aws

开发人员指南

AWS HealthLake

AWS HealthLake: 开发人员指南

Table of Contents

什么是 AWS HealthLake？

AWS HealthLake 是一项HIPAA符合条件的利用医疗保健互操作性 FHIR (R4) 规范进行临床数据摄取、存储和分析的服务。

> **Note**
>
> 2023 年 2 月 20 日之后，默认情况下， HealthLake 数据存储不使用集成的自然语言处理 (NLP)。如果您有兴趣在数据存储上启用此功能，请参阅 “故障排除” 一章如何开启 HealthLake 集成的自然语言处理功能？中的。

Health 数据往往不完整且不一致。它通常也是非结构化的，信息包含在临床记录、实验室报告、保险索赔、医疗图像、录制的对话和时间序列数据（例如心脏ECG或大脑EEG痕迹）中。

医疗保健提供商可以使用 HealthLake 在 AWS 云端存储、转换、查询和分析数据。使用 HealthLake 集成的医学自然语言处理 (NLP) 功能，您可以分析来自不同来源的非结构化临床文本。 HealthLake 使用自然语言处理模型转换非结构化数据，并提供强大的查询和搜索功能。您可以使用 HealthLake 以安全、合规且可审计的方式对患者信息进行组织、索引和结构化。

HealthLake 还与亚马逊 Athena 和 Lake Format AWS ion 集成。您可以使用此集成来查询您的数据存储SQL。

的好处 AWS HealthLake

借助 AWS HealthLake，您可以：

- 快速轻松地摄取健康数据 — 您可以将本地快速医疗互操作性资源 (FHIR) 文件（包括临床记录、实验室报告、保险索赔等）批量导入到亚马逊简单存储服务 (Amazon S3) 存储桶。然后，您可以在下游应用程序或工作流程中使用这些数据。
- 使用FHIRRESTAPI操作- HealthLake 支持使用FHIRRESTAPI操作对数据存储执行 CRUD (Create/Read/Update/Delete) 操作。 FHIR还支持搜索。
- 以@@ 安全、HIPAA符合条件且可审计的方式将您的数据存储在 AWS 云端 — 您可以按该FHIR格式存储数据，以便于查询。 HealthLake 创建按时间顺序排列的每位患者的病史的完整视图，并以 R4 FHIR 标准格式对其进行构建。

- Athena 集成 HealthLake — 与 Athena 的集成意味着您可以创建基于SQL强大功能的查询，用于创建和保存复杂的筛选条件。然后，您可以在下游应用程序中使用这些数据，例如 SageMaker AI 来训练机器学习模型，或者使用 Amazon QuickSight 来创建仪表板和数据可视化。
- 使用专门的机器学习 (ML) 模型转换非结构化数据 — 使用 Amazon Comprehend Medical HealthLake 提供集成的医学自然语言处理 (NLP)。原始医学文本数据使用专门的 ML 模型进行转换。这些模型经过训练，可以理解和从非结构化医疗保健数据中提取有意义的信息。借助综合医疗 NLP，您可以自动从医学文本中提取实体（例如医疗程序和药物）、实体关系（例如药物及其剂量）和实体特征（例如，阳性或阴性检测结果或手术时间）数据。 HealthLake 然后根据特征、症状和状况创建新资源。它们被添加为新的条件、观测值和 MedicationStatement 资源类型。

HealthLake 用例

您可以 HealthLake 用于以下医疗保健应用程序：

- 人口健康管理 — HealthLake 帮助医疗保健组织分析人口健康趋势、结果和成本。这可以帮助组织确定最适合患者群体的干预措施，并选择更好的护理管理方案。
- 提高护理质量 — 通过汇编患者病史的完整视图， HealthLake 帮助医院、健康保险公司和生命科学组织缩小护理差距，提高护理质量并降低成本。
- 优化医院效率 — HealthLake 为医院提供关键分析和机器学习工具，以提高效率和减少医院浪费。

正在访问 HealthLake

您可以 HealthLake 通过 AWS Management Console、 AWS Command Line Interface (AWS CLI) 或 AWS SDKs。

1. AWS Management Console — 提供可用于访问的 Web 界面 HealthLake。
2. AWS Command Line Interface (AWS CLI) — 为各种 AWS 服务提供命令，包括 HealthLake Windows、macOS 和 Linux，并支持这些服务。有关安装的更多信息 AWS CLI，请参阅AWS Command Line Interface。
3. AWS SDKs— AWS 提供SDKs（软件开发套件），其中包括适用于各种编程语言和平台（Java、Python、Ruby 等）的库和示例代码。 NET、iOS、安卓等）。SDKs提供了一种便捷的方法来创建对 HealthLake 和的编程访问 AWS。有关更多信息，请参阅适用AWSSDK于 Python 的。

HIPAA资格和数据安全

这是一项HIPAA符合条件的服务。有关 AWS《1996 年美国健康保险流通与责任法案》(HIPAA) 以及使用 AWS 服务处理、存储和传输受保护的健康信息 (PHI) 的更多信息，请参阅HIPAA概述。

