在上实施基础架构即产品 (IaP) AWS

AWS 规范性指导

AWS 规范性指导: 在上实施基础架构即产品 (IaP) AWS

Table of Contents

在上实现 iAPAWS

Kirsten Kissmeyer，Amazon Web Services (AWS)

2023 年 1 月（文件历史记录）

本指南探讨了将AWS基础架构作为产品 (iAP) 进行管理的方法。IaP 提供的抽象和控制水平高于基础架构即代码 (iaC)，但使用 iaC 方法来实现其目标。该指南还探讨了AWS 服务用于管理 iAP 的工具，并重点介绍了每种工具如何支持您的基础架构管理目标。本指南中的信息基于从一家大型金融部门公司的AWS Service Catalog赋能计划中吸取的经验教训。

本指南适用于想要开发功能性AWS Cloud基础架构服务的用户，这些服务可以根据需要轻松分配和授权给不同的组织用户、业务部门和第三方。

为什么要将基础设施作为产品进行管理？

将基础架构资源作为产品进行管理的好处是，您可以将消费者能力打包为一组具有标准化定义和配置的资源。产品为组织提供了一种便捷的方法来管理和控制AWS权能的分配和使用方式。产品可能仅限于指定的组织单位 (OU) 或需要这些功能的个人。也可以将产品限制AWS 区域为特定产品。

产品配置模型还允许您从中心位置封装和更新产品的定义。然后，您可以一次性或按计划分发产品更新，因为产品的实施会随着时间的推移而发生变化。

有针对性的业务成果

Organizations 一直在寻找更好的方法来管理和配置其AWS基础架构。你的目标可能包括：

- 实现高度的灵活性、可靠性、容错和集中控制，其中单点配置可满足不断变化的内部和外部标准的要求。
- 一种低接触或按键式机制，用于集中分发基础架构，同时允许特定团队或个人在需要时自助访问。
- 能够为内部员工、客户账户和合作伙伴 OU 账户提供AWS基础设施和服务。您可能还需要控制哪些 OU 或组织有权访问特定区域中的特定基础架构组件。
- 如果您使用第三方工具（例如 ServiceNow）或自定义工具来管理访问和配置企业资产和基础设施的请求，则可以轻松地将您的AWS基础架构与这些工具集成。
- 能够同时为数十甚至数百个目标账户配置AWS基础架构。
- Support 调配多个AWS资源以提供单一功能。

- 能够在紧张的时间表内创建具有所需基础架构的新帐户。
- 访问您已配置的基础架构清单，并能够更新或删除基础架构组件。
- 使调配和维护过程更简单、更快速、更安全可靠的方法和技术。

AWS Service Catalog用于管理 iAP

AWS提供一项名为的服务 AWS Service Catalog，该服务支持将AWS基础架构作为产品进行管理和配置。您可以使用 Service Catalog 将需要配置的基础架构快速定义为一组产品，向所需各方授予这些产品的权限，并实现单个产品所需的配置和更新模式。

Service Catalog 由... 提供支持AWS CloudFormation。Service Catalog 产品组合、产品及其配置模板作为 CloudFormation 堆栈进行管理。您可以通过四种方式定义这些堆栈：

- 通过使用标准 CloudFormation 模板。
- 通过使用AWS Cloud Development Kit (AWS CDK)和 Service Catalog 构建库以及您首选的受支持的编程语言。
- 通过使用第三方工具提供的框架从描述 CloudFormation 堆栈的声明性元数据生成堆栈定义。
- 通过使用Serv ice Catalog API。此 API 提供了除构建产品之外的所有方法。您可以将产品添加到产品组合，从产品组合中删除产品，标记产品和产品组合，定义管理和运营产品服务操作，以及浏览和搜索产品组合和产品定义。

Service Catalog 产品的核心是一组或多项AWS资源，这些资源被配置为提供集体、可定制（通过参数化）功能。例如，您可以定义Service Catalog 产品以在目标账户中定义一个私有Amazon Simple Storage Service (Amazon S3) 存储桶。S3 存储桶是一种可能具有输入参数的产品，例如存储段名称、允许访问的互联网地址范围、一组可以访问存储段的用户、生命周期分层策略或存储段版本控制规范。您也可以定义AWS Identity and Access Management (IAM) 角色，作为产品的一部分提供对存储桶的访问权限。

您可以将Service Catalog 产品添加到一个或多个产品组合中。Service Catalog 产品组合是组合在一起的产品的集合，通常是因为它们的用途相似（例如，分析、开发、客户访问服务、合作伙伴访问服务等）。

您为用户、群组或角色提供权限，使其有权在产品组合级别配置产品。在配置方面，产品要么与 Launch IAM 角色（用于以自助方式向任何可以担任该角色的人启动产品）相关联，要么与定义一个或多个账户的堆栈集相关联，该堆栈集定义了产品可以配置到的一个或多个账户。要使用堆栈集，必须在Service Catalog 中心帐户中定义Service Catalog 管理员角色，并在堆栈集的每个目标帐户中定义 Service Catalog 产品配置执行角色。

以下各节将更详细地讨论Service Catalog IAP 功能。

主题

Support 模块化和代码重用

您可以从许多不同的AWS资源中组装产品，甚至可以从其他产品中组装产品。理想情况下，您可以以模块化方式定义资源，以便可以在多个产品中重复使用它们。资源级重用使您能够在一个地方进行任何 future 更改，而不是在使用该资源类型的每个产品上进行更改。

Service Catalog 提供了一项名为链接的功能，以支持产品级别的可重用性。您可以将一个产品链接到一个或多个其他产品。例如，您可能希望将 S3 日志存储桶产品链接到更高级别的监控产品。尽管链式支持模块化，但它会带来一些操作复杂性，因为你必须管理依赖关系。Service Catalog 不会自动维护链接产品之间的版本控制，因此它无法确保对一个产品的更改不会破坏依赖于该产品的其他产品。谨慎使用链接，开发自己的机制来确保版本控制和维护依赖关系。

Service C CloudFormation atalog 原生使用将产品配置模板部署为 CloudFormation堆栈。但是，Service Catalog 对产品堆栈的 CloudFormation 部署施加了一些限制。特别是，Service Catalog 配置不支持用于在多个级别上插入可重复使用的 CloudFormation 脚本段或引用嵌套脚本（或堆栈）的 CloudFormation `include`宏。这些Service Catalog 限制限制了通过可重复使用的 CloudFormation 模板或组件定义产品的能力，这是您在本地定义堆栈时的标准最佳实践 CloudFormation。

> **Note**
>
> Service Catalog 允许您使用使用这些 CloudFormation 结构的配置模板成功定义产品。但是，如果您使用`include`宏或在 Service Catalog CloudFormation 模板中嵌套多个级别的脚本，则会遇到配置时错误。

这些限制可能会使在Service Catalog 中实现模块化和可重复使用的产品变得困难。如果需要模块化，您可以探索使用AWS CDK来实现您的产品及其配置模板，或者使用 AWSLabs Service Catalog Tools 项目中的配置工作流程和引擎。本指南下文中将概述本指南下文中的。

用于在Service Catalog 中定义产品的编程选项

使用Service Catalog 调配AWS基础架构的两个编程选项是 CloudFormation模板或AWS CDK. 当前，没有用于定义Service Catalog 产品的声明性或无代码机制。

CloudFormation 脚本

AWS CloudFormation是一种久经考验的 IaC 原生脚本语言，用于配置AWS基础架构。您可以在AWS Management Console或使用诸如 Visual Studio Code（或简单的文本编辑器）和 () 之类的AWS Command Line Interface开发工具来开发 CloudFormation 脚本。AWS CLI

有关更多信息，请参阅 CloudFormation 文档。有关使用 CloudFormation 模板指定Service Catalog 产品的更多信息，请参阅 CloudFormation 文档中的AWS::ServiceCatalog::CloudFormation产品资源。

采用编程方法AWS CDK

AWS CDK提供了一个优雅而强大的面向对象的编程框架，用于通过使用多种编程语言来定义和维护AWS基础架构。您可以使用开发面向对象的细粒度自定义和对AWS类框架的扩展。AWS CDKAWS CDK适用于想要针对更复杂的基础架构需求AWS 服务进行自定义且具有必要编程技能和经验的用户。

要使用实现Service Catalog 解决方案AWS CDK，您可以使用内置的Service Catalog 类来定义您的产品和产品组合。这些类由AWS CDK aws-cdk-lib.aws_servicecatalog 模块提供。

您可以使用AWS CDK以多种方式实现产品。为避免在中为产品编写配置模板 CloudFormation 并保持可重用性，我们建议您使用AWS CDKProductStack类来表示配置模板。`ProductStack`实例是您以编程方式向其添加资源的AWS CDK堆栈。例如，您可以添加 S3 存储桶、IAM 角色角色或Amazon CloudWatch 日换。当您通过调用将该`ProductStack``servicecatalog.CloudFormationProduct`实例作为其配置模板添加到已定义的实例时`servicecatalog.CloudFormationTemplate.fromProductStack (<ProductStack instance>)`，会AWS CDK自动生成该 CloudFormation模板。

以下是Amazon S3 产品的 Java`ProductStack` 实现示例。

```
import * as s3 from 'aws-cdk-lib/aws-s3';
import * as cdk from 'aws-cdk-lib';
```

```
class S3BucketProduct extends servicecatalog.ProductStack {
  constructor(scope: Construct, id: string) {
    super(scope, id);

    new s3.Bucket(this, 'BucketProduct');
  }
}

const product = new servicecatalog.CloudFormationProduct(this, 'Product', {
  productName: "My Product",
  owner: "Product Owner",
  productVersions: [
    {
      productVersionName: "v1",
      cloudFormationTemplate:
 servicecatalog.CloudFormationTemplate.fromProductStack(new S3BucketProduct(this,
 'S3BucketProduct')),
    },
  ],
});
```

AWS CDK提供内置内置持续集成和持续部署（CI/CD）管道。您可以自定义这些内置管道和软件开发生命周期 (SDLC) 流程，以满足自己的流程标准和目标。

自定义AWS CDK类可以继承其他类以提供专门的函数，并且一个类可以由其他类的实例组成。如果您使用共享AWS CDK类框架来实现多个 Service Catalog 产品，请考虑任何版本控制或兼容性影响，尤其是在多个开发团队中。你必须确保所做的更改是向后兼容的，或者你有一个正在遵循的版本控制方案，这样你对一个产品所做的类别更改就不会破坏另一个产品。

