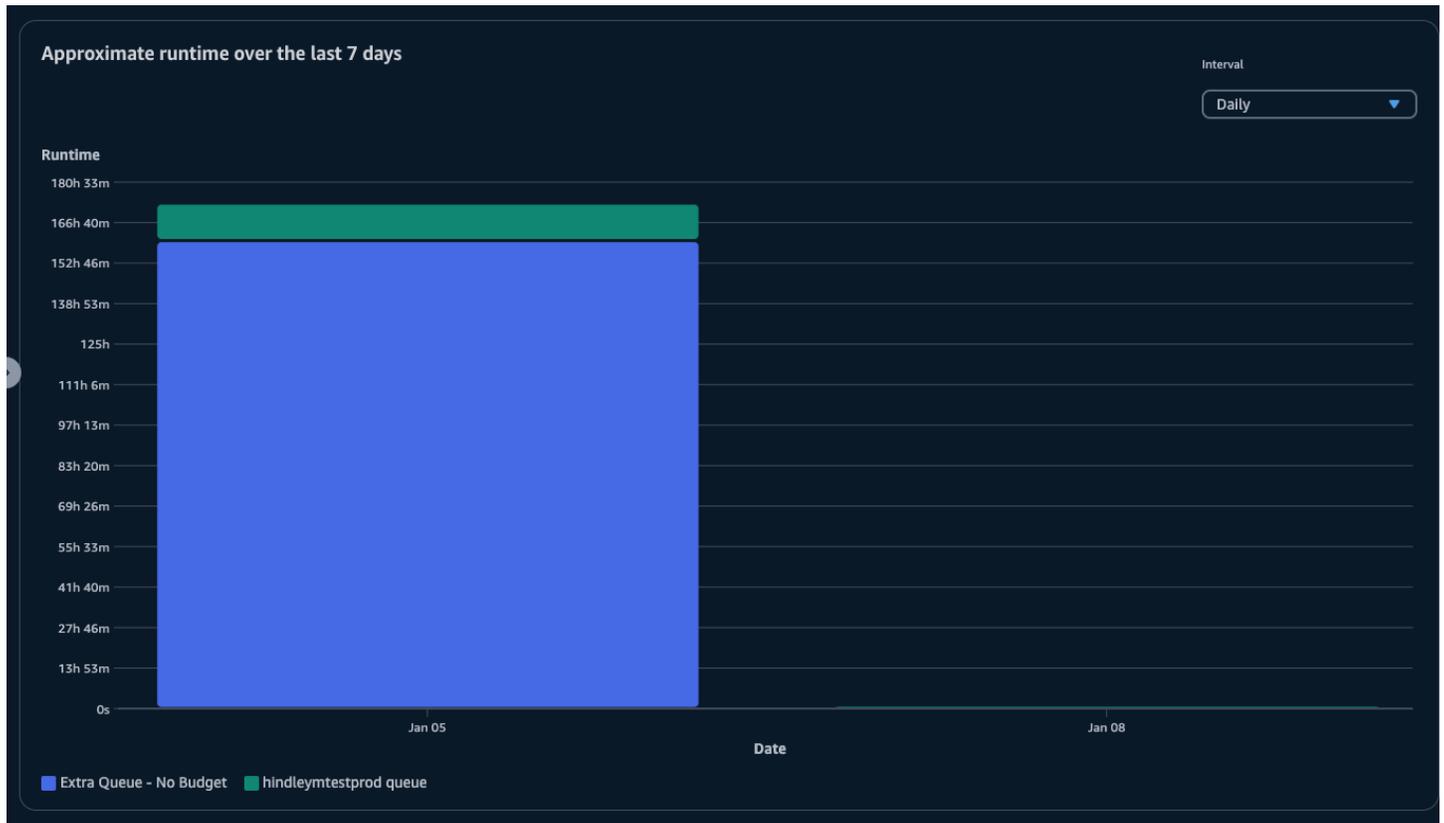
查看指标明细

在饼图下方，使用情况浏览器提供了更详细的特定指标明细，这些指标将随着参数的变化而变化。默认情况下，使用情况资源管理器中会显示五个结果。您可以使用划分部分中的分页箭头滚动浏览结果。

默认情况下，细分最小化。要展开并显示结果，请选择查看所有细分箭头。要下载细目，请选择下载数据。

查看队列的大致运行时间

您还可以根据您指定的不同时间间隔查看队列的大致运行时间。间隔选项包括每小时、每天、每周和每月。选择间隔后，图表将显示队列的大致运行时间。



成本管理

AWS Deadline Cloud 提供预算和使用情况浏览器，可帮助您控制和可视化工作成本。但是，Deadline Cloud 使用其他 AWS 服务，例如亚马逊 S3。这些服务的费用不会反映在 Deadline Cloud 预算或使用量资源管理器中，而是根据使用量单独收费。根据您的配置 Deadline Cloud 的方式，您可以使用以下 AWS 服务以及其他服务：

服务	定价页面
亚马逊 CloudWatch 日志	亚马逊 CloudWatch 日志定价
Amazon Elastic Compute Cloud	Amazon 弹性计算云定价
AWS Key Management Service	AWS Key Management Service 定价
AWS PrivateLink	AWS PrivateLink 定价
Amazon Simple Storage Service	Amazon Simple Storage Service 定价

服务	定价页面
Amazon Virtual Private Cloud	亚马逊 Virtual Private Cloud 定价

成本管理最佳实践

使用以下最佳实践可以帮助您了解和控制使用 Deadline Cloud 时的成本，以及在成本和效率之间可以做出的权衡。

Note

使用 Deadline Cloud 的最终成本取决于多种 AWS 服务之间的交互、您处理的工作量以及您运行作业 AWS 区域的地点。以下最佳做法仅供参考，可能不会显著降低成本。

CloudWatch 日志的最佳实践

Deadline Cloud 将工作人员和任务日志发送到 CloudWatch 日志。您需要收集、存储和分析这些日志。您可以通过仅记录监控任务所需的最低数据量来降低成本。

创建队列或队列时，Deadline Cloud 会使用以下名称创建 CloudWatch 日志组：

- `aws/deadline/<FARM_ID>/<FLEET_ID>`
- `aws/deadline/<FARM_ID>/<QUEUE_ID>`

默认情况下，这些日志永不过期。您可以调整日志组的保留策略以删除旧日志并帮助降低存储成本。您还可以将日志导出到 Amazon S3。Amazon S3 的存储成本低于的存储成本 CloudWatch。有关更多信息，请参阅[将日志数据导出至 Amazon S3](#)。

Amazon EC2 的最佳实践

您可以将 Amazon EC2 实例用于服务托管和客户管理的队列。有三个注意事项：

- 对于服务管理队列，您可以通过设置队列的最低工作人员数量来选择让一个或多个实例始终可用。当您为最小工作人员数设置为 0 以上时，队列中总是有这么多名工作人员在运行。这可以缩短 Deadline Cloud 开始处理任务所需的时间，但是您需要为实例的空闲时间付费。
- 对于服务管理的队列，请设置队列的最大规模。这限制了队列可以自动扩展到的实例数量。即使有更多的工作等待处理，船队也不会超过这个规模。

- 对于服务托管和客户管理的队列，您都可以在队列中指定 Amazon EC2 实例类型。使用较小的实例每分钟的成本较低，但可能需要更长的时间才能完成任务。相反，较大的实例每分钟的成本更高，但可以缩短完成任务的时间。了解您的任务对实例提出的要求有助于降低成本。
- 如果可能，请为您的队列选择 Amazon EC2 竞价型实例。竞价型实例的价格较低，但可能会因按需请求而中断。按需实例按秒计费，不会中断。

的最佳实践 AWS KMS

默认情况下，Deadline Cloud 使用 AWS 自有密钥对您的数据进行加密。您无需为此密钥付费。

您可以选择使用客户管理的密钥来加密您的数据。当您使用自己的密钥时，将根据密钥的使用方式向您收费。如果您使用现有密钥，则额外使用将产生增量成本。

的最佳实践 AWS PrivateLink

您可以使用接口终端节点 AWS PrivateLink 在您的 VPC 和 Deadline Cloud 之间创建连接。创建连接时，您可以调用所有 Deadline Cloud API 操作。对于您创建的每个终端节点，按小时计费。如果使用 PrivateLink，则必须创建至少三个终端节点，根据您的配置，您可能需要多达五个。

亚马逊 S3 的最佳实践

Deadline Cloud 使用 Amazon S3 存储待处理的资产、任务附件、输出和日志。要降低与 Amazon S3 相关的成本，请减少您存储的数据量。一些建议：

- 仅存储当前正在使用或即将使用的资产。
- 使用 [S3 生命周期配置](#) 自动从 S3 存储桶中删除未使用的文件。

亚马逊 VPC 的最佳实践

当您对客户管理的队列使用基于使用量的许可时，您将创建一个 Deadline Cloud 许可证终端节点，即在您的账户中创建的 Amazon VPC 终端节点。此端点按小时费率收费。要降低成本，请在不使用基于使用量的许可证时移除端点。

安全性 Deadline Cloud

云安全 AWS 是重中之重。作为 AWS 客户，您可以受益于专为满足大多数安全敏感型组织的要求而构建的数据中心和网络架构。

安全是双方共同承担 AWS 的责任。[责任共担模式](#)将其描述为云的安全性和云中的安全性：

- 云安全 — AWS 负责保护在云 AWS 服务 中运行的基础架构 AWS Cloud。AWS 还为您提供可以安全使用的服务。作为[AWS 合规计划](#)的一部分，第三方审计师定期测试和验证我们安全的有效性。要了解适用于的合规计划 AWS Deadline Cloud，请参阅“按合规计划划分[AWS 服务的范围](#)”中的“[按合规计划AWS 服务](#)”。
- 云端安全 — 您的责任由您 AWS 服务 使用的内容决定。您还需要对其他因素负责，包括您的数据的敏感性、您的公司的要求以及适用的法律法规。

本文档可帮助您了解在使用时如何应用分担责任模型 Deadline Cloud。以下主题向您介绍如何进行配置 Deadline Cloud 以满足您的安全和合规性目标。您还将学习如何使用其他 AWS 服务 方法来监控和保护您的 Deadline Cloud 资源。

主题

- [中的数据保护 Deadline Cloud](#)
- [Deadline Cloud 中的身份和访问管理](#)
- [合规性验证 Deadline Cloud](#)
- [韧性在 Deadline Cloud](#)
- [截止日期云中的基础设施安全](#)
- [截止日期云中的配置和漏洞分析](#)
- [防止跨服务混淆座席](#)
- [AWS Deadline Cloud 使用接口端点进行访问 \(AWS PrivateLink\)](#)
- [截止日期云的安全最佳实践](#)

中的数据保护 Deadline Cloud

分 AWS [担责任模型](#)适用于中的数据保护 AWS Deadline Cloud。如本模型所述 AWS ，负责保护运行所有内容的全球基础架构 AWS Cloud。您负责维护对托管在此基础架构上的内容的控制。您还负

责您所使用的 AWS 服务 的安全配置和管理任务。有关数据隐私的更多信息，请参阅[数据隐私常见问题](#)。有关欧洲数据保护的信息，请参阅 AWS Security Blog 上的 [AWS Shared Responsibility Model and GDPR](#) 博客文章。

出于数据保护目的，我们建议您保护 AWS 账户 凭证并使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 设置个人用户。这样，每个用户只获得履行其工作职责所需的权限。我们还建议您通过以下方式保护数据：

- 对每个账户使用多重身份验证 (MFA)。
- 使用 SSL/TLS 与资源通信。AWS 我们要求使用 TLS 1.2，建议使用 TLS 1.3。
- 使用设置 API 和用户活动日志 AWS CloudTrail。
- 使用 AWS 加密解决方案以及其中的所有默认安全控件 AWS 服务。
- 使用高级托管安全服务（例如 Amazon Macie），它有助于发现和保护存储在 Amazon S3 中的敏感数据。
- 如果您在 AWS 通过命令行界面或 API 进行访问时需要经过 FIPS 140-2 验证的加密模块，请使用 FIPS 端点。有关可用的 FIPS 端点的更多信息，请参阅 [《美国联邦信息处理标准 \(FIPS\) 第 140-2 版》](#)。