必须对与 HealthLake 包含个人身份信息 (PII) 的连接进行加密。默认情况下，所有连接都要 HealthLake 使用HTTPSTLS。 HealthLake 存储加密的客户内容，并遵循责任AWS共担原则。

定价

有关 HealthLake 定价的信息，请参阅定AWS HealthLake 价页面。为了更好地估算与之相关的潜在成本 HealthLake，您可以使用定HealthLake 价计算器。

如何 AWS HealthLake 运作

AWS HealthLake 使用医疗保健互操作性 FHIR (R4) 规范创建存储健康记录的数据存储。使用 HealthLake，您可以执行以下任务。

> **Note**
>
> 2023 年 2 月 20 日之后，默认情况下， HealthLake 数据存储不使用集成的自然语言处理 (NLP)。如果您有兴趣在数据存储上启用此功能，请参阅 “故障排除” 一章如何开启 HealthLake 集成的自然语言处理功能？中的。

- 创建、监控和删除数据存储。
- 用于`StartFHIRImportJob`将医疗保健数据从亚马逊简单存储服务 (Amazon S3) 存储桶批量导入数据存储。
- 使用创建、读取、更新和删除 (CRUD) 操作来管理存储在数据存储中的数据。
- SQL在 Amazon Athena 中使用来查询您的数据存储。
- 在FHIRRESTAPI操作中使用HTTP客户端搜索您的数据存储。
- 让 Amazon Comprehend M API edical 操作能够使用自然语言处理 () 在您的数据中搜索医学见解。NLP

创建和监控数据存储

使用 HealthLake，您可以创建和监控可以存储快速医疗互操作性资源 (FHIR) 数据的数据存储。

要创建新的数据存储，可以使用 C reateFHIRDatastore 或 HealthLake 控制台。要查看数据存储的状态，请使用 D escribeFHIRDatastore。 要查看多个活动数据存储的状态，请使用 L istFHIRDatastores。 要删除数据存储，请使用 D eleteFHIRDatastore。

FHIRRESTAPI操作

您可以使用这些FHIRRESTAPI操作对 HealthLake 数据存储执行创建、读取、更新、删除 (CRUD) 操作。要详细了解如何 HealthLake 支持这些FHIRRESTAPI操作，请参阅使用与 HealthLake 数据存储的FHIRRESTAPI交互。

通过资源扩展自动生成FHIR DocumentReference 资源

> **Note**
>
> 当您创建 HealthLake 数据存储并添加包含的数据时DocumentReference，您的 AWS 账户将产生费用。有关更多详细信息，请参阅AWS HealthLake 定价。

HealthLake 提供了NLP在DocumentReference资源类型中找到的文档。要分析文本，请 HealthLake 使用以下 Amazon Comprehend Medical 操作。API

- `DetectEntitiesV2`：检查各种医疗实体的临床文本，并返回有关它们的特定信息，例如实体类别、位置和置信度分数。
- `InferICD10CM`: 检查临床文本，将疾病检测为患者记录中列出的实体，并将这些实体与疾病控制中心 ICD -10-CM知识库中的标准化概念标识符关联起来。
- `InferRxNorm`: 检查临床文本，将药物检测为患者记录中列出的实体，并链接到国家医学图书馆 RxNorm 数据库中标准化概念标识符。

HealthLake 将DocumentReference资源类型中的数据添加到您的数据存储时，会自动对其进行分析。原始DocumentReference资源文件保持不变。提取的医疗信息会自动FHIR附加为符合标准的扩展名。要详细了解中的NLP工作原理 HealthLake，请参阅使用基于资源类型的自然语言处理 (NLP) 的自动FHIR DocumentReference 资源生成 AWS HealthLake。

使用SQL基于基础的查询进行搜索

> **Note**
>
> 对于在 2022 年 11 月 14 日之前创建的数据存储，您的搜索仅限于FHIRRESTAPI操作。要对数据存储中的数据使用SQL基于查询的方式，请参阅在 Amazon Athena SQL 中使用查询 AWS HealthLake 数据存储。 HealthLake

Amazon Athena 是一项SQL基于无服务器的查询服务。 HealthLake 数据存储作为 Apache Iceberg 表提取到 Athena 中。这些表旨在支持大型分析数据集。在 Athena 中，FHIR每种资源类型都以表格的形式表示。使用 Athena，您只能在数据存储上READ发出请求。要了解有关SQL基于基础的搜索的更多信息，请参阅使用查询您的 HealthLake 数据存储 SQL。

使用FHIRRESTAPI操作进行搜索

您可以通过使用支持的搜索参数指定资源类型来搜索存储在数据存储中的健康记录，也可以使用在服务器中找到的资源 ID 来搜索存储在数据存储中的健康记录，而无需指定资源类型。要了解有关使用FHIRRESTAPI操作进行搜索的更多信息，请参阅使用与 HealthLake 数据存储的FHIRRESTAPI交互。