有关更多信息，请参阅 AWS CDK 文档。

与外部配置流程和工作流程集成

您可以使用AWS SDK API 或者，与Service Catalog 组件进行交互AWS CLI。您可以使用 AWSSDK Service Catalog API 通过任何可以集成Service Catalog API 调用的工具管理Service Catalog 产品。API 涵盖了Service Catalog 创建和管理的所有方面。例如，Terraform 支持通过在其Launch Wizard 中调用AWS SDK Service Catalog API 来启动（配置）Service Catalog 产品。有关更多信息，请参阅AWS文档中的使用 Terraform 启动AWS Service Catalog产品。

您也可以调用ServAWS CLI ice Catalog 命令对Service Catalog 执行操作。有关支持的命令的更多信息，请参阅《AWS CLI命令参考》中的 s ervicatal og。

产品供应规范

Service Catalog 以 CloudFormation 堆栈集部署的形式启动配置过程，这些资源是在 CloudFormation 配置模板中指定的资源。（模板可以直接在构造中创建，AWS CloudFormation也可以由AWS CDK`ProductStack`构造生成。）Service Catalog 产品配置是一个封闭的过程，您无法对其进行自定义以添加初步或后处理步骤，也无法对其进行调整。但是，您可以修改配置模板以 CloudFormation 资源规范的形式添加步骤。这些可能是AWS Lambda或AWS Step Functions，也可以是 Lambda 支持的自定义资源，用于执行初步步骤（例如自定义引导以设置在配置期间使用的堡垒主机）和后期步骤（例如拆除堡垒主机）。这种实现预置备和预配后步骤的方法受与置备模板相同的`include`嵌套堆栈限制的约束。

您可以将目标帐户指定为个人帐户而不是组织单位 (OU)。您可以编写自定义资源或函数来解决此限制。大多数组织向 OU 而不是个人账户提供产品组合，因为它们会自动生成账户，不想手动维护账户列表。

DevSecOps 支持

目前，使用Service Catalog CloudFormation 脚本置备的产品没有对 CI/CD 流程的内置支持。我们建议您在中创建 CI/CD 流程AWS CodePipeline或其他 DevOps 工具，通过开发、测试、阶段和生产等生命周期环境开发、测试和发布产品。

AWS CDK确实为产品提供了内置的 CI/CD 支持，如本指南前面所述。

自定义重用和账户专属配置

产品应可重复使用，以实现尽可能多的不同定制目的。Service Catalog 支持通过产品参数实现可重用性。您可以在配置时将这些参数作为输入提供给产品。

您还可以在 CloudFormation模板级别将这些参数指定为 Parameter StorAWS Systems Manager e 值，以应用特定于帐户的值和 OU 特定的值。这是 CloudFormation 配置模板设计的最佳实践。预置产品时，将应用目标账户中命名参数的值。例如，您可以将子网参数指定为参数存储值，并在产品配置时将该子网应用于特定 OU 帐户。有关将参数存储值作为 CloudFormation 模板参数的更多信息，请参阅 AWS CloudFormation文档中的使用动态引用指定模板值。

将Service Catalog 产品资源定义和管理为应用程序

AWS Service Catalog AppRegistry提供集中的应用程序搜索、报告和管理功能。 AppRegistry 应用程序可以包括一个或多个预配置的产品堆栈以及独立于Service Catalog 的 CloudFormation堆栈。您可以

对定义为部署目标的所有应用程序资源集合进行AWS 账户分组和查看。这些大厅的的的的的的的的的的的的的的的的的的的的的的的的的

您也可以使用 AppRegistry 将元数据属性与应用程序相关联。您可以分配包含属性集的可重用属性组。然后，您可以使用 AppRegistry 或集成服务搜索具有给定属性的应用程序资源并对其进行操作。这些集成服务包括：

- A@@ pplication ManagerAWS Systems Manager，Application Manager，该功能在您的应用程序集群环境环境环境环境环境环境环境环境环境环境环境环境环境环境环境环境AWS
- AWS Resource Access Manager，与AWS组织负责人共享应用程序和属性组
- 使用 AWS Resource Groups 的AWS服务
- AWS Resilience Hub用于产品结构发现和弹性评估
- AWS 服务管理连接器声明和设置与 ServiceNow JIRA 和其他常用工具的连接

有关的更多信息 AppRegistry，请参阅下文：

- AppRegistry 管理员指南。
- 使用AWS Service Catalog AppRegistry博客文章提高应用程序的可见性和治理。本文概述了如何在基础架构治理 AppRegistry 中使用，并提供了将基础架构注册为应用程序的命令行示例 AppRegistry。
- 使用 AppRegistry 应用程序管理器博客文章集中管理您的应用程序。本文概述了 AppRegistry 如何申请在上注册 LAMP Web 应用程序AWS Management Console并使用应用程序管理器对其进行管理的教程。

管理您的资源管理

Service Catalog 具有自己的内部库存管理功能，可在通过产品共享和自助服务配置产品时对产品进行注册。但是，我们建议您使用AWS Config或AppRegistry和相关服务来管理您的产品预配置资源。这些工具提供了一种更全面、更集成的方法，可将预置的Service Catalog 产品与AWS基础架构的其余部分一起管理。 AWS Config允许您在控制台上或使用AWS SDK API 清点预配置产品并对其执行操作。AppRegistry与应用程序管理器集成，还为Service Catalog 预配置的产品提供库存管理。

使用AWS Service Catalog工具

如果您想以更具声明性的方式配置 iaC 产品中的自定义配置工作流程，则可能需要增强部分Service Catalog 功能。 AWS提供了多种工具来支持这些需求。AWS实验室项目中提供了两个常用的工具：Service Catalog Puppet 和Service Catalog 工厂。

主题

SService Catalog ap

Service Catalog Puppet 是使用AWS Boto3 API 在 Python 中实现的。此工具为配置和配置Service Catalog 产品提供了多项强大的功能。开发人员可以使用用作清单的 YAML 模板来配置Service Catalog 产品和产品组合配置信息。Service Catalog Puppet 配置工作流程支持需要比Service Catalog 更复杂的部署过程的产品。它们还支持性能优化，以便在严格的时间窗口内大规模配置产品。

Service Catalog Puppet 在部署时访问Service Catalog CloudFormation 模板以进行产品配置。它 CloudFormation 直接调用以部署产品的配置模板堆栈，并绕过了 Service Catalog 自己的堆栈集配置过程施加的限制。如果配置模板使用宏来包含其他 CloudFormation 脚本或使用嵌套 CloudFormation脚本，则必须在配置工作流程的引导部分提供对目标账户中这些脚本的访问权限。

有关更多信息：

- 请参阅Service Catalog Puppet 文档和GitHub存储库。
- 如果您想使用Service Catalog Puppet SDK 以编程方式与该工具交互以启动产品和产品组合配置，请参阅 SDK 文档。
- GitOps是管理Service Catalog Puppet 环境的默认机制。

Service Catalog Puppet 对于开发人员来说相当容易学习。它需要熟悉 CloudFormation才能实现产品配置模板，而要实现清单，则需要熟悉 YAML 模板。有不错的研讨会可以让新开发者快速上手，例如自定进度的研讨会。

Support 配置工作流程

Service Catalog Puppet 使用 Python Luigi 任务编排引擎来实现引导和配置工作流程。这些工作流程中的所有步骤均作为 Luigi 工作流程任务实现。有关 Luigi 及其与其他流行工作流程工具的比较的概述，请参阅 Data Re venue 博客上的 Airflow vs Luigi vs Argo vs mlFlow KubeFlow 对比。

Luigi 允许 Service Catalog Puppet 控制与工作流程任务相关的工作人员数量，并控制工作流程的其他方面，以实现更好的扩展和性能。Service Catalog Puppet 还提供了 depends_on 机制，用于管理产品和步骤依赖关系以及协调产品配置。此功能可帮助您实现和操作管理精细的产品定义和复杂的依赖关系。

配置模式

Service Catalog Puppet 支持三种执行模式：集线器、分支和异步。所有三种模式均在Service Catalog 中已定义的产品组合中配置产品。他们依靠与目标帐户共享Service Catalog 产品，并使用Service Catalog 管理员和启动角色在这些目标中实现配置。Service Catalog Puppet 根据 YAML 配置文件中提供的角色配置在同一组织内执行引导步骤。该工具还支持从单个中心账户向多个组织进行配置。在这种情况下，必须在外部组织中手动执行引导，以允许 Service Catalog Puppet 在外部组织的帐户中执行所需的配置操作。

在所有配置模式下，Service Catalog Puppet 无需调用服务目录的配置过程即可直接实现产品配置。您可以配置配置配置清单，以使用现有Service Catalog 堆栈集约束中的角色和目标账户规范。Service Catalog Puppet 使用这些信息对 Luigi 工作流程进行自己的配置。

除了直接指定 OU 或账户外，您还可以根据账户标记方法定义产品组合配置目标。在基于账户标签的配置中，将向具有指定清单配置标签集中的所有标签的所有账户配置产品组合。例如，如果您想向美国东部地区的所有机构生产账户发行投资组合产品，则可以指定标签`type:prod partition:us-east`、和`scope:institutional-client`。您还可以声明帐户和 OU 排除项，以禁止向具有您指定标签的 OU 或帐户进行配置，或禁止向属于 OU 指定目标的成员的帐户进行配置。有关帐户标记的更多信息，请参阅Serv ice Catalog 工具文档。

Hub

在集线器配置模式下，分支账户的所有 Luigi 工作流程均由指定的中央中心账户管理。中心账户担任 IAM 角色，允许其在分支账户中执行操作，但任务管理在中心账户内进行。中心账户会同步等待，直到所有分支账户配置任务成功或失败完成。然后，它会报告最终状态。中心账户模式是最古老、最成熟的配置模式。但是，许多用户已转向分支配置模式，以实现更高的配置并发性和速度。

分支模式

在分支模式下，Service Catalog 中心账户启动 Luigi 工作流程，使其在指定的引导分支账户中运行。分支工作流程完成后，中心账户会收到通知。分支账户中的故障上升到中心账户。中心账户对分支账户进行轮询以查看分支账户是否已完成并确定状态。