我们强烈建议您切勿将机密信息或敏感信息（如您客户的电子邮件地址）放入标签或自由格式文本字段（如名称字段）。这包括您使用控制台、API Deadline Cloud 或 SDK 或以其他 AWS 服务 方式使用控制台 AWS CLI、API 或 AWS SDK 的情况。在用于名称的标签或自由格式文本字段中输入的任何数据都可能会用于计费或诊断日志。如果您向外部服务器提供网址，强烈建议您不要在网址中包含凭证信息来验证对该服务器的请求。

主题

- [静态加密](#)
- [传输中加密](#)
- [密钥管理](#)
- [互连网络流量隐私](#)
- [选择退出](#)

静态加密

AWS Deadline Cloud 使用存储在 [AWS Key Management Service \(AWS KMS\)](#) 中的加密密钥对静态数据进行加密，从而保护敏感数据。所有可用 AWS 区域 的地方 Deadline Cloud 都提供静态加密。

加密数据意味着如果没有有效的密钥，用户或应用程序就无法读取保存在磁盘上的敏感数据。只有拥有有效托管密钥的一方才能解密数据。

有关如何 Deadline Cloud 使用 AWS KMS 静态加密数据的信息，请参阅[密钥管理](#)。

传输中加密

对于传输中的数据，AWS Deadline Cloud 使用传输层安全 (TLS) 1.2 或 1.3 来加密在服务和工作程序之间发送的数据。我们要求使用 TLS 1.2，建议使用 TLS 1.3。此外，如果您使用虚拟私有云 (VPC)，则可以使用 AWS PrivateLink 在您的 VPC 和之间建立私有连接 Deadline Cloud。

密钥管理

创建新服务器场时，您可以选择以下密钥之一来加密服务器场数据：

- AWS 拥有的 KMS 密钥-如果您在创建服务器场时未指定密钥，则为默认加密类型。KMS 密钥归所有者 AWS Deadline Cloud。您无法查看、管理或使用 AWS 自有密钥。但是，您无需采取任何措施来保护加密数据的密钥。有关更多信息，请参阅AWS Key Management Service 开发者指南中的[AWS 自有密钥](#)。
- 客户托管的 KMS 密钥-您在创建服务器场时指定客户托管密钥。服务器场中的所有内容均使用 KMS 密钥进行加密。密钥存储在您的账户中，由您创建、拥有和管理，并 AWS KMS 收取费用。您对 KMS 密钥拥有完全控制权。您可以执行以下任务：
 - 制定和维护关键政策
 - 建立和维护 IAM 策略和授权
 - 启用和禁用密钥策略
 - 添加标签
 - 创建密钥别名

您无法手动轮换用于 Deadline Cloud 服务器场的客户拥有的密钥。支持密钥的自动轮换。

有关更多信息，请参阅《AWS Key Management Service 开发者指南》中的[客户拥有的密钥](#)。

要创建客户托管密钥，请按照《AWS Key Management Service 开发人员指南》中[创建对称客户托管密钥](#)的步骤进行操作。

如何 Deadline Cloud 使用 AWS KMS 补助金

Deadline Cloud 需要获得[授权](#)才能使用您的客户托管密钥。当您创建使用客户托管密钥加密的场时，Deadline Cloud 会向发送[CreateGrant](#)请求 AWS KMS 以获取您指定的 KMS 密钥的访问权限，从而代表您创建授权。

Deadline Cloud 使用多个授权。每项拨款都由需要加密或解密您的数据的不同部分使用。Deadline Cloud 还使用授权来允许访问用于代表您存储数据的其他 AWS 服务，例如亚马逊简单存储服务、Amazon Elastic Block Store 或 OpenSearch。

Deadline Cloud 允许管理服务管理队列中的计算机的授权包括 Deadline Cloud 账号和角色，`GranteePrincipal`而不是服务委托人。虽然不常见，但这是使用为服务器场指定的客户托管 KMS 密钥为服务托管队伍中的工作人员加密 Amazon EBS 卷所必需的。

客户托管密钥策略

密钥策略控制对客户托管密钥的访问。每个密钥必须只有一个密钥策略，其中包含用于确定谁可以使用密钥以及如何使用密钥的声明。在创建客户托管密钥时，您可以指定密钥策略。有关更多信息，请参阅《AWS Key Management Service 开发人员指南》中的[管理对客户托管密钥的访问](#)。

适用的最低 IAM 政策 `CreateFarm`

要使用您的客户托管密钥通过控制台或 [CreateFarm](#) API 操作创建农场，必须允许以下 AWS KMS API 操作：

- [kms:CreateGrant](#) – 添加客户托管密钥授权。授予对指定 AWS KMS 密钥的控制台访问权限。有关更多信息，请参阅AWS Key Management Service 开发者指南中的[使用授权](#)。
- [kms:Decrypt](#)— Deadline Cloud 允许解密服务器场中的数据。
- [kms:DescribeKey](#)— 提供客户管理的密钥详细信息 Deadline Cloud 以允许验证密钥。
- [kms:GenerateDataKey](#)— Deadline Cloud 允许使用唯一的数据密钥加密数据。

以下策略声明授予`CreateFarm`操作所需的权限。

```
{
  "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"
        }
    }
}
]
}

```

只读操作的最低 IAM 政策

使用您的客户托管密钥进行只读 Deadline Cloud 操作，例如获取有关农场、队列和队列的信息。必须允许以下 AWS KMS API 操作：

- [kms:Decrypt](#)— Deadline Cloud 允许解密服务器场中的数据。
- [kms:DescribeKey](#)— 提供客户管理的密钥详细信息 Deadline Cloud 以允许验证密钥。

以下策略声明授予只读操作所需的权限。

```

{
  "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"
        }
      }
    }
  ]
}

```

```

    }
  ]
}

```

用于读写操作的最低 IAM 策略

使用您的客户托管密钥进行读写 Deadline Cloud 操作，例如创建和更新服务器场、队列和队列。必须允许以下 AWS KMS API 操作：

- [kms:Decrypt](#)— Deadline Cloud 允许解密服务器场中的数据。
- [kms:DescribeKey](#)— 提供客户管理的密钥详细信息 Deadline Cloud 以允许验证密钥。
- [kms:GenerateDataKey](#)— Deadline Cloud 允许使用唯一的数据密钥加密数据。

以下策略声明授予 CreateFarm 操作所需的权限。

```

{
  "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"
        }
      }
    }
  ]
}

```

监控您的加密密钥

当您在 Deadline Cloud 服务器场中使用 AWS KMS 客户托管密钥时，您可以使用 [AWS CloudTrail](#) 或 [Amazon CloudWatch Logs](#) 来跟踪 Deadline Cloud 发送到的请求 AWS KMS。

CloudTrail 补助金活动

以下示例 CloudTrail 事件发生在创建授权时，通常是在您调用CreateFarmCreateMonitor、或CreateFleet操作时。

```
{
  "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 用于解密的事件

使用客户托管的 KMS 密钥解密值时会发生以下示例 CloudTrail 事件。

```

{
  "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 加密事件

使用客户托管的 KMS 密钥对值进行加密时，会发生以下示例 CloudTrail 事件。

```
{
  "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"
}

```

删除客户托管的 KMS 密钥

删除 AWS Key Management Service (AWS KMS) 中客户管理的 KMS 密钥具有破坏性，并且具有潜在的危险。这将删除密钥材料以及与此密钥关联的所有元数据，并且不可撤销。删除客户托管 KMS 密钥后，您不能再解密用该此密钥加密的数据。这表示无法恢复此数据。

这就是为什么客户 AWS KMS 在删除 KMS 密钥之前有长达 30 天的等待期。默认的等待期限为 30 天。

关于等待期限

由于删除客户管理的 KMS 密钥具有破坏性和潜在危险，因此我们要求您将等待期设置为 7-30 天。默认的等待期限为 30 天。

但是，实际等待时间可能比您预定的时间长达 24 小时。要获取删除密钥的实际日期和时间，请使用 [DescribeKey](#) 操作。您还可以在 [AWS KMS 控制台](#) 中的密钥详细信息页面的常规配置部分中参阅密钥计划删除日期。注意时区。

在等待期限内，客户托管密钥状态和密钥状态为等待删除。

- 待删除的客户托管 KMS 密钥不能用于任何[加密操作](#)。
- AWS KMS 不会[轮换待删除的客户托管 KMS 密钥的支持](#)密钥。

有关删除客户托管的 KMS 密钥的更多信息，请参阅《AWS Key Management Service 开发人员指南》中的[删除客户主密钥](#)。

互连网络流量隐私

AWS Deadline Cloud 支持亚马逊 Virtual Private Cloud (亚马逊 VPC) 来保护连接。Amazon VPC 提供三种功能，以供您用来提高和监控虚拟私有云 (VPC) 的安全性：

您可以使用在 VPC 内运行的亚马逊弹性计算云 (Amazon EC2) 实例来设置客户托管队列 (CMF)。通过部署要使用的 Amazon VPC 终端节点 AWS PrivateLink，您的 CMF 中的工作人员与 Deadline Cloud 终端节点之间的流量将保留在您的 VPC 内。此外，您可以将您的 VPC 配置为限制您的实例访问互联网。

在服务管理的车队中，无法通过互联网联系到员工，但他们确实可以访问互联网并通过互联网连接到 Deadline Cloud 服务。

选择退出

AWS Deadline Cloud 收集某些运营信息以帮助我们发展和改进 Deadline Cloud。收集的数据包括您的 AWS 帐户 ID 和用户 ID 之类的信息，以便在您遇到问题时我们可以正确识别您的身份 Deadline Cloud。我们还收集 Deadline Cloud 特定信息，例如资源 ID (适用时为 farmID 或 queueID)、产品名称 (例如 JobAttachments WorkerAgent、等) 和产品版本。

您可以使用应用程序配置选择退出此数据收集。与之交互的每台计算机 Deadline Cloud，包括客户工作站和车队员工，都需要单独选择退出。

Deadline Cloud 显示器-台式机

Deadline Cloud monitor-desktop 会收集操作信息，例如何时发生崩溃以及何时打开应用程序，以帮助我们知道您的应用程序何时出现问题。要选择收集这些操作信息，请前往设置页面并清除“开启数据收集以衡量 Deadline Cloud Monitor 的性能”。

在您选择退出后，桌面显示器将不再发送操作数据。之前收集的所有数据都将被保留，并且仍可用于改进服务。有关更多信息，请参阅 [数据隐私 FAQ](#)。

AWS Deadline Cloud CLI 和工具

AWS Deadline Cloud CLI、提交者和工作人员代理都会收集操作信息，例如何时发生崩溃以及何时提交作业，以帮助我们知道您在使用这些应用程序时遇到问题。要选择不收集此操作信息，请使用以下任一方法：

- 在终端中输入 **deadline config set telemetry.opt_out true**。

当以当前用户身份运行时，这将选择退出 CLI、提交者和工作器代理。

- 安装 Deadline Cloud 工作器代理时，添加 **--telemetry-opt-out** 命令行参数。例如， **./install.sh --farm-id \$FARM_ID --fleet-id \$FLEET_ID --telemetry-opt-out**。
- 在运行工作器代理、CLI 或提交器之前，请设置环境变量：**DEADLINE_CLOUD_TELEMETRY_OPT_OUT=true**

在您选择退出后，这些 Deadline Cloud 工具将不再发送操作数据。之前收集的所有数据都将被保留，并且仍可用于改进服务。有关更多信息，请参阅 [数据隐私 FAQ](#)。

Deadline Cloud 中的身份和访问管理

AWS Identity and Access Management (IAM) AWS 服务 可帮助管理员安全地控制对 AWS 资源的访问权限。IAM 管理员控制谁可以进行身份验证（登录）和授权（有权限）使用 Deadline Cloud 资源。您可以使用 IAM AWS 服务，无需支付额外费用。

主题

- [受众](#)
- [使用身份进行身份验证](#)
- [使用策略管理访问](#)
- [截止日期云如何与 IAM 配合使用](#)
- [Deadline Cloud 基于身份的策略示例](#)
- [AWS 截止日期云的托管策略](#)
- [故障排除 De AWS adline Cloud](#)

受众

您的使用方式 AWS Identity and Access Management (IAM) 会有所不同，具体取决于您在 Deadline Cloud 中所做的工作。