数据导入操作

AWS HealthLake 用于从 Amazon S3 存储桶中批量导入您的文件。使用控制台或 S tartFHIRImport Job 开始导入任务。导入文件后，您可以使用 D escribeFHIRImport Job 监视作业的状态。导入任务完成后，可以将数据添加到 Athena、进行转换或分析并在下游应用程序中使用。

数据导出操作

用于 HealthLake 将您的文件批量导出到 Amazon S3 存储桶。使用控制台或 S tartFHIRExport Job 开始导出任务。导出文件后，您可以使用 D escribeFHIRExport Job 监控作业的状态并查看其属性。导出任务完成后，您可以使用 Amazon QuickSight 对数据进行可视化，也可以使用其他 AWS 服务进行访问。

AWS HealthLake 支持的FHIR配置文件验证

HealthLake 支持基本的 FHIRR4 规范。R4 规范中包括FHIR配置文件。配置文件用于FHIR资源类型，使用基本资源类型的约束和/或扩展来定义更具体的资源类型定义。例如，FHIR配置文件可以识别必填字段，例如扩展名和值集。一个资源可以支持多个配置文件。所有 HealthLake 数据存储都支持使用FHIR配置文件。

向数据存储中添加数据时，不需要添加FHIR HealthLake 配置文件。如果在添加或更新资源时未指定FHIR配置文件，则仅根据基本 FHIR R4 架构对资源进行验证。

FHIR资源符合的配置文件在被摄取到资源中之前会包含在资源中。 HealthLake HealthLake 将指定的FHIR配置文件添加到您的 HealthLake 数据存储中时对其进行验证。

FHIR配置文件在实施指南中指定。 HealthLake 验证以下实现指南中定义的FHIR配置文件。

支持的FHIR配置文件由 HealthLake

名称	版本	实施指南	能力
美国核心	3.1.1	http://hl7.org/fhir/us/core/STU3.1.1/	默认
美国核心	4.0.0	https://hl7.org/fhir/us/core/STU4/index.html	支持
CARIN蓝色按钮	1.1.0	http://hl7.org/fhir/us/carin-bb/STU1.1/	默认
CARIN蓝色按钮	1.0.0	https://hl7.org/fhir/us/carin-bb/STU1/	支持
Da Vinci Payer Data Exchange	1.0.0	https://hl7.org/fhir/us/davinci-pdex/	默认
Da Vinci Health 记录交易所 () HRex	0.2.0	https://hl7.org/fhir/us/davinci-hrex/2020Sep/	默认
DaVinci PDEX计划网	1.1.0	https://hl7.org/fhir/us/davinci-pdex-plan-net/STU1.1/	默认

名称	版本	实施指南	能力
DaVinci PDEX计划网	1.0.0	https://hl7.org/fhir/us/davinci-pdex-plan-net/STU1/	支持
DaVinci Payer Data Exchange (PDex)《美国药品处方集》	1.1.0	https://hl7.org/fhir/us/davinci-drug-formulary/STU1.1/	默认
DaVinci Payer Data Exchange (PDex)《美国药品处方集》	1.0.1	https://hl7.org/fhir/us/davinci-drug-formulary/STU1.0.1/	支持
国家卫生局的Ayushman Bharat数字使命 () ABDM	2.0	https://www.nrces.in/ndhm/fhir/r4/index.html	默认

验证资源中指定的FHIR配置文件

要验证FHIR配置文件，请使用实施指南中URL指定的配置文件将其添加到单个资源的`profile`元素中。

FHIR向数据存储中添加新资源时，配置文件会被验证。要添加新资源，您可以使用 S tartFHIRImport Job API 操作，POST请求添加新资源， PUT 或者请求更新现有资源。

Example — 查看资源中引用了哪个FHIR配置文件

配置文件URL将添加到`"meta" : "profile"`键值对中的`profile`元素中。为清楚起见，此资源已被截断。

```
{
 "resourceType": "Patient",
 "id": "abcd1234efgh5678hijk9012",
 "meta": {
  "lastUpdated": "2023-05-30T00:48:07.8443764-07:00",
```

```
            "profile": [
             "http://hl7.org/fhir/us/core/StructureDefinition/us-core-patient"
            ]
           }
          }
```

Example — 如何引用非默认支持的配置文件 FHIR

根据支持的非默认配置文件进行验证（例如 carinBB 1.0.0)-在元素中添加URL带有版本（用 “|” 分隔）的配置文件和基本配置文件URL。`meta.profile`为清楚起见，此示例资源已被截断。

```
          {
           "resourceType": "ExplanationOfBenefit",
           "id": "sample-EOB",
           "meta": {
            "lastUpdated": "2024-02-02T05:56:09.4+00:00",
            "profile": [
             "http://hl7.org/fhir/us/carin-bb/StructureDefinition/C4BB-
ExplanationOfBenefit-Pharmacy|1.0.0",
              "http://hl7.org/fhir/us/carin-bb/StructureDefinition/C4BB-
ExplanationOfBenefit-Pharmacy“
            ]
           }
          }
```

预加载的数据类型

HealthLake 仅支持SYNTHEA作为预加载的数据类型。S@@ ynthea 是一种合成患者生成器，用于对模型生成的患者的病史进行建模。它是一个开源 Git 存储库， HealthLake 允许生成FHIR符合 R4 标准的资源包，这样用户就可以在不使用实际患者数据的情况下测试模型。