分支模式不需要增加AWS 服务配额的可能性最小，因为几乎所有内容都在单独的分支账户中运行。分支模式在保持中央控制的同时，还提供了比集线器模式更高的并发性。与集线器模式相比，它可以将配置速度提高 800%。分支模式支持通过产品之间的DependsOn关系链接产品，从而确保依赖的产品已经配置完毕。包含链接产品的产品也可以提供组件链产品。您也可以使用专门的AWS Lambda函数调用来执行所需的步骤。一个辐条的故障与其他辐条是隔离的。

分支模式由在多达 7 个区域拥有 980 多个账户的企业使用。这些企业通常能够在一小时内向其基础架构中的所有地区和账户预置产品。

> **Note**
>
> 这些结果可能会因网络基础架构、工作负载以及AWS组织中心和分支账户的现有配额等因素而有所不同。它们还取决于所配置的产品资源、其固有的创建时间以及它们对其他资源的依赖关系。

Ayservice

异步模式在分支账户中启动配置工作流程，但它不会等待或接收来自分支账户的完成响应。

缓存

Service Catalog Puppet 用来优化工作流程速度的另一种机制是缓存代表工作流程步骤的常见任务。缓存的作业写入Amazon Simple Storage Service (Amazon S3)。下次使用相同参数在同一会话中调用任务时，Service Catalog Puppet 使用缓存的值而不是重新运行任务。有关更多信息，请参阅 Service C erv ice Service Service Cat

DevSecOps 生命转

Service Catalog Puppet 包括对管理 DevSecOps 管道的支持。您可以使用 Service Catalog Tools 操作（如Serv ice Catalog Puppet 概述所示）自动测试和在整个AWS生命周期账户（包括推荐的金丝雀账户）中推广产品。有关更多信息，请参阅Servic e Catalog Puppet 文档中的管理环境。

为确保在广泛生产使用之前检测到与产品变更相关的任何问题，Service Catalog Puppet 需要至少一个金丝雀帐户进行初始部署。测试新版本并获得对新版本的信心后，您可以将其推广到非 Canary 生产账户。如果您发现任何问题，可以回滚发行版，并在问题解决后重新引入该版本。使用这种方法时，如果您发布的金丝雀版本对生产账户有问题，则可能会出现生产问题。作为替代方法，您可以在将更改发布到生产环境之前，对每个产品变更进行全面的回归测试。这会在 CI/CD 过程中带来额外的开销，但有助于避免生产问题。 DevSecOps 管理员有责任为其开发团队确定最佳功能发布场景和方法。

Service Catalog Puppet 允许多个团队同时开发和测试Service Catalog 产品解决方案的配置。作为最佳实践，一个产品不应由多个开发人员同时更改。相反，您可以将产品分解为更细粒度的组件，进行单独的同步修改。

Service Catalog Puppet 还通过提供静态和单元测试功能的断言语句帮助自动化测试。您可以测试使用策略SCP (SCP) 和 IAM 策略。这些是技术 end-to-end 测试，但可用于系统集成测试 (SIT) 环境。有关更多信息，请参阅 Service Catalog Puppet 文档中的使用策略模拟和应用服务控制策略。

成熟度、完整性和支持

尽管Service Catalog Puppet 不是官方支持AWS 服务的，但它已被广泛采用。在过去的几年中，大型组织一直使用此工具在所需的配置时间窗口内成功地向数百个 OU 账户集中调配产品。事实证明，它可以大规模提供容错产品配置。在使用 Service Catalog Puppet 时遇到任何问题的用户可以将这些问题登录到GitHub 存储库中，以供此AWS Labs 解决方案的贡献者解决。

Service Catalog 工

Service Catalog 工厂是AWS实验室提供的另一种工具。它类似于AWS Control Tower —它生成帐户并调用Service Catalog（可能通过 Puppet）在这些帐户中配置 iAP。它使用许多与 Service Catalog Puppet 相同的机制来实现其功能。Service Catalog 工厂可以调用 Service Catalog 或 Service Catalog Puppet 为账户中的产品配置基础架构。该工具还支持在多个AWS 区域和组织中生成帐户。有关更多信息，请参阅Service Catalog 工厂文档和GitHub 存储库。

回顾和后续步骤

Service Catalog 可帮助您快速可靠地将基础架构作为产品进行配置。您可以从定义的产品目录中自助服务基础架构，也可以将产品推送到 hub-and-spoke 模型中的指定目标客户。您可以使用 CloudFormation 脚本或使用来定义Service Catalog 产品及其配置模板AWS CDK。在这两种方法中，Service Catalog 都通过调用部署代表产品配置模板的堆栈 CloudFormation 来配置产品。堆栈将部署到堆 CloudFormation 栈集内的所有指定目标账户。

与之相比 CloudFormation，Service Catalog 开发AWS CDK方法支持更高的模块化和重用性，因为您可以使用预定义的Service Catalog 产品和产品组合类以及预定义的资源类型来定义产品及其资源。AWS CDK实现需要更高级的编程技能。如果您的组织想要建立自己的具有标准化资源配置和行为的可重用产品框架作为AWS基础架构开发的基础，那么这可能是合理的。

您可以使用Service Catalog Puppet 和Service Catalog 工厂来增强Service Catalog 功能，主要用于配置。Service Catalog Puppet 具有声明性和基于标签的产品配置规范；内置、可定制、高性能和专门构建的配置工作流程；以及内置、可定制、基于操作的 CI/CD 和 SDLC 管道。通过使用工作流程依赖关系管理和内置的测试自动化功能，您可以将 Service Catalog 产品链接起来，降低运营风险。Service Catalog Puppet 可帮助您在严苛的时间窗口内可靠地向数百个账户配置产品。Service Catalog 工厂类似于AWS Control Tower。它生成帐户并调用 Service Catalog 以在这些账户中配置 iAP。

Service Catalog 和Service Catalog 工具提供了广泛的功能，可帮助您管理 iAPAWS。Service Catalog 和这些工具正在不断改进。有关最新功能，请参阅AWS Service Catalog功能和AWS Service Catalog 产品存储库。

资源

参考

- Service Catalog 文档
- Service Catalog API
- AppRegistry
- AWS CloudFormation 文档
- AWS CloudFormation堆栈集
- AWS::ServiceCatalog::CloudFormation产品资源
- 使用 Terraform 推出AWS Service Catalog产品
- AWS Cloud Development Kit (AWS CDK)
- Service Catalog 构造库
- AWS CDK ProductStack阶级
- AWS Organizations 文档

工具

- Service Catalog Puppet 文档
- Service Catalog Puppet GitHub 存储库
- Service Catalog 工厂文档
- Service Catalog 工厂 GitHub存储库

AWS规范性指导模式

- 以多种AWS 账户方式管理AWS Service Catalog产品AWS 区域
- 在不同的AWS 账户地区复制AWS Service Catalog产品AWS 区域
- 使用以下方法自动部署产品AWS Service Catalog组合和产品AWS CDK

文档历史记录

下表描述了本指南的重大更改。如果您想收到有关future 更新的通知，可以订阅 RSS 提要。

变更	说明	日期
首次出版	—	2022 年 1 月 30 日

AWS 规范性指导词汇表

以下是 AWS 规范性指导提供的策略、指南和模式中的常用术语。若要推荐词条，请使用术语表末尾的提供反馈链接。

数字

7 R

将应用程序迁移到云中的 7 种常见迁移策略。这些策略以 Gartner 于 2011 年确定的 5 R 为基础，包括以下内容：

- 重构/重新架构 - 充分利用云原生功能来提高敏捷性、性能和可扩展性，以迁移应用程序并修改其架构。这通常涉及到移植操作系统和数据库。示例：将您的本地 Oracle 数据库迁移到兼容 Amazon Aurora PostgreSQL 的版本。
- 更换平台 - 将应用程序迁移到云中，并进行一定程度的优化，以利用云功能。示例：在中将您的本地 Oracle 数据库迁移到适用于 Oracle 的亚马逊关系数据库服务 (Amazon RDS) AWS Cloud。
- 重新购买 - 转换到其他产品，通常是从传统许可转向 SaaS 模式。示例：将您的客户关系管理 (CRM) 系统迁移到 Salesforce.com。
- 更换主机（直接迁移）- 将应用程序迁移到云中，无需进行任何更改即可利用云功能。示例：在中的 EC2 实例上将您的本地 Oracle 数据库迁移到 Oracle AWS Cloud。
- 重新定位（虚拟机监控器级直接迁移）：将基础设施迁移到云中，无需购买新硬件、重写应用程序或修改现有操作。您可以将服务器从本地平台迁移到同一平台的云服务。示例：将Microsoft Hyper-V应用程序迁移到 AWS。
- 保留（重访）- 将应用程序保留在源环境中。其中可能包括需要进行重大重构的应用程序，并且您希望将工作推迟到以后，以及您希望保留的遗留应用程序，因为迁移它们没有商业上的理由。
- 停用 - 停用或删除源环境中不再需要的应用程序。

A

ABAC

请参阅基于属性的访问控制。

抽象服务

参见托管服务。

酸

参见原子性、一致性、隔离性、耐久性。

主动-主动迁移

一种数据库迁移方法，在这种方法中，源数据库和目标数据库保持同步（通过使用双向复制工具或双写操作），两个数据库都在迁移期间处理来自连接应用程序的事务。这种方法支持小批量、可控的迁移，而不需要一次性割接。与主动-被动迁移相比，它更灵活，但需要更多的工作。

主动-被动迁移

一种数据库迁移方法，在这种方法中，源数据库和目标数据库保持同步，但在将数据复制到目标数据库时，只有源数据库处理来自连接应用程序的事务。目标数据库在迁移期间不接受任何事务。

聚合函数

一个 SQL 函数，它对一组行进行操作并计算该组的单个返回值。聚合函数的示例包括SUM和MAX。

AI

参见人工智能。

AIOps

参见人工智能操作。

匿名化

永久删除数据集中个人信息的过程。匿名化可以帮助保护个人隐私。匿名化数据不再被视为个人数据。

反模式

一种用于解决反复出现的问题的常用解决方案，而在这类问题中，此解决方案适得其反、无效或不如替代方案有效。

应用程序控制

一种安全方法，仅允许使用经批准的应用程序，以帮助保护系统免受恶意软件的侵害。

应用程序组合

有关组织使用的每个应用程序的详细信息的集合，包括构建和维护该应用程序的成本及其业务价值。这些信息是产品组合发现和分析过程的关键，有助于识别需要进行迁移、现代化和优化的应用程序并确定其优先级。