服务用户-如果您使用 Deadline Cloud 服务完成工作，则您的管理员会为您提供所需的凭据和权限。当您使用更多的 Deadline Cloud 功能来完成工作时，您可能需要额外的权限。了解如何管理访问权限有助于您向管理员请求适合的权限。如果您无法访问 Deadline Cloud 中的某项功能，请参阅[故障排除 De AWS adline Cloud](#)。

服务管理员 — 如果您负责公司的 Deadline Cloud 资源，则可能拥有对 Deadline Cloud 的完全访问权限。您的工作是确定您的服务用户应访问哪些 Deadline Cloud 功能和资源。然后，您必须向 IAM 管理员提交请求以更改服务用户的权限。请查看该页面上的信息以了解 IAM 的基本概念。要详细了解贵公司如何将 IAM 与 Deadline Cloud 配合使用，请参阅[截止日期云如何与 IAM 配合使用](#)。

IAM 管理员 — 如果您是 IAM 管理员，则可能需要详细了解如何编写策略来管理 Deadline Cloud 的访问权限。要查看您可以在 IAM 中使用的基于身份的 Deadline Cloud 策略示例，请参阅。[Deadline Cloud 基于身份的策略示例](#)

使用身份进行身份验证

身份验证是您 AWS 使用身份凭证登录的方式。您必须以 IAM 用户身份或通过担 AWS 账户根用户任 IAM 角色进行身份验证 (登录 AWS)。

您可以使用通过身份源提供的凭据以 AWS 联合身份登录。AWS IAM Identity Center (IAM Identity Center) 用户、贵公司的单点登录身份验证以及您的 Google 或 Facebook 凭据就是联合身份的示例。当您以联合身份登录时，您的管理员以前使用 IAM 角色设置了身份联合验证。当您使用联合访问 AWS 时，您就是在间接扮演一个角色。

根据您的用户类型，您可以登录 AWS Management Console 或 AWS 访问门户。有关登录的更多信息 AWS，请参阅《AWS 登录 用户指南》[中的如何登录到您 AWS 账户](#)的。

如果您 AWS 以编程方式访问，则会 AWS 提供软件开发套件 (SDK) 和命令行接口 (CLI)，以便使用您的凭据对请求进行加密签名。如果您不使用 AWS 工具，则必须自己签署请求。有关使用推荐的方法自行签署请求的更多信息，请参阅 IAM 用户指南中的[签署 AWS API 请求](#)。

无论使用何种身份验证方法，您可能需要提供其他安全信息。例如，AWS 建议您使用多重身份验证 (MFA) 来提高账户的安全性。要了解更多信息，请参阅《AWS IAM Identity Center 用户指南》中的 [多重身份验证](#)和《IAM 用户指南》中的 [在 AWS 中使用多重身份验证 \(MFA \)](#)。

AWS 账户 root 用户

创建时 AWS 账户，首先要有一个登录身份，该身份可以完全访问账户中的所有资源 AWS 服务和资源。此身份被称为 AWS 账户 root 用户，使用您创建帐户时使用的电子邮件地址和密码登录即可访问该身份。强烈建议您不要使用根用户执行日常任务。保护好根用户凭证，并使用这些凭证来执行仅根用户可以执行的任务。有关要求您以根用户身份登录的任务的完整列表，请参阅《IAM 用户指南》中的[需要根用户凭证的任务](#)。

联合身份

作为最佳实践，要求人类用户（包括需要管理员访问权限的用户）使用与身份提供商的联合身份验证 AWS 服务 通过临时证书进行访问。

联合身份是指您的企业用户目录、Web 身份提供商、Identity Center 目录中的用户，或者任何使用 AWS 服务 通过身份源提供的凭据进行访问的用户。AWS Directory Service 当联合身份访问时 AWS 账户，他们将扮演角色，角色提供临时证书。

要集中管理访问权限，建议您使用 AWS IAM Identity Center。您可以在 IAM Identity Center 中创建用户和群组，也可以连接并同步到您自己的身份源中的一组用户和群组，以便在您的所有 AWS 账户 和应用程序中使用。有关 IAM Identity Center 的信息，请参阅《AWS IAM Identity Center 用户指南》中的[什么是 IAM Identity Center ?](#)

IAM 用户和群组

[IAM 用户](#)是您 AWS 账户 内部对个人或应用程序具有特定权限的身份。在可能的情况下，我们建议使用临时凭证，而不是创建具有长期凭证（如密码和访问密钥）的 IAM 用户。但是，如果您有一些特定的使用场景需要长期凭证以及 IAM 用户，建议您轮换访问密钥。有关更多信息，请参阅《IAM 用户指南》中的[对于需要长期凭证的使用场景定期轮换访问密钥](#)。

[IAM 组](#)是一个指定一组 IAM 用户的身份。您不能使用组的身份登录。您可以使用组来一次性为多个用户指定权限。如果有大量用户，使用组可以更轻松地管理用户权限。例如，您可能具有一个名为 IAMAdmins 的组，并为该组授予权限以管理 IAM 资源。

用户与角色不同。用户唯一地与某个人员或应用程序关联，而角色旨在让需要它的任何人代入。用户具有永久的长期凭证，而角色提供临时凭证。要了解更多信息，请参阅《IAM 用户指南》中的[何时创建 IAM 用户（而不是角色）](#)。

IAM 角色

[IAM 角色](#)是您内部具有特定权限 AWS 账户 的身份。它类似于 IAM 用户，但与特定人员不关联。您可以 AWS Management Console 通过[切换角色在中临时担任 IAM 角色](#)。您可以通过调用 AWS CLI 或

AWS API 操作或使用自定义 URL 来代入角色。有关使用角色的方法的更多信息，请参阅《IAM 用户指南》中的[使用 IAM 角色](#)。

具有临时凭证的 IAM 角色在以下情况下很有用：

- 联合用户访问 – 要向联合身份分配权限，请创建角色并为角色定义权限。当联合身份进行身份验证时，该身份将与角色相关联并被授予由此角色定义的权限。有关联合身份验证的角色的信息，请参阅《IAM 用户指南》中的[为第三方身份提供商创建角色](#)。如果您使用 IAM Identity Center，则需要配置权限集。为控制您的身份在进行身份验证后可以访问的内容，IAM Identity Center 将权限集与 IAM 中的角色相关联。有关权限集的信息，请参阅《AWS IAM Identity Center 用户指南》中的[权限集](#)。
- 临时 IAM 用户权限 – IAM 用户可代入 IAM 用户或角色，以暂时获得针对特定任务的不同权限。
- 跨账户存取 – 您可以使用 IAM 角色以允许不同账户中的某个人（可信主体）访问您的账户中的资源。角色是授予跨账户访问权限的主要方式。但是，对于某些资源 AWS 服务，您可以将策略直接附加到资源（而不是使用角色作为代理）。要了解角色和基于资源的跨账户访问策略之间的区别，请参阅[IAM 用户指南中的跨账户资源访问](#)。
- 跨服务访问 — 有些 AWS 服务使用其他 AWS 服务服务中的功能。例如，当您在某个服务中进行调用时，该服务通常会在 Amazon EC2 中运行应用程序或在 Simple Storage Service (Amazon S3) 中存储对象。服务可能会使用发出调用的主体的权限、使用服务角色或使用服务相关角色来执行此操作。
- 转发访问会话 (FAS) — 当您使用 IAM 用户或角色在中执行操作时 AWS，您被视为委托人。使用某些服务时，您可能会执行一个操作，然后此操作在其他服务中启动另一个操作。FAS 使用调用委托人的权限以及 AWS 服务 向下游服务发出请求的请求。AWS 服务只有当服务收到需要与其他 AWS 服务 或资源交互才能完成的请求时，才会发出 FAS 请求。在这种情况下，您必须具有执行这两个操作的权限。有关发出 FAS 请求时的策略详情，请参阅[转发访问会话](#)。
- 服务角色 - 服务角色是服务代表您在您的账户中执行操作而分派的 [IAM 角色](#)。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的[创建向 AWS 服务委派权限的角色](#)。
- 服务相关角色-服务相关角色是一种链接到的服务角色。AWS 服务服务可以代入代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。
- 在 A@@@ mazon EC2 上运行的应用程序 — 您可以使用 IAM 角色管理在 EC2 实例上运行并发出 AWS CLI 或 AWS API 请求的应用程序的临时证书。这优先于在 EC2 实例中存储访问密钥。要向 EC2 实例分配 AWS 角色并使其可供其所有应用程序使用，您需要创建附加到该实例的实例配置文件。实例配置文件包含角色，并使 EC2 实例上运行的程序能够获得临时凭证。有关更多信息，请参阅《IAM 用户指南》中的[使用 IAM 角色为 Amazon EC2 实例上运行的应用程序授予权限](#)。

要了解是使用 IAM 角色还是 IAM 用户，请参阅 IAM 用户指南中的[何时创建 IAM 角色（而不是用户）](#)。

使用策略管理访问

您可以 AWS 通过创建策略并将其附加到 AWS 身份或资源来控制中的访问权限。策略是其中的一个对象 AWS，当与身份或资源关联时，它会定义其权限。AWS 在委托人（用户、root 用户或角色会话）发出请求时评估这些策略。策略中的权限确定是允许还是拒绝请求。大多数策略都以 JSON 文档的 AWS 形式存储在中。有关 JSON 策略文档的结构和内容的更多信息，请参阅 IAM 用户指南中的[JSON 策略概览](#)。

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

默认情况下，用户和角色没有权限。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM 策略。管理员随后可以向角色添加 IAM policy，用户可以代入角色。

IAM 策略定义操作的权限，无关乎您使用哪种方法执行操作。例如，假设您有一个允许 iam:GetRole 操作的策略。拥有该策略的用户可以从 AWS Management Console AWS CLI、或 AWS API 获取角色信息。

基于身份的策略

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的[创建 IAM policy](#)。

基于身份的策略可以进一步归类为内联策略或托管策略。内联策略直接嵌入单个用户、组或角色中。托管策略是独立的策略，您可以将其附加到中的多个用户、群组和角色 AWS 账户。托管策略包括 AWS 托管策略和客户托管策略。要了解如何在托管式策略和内联策略之间进行选择，请参阅 IAM 用户指南中的[在托管式策略与内联策略之间进行选择](#)。

基于资源的策略

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。对于在其中附加策略的资源，策略定义指定主体可以对该资源执行哪些操作以及在什么条件下执行。您必须在基于资源的策略中[指定主体](#)。委托人可以包括账户、用户、角色、联合用户或 AWS 服务。

基于资源的策略是位于该服务中的内联策略。您不能在基于资源的策略中使用 IAM 中的 AWS 托管策略。

访问控制列表 (ACL)

访问控制列表 (ACL) 控制哪些主体 (账户成员、用户或角色) 有权访问资源。ACL 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

Amazon S3 和 Amazon VPC 就是支持 ACL 的服务示例。AWS WAF 要了解有关 ACL 的更多信息，请参阅《Amazon Simple Storage Service 开发人员指南》中的[访问控制列表 \(ACL \) 概览](#)。

其他策略类型

AWS 支持其他不太常见的策略类型。这些策略类型可以设置更常用的策略类型向您授予的最大权限。

- **权限边界**：权限边界是一个高级特征，用于设置基于身份的策略可以为 IAM 实体 (IAM 用户或角色) 授予的最大权限。您可为实体设置权限边界。这些结果权限是实体基于身份的策略及其权限边界的交集。在 Principal 中指定用户或角色的基于资源的策略不受权限边界限制。任一项策略中的显式拒绝将覆盖允许。有关权限边界的更多信息，请参阅《IAM 用户指南》中的[IAM 实体的权限边界](#)。
- **服务控制策略 (SCP)**-SCP 是 JSON 策略，用于指定组织或组织单位 (OU) 的最大权限。AWS Organizations AWS Organizations 是一项用于对您的企业拥有的多 AWS 账户 项进行分组和集中管理的服务。如果在组织内启用了所有功能，则可对任意或全部账户应用服务控制策略 (SCP)。SCP 限制成员账户中的实体 (包括每个 AWS 账户根用户实体) 的权限。有关 Organizations 和 SCP 的更多信息，请参阅《AWS Organizations 用户指南》中的[SCP 的工作原理](#)。
- **会话策略** – 会话策略是当您以编程方式为角色或联合用户创建临时会话时作为参数传递的高级策略。结果会话的权限是用户或角色的基于身份的策略和会话策略的交集。权限也可以来自基于资源的策略。任一项策略中的显式拒绝将覆盖允许。有关更多信息，请参阅《IAM 用户指南》中的[会话策略](#)。