预加载的数据存储中提供以下资源类型。

支持的 Synthea 资源类型

AllergyIntolerance	位置
CarePlan	MedicationAdministration
CareTeam	MedicationRequest
Claim	观察
状况	组织
设备	病人
DiagnosticReport	从业者
遭遇	PractitionerRole
ExplanationofBenefit	过程
ImagingStudy	出处
免疫接种	

设置开始使用的权限 AWS HealthLake

在本章中，您将使用 AWS Management Console 来设置开始使用 AWS HealthLake 和创建数据存储所需的权限。要设置创建数据存储的权限，您需要创建一个既是数据湖管理员又 HealthLake 是管理员的IAM用户或角色。您可以在 Lake Formation 中将此用户设置为数据 AWS 湖管理员。数据湖管理员授予 Lake Formation 访问使用亚马逊 Athena 查询数据存储所需的资源的权限。

在中创建数据存储后 HealthLake，您可以设置将文件导入数据存储或导出文件的权限。有关设置导入文件权限的信息，请参见为导入任务设置权限。有关设置导出文件权限的信息，请参阅为导出任务设置权限。

主题

- 注册获取 AWS 账户
- 创建具有管理访问权限的用户
- 配置要使用的IAM用户或角色 HealthLake （ IAM管理员 ）
- 在 Lake Formation 中添加用户或角色作为数据湖IAM管理员（ 管理员 ）

注册获取 AWS 账户

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/注册。
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。

AWS 注册过程完成后会向您发送一封确认电子邮件。您可以随时前往 https://aws.amazon.com/并选择“我的账户”，查看您当前的账户活动并管理您的账户。

创建具有管理访问权限的用户

注册后，请保护您的安全 AWS 账户 AWS 账户根用户 AWS IAM Identity Center，启用并创建管理用户，这样您就不会使用 root 用户执行日常任务。

保护你的 AWS 账户根用户

1. 选择 Root 用户并输入您的 AWS 账户 电子邮件地址，以账户所有者的身份登录。AWS Management Console在下一页上，输入您的密码。

   要获取使用根用户登录方面的帮助，请参阅《AWS 登录 用户指南》中的以根用户身份登录。
2. 为您的 root 用户开启多重身份验证 (MFA)。

   有关说明，请参阅《用户指南》中的 “为 AWS 账户 root 用户（控制台）启用虚拟MFA设备” IAM。

创建具有管理访问权限的用户

1. 启用IAM身份中心。

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的启用 AWS IAM Identity Center。
2. 在 IAM Identity Center 中，向用户授予管理访问权限。

   有关使用 IAM Identity Center 目录 作为身份源的教程，请参阅《用户指南》 IAM Identity Center 目录中的使用默认设置配置AWS IAM Identity Center 用户访问权限。

以具有管理访问权限的用户身份登录

- 要使用您的 Ident IAM ity Center 用户登录URL，请使用您在创建 Ident IAM ity Center 用户时发送到您的电子邮件地址的登录信息。

  有关使用 Ident IAM ity Center 用户登录的帮助，请参阅AWS 登录 用户指南中的登录 AWS 访问门户。

将访问权限分配给其他用户

1. 在 IAM Identity Center 中，创建一个遵循应用最低权限权限的最佳实践的权限集。

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的创建权限集。

2. 将用户分配到一个组，然后为该组分配单点登录访问权限。

   有关说明，请参阅《AWS IAM Identity Center 用户指南》中的添加组。

配置要使用的IAM用户或角色 HealthLake （ IAM管理员 ）

> **角色：IAM管理员**
>
> 可以创建IAM用户和角色并可以添加数据湖管理员的用户。

本主题中的这些步骤必须由IAM管理员执行。

要将您的 HealthLake 数据存储连接到 Athena，您需要IAM创建一个既是数据湖管理员又是管理员的用户或角色。 HealthLake 此新用户或角色授予通过 AWS Lake Formation 访问数据存储中资源的权限，并将`AmazonHealthLakeFullAccess` AWS 托管策略添加到其用户或角色中。

> **Important**
>
> 作为数据湖管理员的IAM用户或角色无法创建新的数据湖管理员。要添加其他数据湖管理员，您必须使用已被授予`AdministratorAccess`访问权限的IAM用户或角色。

创建管理员

1. 将**`AmazonHealthlakeFullAccess`**IAM AWS 托管策略添加到组织中的用户或角色。

   如果您不熟悉创建IAM用户，请参阅用户指南中的创建IAM用户和 AWS IAM策略概述。IAM

2. 向IAM用户或角色授予访问 AWS Lake Formation 的访问权限。

   - 将以下IAM AWS 托管策略添加到组织中的用户或角色：**`AWSLakeFormationDataAdmin`**

     > **Note**
     >
     > 该`AWSLakeFormationDataAdmin`政策允许访问所有 AWS Lake Formation 资源。建议您始终使用完成任务所需的最低权限。有关更多信息，请参阅《IAM用户指南》中的IAM最佳实践。