人工智能（AI）

计算机科学领域致力于使用计算技术执行通常与人类相关的认知功能，例如学习、解决问题和识别模式。有关更多信息，请参阅什么是人工智能？

人工智能运营（AIOps）

使用机器学习技术解决运营问题、减少运营事故和人为干预以及提高服务质量的过程。有关如何在 AWS 迁移策略中使用 AIOps 的更多信息，请参阅运营集成指南。

非对称加密

一种加密算法，使用一对密钥，一个公钥用于加密，一个私钥用于解密。您可以共享公钥，因为它不用于解密，但对私钥的访问应受到严格限制。

原子性、一致性、隔离性、持久性（ACID）

一组软件属性，即使在出现错误、电源故障或其他问题的情况下，也能保证数据库的数据有效性和操作可靠性。

基于属性的访问权限控制（ABAC）

根据用户属性（如部门、工作角色和团队名称）创建精细访问权限的做法。有关更多信息，请参阅 AWS Identity and Access Management (I AM) 文档 AWS中的 AB AC。

权威数据源

存储主要数据版本的位置，被认为是最可靠的信息源。您可以将数据从权威数据源复制到其他位置，以便处理或修改数据，例如对数据进行匿名化、编辑或假名化。

可用区

中的一个不同位置 AWS 区域 ，不受其他可用区域故障的影响，并向同一区域中的其他可用区提供低成本、低延迟的网络连接。

AWS 云采用框架 (AWS CAF)

该框架包含指导方针和最佳实践 AWS ，可帮助组织制定高效且有效的计划，以成功迁移到云端。AWS CAF将指导分为六个重点领域，称为视角：业务、人员、治理、平台、安全和运营。业务、人员和治理角度侧重于业务技能和流程；平台、安全和运营角度侧重于技术技能和流程。例如，人员角度针对的是负责人力资源（HR）、人员配置职能和人员管理的利益相关者。从这个角度来看，AWS CAF 为人员发展、培训和沟通提供了指导，以帮助组织为成功采用云做好准备。有关更多信息，请参阅 AWS CAF 网站和 AWS CAF 白皮书。

AWS 工作负载资格框架 (AWS WQF)

一种评估数据库迁移工作负载、推荐迁移策略和提供工作估算的工具。 AWS WQF 包含在 AWS Schema Conversion Tool (AWS SCT) 中。它用来分析数据库架构和代码对象、应用程序代码、依赖关系和性能特征，并提供评测报告。

B

坏机器人

旨在破坏个人或组织或对其造成伤害的机器人。

BCP

参见业务连续性计划。

行为图

一段时间内资源行为和交互的统一交互式视图。您可以使用 Amazon Detective 的行为图来检查失败的登录尝试、可疑的 API 调用和类似的操作。有关更多信息，请参阅 Detective 文档中的行为图中的数据。

大端序系统

一个先存储最高有效字节的系统。另请参见字节顺序。

二进制分类

一种预测二进制结果（两个可能的类别之一）的过程。例如，您的 ML 模型可能需要预测诸如“该电子邮件是否为垃圾邮件？”或“这个产品是书还是汽车？”之类的问题

bloom 筛选条件

一种概率性、内存高效的数据结构，用于测试元素是否为集合的成员。

蓝/绿部署

一种部署策略，您可以创建两个独立但完全相同的环境。在一个环境中运行当前的应用程序版本（蓝色），在另一个环境中运行新的应用程序版本（绿色）。此策略可帮助您在影响最小的情况下快速回滚。

自动程序

一种通过互联网运行自动任务并模拟人类活动或互动的软件应用程序。有些机器人是有用或有益的，例如在互联网上索引信息的网络爬虫。其他一些被称为恶意机器人的机器人旨在破坏个人或组织或对其造成伤害。

僵尸网络

被恶意软件感染并受单方（称为机器人牧民或机器人操作员）控制的机器人网络。僵尸网络是最著名的扩展机器人及其影响力的机制。

分支

代码存储库的一个包含区域。在存储库中创建的第一个分支是主分支。您可以从现有分支创建新分支，然后在新分支中开发功能或修复错误。为构建功能而创建的分支通常称为功能分支。当功能可以发布时，将功能分支合并回主分支。有关更多信息，请参阅关于分支（GitHub 文档）。

破碎的玻璃通道

在特殊情况下，通过批准的流程，用户 AWS 账户 可以快速访问他们通常没有访问权限的内容。有关更多信息，请参阅 Well -Architected 指南中的 “实施破碎玻璃程序” 指示 AWS 器。

棕地策略

您环境中的现有基础设施。在为系统架构采用棕地策略时，您需要围绕当前系统和基础设施的限制来设计架构。如果您正在扩展现有基础设施，则可以将棕地策略和全新策略混合。

缓冲区缓存

存储最常访问的数据的内存区域。

业务能力

企业如何创造价值（例如，销售、客户服务或营销）。微服务架构和开发决策可以由业务能力驱动。有关更多信息，请参阅在 AWS上运行容器化微服务白皮书中的围绕业务能力进行组织部分。

业务连续性计划（BCP）

一项计划，旨在应对大规模迁移等破坏性事件对运营的潜在影响，并使企业能够快速恢复运营。

C

CAF

参见AWS 云采用框架。

金丝雀部署

向最终用户缓慢而渐进地发布版本。当你有信心时，你可以部署新版本并全部替换当前版本。

CCoE

参见云卓越中心。

CDC

参见变更数据捕获。

更改数据捕获（CDC）

跟踪数据来源（如数据库表）的更改并记录有关更改的元数据的过程。您可以将 CDC 用于各种目的，例如审计或复制目标系统中的更改以保持同步。

混沌工程

故意引入故障或破坏性事件来测试系统的弹性。您可以使用 AWS Fault Injection Service (AWS FIS) 来执行实验，对您的 AWS 工作负载施加压力并评估其响应。

CI/CD

查看持续集成和持续交付。

分类

一种有助于生成预测的分类流程。分类问题的 ML 模型预测离散值。离散值始终彼此不同。例如，一个模型可能需要评估图像中是否有汽车。

客户端加密

在目标 AWS 服务 收到数据之前，对数据进行本地加密。

云卓越中心（CCoE）

一个多学科团队，负责推动整个组织的云采用工作，包括开发云最佳实践、调动资源、制定迁移时间表、领导组织完成大规模转型。有关更多信息，请参阅 AWS Cloud 企业战略博客上的 CCoE 帖子。

云计算

通常用于远程数据存储和 IoT 设备管理的云技术。云计算通常与边缘计算技术相关。

云运营模型

在 IT 组织中，一种用于构建、完善和优化一个或多个云环境的运营模型。有关更多信息，请参阅构建您的云运营模型。

云采用阶段

组织迁移到以下阶段时通常会经历四个阶段 AWS Cloud：

- 项目 - 出于概念验证和学习目的，开展一些与云相关的项目
- 基础 - 进行基础投资以扩大云采用率（例如，创建登录区、定义 CCoE、建立运营模型）

- 迁移 - 迁移单个应用程序
- 重塑 - 优化产品和服务，在云中创新

Stephen Orban在 AWS Cloud 企业战略博客的博客文章《云优先之旅和采用阶段》中定义了这些阶段。有关它们与 AWS 迁移策略的关系的信息，请参阅迁移准备指南。

CMDB

参见配置管理数据库。

代码存储库

通过版本控制过程存储和更新源代码和其他资产（如文档、示例和脚本）的位置。常见的云存储库包括 GitHub 或 AWS CodeCommit。每个版本的代码都称为一个分支。在微服务结构中，每个存储库都专门用于一个功能。单个 CI/CD 管道可以使用多个存储库。

冷缓存

一种空的、填充不足或包含过时或不相关数据的缓冲区缓存。这会影响性能，因为数据库实例必须从主内存或磁盘读取，这比从缓冲区缓存读取要慢。

冷数据

很少访问的数据，且通常是历史数据。查询此类数据时，通常可以接受慢速查询。将这些数据转移到性能较低且成本更低的存储层或类别可以降低成本。

计算机视觉 (CV)

人工智能领域，使用机器学习来分析和提取数字图像和视频等视觉格式中的信息。例如， AWS Panorama 提供将 CV 添加到本地摄像机网络的设备，而 Amazon 则为 CV SageMaker 提供图像处理算法。

配置偏差

对于工作负载，配置会从预期状态发生变化。这可能会导致工作负载变得不合规，而且通常是渐进的，不是故意的。

配置管理数据库（CMDB）

一种存储库，用于存储和管理有关数据库及其 IT 环境的信息，包括硬件和软件组件及其配置。您通常在迁移的产品组合发现和分析阶段使用来自 CMDB 的数据。

合规性包

一系列 AWS Config 规则和补救措施，您可以汇编这些规则和补救措施，以自定义合规性和安全性检查。您可以使用 YAML 模板将一致性包作为单个实体部署在 AWS 账户 和区域或整个组织中。有关更多信息，请参阅 AWS Config 文档中的一致性包。

持续集成和持续交付（CI/CD）

自动执行软件发布过程的源代码、构建、测试、暂存和生产阶段的过程。CI/CD 通常被描述为管道。CI/CD 可以帮助您实现流程自动化、提高工作效率、改善代码质量并加快交付速度。有关更多信息，请参阅持续交付的优势。CD 也可以表示持续部署。有关更多信息，请参阅持续交付与持续部署。

CV

参见计算机视觉。

D

静态数据

网络中静止的数据，例如存储中的数据。

数据分类

根据网络中数据的关键性和敏感性对其进行识别和分类的过程。它是任何网络安全风险管理策略的关键组成部分，因为它可以帮助您确定对数据的适当保护和保留控制。数据分类是 Well-Architecte AWS d Framework 中安全支柱的一个组成部分。有关详细信息，请参阅数据分类。