多个策略类型

当多个类型的策略应用于一个请求时，生成的权限更加复杂和难以理解。要了解在涉及多种策略类型时如何 AWS 确定是否允许请求，请参阅 IAM 用户指南中的[策略评估逻辑](#)。

截止日期云如何与 IAM 配合使用

在使用 IAM 管理 Deadline Cloud 的访问权限之前，请先了解哪些可用于 Deadline Cloud 的 IAM 功能。

您可以在 Deadline Cloud 上 AWS 使用的 IA

IAM 功能	截止日期云支持
基于身份的策略	是
基于资源的策略	否
策略操作	是
策略资源	是
策略条件键 (特定于服务)	是
ACL	否
ABAC (策略中的标签)	是
临时凭证	是
转发访问会话 (FAS)	是
服务角色	是
服务相关角色	否

要全面了解 Deadline Cloud 和其他功能如何 AWS 服务 与大多数 IAM 功能配合使用，请参阅 [IAM 用户指南中与 IAM 配合使用的AWS 服务](#)。

Deadline Cloud 基于身份的策略

支持基于身份的策略	是
-----------	---

基于身份的策略是可附加到身份 (如 IAM 用户、用户组或角色) 的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的[创建 IAM policy](#)。

通过使用 IAM 基于身份的策略，您可以指定允许或拒绝的操作和资源以及允许或拒绝操作的条件。您无法在基于身份的策略中指定主体，因为它适用于其附加的用户或角色。要了解可在 JSON 策略中使用的所有元素，请参阅《IAM 用户指南》中的 [IAM JSON 策略元素引用](#)。

Deadline Cloud 基于身份的策略示例

要查看 Deadline Cloud 基于身份的策略的示例，请参阅 [Deadline Cloud 基于身份的策略示例](#)

截止日期云中基于资源的政策

支持基于资源的策略	否
-----------	---

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。对于在其中附加策略的资源，策略定义指定主体可以对该资源执行哪些操作以及在什么条件下执行。您必须在基于资源的策略中 [指定主体](#)。委托人可以包括账户、用户、角色、联合用户或 AWS 服务。

要启用跨账户存取，您可以将整个账户或其他账户中的 IAM 实体指定为基于资源的策略中的主体。将跨账户主体添加到基于资源的策略只是建立信任关系工作的一半而已。当委托人和资源处于不同位置时 AWS 账户，可信账户中的 IAM 管理员还必须向委托人实体（用户或角色）授予访问资源的权限。他们通过将基于身份的策略附加到实体以授予权限。但是，如果基于资源的策略向同一个账户中的主体授予访问权限，则不需要额外的基于身份的策略。有关更多信息，请参阅 IAM 用户指南中的跨账户访问 [IAM 中的资源](#)。

截止日期云的政策行动

支持策略操作	是
--------	---

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

JSON 策略的 Action 元素描述可用于在策略中允许或拒绝访问的操作。策略操作通常与关联的 AWS API 操作同名。有一些例外情况，例如没有匹配 API 操作的仅限权限操作。还有一些操作需要在策略中执行多个操作。这些附加操作称为相关操作。

在策略中包含操作以授予执行关联操作的权限。

要查看 Deadline Cloud 操作列表，请参阅《服务授权参考》中的 [De AWS adline Cloud 定义的操作](#)。

Deadline Cloud 中的策略操作在操作前使用以下前缀：

```
deadline
```

要在单个语句中指定多项操作，请使用逗号将它们隔开。

```
"Action": [  
  "deadline:action1",  
  "deadline:action2"  
]
```

要查看 Deadline Cloud 基于身份的策略的示例，请参阅 [Deadline Cloud 基于身份的策略示例](#)

截止日期云的政策资源

支持策略资源

是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

Resource JSON 策略元素指定要向其应用操作的一个或多个对象。语句必须包含 Resource 或 NotResource 元素。作为最佳实践，请使用其 [Amazon 资源名称 \(ARN\)](#) 指定资源。对于支持特定资源类型（称为资源级权限）的操作，您可以执行此操作。

对于不支持资源级权限的操作（如列出操作），请使用通配符 (*) 指示语句应用于所有资源。

```
"Resource": "*"
```

要查看 Deadline Cloud 资源类型及其 ARN 的列表，请参阅《服务授权参考》中的 De [AWS adline Cloud 定义的资源](#)。要了解您可以使用哪些操作来指定每种资源的 ARN，请参阅 Deadline Clou [d 定义的 AWS 操作](#)。

要查看 Deadline Cloud 基于身份的策略的示例，请参阅 [Deadline Cloud 基于身份的策略示例](#)

截止日期云的策略条件密钥

支持特定于服务的策略条件密钥 **是**

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

在 Condition 元素 (或 Condition 块) 中，可以指定语句生效的条件。Condition 元素是可选的。您可以创建使用[条件运算符](#) (例如，等于或小于) 的条件表达式，以使策略中的条件与请求中的值相匹配。

如果您在一个语句中指定多个 Condition 元素，或在单个 Condition 元素中指定多个键，则 AWS 使用逻辑 AND 运算评估它们。如果您为单个条件键指定多个值，则使用逻辑 OR 运算来 AWS 评估条件。在授予语句的权限之前必须满足所有的条件。

在指定条件时，您也可以使用占位符变量。例如，只有在使用 IAM 用户名标记 IAM 用户时，您才能为其授予访问资源的权限。有关更多信息，请参阅《IAM 用户指南》中的[IAM policy 元素：变量和标签](#)。

AWS 支持全局条件密钥和特定于服务的条件密钥。要查看所有 AWS 全局条件键，请参阅 IAM 用户指南中的[AWS 全局条件上下文密钥](#)。

要查看 Deadline Cloud 条件密钥列表，请参阅《服务授权参考》中的 [De AWS adline Cloud 条件密钥](#)。要了解可以使用条件键的操作和资源，请参阅 De [AWS adline Cloud 定义的操作](#)。

要查看 Deadline Cloud 基于身份的策略的示例，请参阅。[Deadline Cloud 基于身份的策略示例](#)

截止日期云中的 ACL

支持 ACL **否**

访问控制列表 (ACL) 控制哪些主体 (账户成员、用户或角色) 有权访问资源。ACL 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

带有截止日期云的 ABAC

支持 ABAC (策略中的标签) **是**

基于属性的访问控制 (ABAC) 是一种授权策略，该策略基于属性来定义权限。在中 AWS，这些属性称为标签。您可以将标签附加到 IAM 实体（用户或角色）和许多 AWS 资源。标记实体和资源是 ABAC 的第一步。然后设计 ABAC 策略，以在主体的标签与他们尝试访问的资源标签匹配时允许操作。

ABAC 在快速增长的环境中非常有用，并在策略管理变得繁琐的情况下可以提供帮助。

要基于标签控制访问，您需要使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 条件键在策略的 [条件元素](#) 中提供标签信息。

如果某个服务对于每种资源类型都支持所有这三个条件键，则对于该服务，该值为是。如果某个服务仅对于部分资源类型支持所有这三个条件键，则该值为部分。

有关 ABAC 的更多信息，请参阅《IAM 用户指南》中的 [什么是 ABAC？](#)。要查看设置 ABAC 步骤的教程，请参阅《IAM 用户指南》中的 [使用基于属性的访问权限控制 \(ABAC\)](#)。

在截止日期云中使用临时证书

支持临时凭证 是

当你使用临时证书登录时，有些 AWS 服务不起作用。有关更多信息，包括哪些 AWS 服务适用于临时证书，请参阅 IAM 用户指南中的 [AWS 服务与 IAM 配合使用的信息](#)。

如果您使用除用户名和密码之外的任何方法登录，则 AWS Management Console 使用的是临时证书。例如，当您 AWS 使用公司的单点登录 (SSO) 链接进行访问时，该过程会自动创建临时证书。当您以用户身份登录控制台，然后切换角色时，您还会自动创建临时凭证。有关切换角色的更多信息，请参阅《IAM 用户指南》中的 [切换到角色 \(控制台\)](#)。

您可以使用 AWS CLI 或 AWS API 手动创建临时证书。然后，您可以使用这些临时证书进行访问 AWS。AWS 建议您动态生成临时证书，而不是使用长期访问密钥。有关更多信息，请参阅 [IAM 中的临时安全凭证](#)。

截止日期云的转发访问会话

支持转发访问会话 (FAS) 是

当您使用 IAM 用户或角色在中执行操作时 AWS，您被视为委托人。使用某些服务时，您可能会执行一个操作，然后此操作在其他服务中启动另一个操作。FAS 使用调用委托人的权限以及 AWS 服务向下游服务发出请求的请求。AWS 服务只有当服务收到需要与其他 AWS 服务或资源交互才能完成的请求

时，才会发出 FAS 请求。在这种情况下，您必须具有执行这两个操作的权限。有关发出 FAS 请求时的策略详情，请参阅[转发访问会话](#)。

截止日期云的服务角色

支持服务角色	是
--------	---

服务角色是由一项服务担任、代表您执行操作的 [IAM 角色](#)。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的[创建向 AWS 服务委派权限的角色](#)。

Warning

更改服务角色的权限可能会中断 Deadline Cloud 的功能。仅当 Deadline Cloud 提供相关指导时才编辑服务角色。

截止日期云的服务相关角色

支持服务相关角色	否
----------	---

服务相关角色是一种与服务相关联的 AWS 服务服务角色。服务可以代入代表您执行操作的角色。服务相关角色出现在您的 AWS 账户，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

有关创建或管理服务相关角色的详细信息，请参阅[能够与 IAM 搭配使用的 AWS 服务](#)。在表中查找服务相关角色列中包含 Yes 的表。选择是链接以查看该服务的服务相关角色文档。

Deadline Cloud 基于身份的策略示例

默认情况下，用户和角色无权创建或修改 Deadline Cloud 资源。他们也无法使用 AWS Management Console、AWS Command Line Interface (AWS CLI) 或 AWS API 执行任务。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM policy。管理员随后可以向角色添加 IAM 策略，用户可以代入角色。

要了解如何使用这些示例 JSON 策略文档创建基于 IAM 身份的策略，请参阅 IAM 用户指南中的[创建 IAM 策略](#)。

有关 Deadline Cloud 定义的操作和资源类型（包括每种资源类型的 ARN 格式）的详细信息，请参阅《服务授权参考》中的 [De AWS adline Cloud 的操作、资源和条件密钥](#)。

主题

- [策略最佳实践](#)
- [使用截止日期云控制台](#)
- [向队列提交作业的政策](#)
- [允许创建许可证端点的策略](#)
- [允许监控特定服务器场队列的策略](#)

策略最佳实践

基于身份的策略决定了某人是否可以在您的账户中创建、访问或删除 Deadline Cloud 资源。这些操作可能会使 AWS 账户产生成本。创建或编辑基于身份的策略时，请遵循以下指南和建议：

- 开始使用 AWS 托管策略并转向最低权限权限 — 要开始向用户和工作负载授予权限，请使用为许多常见用例授予权限的 AWS 托管策略。它们在你的版本中可用 AWS 账户。我们建议您通过定义针对您的用例的 AWS 客户托管策略来进一步减少权限。有关更多信息，请参阅《IAM 用户指南》中的 [AWS 托管策略](#) 或 [工作职能的 AWS 托管策略](#)。
- 应用最低权限 – 在使用 IAM policy 设置权限时，请仅授予执行任务所需的权限。为此，您可以定义在特定条件下可以对特定资源执行的操作，也称为最低权限许可。有关使用 IAM 应用权限的更多信息，请参阅《IAM 用户指南》中的 [IAM 中的策略和权限](#)。
- 使用 IAM policy 中的条件进一步限制访问权限 – 您可以向策略添加条件来限制对操作和资源的访问。例如，您可以编写策略条件来指定必须使用 SSL 发送所有请求。如果服务操作是通过特定的方式使用的，则也可以使用条件来授予对服务操作的访问权限 AWS 服务，例如 AWS CloudFormation。有关更多信息，请参阅《IAM 用户指南》中的 [IAM JSON 策略元素：条件](#)。
- 使用 IAM Access Analyzer 验证您的 IAM policy，以确保权限的安全性和功能性 – IAM Access Analyzer 会验证新策略和现有策略，以确保策略符合 IAM policy 语言 (JSON) 和 IAM 最佳实践。IAM Access Analyzer 提供 100 多项策略检查和可操作的建议，以帮助您制定安全且功能性强的策略。有关更多信息，请参阅《IAM 用户指南》中的 [IAM Access Analyzer 策略验证](#)。
- 需要多重身份验证 (MFA)-如果 AWS 账户您的场景需要 IAM 用户或根用户，请启用 MFA 以提高安全性。若要在调用 API 操作时需要 MFA，请将 MFA 条件添加到您的策略中。有关更多信息，请参阅《IAM 用户指南》中的 [配置受 MFA 保护的 API 访问](#)。