3. 向用户或角色添加以下内联策略。有关更多信息，请参阅《IAM用户指南》中的内联策略。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-source-bucket/*",
                "arn:aws:s3:::amzn-s3-demo-logging-bucket/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ram:GetResourceShareInvitations",
                "ram:AcceptResourceShareInvitation",
                "glue:CreateDatabase",
                "glue:DeleteDatabase"
            ],
            "Resource": "*"
        }
    ]
}
```

有关该AWSLakeFormationDataAdmin政策的更多信息，请参阅 Lake Formation 开发者指南中的 La AWS ke Formation 角色和IAM权限参考。

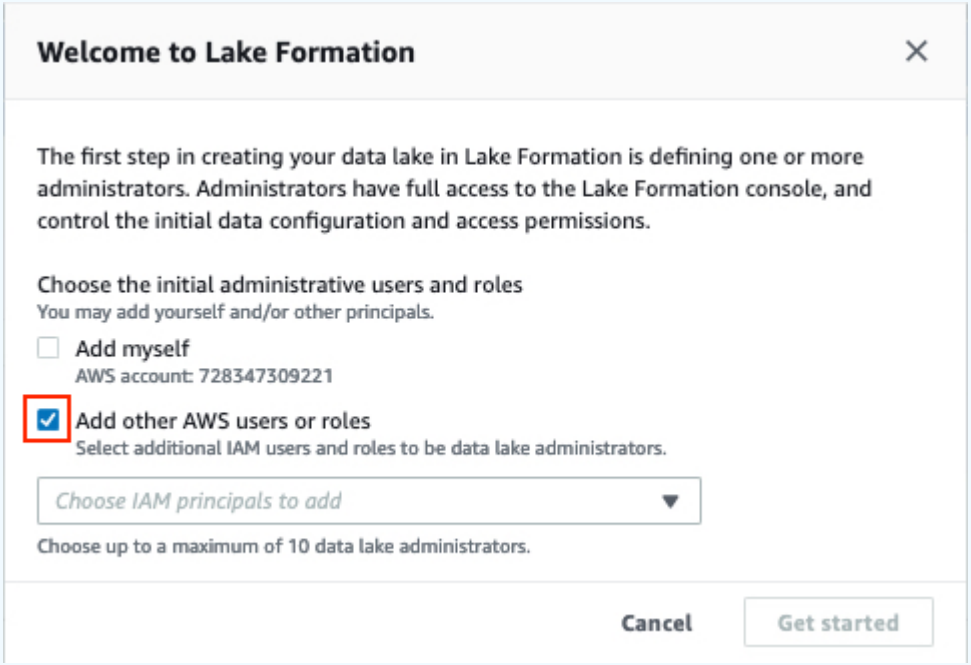
在 Lake Formation 中添加用户或角色作为数据湖IAM管理员（管理员）

接下来，IAM管理员需要将步骤 1 中创建的用户或角色添加为 Lake Formation 中的数据湖管理员。

将IAM用户或角色添加为数据湖管理员

1. 打开 AWS Lake Formation 控制台：https://console.aws.amazon.com/lakeformation/

> **Note**
>
> 如果这是你第一次访问 Lake Formation，则会出现一个 “欢迎来到 Lake Form ation” 对话框，要求你定义 Lake Formation 管理员。



2. 将新用户或角色分配为 AWS Lake Formation 数据湖管理员。

   - 选项 1：如果你收到了 “欢迎来到 Lake Formation” 对话框。

     1. 选择添加其他 AWS 用户或角色。
     2. 选择向下箭头 (▾)。
     3. 选择你想同时成为 Lake Formation 管理员的管理员。 HealthLake
     4. 选择开始。

   - 选项 2：使用导航窗格 (☰)。

     1. 选择导航窗格 (☰)。
     2. 在 “权限” 下，选择 “管理角色和任务”。
     3. 在数据湖管理员部分中，选择选择管理员。
     4. 在 “管理数据湖管理员” 对话框中，选择向下箭头 (▾)。
     5. 接下来，选择或搜索您也想成为 Lake Formation HealthLake 管理员的管理员用户或角色。
     6. 选择保存。

3. 将默认安全设置更改为由 Lake Formation 管理。 HealthLake 数据存储资源不需要由 Lake Formation 管理IAM。要进行更新，请参阅 La AWS ke Formation 开发者指南中的更改默认权限模型。

在中创建数据存储 AWS HealthLake

完成后设置开始使用的权限 AWS HealthLake，就可以创建数据存储了。在中 AWS HealthLake，您可以使用数据存储以 HL7 FHIR (R4) 格式存储数据。本章中的主题介绍如何创建数据存储。

要在 Athena 中创建支持分析的数据存储并授予访问权限，请将托管策略
添加到AWSLakeFormationDataAdminIAM您的用户、群组或角色。
该AWSLakeFormationDataAdmin策略允许您创建数据湖管理员并授予对 Athena 中数据存储的访问权限。有关设置权限的信息，请参阅设置开始使用的权限 AWS HealthLake。

HealthLake 还集成了 AWS CloudTrail。您可以使用 CloudTrail 来记录用户、角色或 AWS 服务在中执行的操作 HealthLake。 CloudTrail 将所有API呼叫和控制台操作捕获 HealthLake 为事件。要了解更多信息，请参阅 使用 AWS CloudTrail记录 AWS HealthLake API 调用。

要详细了解支持的快速医疗互操作性资源 (FHIR) 资源类型 HealthLake，请参阅中支持的FHIR资源类型 AWS HealthLake。

> ⚠ 亚马逊 Athena 兼容性
>
> HealthLake 2022 年 11 月 14 日之前创建的日期存储无法使用 Athena 执行SQL查询。要在先前存在的数据存储上使用 Athena 搜索功能，请先将数据迁移到新的数据存储。要了解有关迁移先前存在的数据存储的更多信息，请参阅迁移现有数据存储以使用 Amazon Athena。

创建数据存储后，您可以使用API_DescribeFHIRDatastore或 API_ListFHIRDatastores.html API 操作获取其属性，包括其状态。或者，您可以在 HealthLake 控制台的数据存储页面上找到数据存储状态和其他详细信息。

HealthLake 数据存储可以具有以下状态：

- 正在@@ 创建-正在创建您的数据存储。
- 活动-您的数据存储处于活动状态。您可以从中导入和导出数据。您还可以管理和搜索存储在数据存储中的FHIR资源。
- 正在删除-正在删除您的数据存储。
- 已删除-您的数据存储已删除。

主题

创建数据存储 (AWS Management Console)

> **⚠ HealthLake 控制台差异**
>
> HealthLake 控制台不支持创建FHIR已SMART启用的数据存储。要创建FHIR未SMART启用的数据存储，必须使用 AWS CLI 或 AWS 支持的数据存储SDKS。要了解更多信息，请参阅继续FHIR与SMART集成 AWS HealthLake。此外，当您查看单个数据存储的详细信息页面 HealthLake 时，控制台不区分所支持的两种数据存储。

创建 HealthLake 数据存储

1. 在家中打开 https://console.aws.amazon.com//healthlake/主 HealthLake 机。
2. 打开导航窗格 (►)。
3. 然后，选择数据存储。
4. 接下来，选择创建数据存储。
5. 在数据存储设置部分，为数据存储名称指定一个名称。
6. （可选）在数据存储设置部分中，对于预加载样本数据，选中预加载合成数据的复选框。
   - 合成数据是一个预加载的样本数据集。有关更多信息，请参阅 预加载的数据类型。
7. 在数据存储加密部分，选择使用AWS自有密钥（默认）或选择其他AWSKMS密钥（高级）。
8. 在标签-可选部分中，您可以向数据存储中添加标签。
   - 要了解有关为数据存储添加标签的更多信息，请参阅向数据存储添加标签。
9. 接下来，选择创建数据存储。数据存储的状态可在数据存储页面上找到。

创建数据存储（ AWS CLI 和 AWS SDKs ）

您可以使用以下代码示例来创建 HealthLake 数据存储。

AWS CLI

以下示例演示了如何在 AWS CLI中使用 `CreateFHIRDatastore` 操作。要运行示例，您必须安装 AWS CLI。创建数据存储时，除非另有说明，否则静态加密默认为AWS拥有的KMS密钥。要了解有关加密REST的更多信息， HealthLake 请参阅在 for 处REST加密 AWS HealthLake。

此示例的格式适用于 Unix、Linux 和 macOS。对于 Windows，将每行末尾的反斜杠 (\) Unix 延续字符替换为尖号 ()。^