数据漂移

生产数据与用来训练机器学习模型的数据之间的有意义差异，或者输入数据随时间推移的有意义变化。数据漂移可能降低机器学习模型预测的整体质量、准确性和公平性。

传输中数据

在网络中主动移动的数据，例如在网络资源之间移动的数据。

数据网格

一种架构框架，可提供分布式、去中心化的数据所有权以及集中式管理和治理。

数据最少化

仅收集并处理绝对必要数据的原则。在中进行数据最小化 AWS Cloud 可以降低隐私风险、成本和分析碳足迹。

数据边界

AWS 环境中的一组预防性防护措施，可帮助确保只有可信身份才能访问来自预期网络的可信资源。有关更多信息，请参阅在上构建数据边界。 AWS

数据预处理

将原始数据转换为 ML 模型易于解析的格式。预处理数据可能意味着删除某些列或行，并处理缺失、不一致或重复的值。

数据溯源

在数据的整个生命周期跟踪其来源和历史的过程，例如数据如何生成、传输和存储。

数据主体

正在收集和处理其数据的个人。

数据仓库

一种支持商业智能（例如分析）的数据管理系统。数据仓库通常包含大量历史数据，通常用于查询和分析。

数据库定义语言（DDL）

在数据库中创建或修改表和对象结构的语句或命令。

数据库操作语言（DML）

在数据库中修改（插入、更新和删除）信息的语句或命令。

DDL

参见数据库定义语言。

深度融合

组合多个深度学习模型进行预测。您可以使用深度融合来获得更准确的预测或估算预测中的不确定性。

深度学习

一个 ML 子字段使用多层人工神经网络来识别输入数据和感兴趣的目标变量之间的映射。

defense-in-depth

一种信息安全方法，经过深思熟虑，在整个计算机网络中分层实施一系列安全机制和控制措施，以保护网络及其中数据的机密性、完整性和可用性。当你采用这种策略时 AWS，你会在 AWS Organizations 结构的不同层面添加多个控件来帮助保护资源。例如，一种 defense-in-depth 方法可以结合多因素身份验证、网络分段和加密。

委托管理员

在中 AWS Organizations，兼容的服务可以注册 AWS 成员帐户来管理组织的帐户并管理该服务的权限。此账户被称为该服务的委托管理员。有关更多信息和兼容服务列表，请参阅 AWS Organizations 文档中使用 AWS Organizations的服务。

部署

使应用程序、新功能或代码修复在目标环境中可用的过程。部署涉及在代码库中实现更改，然后在应用程序的环境中构建和运行该代码库。

开发环境

参见环境。

侦测性控制

一种安全控制，在事件发生后进行检测、记录日志和发出警报。这些控制是第二道防线，提醒您注意绕过现有预防性控制的安全事件。有关更多信息，请参阅在 AWS上实施安全控制中的侦测性控制。

开发价值流映射 (DVSM)

用于识别对软件开发生命周期中的速度和质量产生不利影响的限制因素并确定其优先级的流程。DVSM 扩展了最初为精益生产实践设计的价值流映射流程。其重点关注在软件开发过程中创造和转移价值所需的步骤和团队。

数字孪生

真实世界系统的虚拟再现，如建筑物、工厂、工业设备或生产线。数字孪生支持预测性维护、远程监控和生产优化。

维度表

在星型架构中，一种较小的表，其中包含事实表中定量数据的数据属性。维度表属性通常是文本字段或行为类似于文本的离散数字。这些属性通常用于查询约束、筛选和结果集标注。

灾难

阻止工作负载或系统在其主要部署位置实现其业务目标的事件。这些事件可能是自然灾害、技术故障或人为操作的结果，例如无意的配置错误或恶意软件攻击。

灾难恢复 (DR)

您用来最大限度地减少灾难造成的停机时间和数据丢失的策略和流程。有关更多信息，请参阅 Well-Architected Fr ame AWS work 中的 “工作负载灾难恢复：云端 AWS 恢复”。

DML

参见数据库操作语言。

领域驱动设计

一种开发复杂软件系统的方法，通过将其组件连接到每个组件所服务的不断发展的领域或核心业务目标。Eric Evans 在其著作领域驱动设计：软件核心复杂性应对之道（Boston: Addison-Wesley Professional, 2003）中介绍了这一概念。有关如何将领域驱动设计与 strangler fig 模式结合使用的信息，请参阅使用容器和 Amazon API Gateway 逐步将原有的 Microsoft ASP.NET（ASMX）Web 服务现代化。

DR

参见灾难恢复。

漂移检测

跟踪与基准配置的偏差。例如，您可以使用 AWS CloudFormation 来检测系统资源中的偏差，也可以使用 AWS Control Tower 来检测着陆区中可能影响监管要求合规性的变化。

DVSM

参见开发价值流映射。

E

EDA

参见探索性数据分析。

边缘计算

该技术可提高位于 IoT 网络边缘的智能设备的计算能力。与云计算相比，边缘计算可以减少通信延迟并缩短响应时间。

加密

一种将人类可读的纯文本数据转换为密文的计算过程。

加密密钥

由加密算法生成的随机位的加密字符串。密钥的长度可能有所不同，而且每个密钥都设计为不可预测且唯一。

字节顺序

字节在计算机内存中的存储顺序。大端序系统先存储最高有效字节。小端序系统先存储最低有效字节。

端点

参见服务端点。

端点服务

一种可以在虚拟私有云（VPC）中托管，与其他用户共享的服务。您可以使用其他 AWS 账户 或 AWS Identity and Access Management (IAM) 委托人创建终端节点服务， AWS PrivateLink 并向其授予权限。这些账户或主体可通过创建接口 VPC 端点来私密地连接到您的端点服务。有关更多信息，请参阅 Amazon Virtual Private Cloud（Amazon VPC）文档中的创建端点服务。

企业资源规划 (ERP)

一种自动化和管理企业关键业务流程（例如会计、MES 和项目管理）的系统。

信封加密

用另一个加密密钥对加密密钥进行加密的过程。有关更多信息，请参阅 AWS Key Management Service (AWS KMS) 文档中的信封加密。

environment

正在运行的应用程序的实例。以下是云计算中常见的环境类型：

- 开发环境 — 正在运行的应用程序的实例，只有负责维护应用程序的核心团队才能使用。开发环境用于测试更改，然后再将其提升到上层环境。这类环境有时称为测试环境。
- 下层环境 — 应用程序的所有开发环境，比如用于初始构建和测试的环境。
- 生产环境 — 最终用户可以访问的正在运行的应用程序的实例。在 CI/CD 管道中，生产环境是最后一个部署环境。
- 上层环境 — 除核心开发团队以外的用户可以访问的所有环境。这可能包括生产环境、预生产环境和用户验收测试环境。

epic

在敏捷方法学中，有助于组织工作和确定优先级的功能类别。epics 提供了对需求和实施任务的总体描述。例如， AWS CAF 安全史诗包括身份和访问管理、侦探控制、基础设施安全、数据保护和事件响应。有关 AWS 迁移策略中 epics 的更多信息，请参阅计划实施指南。

ERP

参见企业资源规划。

探索性数据分析（EDA）

分析数据集以了解其主要特征的过程。您收集或汇总数据，并进行初步调查，以发现模式、检测异常并检查假定情况。EDA 通过计算汇总统计数据和创建数据可视化得以执行。

F

事实表

星形架构中的中心表。它存储有关业务运营的定量数据。通常，事实表包含两种类型的列：包含度量的列和包含维度表外键的列。

失败得很快

一种使用频繁和增量测试来缩短开发生命周期的理念。这是敏捷方法的关键部分。

故障隔离边界

在中 AWS Cloud，诸如可用区 AWS 区域、控制平面或数据平面之类的边界，它限制了故障的影响并有助于提高工作负载的弹性。有关更多信息，请参阅AWS 故障隔离边界。

功能分支

参见分支。

特征

您用来进行预测的输入数据。例如，在制造环境中，特征可能是定期从生产线捕获的图像。

特征重要性

特征对于模型预测的重要性。这通常表示为数值分数，可以通过各种技术进行计算，例如 Shapley 加法解释（SHAP）和积分梯度。有关更多信息，请参阅机器学习模型的可解释性：AWS。

功能转换

为 ML 流程优化数据，包括使用其他来源丰富数据、扩展值或从单个数据字段中提取多组信息。这使得 ML 模型能从数据中获益。例如，如果您将“2021-05-27 00:15:37”日期分解为“2021”、“五月”、“星期四”和“15”，则可以帮助学习与不同数据成分相关的算法学习精细模式。

FGAC

请参阅精细的访问控制。

精细访问控制 (FGAC)

使用多个条件允许或拒绝访问请求。

快闪迁移

一种数据库迁移方法，它使用连续的数据复制，通过更改数据捕获在尽可能短的时间内迁移数据，而不是使用分阶段的方法。目标是将停机时间降至最低。

G

地理封锁

请参阅地理限制。

地理限制（地理阻止）

在 Amazon 中 CloudFront，一种阻止特定国家/地区的用户访问内容分发的选项。您可以使用允许列表或阻止列表来指定已批准和已禁止的国家/地区。有关更多信息，请参阅 CloudFront 文档中的限制内容的地理分布。

GitFlow 工作流程

一种方法，在这种方法中，下层和上层环境在源代码存储库中使用不同的分支。Gitflow 工作流程被认为是传统的，而基于主干的工作流程是现代的首选方法。

全新策略

在新环境中缺少现有基础设施。在对系统架构采用全新策略时，您可以选择所有新技术，而不受对现有基础设施（也称为棕地）兼容性的限制。如果您正在扩展现有基础设施，则可以将棕地策略和全新策略混合。

防护机制

一种高级规则，用于跨组织单位（OU）管理资源、策略和合规性。预防性防护机制会执行策略以确保符合合规性标准。它们是使用服务控制策略和 IAM 权限边界实现的。侦测性防护机制会检测策略违规和合规性问题，并生成警报以进行修复。它们通过使用 AWS Config、、Amazon、 AWS Security Hub GuardDuty AWS Trusted Advisor、Amazon Inspector 和自定义 AWS Lambda 支票来实现。

H

HA

参见高可用性。

异构数据库迁移

将源数据库迁移到使用不同数据库引擎的目标数据库（例如，从 Oracle 迁移到 Amazon Aurora）。异构迁移通常是重新架构工作的一部分，而转换架构可能是一项复杂的任务。AWS 提供了 AWS SCT 来帮助实现架构转换。