有关 IAM 中的最佳实操的更多信息，请参阅《IAM 用户指南》中的 [IAM 中的安全最佳实操](#)。

使用截止日期云控制台

要访问 De AWS adline Cloud 控制台，您必须拥有一组最低权限。这些权限必须允许您列出和查看有关您的 Deadline Cloud 资源的详细信息 AWS 账户。如果创建比必需的最低权限更为严格的基于身份的策略，对于附加了该策略的实体（用户或角色），控制台将无法按预期正常运行。

对于仅调用 AWS CLI 或 AWS API 的用户，您无需为其设置最低控制台权限。相反，只允许访问与其尝试执行的 API 操作相匹配的操作。

为确保用户和角色仍然可以使用 Deadline Cloud 控制台，还需要将 Deadline Cloud *ConsoleAccess* 或 *ReadOnly* AWS 托管策略附加到实体。有关更多信息，请参阅《IAM 用户指南》中的[为用户添加权限](#)。

向队列提交作业的政策

在此示例中，您创建了一个范围缩小策略，该策略授予向特定服务器场中的特定队列提交作业的权限。

```
{
  "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/*"
    }
  ]
}
```

允许创建许可证端点的策略

在此示例中，您将创建一个范围缩小策略，该策略授予创建和管理许可证端点所需的权限。使用此策略为与您的服务器场关联的 VPC 创建许可证终端节点。

```
{
  "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": "*"
}]
}

```

允许监控特定服务器场队列的策略

在此示例中，您创建了一个范围缩小策略，该策略授予监视特定服务器场特定队列中作业的权限。

```

{
  "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 截止日期云的托管策略

AWS 托管策略是由创建和管理的独立策略 AWS。AWS 托管策略旨在为许多常见用例提供权限，以便您可以开始为用户、组和角色分配权限。

请记住，AWS 托管策略可能不会为您的特定用例授予最低权限权限，因为它们可供所有 AWS 客户使用。我们建议通过定义特定于您的使用场景的[客户管理型策略](#)来进一步减少权限。

您无法更改 AWS 托管策略中定义的权限。如果 AWS 更新 AWS 托管策略中定义的权限，则更新会影响该策略所关联的所有委托人身份（用户、组和角色）。AWS 最有可能在启动新的 API 或现有服务可以使用新 AWS 服务的 API 操作时更新 AWS 托管策略。

有关更多信息，请参阅《IAM 用户指南》中的[AWS 托管式策略](#)。

AWS 托管策略：AWSDeadlineCloud-FleetWorker

您可以将AWSDeadlineCloud-FleetWorker策略附加到您的 AWS Identity and Access Management (IAM) 身份。

此策略向该队列中的工作人员授予连接服务并从该服务接收任务所需的权限。

权限详细信息

该策略包含以下权限：

- deadline— 允许校长管理车队中的员工。

有关策略详情的 JSON 列表，请参阅 AWS 托管策略参考指南FleetWorker中的[AWSDeadlineCloud-](#)。

AWS 托管策略：AWSDeadlineCloud-WorkerHost

您可以将 AWSDeadlineCloud-WorkerHost 策略附加到 IAM 身份。

此策略授予最初连接到服务所需的权限。它可以用作亚马逊弹性计算云 (Amazon EC2) 实例配置文件。

权限详细信息

该策略包含以下权限：

- `deadline`— 允许委托人创建工作人员。

有关策略详情的 JSON 列表，请参阅 AWS 托管策略参考指南 WorkerHost 中的 [AWSDeadlineCloud-](#)。

AWS 托管策略：AWSDeadlineCloud-UserAccessFarms

您可以将 `AWSDeadlineCloud-UserAccessFarms` 策略附加到 IAM 身份。

此策略允许用户根据其所属的服务器场及其成员级别访问服务器场数据。

权限详细信息

该策略包含以下权限：

- `deadline`— 允许用户访问服务器场数据。
- `ec2`— 允许用户查看有关 Amazon EC2 实例类型的详细信息。
- `identitystore`— 允许用户查看用户名和组名。

有关策略详情的 JSON 列表，请参阅 AWS 托管策略参考指南中的 [AWSDeadlineCloud-UserAccessFarm s](#)。

AWS 托管策略：AWSDeadlineCloud-UserAccessFleets

您可以将 `AWSDeadlineCloud-UserAccessFleets` 策略附加到 IAM 身份。

此政策允许用户根据其所属的农场及其成员级别访问舰队数据。

权限详细信息

该策略包含以下权限：

- `deadline`— 允许用户访问服务器场数据。
- `ec2`— 允许用户查看有关 Amazon EC2 实例类型的详细信息。
- `identitystore`— 允许用户查看用户名和组名。

有关策略详情的 JSON 列表，请参阅 AWS 托管策略参考指南中的 [AWSDeadlineCloud-UserAccess Fleets](#)。

AWS 托管策略：AWSDeadlineCloud-UserAccessJobs

您可以将 AWSDeadlineCloud-UserAccessJobs 策略附加到 IAM 身份。

此策略允许用户根据其所属的农场及其成员级别访问作业数据。

权限详细信息

该策略包含以下权限：

- `deadline`— 允许用户访问服务器场数据。
- `ec2`— 允许用户查看有关 Amazon EC2 实例类型的详细信息。
- `identitystore`— 允许用户查看用户名和组名。

有关策略详情的 JSON 列表，请参阅 AWSDeadlineCloud AWS 托管策略参考指南中的 [职UserAccess 位](#)。

AWS 托管策略：AWSDeadlineCloud-UserAccessQueues

您可以将 AWSDeadlineCloud-UserAccessQueues 策略附加到 IAM 身份。

此策略允许用户根据其所属服务器场及其成员级别访问队列数据。

权限详细信息

该策略包含以下权限：

- `deadline`— 允许用户访问服务器场数据。
- `ec2`— 允许用户查看有关 Amazon EC2 实例类型的详细信息。
- `identitystore`— 允许用户查看用户名和组名。

有关策略详情的 JSON 列表，请参阅 AWSDeadlineCloud AWS 托管策略参考指南中的 [AWSDeadlineCloud-UserAccessQueues](#)。

截止日期云更新托 AWS 管策略

查看自该服务开始跟踪这些更改以来 Deadline Cloud AWS 托管政策更新的详细信息。要获得有关此页面变更的自动提醒，请在 Deadline Cloud 文档历史记录页面上订阅 RSS 提要。

更改	描述	日期
截止日期云开始跟踪变更	Deadline Cloud 开始跟踪其 AWS 托管政策的变更。	2024 年 4 月 2 日

故障排除 De AWS adline Cloud

使用以下信息来帮助您诊断和修复在使用 Deadline Cloud 和 IAM 时可能遇到的常见问题。

主题

- [我无权在 Deadline Cloud 中执行操作](#)
- [我无权执行 iam : PassRole](#)
- [我想允许我以外的人访问我的 Dead AWS 账户 line Cloud 资源](#)

我无权在 Deadline Cloud 中执行操作

如果您收到错误提示，表明您无权执行某个操作，则您必须更新策略以允许执行该操作。

当 mateojackson IAM 用户尝试使用控制台查看有关虚构 *my-example-widget* 资源的详细信息，但不拥有虚构 `deadline:GetWidget` 权限时，会发生以下示例错误。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
deadline:GetWidget on resource: my-example-widget
```

在此情况下，必须更新 mateojackson 用户的策略，以允许使用 `deadline:GetWidget` 操作访问 *my-example-widget* 资源。

如果您需要帮助，请联系您的 AWS 管理员。您的管理员是提供登录凭证的人。

我无权执行 iam : PassRole

如果您收到错误消息，说您无权执行该 `iam:PassRole` 操作，则必须更新您的策略以允许您将角色传递给 Deadline Cloud。

有些 AWS 服务 允许您将现有角色传递给该服务，而不是创建新的服务角色或服务相关角色。为此，您必须具有将角色传递到服务的权限。

当名为 Deadline Cloud 的 IAM 用户 marymajor 尝试使用控制台在 Deadline Cloud 中执行操作时，会发生以下示例错误。但是，服务必须具有服务角色所授予的权限才可执行此操作。Mary 不具有将角色传递到服务的权限。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

在这种情况下，必须更新 Mary 的策略以允许她执行 iam:PassRole 操作。

如果您需要帮助，请联系您的 AWS 管理员。您的管理员是提供登录凭证的人。

我想允许我以外的人访问我的 Dead AWS 账户 line Cloud 资源

您可以创建一个角色，以便其他账户中的用户或您组织外的人员可以使用该角色来访问您的资源。您可以指定谁值得信赖，可以担任角色。对于支持基于资源的策略或访问控制列表 (ACL) 的服务，您可以使用这些策略向人员授予对您的资源的访问权。

要了解更多信息，请参阅以下内容：

- 要了解 Deadline Cloud 是否支持这些功能，请参阅[截止日期云如何与 IAM 配合使用](#)。
- 要了解如何提供对您拥有的资源的访问权限 AWS 账户，请参阅[IAM 用户指南中的向您拥有 AWS 账户的另一个 IAM 用户提供访问权限](#)。
- 要了解如何向第三方提供对您的资源的访问权限 AWS 账户，请参阅[IAM 用户指南中的向第三方提供访问权限](#)。AWS 账户
- 要了解如何通过联合身份验证提供访问权限，请参阅《IAM 用户指南》中的[为经过外部身份验证的用户 \(联合身份验证 \) 提供访问权限](#)。
- 要了解使用角色和基于资源的策略进行跨账户访问的区别，请参阅[IAM 用户指南中的跨账户资源访问](#)。

合规性验证 Deadline Cloud

要了解是否属于特定合规计划的范围，请参阅AWS 服务“[按合规计划划分的范围](#)”，然后选择您感兴趣的合规计划。AWS 服务 有关一般信息，请参阅[AWS 合规计划AWS](#)。

您可以使用下载第三方审计报告 AWS Artifact。有关更多信息，请参阅中的“[下载报告](#)”中的“[AWS Artifact](#)”。

您在使用 AWS 服务 时的合规责任取决于您的数据的敏感性、贵公司的合规目标以及适用的法律和法规。AWS 提供了以下资源来帮助实现合规性：

- [安全与合规性快速入门指南](#) — 这些部署指南讨论了架构注意事项，并提供了在这些基础上 AWS 部署以安全性和合规性为重点的基准环境的步骤。
- 在 [Amazon Web Services 上构建 HIPAA 安全与合规性](#) — 本白皮书描述了各公司如何使用 AWS 来创建符合 HIPAA 资格的应用程序。