```
aws healthlake create-fhir-datastore \
  --datastore-type-version R4 \
  --preload-data-config PreloadDataType="SYNTHEA" \
  --datastore-name "your-data-store-name"
```

成功后，您将收到以下JSON响应。当您的数据存储准备好摄取数据时，状态将更改为。ACTIVE要了解有关将数据导入 HealthLake 数据存储的更多信息，请参阅将文件导入 HealthLake 数据存储。

```
{
    "DatastoreId": "eeb8005725ae22b35b4edbdc68cf2dfd",
 "DatastoreArn": "arn:aws:healthlake:us-west-2:111122223333:datastore/fhir/
eeb8005725ae22b35b4edbdc68cf2dfd",
    "DatastoreStatus": "CREATING",
    "DatastoreEndpoint": "https://healthlake.us-west-2.amazonaws.com/datastore/
eeb8005725ae22b35b4edbdc68cf2dfd/r4/"
}
```

要查看所有数据存储/数据存储的列表，您可以使用操作。`ListFHIRDataStore`您还可以在 HealthLake 控制台中查看活动数据存储列表。

Python (boto3)

以下示例演示如何使用`create_fhir_datastore`操作创建 HealthLake 数据存储。除非另有说明，否则在创建数据存储时，静态加密默认为 AWS拥有的 AWS KMS 密钥。要了解有关加密REST的更多信息， HealthLake 请参阅在 for 处REST加密 AWS HealthLake。