高可用性 (HA)

在遇到挑战或灾难时，工作负载无需干预即可连续运行的能力。HA 系统旨在自动进行故障转移、持续提供良好性能，并以最小的性能影响处理不同负载和故障。

历史数据库现代化

一种用于实现运营技术 (OT) 系统现代化和升级以更好满足制造业需求的方法。历史数据库是一种用于收集和存储工厂中各种来源数据的数据库。

同构数据库迁移

将源数据库迁移到共享同一数据库引擎的目标数据库（例如，从 Microsoft SQL Server 迁移到 Amazon RDS for SQL Server）。同构迁移通常是更换主机或更换平台工作的一部分。您可以使用本机数据库实用程序来迁移架构。

热数据

经常访问的数据，例如实时数据或近期的转化数据。这些数据通常需要高性能存储层或存储类别才能提供快速的查询响应。

修补程序

针对生产环境中关键问题的紧急修复。由于其紧迫性，修补程序通常是在典型的 DevOps 发布工作流程之外进行的。

hypercare 周期

割接之后，迁移团队立即管理和监控云中迁移的应用程序以解决任何问题的时间段。通常，这个周期持续 1-4 天。在 hypercare 周期结束时，迁移团队通常会将应用程序的责任移交给云运营团队。

I

IaC

参见基础架构即代码。

基于身份的策略

附加到一个或多个 IAM 委托人的策略，用于定义他们在 AWS Cloud 环境中的权限。

空闲应用程序

90 天内平均 CPU 和内存使用率在 5% 到 20% 之间的应用程序。在迁移项目中，通常会停用这些应用程序或将其保留在本地。

IIoT

参见工业物联网。

不可变的基础架构

一种为生产工作负载部署新基础架构，而不是更新、修补或修改现有基础架构的模型。不可变基础架构本质上比可变基础架构更一致、更可靠、更可预测。有关更多信息，请参阅 Well-Architected Framework 中的使用不可变基础架构 AWS 部署最佳实践。

入站（入口）VPC

在 AWS 多账户架构中，一种接受、检查和路由来自应用程序外部的网络连接的 VPC。AWS 安全参考架构建议使用入站、出站和检查 VPC 设置网络账户，保护应用程序与广泛的互联网之间的双向接口。

增量迁移

一种割接策略，在这种策略中，您可以将应用程序分成小部分进行迁移，而不是一次性完整割接。例如，您最初可能只将几个微服务或用户迁移到新系统。在确认一切正常后，您可以逐步迁移其他微服务或用户，直到停用遗留系统。这种策略降低了大规模迁移带来的风险。

工业 4.0

该术语由克劳斯·施瓦布（Klaus Schwab）于2016年推出，指的是通过连接、实时数据、自动化、分析和人工智能/机器学习的进步实现制造流程的现代化。

基础设施

应用程序环境中包含的所有资源和资产。

基础设施即代码（IaC）

通过一组配置文件预置和管理应用程序基础设施的过程。IaC 旨在帮助您集中管理基础设施、实现资源标准化和快速扩展，使新环境具有可重复性、可靠性和一致性。

工业物联网（IIoT）

在工业领域使用联网的传感器和设备，例如制造业、能源、汽车、医疗保健、生命科学和农业。有关更多信息，请参阅制定工业物联网（IIoT）数字化转型策略。

检查 VPC

在 AWS 多账户架构中，一种集中式 VPC，用于管理 VPC（相同或不同 AWS 区域）、互联网和本地网络之间的网络流量检查。AWS 安全参考架构建议使用入站、出站和检查 VPC 设置网络账户，保护应用程序与广泛的互联网之间的双向接口。

物联网（IoT）

由带有嵌入式传感器或处理器的连接物理对象组成的网络，这些传感器或处理器通过互联网或本地通信网络与其他设备和系统进行通信。有关更多信息，请参阅什么是 IoT？

可解释性

它是机器学习模型的一种特征，描述了人类可以理解模型的预测如何取决于其输入的程度。有关更多信息，请参阅使用 AWS 实现机器学习模型的可解释性。

IoT

参见物联网。

IT 信息库（ITIL）

提供 IT 服务并使这些服务符合业务要求的一套最佳实践。ITIL 是 ITSM 的基础。

IT 服务管理（ITSM）

为组织设计、实施、管理和支持 IT 服务的相关活动。有关将云运营与 ITSM 工具集成的信息，请参阅运营集成指南。

ITIL

请参阅 IT 信息库。

ITSM

请参阅 IT 服务管理。

L

基于标签的访问控制 (LBAC)

强制访问控制 (MAC) 的一种实施方式，其中明确为用户和数据本身分配了安全标签值。用户安全标签和数据安全标签之间的交集决定了用户可以看到哪些行和列。

登录区

landing zone 是一个架构精良的多账户 AWS 环境，具有可扩展性和安全性。这是一个起点，您的组织可以从这里放心地在安全和基础设施环境中快速启动和部署工作负载和应用程序。有关登录区的更多信息，请参阅设置安全且可扩展的多账户 AWS 环境。

大规模迁移

迁移 300 台或更多服务器。

LBAC

参见基于标签的访问控制。

最低权限

授予执行任务所需的最低权限的最佳安全实践。有关更多信息，请参阅 IAM 文档中的应用最低权限许可。

直接迁移

见 7 R。

小端序系统

一个先存储最低有效字节的系统。另请参见字节顺序。

下层环境

参见环境。

M

机器学习（ML）

一种使用算法和技术进行模式识别和学习的人工智能。ML 对记录的数据（例如物联网（IoT）数据）进行分析和学习，以生成基于模式的统计模型。有关更多信息，请参阅机器学习。

主分支

参见分支。

恶意软件

旨在危害计算机安全或隐私的软件。恶意软件可能会破坏计算机系统、泄露敏感信息或获得未经授权的访问。恶意软件的示例包括病毒、蠕虫、勒索软件、特洛伊木马、间谍软件和键盘记录器。

托管服务

AWS 服务 它 AWS 运行基础设施层、操作系统和平台，您可以访问端点来存储和检索数据。亚马逊简单存储服务 (Amazon S3) Service 和 Amazon DynamoDB 就是托管服务的示例。这些服务也称为抽象服务。

制造执行系统 (MES)

一种软件系统，用于跟踪、监控、记录和控制在车间将原材料转化为成品的生产过程。

MAP

参见迁移加速计划。

机制

一个完整的过程，在此过程中，您可以创建工具，推动工具的采用，然后检查结果以进行调整。机制是一种在运行过程中自我增强和改进的循环。有关更多信息，请参阅在 Well-Architect AWS ed 框架中构建机制。

成员账户

AWS 账户 除属于组织中的管理账户之外的所有账户 AWS Organizations。一个账户一次只能是一个组织的成员。

MES

参见制造执行系统。

消息队列遥测传输 (MQTT)

一种基于发布/订阅模式的轻量级 machine-to-machine (M2M) 通信协议，适用于资源受限的物联网设备。

微服务

一种小型独立服务，通过明确定义的 API 进行通信，通常由小型独立团队拥有。例如，保险系统可能包括映射到业务能力（如销售或营销）或子域（如购买、理赔或分析）的微服务。微服务的好处包括敏捷、灵活扩展、易于部署、可重复使用的代码和恢复能力。有关更多信息，请参阅使用 AWS 无服务器服务集成微服务。

微服务架构

一种使用独立组件构建应用程序的方法，这些组件将每个应用程序进程作为微服务运行。这些微服务使用轻量级 API 通过明确定义的接口进行通信。该架构中的每个微服务都可以更新、部署和扩展，以满足对应用程序特定功能的需求。有关更多信息，请参阅在上实现微服务。 AWS

迁移加速计划（MAP）

AWS 该计划提供咨询支持、培训和服务，以帮助组织为迁移到云奠定坚实的运营基础，并帮助抵消迁移的初始成本。MAP 提供了一种以系统的方式执行遗留迁移的迁移方法，以及一套用于自动执行和加速常见迁移场景的工具。

大规模迁移

将大部分应用程序组合分波迁移到云中的过程，在每一波中以更快的速度迁移更多应用程序。本阶段使用从早期阶段获得的最佳实践和经验教训，实施由团队、工具和流程组成的迁移工厂，通过自动化和敏捷交付简化工作负载的迁移。这是 AWS 迁移策略的第三阶段。

迁移工厂

跨职能团队，通过自动化、敏捷的方法简化工作负载迁移。迁移工厂团队通常包括运营、业务分析师和所有者、迁移工程师、开发 DevOps 人员和冲刺专业人员。20% 到 50% 的企业应用程序组合由可通过工厂方法优化的重复模式组成。有关更多信息，请参阅本内容集中有关迁移工厂的讨论和云迁移工厂指南。

迁移元数据

有关完成迁移所需的应用程序和服务器的信息。每种迁移模式都需要一套不同的迁移元数据。迁移元数据的示例包括目标子网、安全组和 AWS 账户。

迁移模式

一种可重复的迁移任务，详细列出了迁移策略、迁移目标以及所使用的迁移应用程序或服务。示例：使用 AWS 应用程序迁移服务重新托管向 Amazon EC2 的迁移。

迁移组合评测（MPA）

一种在线工具，可提供信息，用于验证迁移到的业务案例。 AWS Cloud MPA 提供了详细的组合评测（服务器规模调整、定价、TCO 比较、迁移成本分析）以及迁移计划（应用程序数据分析和数据收集、应用程序分组、迁移优先级排序和波次规划）。所有 AWS 顾问和 APN 合作伙伴顾问均可免费使用 MPA 工具（需要登录）。

迁移准备情况评测（MRA）

使用 AWS CAF 深入了解组织的云就绪状态、确定优势和劣势以及制定行动计划以缩小已发现差距的过程。有关更多信息，请参阅迁移准备指南。MRA 是 AWS 迁移策略的第一阶段。

迁移策略

用于将工作负载迁移到的方法 AWS Cloud。有关更多信息，请参阅此词汇表中的 7 R 条目和动员组织以加快大规模迁移。

ML

参见机器学习。

现代化

将过时的（原有的或单体）应用程序及其基础设施转变为云中敏捷、弹性和高度可用的系统，以降低成本、提高效率和利用创新。有关更多信息，请参阅中的应用程序现代化策略。 AWS Cloud