Note

并非所有 AWS 服务 人都符合 HIPAA 资格。有关更多信息，请参阅[符合 HIPAA 要求的服务参考](#)。

- [AWS 合规资源](#) — 此工作簿和指南集可能适用于您的行业和所在地区。
- [AWS 客户合规指南](#) — 从合规角度了解责任共担模式。这些指南总结了保护的最佳实践，AWS 服务 并将指南映射到跨多个框架（包括美国国家标准与技术研究院 (NIST)、支付卡行业安全标准委员会 (PCI) 和国际标准化组织 (ISO)）的安全控制。
- [使用 AWS Config 开发人员指南中的规则评估资源](#) — 该 AWS Config 服务 评估您的资源配置在多大程度上符合内部实践、行业指导方针和法规。
- [AWS Security Hub](#) — 这 AWS 服务 可以全面了解您的安全状态 AWS。Security Hub 通过安全控件评估您的 AWS 资源并检查其是否符合安全行业标准和最佳实践。有关受支持服务及控件的列表，请参阅 [Security Hub 控件参考](#)。
- [Amazon GuardDuty](#) — 它通过监控您的 AWS 账户环境中是否存在可疑和恶意活动，来 AWS 服务 检测您的工作负载、容器和数据面临的潜在威胁。GuardDuty 通过满足某些合规性框架规定的入侵检测要求，可以帮助您满足各种合规性要求，例如 PCI DSS。
- [AWS Audit Manager](#) — 这 AWS 服务 可以帮助您持续审计 AWS 使用情况，从而简化风险管理以及对法规和行业标准的合规性。

韧性在 Deadline Cloud

AWS 全球基础设施是围绕 AWS 区域 可用区构建的。AWS 区域 提供多个物理隔离和隔离的可用区，这些可用区通过低延迟、高吞吐量和高度冗余的网络连接。利用可用区，您可以设计和操作在可用区之间无中断地自动实现失效转移的应用程序和数据库。与传统的单个或多个数据中心基础设施相比，可用区具有更高的可用性、容错性和可扩展性。

有关 AWS 区域 和可用区的更多信息，请参阅[AWS 全球基础设施](#)。

AWS Deadline Cloud 不会备份存储在任务附件 S3 存储桶中的数据。您可以使用任何标准 Amazon S3 备份机制（例如 [S3 版本控制](#) 或 [AWS Backup](#)）启用任务附件数据的备份。

截止日期云中的基础设施安全

作为一项托管服务，De AWS adline Cloud 受到 AWS 全球网络安全的保护。有关 AWS 安全服务以及如何 AWS 保护基础设施的信息，请参阅[AWS 云安全](#)。要使用基础设施安全的最佳实践来设计您的 AWS 环境，请参阅 [AWS security Pillar Well-Architected Fram ework](#) 中的[基础设施保护](#)。

您可以使用 AWS 已发布的 API 调用通过网络访问 Deadline Cloud。客户端必须支持以下内容：

- 传输层安全性协议 (TLS) 我们要求使用 TLS 1.2，建议使用 TLS 1.3。
- 具有完全向前保密 (PFS) 的密码套件，例如 DHE（临时 Diffie-Hellman）或 ECDHE（临时椭圆曲线 Diffie-Hellman）。大多数现代系统（如 Java 7 及更高版本）都支持这些模式。

此外，必须使用访问密钥 ID 和与 IAM 主体关联的秘密访问密钥来对请求进行签名。或者，您可以使用 [AWS Security Token Service](#)（AWS STS）生成临时安全凭证来对请求进行签名。

Deadline Cloud 不支持使用 AWS PrivateLink 虚拟私有云 (VPC) 端点策略。它使用 AWS PrivateLink 默认策略，即授予对终端节点的完全访问权限。有关更多信息，请参阅[AWS PrivateLink 用户指南中的默认终端节点策略](#)。

截止日期云中的配置和漏洞分析

AWS 处理基本的安全任务，例如客户机操作系统 (OS) 和数据库修补、防火墙配置和灾难恢复。这些流程已通过相应第三方审核和认证。有关更多详细信息，请参阅以下资源：

- [责任共担模式](#)
- [Amazon Web Services：安全过程概述](#)（白皮书）

AWS Deadline Cloud 管理服务管理或客户管理的车队上的任务：

- 对于服务管理的舰队，Deadline Cloud 管理客户机操作系统。
- 对于客户管理的车队，您负责管理操作系统。

有关 De AWS adline Cloud 的配置和漏洞分析的更多信息，请参阅

- [截止日期云的安全最佳实践](#)

防止跨服务混淆座席

混淆代理问题是一个安全性问题，即不具有操作执行权限的实体可能会迫使具有更高权限的实体执行该操作。在中 AWS，跨服务模仿可能会导致混乱的副手问题。一个服务（呼叫服务）调用另一项服务（所谓的“服务”）时，可能会发生跨服务模拟。可以操纵调用服务，使用其权限以在其他情况下该服务不应有访问权限的方式对另一个客户的资源进行操作。为防止这种情况，AWS 提供可帮助您保护所有服务的数据的工具，而这些服务中的服务主体有权限访问账户中的资源。

我们建议在资源策略中使用 `aws:SourceArn` 和 `aws:SourceAccount` 全局条件上下文密钥来限制为资源 AWS Deadline Cloud 提供其他服务的权限。如果您只希望将一个资源与跨服务访问相关联，请使用 `aws:SourceArn`。如果您想允许该账户中的任何资源与跨服务使用操作相关联，请使用 `aws:SourceAccount`。

防止混淆代理问题最有效的方法是使用具有资源完整 Amazon Resource Name (ARN) 的 `aws:SourceArn` 全局条件上下文键。如果不知道资源的完整 ARN，或者正在指定多个资源，请针对 ARN 未知部分使用带有通配符字符 (*) 的 `aws:SourceArn` 全局上下文条件键。例如，`arn:aws:deadline:*:123456789012:*`。

如果 `aws:SourceArn` 值不包含账户 ID，例如 Amazon S3 存储桶 ARN，您必须使用两个全局条件上下文键来限制权限。

以下示例显示了如何在中使用 `aws:SourceArn` 和 `aws:SourceAccount` 全局条件上下文键 Deadline Cloud 来防止出现混淆的副手问题。

```
{
  "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"
  }
}
}
```

AWS Deadline Cloud 使用接口端点进行访问 (AWS PrivateLink)

您可以使用 AWS PrivateLink 在您的 VPC 和之间创建私有连接 AWS Deadline Cloud。您可以像在 VPC 中 Deadline Cloud 一样进行访问，无需使用互联网网关、NAT 设备、VPN 连接或 AWS Direct Connect 连接。VPC 中的实例不需要公有 IP 地址即可访问 Deadline Cloud。

您可以通过创建由 AWS PrivateLink 提供支持的接口端点来建立此私有连接。我们将在您为接口端点启用的每个子网中创建一个端点网络接口。这些是请求者托管的网络接口，用作发往 Deadline Cloud 的流量的入口点。

有关更多信息，请参阅《AWS PrivateLink 指南》中的[通过 AWS PrivateLink 访问 AWS 服务](#)。

的注意事项 Deadline Cloud

在为设置接口终端节点之前 Deadline Cloud，请参阅 AWS PrivateLink 指南中的[使用接口 VPC 终端节点访问 AWS 服务](#)。

Deadline Cloud 支持通过接口端点调用其所有 API 操作。

默认情况下，允许通过接口终端节点进行完全访问。Deadline Cloud 或者，您可以将安全组与终端节点网络接口相关联，以控制 Deadline Cloud 通过该接口终端节点的流量。

Deadline Cloud 不支持 VPC 终端节点策略。有关更多信息，请参阅 AWS PrivateLink 指南中的[使用端点策略控制对 VPC 端点的访问权限](#)。

Deadline Cloud 端点

Deadline Cloud 使用两个终端节点访问服务 AWS PrivateLink。

工作人员使用 `com.amazonaws.region.deadline.scheduling` 端点从队列中获取任务、向其 Deadline Cloud 报告进度以及将任务输出发送回去。如果您使用的是客户管理的队列，则调度终端节点

是您唯一需要创建的终端节点，除非您使用的是管理操作。例如，如果一个任务创建了更多作业，则需要启用管理端点才能调用该CreateJob操作。

Deadline Cloud 监视器使用`com.amazonaws.region.deadline.management`来管理服务器场中的资源，例如创建和修改队列和队列或获取作业、步骤和任务的列表。

Deadline Cloud 还需要以下 AWS 服务端点的终端节点：

- Deadline Cloud 用于 AWS STS 对工作人员进行身份验证，以便他们可以访问工作资产。有关更多信息 AWS STS，请参阅《AWS Identity and Access Management 用户指南》中的 [IAM 中的临时安全证书](#)。
- 如果您在没有互联网连接的子网中设置客户管理的队列，则必须为 Amazon L CloudWatch ogs 创建 VPC 终端节点，以便工作人员可以写入日志。有关更多信息，请参阅[使用进行监控 CloudWatch](#)。
- 如果您使用任务附件，则必须为亚马逊简单存储服务 (Amazon S3) Simple Storage S3 创建 VPC 终端节点，以便工作人员可以访问附件。有关更多信息，请参阅[中的 Job 附件 Deadline Cloud](#)。

为创建终端节点 Deadline Cloud

您可以创建用于 Deadline Cloud 使用 Amazon VPC 控制台或 AWS Command Line Interface (AWS CLI) 的接口终端节点。有关更多信息，请参阅《AWS PrivateLink 指南》中的[创建接口端点](#)。

Deadline Cloud 使用以下服务名称创建管理和调度端点。将`##`替换为您部署 AWS 区域的地点 Deadline Cloud。

```
com.amazonaws.region.deadline.management
```

```
com.amazonaws.region.deadline.scheduling
```

如果您为接口终端节点启用私有 DNS，则 Deadline Cloud 可以使用其默认区域 DNS 名称向发出 API 请求。例如，`worker.deadline.us-east-1.amazonaws.com`用于工作人员操作或`management.deadline.us-east-1.amazonaws.com`所有其他操作。

您还必须 AWS STS 使用以下服务名称创建终端节点：

```
com.amazonaws.region.sts
```

如果您的客户管理的队列位于没有 Internet 连接的子网上，则必须使用以下服务名称创建 L CloudWatch ogs 端点：

```
com.amazonaws.region.logs
```

如果您使用任务附件传输文件，则必须使用以下服务名称创建 Amazon S3 终端节点：

```
com.amazonaws.region.s3
```

截止日期云的安全最佳实践

AWS Deadline Cloud (Deadline Cloud) 提供了许多安全功能，供您在制定和实施自己的安全策略时考虑。以下最佳实践是一般指导原则，并不代表完整安全解决方案。这些最佳实践可能不适合环境或不满足环境要求，请将其视为有用的考虑因素而不是惯例。

Note

有关许多安全主题的重要性的更多信息，请参阅[责任共担模型](#)。

数据保护

出于数据保护目的，我们建议您保护 AWS 账户 凭证并使用 AWS Identity and Access Management (IAM) 设置个人账户。这样，每个用户只获得履行其工作职责所需的权限。我们还建议您通过以下方式保护数据：

- 对每个账户使用多重身份验证 (MFA)。
- 使用 SSL/TLS 与资源通信。AWS 我们要求使用 TLS 1.2，建议使用 TLS 1.3。
- 使用设置 API 和用户活动日志 AWS CloudTrail。
- 使用 AWS 加密解决方案以及其中的所有默认安全控件 AWS 服务。
- 使用高级托管安全服务 (例如 Amazon Macie)，它有助于发现和保护存储在 Amazon Simple Storage Service (Amazon S3) 中的个人数据。
- 如果在通过命令行界面或 API 访问 AWS 时需要经过 FIPS 140-2 验证的加密模块，请使用 FIPS 端点。有关可用的 FIPS 端点的更多信息，请参阅[美国联邦信息处理标准 \(FIPS \) 第 140-2 版](#)。

我们强烈建议您切勿将敏感的可识别信息 (例如您客户的账号) 放入自由格式字段 (例如名称字段)。这包括你使用控制台、API 或 SD AWS K AWS 服务使用 Deadline Cloud 或其他云时。AWS CLI您输入到Deadline Cloud或其他服务中的任何数据都可能被提取以包含在诊断日志中。当您向外部服务器提供 URL 时，请勿在 URL 中包含凭证信息来验证您对该服务器的请求。

AWS Identity and Access Management 权限

使用用户、AWS Identity and Access Management (IAM) 角色并通过向用户授予最低权限来管理对 AWS 资源的访问权限。制定用于创建、分发、轮换和撤消 AWS 访问凭证的凭证管理策略和程序。有关更多信息，请参阅《IAM 用户指南》中的 [IAM 最佳实操](#)。