```
import boto3
import logging #built in logging library
from botocore.exceptions import ClientError, ValidationError #specific exception
 ClientError from the boto3 library

def create_healthlake_datastore(DatastoreName=None):
```

```
    '''
    :param DatastoreName: the name of the data store, string
    :param:
    :return: True if the data store is created, else False
    '''

    # Create an Amazon Healthlake data store
    # Should we say something about region setting?
    # Should this example have some handling KMS keys

    try:
        if DatastoreName is None:
            healthlake_client = boto3.client('healthlake')
            healthlake_client.create_fhir_datastore(DatastoreTypeVersion='R4')

        else:
            healthlake_client = boto3.client('healthlake')
            healthlake_client.create_fhir_datastore(DatastoreTypeVersion='R4',
                                                    DatastoreName=DatastoreName)
    except (ClientError, ValidationError) as e:
        logging.error(e)
        return False

    return True

# Run the function above
create_healthlake_datastore(DatastoreName='test-datastore-delete-me-2')
```

数据存储可以有四种状态之一。`list_fhir_datastores`用于查看您的 HealthLake 数据存储列表，无论其状态如何。此示例说明如何根据数据存储的状态进行筛选。

```
import boto3

healthlake_client = boto3.client('healthlake')
data_store_list = healthlake_client.list_fhir_datastores(Filter={'DatastoreStatus':
 'ACTIVE'})
print(data_store_list)
```

要了解更多信息，请参阅 Boto3 文档*list_fhir_datastore*中的。

将文件导入 HealthLake 数据存储

完成后在中创建数据存储 AWS HealthLake，您可以将文件从亚马逊简单存储服务 (Amazon S3) 存储桶导入数据存储。要导入文件，您可以使用 HealthLake 控制台或`StartFHIRImportJob`API操作启动导入任务。

创建导入任务时，您可以指定输入数据在 Amazon S3 中的位置、输出日志文件的 Amazon S3 存储桶位置、授予存储桶 HealthLake 访问权限的IAM角色以及客户拥有或 AWS 拥有的 AWS Key Management Service 密钥。 HealthLake 使用此密钥在源位置加密您的数据，并将用于对其进行解密 HealthLake 以允许导入。有关为导入任务设置权限的信息，请参阅为导入任务设置权限。要了解有关创建和使用 AWS KMS 密钥的更多信息，请参阅密AWS钥管理服务开发人员指南中的创建密钥。

HealthLake 接受换行符分隔 JSON (`.ndjson`) 格式的输入文件，其中每行都包含一个有效的FHIR资源。您可以使用API操作`DescribeFHIRImportJob`和`ListFHIRImportJobs`来描述和列出正在进行的导入任务。

为每个导入任务 HealthLake 生成一个`manifest.json`文件。此日志描述了导入任务的成功和失败。HealthLake 将文件输出到您在创建导入任务时指定的 Amazon S3 存储桶。有关更多信息，请参阅 清单JSON文件。

您可以将导入或导出任务加入队列。这些异步导入或导出任务以FIFO（先入先出）的方式处理。在导入或导出任务进行期间，您可以创建、读取、更新或删除FHIR资源。

使用预加载的数据或导入数据填充数据存储后，您可以开始在 Amazon Athena SQL 中使用查询您的数据存储。有关更多信息，请参阅 在 Amazon Athena SQL 中使用查询 AWS HealthLake 数据存储。

主题

为导入任务设置权限

在将文件导入数据存储之前，必须授予访问您在 Amazon S3 中的输入和输出存储桶的 HealthLake 权限。要授予 HealthLake 访问权限，您需要为创建一个IAM服务角色 HealthLake，向该角色添加信任策

略以授予 HealthLake 代入角色权限，并向角色附加权限策略，以授予其访问您的 Amazon S3 存储桶的权限。

创建导入任务时，您可以为指定该角色的 Amazon 资源名称 (ARN) `DataAccessRoleArn`。有关IAM角色和信任策略的更多信息，请参阅IAM角色。

设置权限后，您就可以通过导入任务将文件导入数据存储了。有关更多信息，请参阅 在中启动导入任务 HealthLake。

设置导入权限

1. 如果还没有，请为输出日志文件创建一个目标 Amazon S3 存储桶。Amazon S3 存储桶必须与服务位于同一个 AWS 区域，并且必须为所有选项开启阻止公共访问。要了解更多信息，请参阅使用 Amazon S3 阻止公共访问。还必须使用亚马逊拥有或客户拥有的KMS密钥进行加密。要了解有关使用KMS密钥的更多信息，请参阅 Amazon 密钥管理服务。
2. 使用以下信任策略为其创建数据访问服务角色， HealthLake 并向该 HealthLake 服务授予代入该角色的权限。 HealthLake 使用它来写入输出 Amazon S3 存储桶。

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Principal": {
            "Service": ["healthlake.amazonaws.com"]
        },
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": {
                "aws:SourceAccount": "your-account-id"
            },
            "ArnEquals": {
                "aws:SourceArn": "arn:aws:healthlake:us-west-2:account:datastore/
fhir/data store ID"
            }
        }
    }]
}
```