现代化准备情况评估

一种评估方式，有助于确定组织应用程序的现代化准备情况；确定收益、风险和依赖关系；确定组织能够在多大程度上支持这些应用程序的未来状态。评估结果是目标架构的蓝图、详细说明现代化进程发展阶段和里程碑的路线图以及解决已发现差距的行动计划。有关更多信息，请参阅中的评估应用程序的现代化准备情况 AWS Cloud。

单体应用程序（单体式）

作为具有紧密耦合进程的单个服务运行的应用程序。单体应用程序有几个缺点。如果某个应用程序功能的需求激增，则必须扩展整个架构。随着代码库的增长，添加或改进单体应用程序的功能也会变得更加复杂。若要解决这些问题，可以使用微服务架构。有关更多信息，请参阅将单体分解为微服务。

MPA

参见迁移组合评估。

MQTT

请参阅消息队列遥测传输。

多分类器

一种帮助为多个类别生成预测（预测两个以上结果之一）的过程。例如，ML 模型可能会询问“这个产品是书、汽车还是手机？”或“此客户最感兴趣什么类别的产品？”

可变基础架构

一种用于更新和修改现有生产工作负载基础架构的模型。为了提高一致性、可靠性和可预测性，Well-Architect AWS ed Framework 建议使用不可变基础设施作为最佳实践。

O

OAC

请参阅源站访问控制。

OAI

参见源访问身份。

OCM

参见组织变更管理。

离线迁移

一种迁移方法，在这种方法中，源工作负载会在迁移过程中停止运行。这种方法会延长停机时间，通常用于小型非关键工作负载。

OI

参见运营集成。

OLA

参见运营层协议。

在线迁移

一种迁移方法，在这种方法中，源工作负载无需离线即可复制到目标系统。在迁移过程中，连接工作负载的应用程序可以继续运行。这种方法的停机时间为零或最短，通常用于关键生产工作负载。

OPC-UA

参见开放流程通信-统一架构。

开放流程通信-统一架构 (OPC-UA)

一种用于工业自动化的 machine-to-machine （M2M）通信协议。OPC-UA 提供了数据加密、身份验证和授权方案的互操作性标准。

运营级别协议（OLA）

一项协议，阐明了 IT 职能部门承诺相互交付的内容，以支持服务水平协议（SLA）。

运营准备情况审查 (ORR)

一份问题清单和相关的最佳实践，可帮助您理解、评估、预防或缩小事件和可能的故障的范围。有关更多信息，请参阅 Well-Architecte AWS d Frame work 中的运营准备情况评估 (ORR)。

操作技术 (OT)

与物理环境配合使用以控制工业运营、设备和基础设施的硬件和软件系统。在制造业中，OT 和信息技术 (IT) 系统的集成是工业 4.0 转型的重点。

运营整合（OI）

在云中实现运营现代化的过程，包括就绪计划、自动化和集成。有关更多信息，请参阅运营整合指南。

组织跟踪

由 AWS CloudTrail 此创建的跟踪记录组织 AWS 账户 中所有人的所有事件 AWS Organizations。该跟踪是在每个 AWS 账户 中创建的，属于组织的一部分，并跟踪每个账户的活动。有关更多信息，请参阅 CloudTrail文档中的为组织创建跟踪。

组织变革管理（OCM）

一个从人员、文化和领导力角度管理重大、颠覆性业务转型的框架。OCM 通过加快变革采用、解决过渡问题以及推动文化和组织变革，帮助组织为新系统和战略做好准备和过渡。在 AWS 迁移策略中，该框架被称为人员加速，因为云采用项目需要变更的速度。有关更多信息，请参阅 OCM 指南。

来源访问控制（OAC）

在中 CloudFront，一个增强的选项，用于限制访问以保护您的亚马逊简单存储服务 (Amazon S3) 内容。OAC 全部支持所有 S3 存储桶 AWS 区域、使用 AWS KMS (SSE-KMS) 进行服务器端加密，以及对 S3 存储桶的动态PUT和DELETE请求。

来源访问身份（OAI）

在中 CloudFront，一个用于限制访问权限以保护您的 Amazon S3 内容的选项。当您使用 OAI 时，CloudFront 会创建一个 Amazon S3 可以对其进行身份验证的委托人。经过身份验证的委托人只能通过特定 CloudFront 分配访问 S3 存储桶中的内容。另请参阅 OAC，其中提供了更精细和增强的访问控制。

或者

参见运营准备情况审查。

OT

参见运营技术。

出站（出口）VPC

在 AWS 多账户架构中，一种处理从应用程序内部启动的网络连接的 VPC。AWS 安全参考架构建议使用入站、出站和检查 VPC 设置网络账户，保护应用程序与广泛的互联网之间的双向接口。

P

权限边界

附加到 IAM 主体的 IAM 管理策略，用于设置用户或角色可以拥有的最大权限。有关更多信息，请参阅 IAM 文档中的权限边界。

个人身份信息 (PII)

直接查看其他相关数据或与之配对时可用于合理推断个人身份的信息。PII 的示例包括姓名、地址和联系信息。

PII

查看个人身份信息。

playbook

一套预定义的步骤，用于捕获与迁移相关的工作，例如在云中交付核心运营功能。playbook 可以采用脚本、自动化运行手册的形式，也可以是操作现代化环境所需的流程或步骤的摘要。

PLC

参见可编程逻辑控制器。

PLM

参见产品生命周期管理。

策略

一个对象，可以在中定义权限（参见基于身份的策略）、指定访问条件（参见基于资源的策略）或定义组织中所有账户的最大权限 AWS Organizations （参见服务控制策略）。

多语言持久性

根据数据访问模式和其他要求，独立选择微服务的数据存储技术。如果您的微服务采用相同的数据存储技术，它们可能会遇到实现难题或性能不佳。如果微服务使用最适合其需求的数据存储，则可以更轻松地实现微服务，并获得更好的性能和可扩展性。有关更多信息，请参阅在微服务中实现数据持久性。

组合评测

一个发现、分析和确定应用程序组合优先级以规划迁移的过程。有关更多信息，请参阅评估迁移准备情况。

谓词

返回`true`或的查询条件`false`，通常位于子`WHERE`句中。

谓词下推

一种数据库查询优化技术，可在传输前筛选查询中的数据。这减少了必须从关系数据库检索和处理的数据量，并提高了查询性能。

预防性控制

一种安全控制，旨在防止事件发生。这些控制是第一道防线，帮助防止未经授权的访问或对网络的意外更改。有关更多信息，请参阅在 AWS上实施安全控制中的预防性控制。

主体

中 AWS 可以执行操作和访问资源的实体。此实体通常是 IAM 角色的根用户或用户。 AWS 账户有关更多信息，请参阅 IAM 文档中角色术语和概念中的主体。

隐私设计

一种贯穿整个工程化过程考虑隐私的系统工程方法。

私有托管区

私有托管区就是一个容器，其中包含的信息说明您希望 Amazon Route 53 如何响应一个或多个 VPC 中的某个域及其子域的 DNS 查询。有关更多信息，请参阅 Route 53 文档中的私有托管区的使用。

主动控制

一种安全控制措施，旨在防止部署不合规的资源。这些控件会在资源置备之前对其进行扫描。如果资源与控件不兼容，则不会对其进行配置。有关更多信息，请参阅 AWS Control Tower 文档中的控制参考指南，并参见在上实施安全控制中的主动控制 AWS。

产品生命周期管理 (PLM)

在产品的整个生命周期中，从设计、开发和上市，到成长和成熟，再到衰落和移除，对产品进行数据和流程的管理。

生产环境

参见环境。

可编程逻辑控制器 (PLC)

在制造业中，一种高度可靠、适应性强的计算机，用于监控机器并实现制造过程自动化。

假名化

用占位符值替换数据集中个人标识符的过程。假名化可以帮助保护个人隐私。假名化数据仍被视为个人数据。

发布/订阅（发布/订阅）

一种支持微服务间异步通信的模式，以提高可扩展性和响应能力。例如，在基于微服务的 MES 中，微服务可以将事件消息发布到其他微服务可以订阅的频道。系统可以在不更改发布服务的情况下添加新的微服务。

Q

查询计划

一系列步骤，例如指令，用于访问 SQL 关系数据库系统中的数据。

查询计划回归

当数据库服务优化程序选择的最佳计划不如数据库环境发生特定变化之前时。这可能是由统计数据、约束、环境设置、查询参数绑定更改和数据库引擎更新造成的。

R

RACI 矩阵

参见 “负责任、负责、咨询、知情”（RACI）。

勒索软件

一种恶意软件，旨在阻止对计算机系统或数据的访问，直到付款为止。

RASCI 矩阵

参见 “负责任、负责、咨询、知情”（RACI）。

RCAC

请参阅行和列访问控制。

只读副本

用于只读目的的数据库副本。您可以将查询路由到只读副本，以减轻主数据库的负载。

重新架构师

见 7 R。

恢复点目标 (RPO)

自上一个数据恢复点以来可接受的最长时间。这决定了从上一个恢复点到服务中断之间可接受的数据丢失情况。

恢复时间目标 (RTO)

服务中断和服务恢复之间可接受的最大延迟。

重构

见 7 R。

区域

地理区域内的 AWS 资源集合。每一个 AWS 区域 都相互隔离，彼此独立，以提供容错、稳定性和弹性。有关更多信息，请参阅指定 AWS 区域 您的账户可以使用的账户。

回归

一种预测数值的 ML 技术。例如，要解决“这套房子的售价是多少？”的问题 ML 模型可以使用线性回归模型，根据房屋的已知事实（如建筑面积）来预测房屋的销售价格。

重新托管

见 7 R。

版本

在部署过程中，推动生产环境变更的行为。

搬迁

见 7 R。

更换平台

见 7 R。

回购

见 7 R。

故障恢复能力

应用程序抵御中断或从中断中恢复的能力。在中规划弹性时，高可用性和灾难恢复是常见的考虑因素。 AWS Cloud有关更多信息，请参阅AWS Cloud 弹性。

基于资源的策略

一种附加到资源的策略，例如 AmazonS3 存储桶、端点或加密密钥。此类策略指定了允许哪些主体访问、支持的操作以及必须满足的任何其他条件。

责任、问责、咨询和知情（RACI）矩阵

定义参与迁移活动和云运营的所有各方的角色和责任的矩阵。矩阵名称源自矩阵中定义的责任类型：负责 (R)、问责 (A)、咨询 (C) 和知情 (I)。支持 (S) 类型是可选的。如果包括支持，则该矩阵称为 RASCI 矩阵，如果将其排除在外，则称为 RACI 矩阵。