以用户和群组的身份运行作业

在 Deadline Cloud 中使用队列功能时，最佳做法是指定操作系统 (OS) 用户及其主组，以便操作系统用户对队列的作业拥有最低权限权限。

当您指定“以用户身份运行”（和组）时，提交到队列的作业的所有进程都将使用该操作系统用户运行，并将继承该用户的关联操作系统权限。

队列和队列配置相结合，可以建立安全态势。在队列方面，可以指定“Job 以用户身份运行”和 IAM 角色来使用队列任务的操作系统和 AWS 权限。队列定义了基础架构（工作主机、网络、已安装的共享存储），当这些基础架构与特定队列关联时，将在队列中运行作业。工作服务器主机上的可用数据需要一个或多个关联队列中的作业访问。指定用户或组有助于保护作业中的数据免受其他队列、其他已安装的软件或其他有权访问工作主机的用户的侵害。当队列没有用户时，它会以代理用户身份运行，代理用户可以模仿 (sudo) 任何队列用户。这样，没有用户的队列可以将权限升级到另一个队列。

联网

为防止流量被拦截或重定向，必须确保网络流量的路由方式和位置安全。

我们建议您通过以下方式保护您的网络环境：

- 保护亚马逊虚拟私有云 (Amazon VPC) 子网路由表，以控制 IP 层流量的路由方式。
- 如果您在服务器场或工作站设置中使用亚马逊 Route 53 (Route 53) 作为 DNS 提供商，请安全访问 Route 53 API。
- 如果您使用本地工作站或其他数据中心 AWS 等外部连接到 Deadline Cloud，请保护任何本地网络基础设施。这包括路由器、交换机和其他网络设备上的 DNS 服务器和路由表。

工作和工作数据

Deadline Cloud 作业在工作主机的会话中运行。每个会话在工作主机上运行一个或多个进程，这通常需要您输入数据才能生成输出。

为了保护这些数据，您可以为操作系统用户配置队列。工作器代理使用队列操作系统用户来运行会话子进程。这些子进程继承队列操作系统用户的权限。

我们建议您遵循最佳实践，以保护对这些子流程访问的数据的访问。有关更多信息，请参阅[责任共担模式](#)。

农场结构

您可以通过多种方式安排 Deadline Cloud 舰队和队列。但是，某些安排会涉及安全问题。

服务器场具有最安全的边界之一，因为它无法与其他服务器场共享 Deadline Cloud 资源，包括队列、队列和存储配置文件。但是，您可以在服务器场内共享外部 AWS 资源，这会影响安全边界。

您还可以使用适当的配置在同一服务器场内的队列之间建立安全边界。

按照以下最佳实践在同一个服务器场中创建安全队列：

- 仅将队列与相同安全边界内的队列关联。请注意以下几点：
 - 在工作主机上运行作业后，数据可能会留在后面，例如在临时目录或队列用户的主目录中。
 - 无论您将任务提交到哪个队列，都由同一个操作系统用户在服务拥有的队列工作人员主机上运行所有作业。
 - 作业可能会使进程在工作主机上运行，从而使来自其他队列的作业可以观察其他正在运行的进程。
- 确保只有处于相同安全边界内的队列才能共享用于存放任务附件的 Amazon S3 存储桶。
- 确保只有相同安全边界内的队列共享操作系统用户。
- 将集成到服务器场中的任何其他 AWS 资源保护到边界。

Job 附件队列

Job 附件与队列相关联，该队列使用您的 Amazon S3 存储桶。

- Job 附件对 Amazon S3 存储桶中的根前缀进行写入和读取。您可以在 CreateQueue API 调用中指定此根前缀。
- 存储桶有一个对应的 Queue Role，它指定了向队列用户授予存储桶访问权限的角色和根前缀。创建队列时，您可以在任务附件存储桶和根前缀旁边指定 A Queue Role mazon 资源名称 (ARN)。
- 对 AssumeQueueRoleForReadAssumeQueueRoleForUser、和 AssumeQueueRoleForWorker API 操作的授权调用会返回一组临时安全证书 Queue Role。

如果您创建队列并重复使用 Amazon S3 存储桶和根前缀，则存在信息被泄露给未授权方的风险。例如，queueA 和 queueB 共享相同的存储桶和根前缀。在安全的工作流程中，ArtistA 可以访问 QueueA，但不能访问 queueB。但是，当多个队列共享一个存储桶时，ArtistA 可以访问 QueueB 数据中的数据，因为它使用的存储桶和根前缀与 queueA 相同。

控制台设置的队列在默认情况下是安全的。除非队列属于共同安全边界，否则请确保队列具有 Amazon S3 存储桶和根前缀的独特组合。

要隔离队列，必须将配置 Queue Role 为仅允许队列访问存储桶和根前缀。在以下示例中，用您的资源特定信息替换每个###。

```
{
  "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/*"
    }
  ]
}
```

您还必须为该角色设置信任策略。在以下示例中，用您的资源特定信息替换###文本。

```
{
  "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"
      }
    }
  }
]
}

```

定制软件 Amazon S3 存储桶

您可以在中添加以下语句Queue Role以访问您的 Amazon S3 存储桶中的自定义软件。在以下示例中，将 *SOFTWARE_BUCKET_NAME* ##### *S3* ###的名称。

```

"Statement": [
  {
    "Action": [
      "s3:GetObject",
      "s3:ListBucket"
    ],
    "Effect": "Allow",
    "Resource": [
      "arn:aws:s3:::SOFTWARE_BUCKET_NAME",
      "arn:aws:s3:::SOFTWARE_BUCKET_NAME/*"
    ]
  }
]
}

```

]

有关 Amazon S3 安全最佳实践的更多信息，请参阅《[亚马逊简单存储服务用户指南](#)》中的 [Amazon S3 安全最佳实践](#)。

工作人员主机

保护工作人员主机，以帮助确保每个用户只能为其分配的角色执行操作。

我们建议采用以下最佳做法来保护工作主机：

- 除非提交给这些队列的任务在相同的安全边界内，否则不要对多个队列使用相同的 `jobRunAsUser` 值。
- 不要将队列设置 `jobRunAsUser` 为工作代理运行的操作系统用户的姓名。
- 向队列用户授予目标队列工作负载所需的最低权限操作系统权限。确保他们没有工作代理程序文件或其他共享软件的文件系统写入权限。
- 确保只有 Administrator 有 root 用户开启 Linux 且拥有者账户 Windows 拥有并可以修改工作代理程序文件。
- 在 Linux 工作服务器主机上，可以考虑在中配置一个 `umask` 替代项 `/etc/sudoers`，允许工作器代理用户以队列用户身份启动进程。此配置有助于确保其他用户无法访问写入队列的文件。
- 向受信任的个人授予对工作人员主机的最低权限访问权限。
- 限制对本地 DNS 覆盖配置文件（开 `/etc/hosts` 启 Linux 和 `C:\Windows\system32\etc\hosts` 开 Windows 启）以及工作站和工作主机操作系统上的路由表的权限。
- 限制工作站和工作主机操作系统上的 DNS 配置权限。
- 定期修补操作系统和所有已安装的软件。这种方法包括专门用于 Deadline Cloud 的软件，例如提交者、适配器、工作人员代理、OpenJD 包等。
- 为 Windows 队列使用强密码 `jobRunAsUser`。
- 定期轮换队列的密码 `jobRunAsUser`。
- 确保对 Windows 密码密钥的访问权限最低，并删除未使用的密钥。
- 不要向队列 `jobRunAsUser` 授予将来运行的计划命令的权限：
 - 开启 Linux，拒绝这些账户访问 `cron` 和 `at`。
 - 开启 Windows，拒绝这些账户访问 Windows 任务计划程序。

Note

有关定期修补操作系统和已安装软件的重要性的更多信息，请参阅[责任共担模型](#)。

工作站

保护能够访问 Deadline Cloud 的工作站非常重要。这种方法有助于确保你提交给 Deadline Cloud 的任何任务都无法运行向你 AWS 账户计费的任意工作负载。

我们建议采用以下最佳做法来保护艺术家工作站的安全。有关更多信息，请参阅[责任共担模式](#)。

- 保护所有提供访问权限的永久凭证，包括 Deadline AWS e Cloud。有关更多信息，请参阅《IAM 用户指南》中的[管理 IAM 用户的访问密钥](#)。
- 仅安装可信、安全的软件。
- 要求用户与身份提供商联合使用临时证书 AWS 进行访问。
- 对 Deadline Cloud 提交者程序文件使用安全权限以防止篡改。
- 向受信任的个人授予访问艺术家工作站的最低权限。
- 仅使用您通过 Deadline Cloud Monitor 获得的提交者和适配器。
- 限制工作站/etc/hosts和工作主机操作系统的权限和路由表。
- 将权限限制/etc/resolv.conf在工作站和工作主机操作系统上。
- 定期修补操作系统和所有已安装的软件。这种方法包括专门用于 Deadline Cloud 的软件，例如提交者、适配器、工作人员代理、OpenJD包等。

监控 AWS 截止日期云

监控是维护 Deadline Cloud (De AWS adline Cloud) 和您的 AWS 解决方案的可靠性、可用性和性能的重要组成部分。从 AWS 解决方案的所有部分收集监控数据，以便在出现多点故障时可以更轻松地进行调试。在开始监控 Deadline Cloud 之前，您应该创建一个包含以下问题的答案的监控计划：

- 监控目的是什么？
- 您将监控哪些资源？
- 监控这些资源的频率如何？
- 您将使用哪些监控工具？
- 谁负责执行监控任务？
- 出现错误时应通知谁？

AWS 和 Deadline Cloud 提供了可用于监控资源和应对潜在事件的工具。其中一些工具可以为您进行监控，有些工具需要手动干预。您应该尽可能自动执行监控任务。

- Amazon 会实时 CloudWatch 监控您的 AWS 资源和您运行 AWS 的应用程序。您可以收集和跟踪指标，创建自定义的控制平面，以及设置警报以在指定的指标达到您指定的阈值时通知您或采取措施。例如，您可以 CloudWatch 跟踪您的 Amazon EC2 实例的 CPU 使用率或其他指标，并在需要时自动启动新实例。有关更多信息，请参阅 [Amazon CloudWatch 用户指南](#)。

截止日期云有三个 CloudWatch 指标。

- Amazon CloudWatch Logs 允许您监控、存储和访问来自 Amazon EC2 实例和其他来源的日志文件。CloudTrail CloudWatch 日志可以监视日志文件中的信息，并在达到特定阈值时通知您。您还可以在高持久性存储中检索您的日志数据。有关更多信息，请参阅 [Amazon CloudWatch 日志用户指南](#)。
- Amazon EventBridge 可用于自动化您的 AWS 服务，并自动响应系统事件，例如应用程序可用性问题或资源更改。来自 AWS 服务的事件几乎实时 EventBridge 地传送到。您可以编写简单的规则来指示您关注的事件，并指示要在事件匹配规则时执行的自动化操作。有关更多信息，请参阅 [Amazon EventBridge 用户指南](#)。
- AWS CloudTrail 捕获由您的账户或代表您的 AWS 账户进行的 API 调用和相关事件，并将日志文件传输到您指定的 Amazon S3 存储桶。您可以识别哪些用户和帐户拨打了电话 AWS、发出呼叫的源 IP 地址以及呼叫发生的时间。有关更多信息，请参阅 [《AWS CloudTrail 用户指南》](#)。

主题

- [使用记录通话 CloudTrail](#)
- [使用监控 CloudWatch](#)
- [对 EventBridge 事件采取行动](#)

使用记录通话 CloudTrail

AWS Deadline Cloud 与 AWS CloudTrail 一项服务集成，该服务提供用户、角色或 Deadline Cloud AWS 服务中执行的操作的记录。CloudTrail 将截止日期云的所有 API 调用捕获为事件。捕获的调用包括来自 Deadline Cloud 控制台的调用以及对截止日期云 API 操作的代码调用。

如果您创建跟踪，则可以允许将 CloudTrail 事件持续传输到 Amazon S3 存储桶，包括 Deadline Cloud 的事件。如果您未配置跟踪，您仍然可以在 CloudTrail 控制台的“事件历史记录”中查看最新的事件。使用收集的信息 CloudTrail，您可以确定向 Deadline Cloud 发出的请求、发出请求的 IP 地址、谁发出了请求、何时发出请求以及其他详细信息。