响应性控制

一种安全控制，旨在推动对不良事件或偏离安全基线的情况进行修复。有关更多信息，请参阅在AWS上实施安全控制中的响应性控制。

保留

见 7 R。

退休

见 7 R。

旋转

定期更新密钥以使攻击者更难访问凭据的过程。

行列访问控制 (RCAC)

使用已定义访问规则的基本、灵活的 SQL 表达式。RCAC 由行权限和列掩码组成。

RPO

参见恢复点目标。

RTO

参见恢复时间目标。

运行手册

执行特定任务所需的一套手动或自动程序。它们通常是为了简化重复性操作或高错误率的程序而设计的。

S

SAML 2.0

许多身份提供商 (IdPs) 使用的开放标准。此功能支持联合单点登录 (SSO)，因此用户无需在 IAM 中为组织中的所有人创建用户即可登录 AWS Management Console 或调用 AWS API 操作。有关基于 SAML 2.0 的联合身份验证的更多信息，请参阅 IAM 文档中的关于基于 SAML 2.0 的联合身份验证。

SCADA

参见监督控制和数据采集。

SCP

参见服务控制政策。

secret

在中 AWS Secrets Manager，您以加密形式存储的机密或受限信息，例如密码或用户凭证。它由密钥值及其元数据组成。密钥值可以是二进制、单个字符串或多个字符串。有关更多信息，请参阅 Secret s Manager 密钥中有什么？ 在 Secrets Manager 文档中。

安全控制

一种技术或管理防护机制，可防止、检测或降低威胁行为体利用安全漏洞的能力。安全控制主要有四种类型：预防性、侦测、响应式和主动式。

安全加固

缩小攻击面，使其更能抵御攻击的过程。这可能包括删除不再需要的资源、实施授予最低权限的最佳安全实践或停用配置文件中不必要的功能等操作。

安全信息和事件管理（SIEM）系统

结合了安全信息管理（SIM）和安全事件管理（SEM）系统的工具和服务。SIEM 系统会收集、监控和分析来自服务器、网络、设备和其他来源的数据，以检测威胁和安全漏洞，并生成警报。

安全响应自动化

一种预定义和编程的操作，旨在自动响应或修复安全事件。这些自动化可作为侦探或响应式安全控制措施，帮助您实施 AWS 安全最佳实践。自动响应操作的示例包括修改 VPC 安全组、修补 Amazon EC2 实例或轮换证书。

服务器端加密

在目的地对数据进行加密，由接收数据 AWS 服务 的人加密。

服务控制策略（SCP）

一种策略，用于集中控制 AWS Organizations的组织中所有账户的权限。SCP 为管理员可以委托给用户或角色的操作定义了防护机制或设定了限制。您可以将 SCP 用作允许列表或拒绝列表，指定允许或禁止哪些服务或操作。有关更多信息，请参阅 AWS Organizations 文档中的服务控制策略。

服务端点

的入口点的 URL AWS 服务。您可以使用端点，通过编程方式连接到目标服务。有关更多信息，请参阅 AWS 一般参考 中的 AWS 服务 端点。

服务水平协议（SLA）

一份协议，阐明了 IT 团队承诺向客户交付的内容，比如服务正常运行时间和性能。

服务级别指示器 (SLI)

对服务性能方面的衡量，例如其错误率、可用性或吞吐量。

服务级别目标 (SLO)

代表服务运行状况的目标指标，由服务级别指标衡量。

责任共担模式

描述您在云安全与合规方面共同承担 AWS 的责任的模型。 AWS 负责云的安全，而您则负责云中的安全。有关更多信息，请参阅责任共担模式。

暹粒

参见安全信息和事件管理系统。

单点故障 (SPOF)

应用程序的单个关键组件出现故障，可能会中断系统。

SLA

参见服务级别协议。

SLI

参见服务级别指标。

SLO

参见服务级别目标。

split-and-seed 模型

一种扩展和加速现代化项目的模式。随着新功能和产品发布的定义，核心团队会拆分以创建新的产品团队。这有助于扩展组织的能力和服务，提高开发人员的工作效率，支持快速创新。有关更多信息，请参阅中的分阶段实现应用程序现代化的方法。 AWS Cloud

恶作剧

参见单点故障。

星型架构

一种数据库组织结构，它使用一个大型事实表来存储交易数据或测量数据，并使用一个或多个较小的维度表来存储数据属性。此结构专为在数据仓库中使用或用于商业智能目的而设计。

strangler fig 模式

一种通过逐步重写和替换系统功能直至可以停用原有的系统来实现单体系统现代化的方法。这种模式用无花果藤作为类比，这种藤蔓成长为一棵树，最终战胜并取代了宿主。该模式是由 Martin Fowler 提出的，作为重写单体系统时管理风险的一种方法。有关如何应用此模式的示例，请参阅使用容器和 Amazon API Gateway 逐步将原有的 Microsoft ASP.NET（ASMX）Web 服务现代化。

子网

您的 VPC 内的一个 IP 地址范围。子网必须位于单个可用区中。

监控和数据采集 (SCADA)

在制造业中，一种使用硬件和软件来监控有形资产和生产操作的系统。

对称加密

一种加密算法，它使用相同的密钥来加密和解密数据。

综合测试

以模拟用户交互的方式测试系统，以检测潜在问题或监控性能。你可以使用 Amazon S CloudWatch ynthetics 来创建这些测试。

T

标签

键值对，充当用于组织资源的元数据。 AWS 标签可帮助您管理、识别、组织、搜索和筛选资源。有关更多信息，请参阅标记您的 AWS 资源。

目标变量

您在监督式 ML 中尝试预测的值。这也被称为结果变量。例如，在制造环境中，目标变量可能是产品缺陷。

任务列表

一种通过运行手册用于跟踪进度的工具。任务列表包含运行手册的概述和要完成的常规任务列表。对于每项常规任务，它包括预计所需时间、所有者和进度。

测试环境

参见环境。

训练

为您的 ML 模型提供学习数据。训练数据必须包含正确答案。学习算法在训练数据中查找将输入数据属性映射到目标（您希望预测的答案）的模式。然后输出捕获这些模式的 ML 模型。然后，您可以使用 ML 模型对不知道目标的新数据进行预测。

中转网关

中转网关是网络中转中心，您可用它来互连 VPC 和本地网络。有关更多信息，请参阅 AWS Transit Gateway 文档中的什么是公交网关。

基于中继的工作流程

一种方法，开发人员在功能分支中本地构建和测试功能，然后将这些更改合并到主分支中。然后，按顺序将主分支构建到开发、预生产和生产环境。

可信访问权限

向您指定的服务授予权限，该服务可以代表您在其账户中执行任务。 AWS Organizations 当需要服务相关的角色时，受信任的服务会在每个账户中创建一个角色，为您执行管理任务。有关更多信息，请参阅 AWS Organizations 文档中的AWS Organizations 与其他 AWS 服务一起使用。

优化

更改训练过程的各个方面，以提高 ML 模型的准确性。例如，您可以通过生成标签集、添加标签，并在不同的设置下多次重复这些步骤来优化模型，从而训练 ML 模型。

双披萨团队

一个小 DevOps 团队，你可以用两个披萨来喂食。双披萨团队的规模可确保在软件开发过程中充分协作。

U

不确定性

这一概念指的是不精确、不完整或未知的信息，这些信息可能会破坏预测式 ML 模型的可靠性。不确定性有两种类型：认知不确定性是由有限的、不完整的数据造成的，而偶然不确定性是由数据中固有的噪声和随机性导致的。有关更多信息，请参阅量化深度学习系统中的不确定性指南。

无差别任务

也称为繁重工作，即创建和运行应用程序所必需的工作，但不能为最终用户提供直接价值或竞争优势。无差别任务的示例包括采购、维护和容量规划。

上层环境

参见环境。

V

vacuum 操作

一种数据库维护操作，包括在增量更新后进行清理，以回收存储空间并提高性能。

版本控制

跟踪更改的过程和工具，例如存储库中源代码的更改。

VPC 对等连接

两个 VPC 之间的连接，允许您使用私有 IP 地址路由流量。有关更多信息，请参阅 Amazon VPC 文档中的什么是 VPC 对等连接。

漏洞

损害系统安全的软件缺陷或硬件缺陷。

W

热缓存

一种包含经常访问的当前相关数据的缓冲区缓存。数据库实例可以从缓冲区缓存读取，这比从主内存或磁盘读取要快。

暖数据

不常访问的数据。查询此类数据时，通常可以接受中速查询。

窗口函数

一个 SQL 函数，用于对一组以某种方式与当前记录相关的行进行计算。窗口函数对于处理任务很有用，例如计算移动平均线或根据当前行的相对位置访问行的值。

工作负载

一系列资源和代码，它们可以提供商业价值，如面向客户的应用程序或后端过程。

工作流

迁移项目中负责一组特定任务的职能小组。每个工作流都是独立的，但支持项目中的其他工作流。例如，组合工作流负责确定应用程序的优先级、波次规划和收集迁移元数据。组合工作流将这些资产交付给迁移工作流，然后迁移服务器和应用程序。

蠕虫

参见一次写入，多读。

WQF

请参阅 AWS 工作负载资格框架。

一次写入，多次读取 (WORM)

一种存储模型，它可以一次写入数据并防止数据被删除或修改。授权用户可以根据需要多次读取数据，但他们无法对其进行更改。这种数据存储基础架构被认为是不可变的。

Z

零日漏洞利用

一种利用未修补漏洞的攻击，通常是恶意软件。

零日漏洞

生产系统中不可避免的缺陷或漏洞。威胁主体可能利用这种类型的漏洞攻击系统。开发人员经常因攻击而意识到该漏洞。

僵尸应用程序

平均 CPU 和内存使用率低于 5% 的应用程序。在迁移项目中，通常会停用这些应用程序。

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。