要了解更多信息 CloudTrail，请参阅[AWS CloudTrail 用户指南](#)。

截止日期云中的信息 CloudTrail

CloudTrail 在您创建账户 AWS 账户时已在您的账户上启用。当活动在 Deadline Cloud 中发生时，该活动会与其他 CloudTrail AWS 服务事件一起记录在事件历史记录中。您可以在中查看、搜索和下载最近发生的事件 AWS 账户。有关更多信息，请参阅[使用事件历史记录查看 CloudTrail 事件](#)。

CloudTrail 还会记录用户登录 Deadline Cloud 监控器并接收 AWS 凭据时的事件。当用户登录时，会出现一个包含来源 `signin.amazonaws.com` 和名称 `CloudTrail` 的事件 `UserAuthentication`。当登录用户获得来自来源 `sts.amazonaws.com` 和姓名的 AWS 凭证时，还会发生第二个事件。AssumeRole 用户的 ID 记录在角色会话名称中的第二个事件中。

要持续记录您的事件 AWS 账户，包括 Deadline Cloud 的事件，请创建跟踪。跟踪允许 CloudTrail 将日志文件传输到 Amazon S3 存储桶。预设情况下，在控制台中创建跟踪记录时，此跟踪记录应用于所有 AWS 区域。跟踪记录 AWS 分区中所有区域的事件，并将日志文件传送到您指定的 Amazon S3 存储桶。此外，您可以配置其他 AWS 服务，以进一步分析和处理 CloudTrail 日志中收集的事件数据。

有关更多信息，请参阅下列内容：

[创建跟踪记录概述](#)

[CloudTrail 支持的服务和集成](#)

[配置 Amazon SNS 通知 CloudTrail](#)

[接收来自多个区域的 CloudTrail 日志文件](#)

[接收来自多个账户的 CloudTrail 日志文件](#)

Deadline Cloud 支持将以下操作作为事件 CloudTrail 记录在日志文件中：

- [associate-member-to-farm](#)
- [associate-member-to-fleet](#)
- [associate-member-to-job](#)
- [associate-member-to-queue](#)
- [assume-fleet-role-for-读](#)
- [assume-fleet-role-for-工人](#)
- [assume-queue-role-for-读](#)
- [assume-queue-role-for-用户](#)
- [assume-queue-role-for-工人](#)
- [创建预算](#)
- [创建农场](#)
- [create-fleet](#)
- [create-license-endpoint](#)
- [创建监视器](#)
- [创建队列](#)
- [create-queue-environment](#)
- [create-queue-fleet-association](#)
- [create-storage-profile](#)
- [创建工作者](#)
- [删除预算](#)
- [删除农场](#)
- [delete-fleet](#)
- [delete-license-endpoint](#)
- [delete-metered-product](#)

- [删除监视器](#)
- [删除队列](#)
- [delete-queue-environment](#)
- [delete-queue-fleet-association](#)
- [delete-storage-profile](#)
- [删除工作人员](#)
- [disassociate-member-from-farm](#)
- [disassociate-member-from-fleet](#)
- [disassociate-member-from-job](#)
- [disassociate-member-from-queue](#)
- [get-application-version](#)
- [获取预算](#)
- [get-farm](#)
- [get-feature-map](#)
- [get-fleet](#)
- [get-license-endpoint](#)
- [获取监视器](#)
- [获取队列](#)
- [get-queue-environment](#)
- [get-queue-fleet-association](#)
- [get-sessions-statistics-aggregation](#)
- [get-storage-profile](#)
- [get-storage-profile-for-队列](#)
- [list-available-metered-products](#)
- [清单预算](#)
- [list-farm-members](#)
- [列出农场](#)
- [list-fleet-members](#)
- [列表舰队](#)

- [list-job-members](#)
- [list-license-endpoints](#)
- [list-metered-products](#)
- [列表监视器](#)
- [list-queue-environments](#)
- [list-queue-fleet-associations](#)
- [list-queue-members](#)
- [列表队列](#)
- [list-storage-profiles](#)
- [list-storage-profiles-for-队列](#)
- [list-tags-for-resource](#)
- [put-metered-product](#)
- [start-sessions-statistics-aggregation](#)
- [tag-resource](#)
- [untag-resource](#)
- [更新预算](#)
- [更新农场](#)
- [更新舰队](#)
- [更新监视器](#)
- [更新队列](#)
- [update-queue-environment](#)
- [update-queue-fleet-association](#)
- [update-storage-profile](#)
- [更新工作者](#)

每个事件或日记账条目都包含有关生成请求的人员信息。身份信息有助于您确定以下内容：

- 请求是使用根证书还是 AWS Identity and Access Management (IAM) 用户凭证发出。
- 请求是使用角色还是联合用户的临时安全凭证发出的。
- 请求是否由其它 服务发出。

有关更多信息，请参阅[CloudTrail用户身份元素](#)。

了解截止日期云日志文件条目

跟踪是一种配置，允许将事件作为日志文件传输到您指定的 Amazon S3 存储桶。CloudTrail 日志文件包含一个或多个日志条目。事件代表来自任何来源的单个请求，包括有关请求的操作、操作的日期和时间、请求参数等的信息。CloudTrail 日志文件不是公共 API 调用的有序堆栈跟踪，因此它们不会按任何特定的顺序出现。

以下 JSON 示例显示了通过调用 **CreateFarm** API 生成的日志：

```
{
  "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",
}
```

该示例显示了 AWS 区域、IP 地址以及可以帮助您识别事件的其他 displayName “kmsKeyArn”（例如 “” 和 “”）。requestParameters

使用监控 CloudWatch

Amazon CloudWatch (CloudWatch) 收集原始数据并将其处理成可读的近乎实时的指标。您可以通过 <https://console.aws.amazon.com/cloudwatch/> 打开 CloudWatch 控制台，查看和筛选 Deadline Cloud 指标。

- 在 Deadline Cloud 客户管理的队列中，CloudWatch 向您发送两个指标 UnhealthyWorkerCount 和 RecommendedFleetSize：
- 这些指标的命名空间是 AWS/DeadlineCloud。
- 您可以使用维 farmID 度和 fleetID 来筛选指标。
- 两个指标都使用该单位 count。

这些统计数据会保存 15 个月，因此您可以访问历史信息，从而更好地了解 Web 应用程序或服务的性能。还可以设置特定阈值监视警报，在达到对应阈值时发送通知或采取行动。有关更多信息，请参阅 [Amazon CloudWatch 用户指南](#)。

Deadline Cloud 有两种日志：任务日志和工作人员日志。任务日志是指将执行日志作为脚本运行或以 DCC 运行的形式运行。任务日志可能会显示诸如资源加载、图块渲染或未找到纹理之类的事件。

工作器日志显示工作器代理进程。这些可能包括诸如工作器代理何时启动、注册自身、报告进度、加载配置或完成任务之类的事情。

对于 Deadline Cloud，工作人员将这些日志上传到 CloudWatch 日志。默认情况下，日志永不过期。如果任务输出了大量数据，则可能会产生额外的成本。有关更多信息，请参阅 [Amazon CloudWatch 定价](#)。

您可以调整每个日志组的保留策略。较短的保留期会删除旧日志，并有助于降低存储成本。要保留日志，您可以在删除日志之前将其存档到 Amazon 简单存储服务。有关更多信息，请参阅 [亚马逊 CloudWatch 用户指南中的使用控制台将日志数据导出到 Amazon S3](#)。

Note

CloudWatch 日志读取受以下限制 AWS。如果您计划招募许多艺术家，我们建议您联系 AWS 客户支持并申请增加 GetLogEvents 配额 CloudWatch。此外，我们建议您在不调试时关闭日志跟踪门户。

有关更多信息，请参阅 Amazon CloudWatch 用户指南中的 [CloudWatch 日志配额](#)。

对 EventBridge 事件采取行动

Deadline Cloud EventBridge 向亚马逊发送事件，以通知您服务状态的变化。您可以使用 EventBridge 和这些事件来编写规则，以便在舰队发生变化时采取行动，例如通知您。有关更多信息，请参阅 [什么是亚马逊 EventBridge](#)

舰队规模建议变更

当您将队列配置为使用基于事件的自动缩放时，Deadline Cloud 会发送可用于管理队列的事件。这些事件中的每一个都包含有关舰队当前规模和请求规模的信息。有关使用 EventBridge 事件和示例 Lambda 函数来处理事件的示例，请参阅 [使用 Deadline Cloud 规模推荐功能自动扩展您的 Amazon EC2 机群](#)

发生以下情况时，将发送舰队规模建议更改事件：

- 当建议的舰队规模发生变化 `oldFleetSize` 并且与之不同时 `newFleetSize`。

- 当服务检测到实际舰队规模与建议的舰队规模不匹配时。您可以从[GetFleet](#)操作响应workerCount中获取实际的舰队规模。当活跃的 Amazon EC2 实例未能注册为 Deadline Cloud 工作程序时，可能会发生这种情况。

该活动采用以下格式：

```
{
  "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,
  }
}
```

以下字段定义了事件模式：

"source": "aws.deadline"

标识此事件的来源是 Deadline Cloud。

"detail-type": "Fleet Size Recommendation Change"

识别事件类型。

"detail": { }

提供有关建议对舰队规模进行更改的信息。

"farmId": "farm-12345678900000000000000000000000"

包含舰队的服务器场的标识符。

"fleetId": "fleet-12345678900000000000000000000000"

需要更改规模的舰队的标识符。

```
"oldFleetSize": 1
```

舰队的当前规模。

```
"newFleetSize": 5
```

建议的新舰队规模。

的配额 Deadline Cloud

AWS Deadline Cloud 提供可用于处理作业的资源，例如农场、队列和队列。在您创建时 AWS 账户，我们会为每个资源设置默认配额 AWS 区域。

Service Quotas 是一个中心位置，您可以在其中查看和管理您的配额 AWS 服务。您也可以申请增加您使用的许多资源的配额。

要查看的配额 Deadline Cloud，请打开 [Service Quotas 控制台](#)。在导航窗格中，选择 AWS 服务，然后选择 Deadline Cloud。

要请求提高限额，请参阅《服务限额用户指南》中的[请求提高限额](#)。如果 Service Quotas 中尚无配额，请使用[服务配额增加表格](#)。

使用创建 AWS 截止日期云资源 AWS CloudFormation

AWS Deadline Cloud 与 AWS CloudFormation 一项服务集成，可帮助您对 AWS 资源进行建模和设置，从而减少创建和管理资源和基础设施所花费的时间。您可以创建一个描述所需的所有 AWS 资源（例如服务器场、队列和队列）的模板，并为您预 AWS CloudFormation 置和配置这些资源。

使用时 AWS CloudFormation，您可以重复使用模板来一致且重复地设置 Deadline Cloud 资源。只需描述一次您的资源，然后在多个 AWS 账户 区域中一遍又一遍地配置相同的资源。

截止日期云和 AWS CloudFormation 模板

要为 Deadline Cloud 和相关服务配置和配置资源，您必须了解[AWS CloudFormation 模板](#)。模板是 JSON 或 YAML 格式的文本文件。这些模板描述了您要在 AWS CloudFormation 堆栈中配置的资源。如果你不熟悉 JSON 或 YAML，可以使用 D AWS CloudFormation esigner 来帮助你开始使用 AWS CloudFormation 模板。有关更多信息，请参阅《AWS CloudFormation 用户指南》中的[什么是 AWS CloudFormation Designer ?](#)。

Deadline Cloud 支持在中 AWS CloudFormation 创建农场、队列和队列。有关更多信息，包括用于农场、队列和队列的 JSON 和 YAML 模板示例，请参阅 AWS CloudFormation 用户指南中的 [Dead AWS line Cloud](#)。

了解更多关于 AWS CloudFormation

要了解更多信息 AWS CloudFormation，请参阅以下资源：

- [AWS CloudFormation](#)
- [AWS CloudFormation 用户指南](#)
- [AWS CloudFormation API 引用](#)
- [AWS CloudFormation 命令行界面用户指南](#)

Deadline Cloud 用户指南的文档历史记录

下表描述了每个版本的 De AWS adline Cloud 用户指南中的重要更改。

变更	说明	日期
初始版本	这是 Deadline Cloud 用户指南的初始版本。	2024年4月2日

AWS 词汇表

有关最新 AWS 术语，请参阅《AWS 词汇表 参考资料》中的[AWS 词汇表](#)。

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。