3. 向数据访问角色添加权限策略，使其能够访问 Amazon S3 存储桶。`amzn-s3-demo-bucket`替换为存储桶的名称。

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Action": [
            "s3:ListBucket",
            "s3:GetBucketPublicAccessBlock",
            "s3:GetEncryptionConfiguration"
        ],
        "Resource": [
            "arn:aws:s3:::amzn-s3-demo-source-bucket"
        ],
        "Effect": "Allow"
    },
    {
        "Action": [
            "s3:PutObject"
        ],
        "Resource": [
            "arn:aws:s3:::amzn-s3-demo-logging-bucket/*"
        ],
        "Effect": "Allow"
    },
    {
        "Action": [
            "kms:DescribeKey",
            "kms:GenerateDataKey*"
        ],
        "Resource": [
            "arn:aws:kms:us-east-1:012345678910:key/d330e7fc-b56c-4216-a250-
f4c43ef46e83"
        ],
        "Effect": "Allow"
    }]
}
```

在中启动导入任务 HealthLake

创建数据存储并设置导入任务权限 (为导入任务设置权限) 后，就可以开始使用导入任务导入文件了。您可以使用 AWS HealthLake 控制台或导入来启动 AWS HealthLake 导API入任务start-fhir-import-jobAPI。

主题

使用API操作导入文件

先决条件

使用 AWS HealthLake API操作时，必须先创建 AWS Identity and Access Management (IAM) 策略并将其附加到IAM角色。要了解有关IAM角色和信任策略的更多信息，请参阅IAM策略和权限。客户还必须使用KMS密钥进行加密。要了解有关使用KMS密钥的更多信息，请参阅 Amazon 密钥管理服务。

要导入文件 (API)，请使用以下步骤。

1. 将您的数据上传到 Amazon S3 存储桶。
2. 使用该`start-fhir-import-job API`API操作。启动任务时，请指定包含输入文件的 Amazon S3 存储桶的名称、要用于加密的密KMS钥以及输出数据配置。
3. 要了解有关FHIR导入任务的更多信息，请使用该describe-fhir-import-job操作获取任务的 ID ARN、、名称、开始时间、结束时间和当前状态。list-fhir-import-job用于显示所有导入任务及其状态。

启动导入任务（控制台）

要使用控制台导入文件，您需要将数据上传到 Amazon S3 存储桶，

要导入文件，请使用以下步骤。

1. 将您的数据上传到 Amazon S3 存储桶。
2. 在家中打开 https://console.aws.amazon.com//healthlake/主 HealthLake 机。
3. 转到数据存储的数据存储详细信息页面，然后选择导入。
4. 指定您的 Amazon S3 存储桶，然后创建或确定要使用的IAM角色和KMS密钥。
5. 选择导入数据。

清单JSON文件

为每个导入任务 HealthLake 生成一个manifest.json文件。 HealthLake 将文件输出到您在创建导入任务时指定的 Amazon S3 存储桶。

该manifest.json文件描述了导入任务的成功和失败。日志文件分为两个文件夹，名为SUCCESS和FAILURE。输出文件可能包含敏感信息，因此，在创建导入任务时，必须同时提供输出 Amazon S3 存储桶和加密 AWS KMS 密钥。

以下是输出manifest.json文件的示例。我们建议您使用此文件作为对失败的导入任务进行故障排除的第一步。它提供了有关每个文件以及导入任务失败的原因的详细信息。

```
  {
  "inputDataConfig": {
    "s3Uri": "s3://amzn-s3-demo-source-bucket/healthlake-input/invalidInput/"
  },
  "outputDataConfig": {
    "s3Uri": "s3://amzn-s3-demo-logging-bucket/32839038a2f47f17c2fe0f53f0c3a0ba-
FHIR_IMPORT-19dd7bb7bcc8ee12a09bf6d322744a3d/",
    "encryptionKeyID": "arn:aws:kms:us-west-2:123456789012:key/fbbbfee3-20b3-42a5-a99d-
c48c655ed545"
  },
  "successOutput": {
    "successOutputS3Uri": "s3://amzn-s3-demo-logging-
bucket/32839038a2f47f17c2fe0f53f0c3a0ba-FHIR_IMPORT-19dd7bb7bcc8ee12a09bf6d322744a3d/
SUCCESS/"
  },
  "failureOutput": {
    "failureOutputS3Uri": "s3://amzn-s3-demo-logging-
bucket/32839038a2f47f17c2fe0f53f0c3a0ba-FHIR_IMPORT-19dd7bb7bcc8ee12a09bf6d322744a3d/
FAILURE/"
  },
  "numberOfScannedFiles": 1,
  "numberOfFilesImported": 1,
  "sizeOfScannedFilesInMB": 0.023627,
  "sizeOfDataImportedSuccessfullyInMB": 0.011232,
  "numberOfResourcesScanned": 9,
  "numberOfResourcesImportedSuccessfully": 4,
  "numberOfResourcesWithCustomerError": 5,
  "numberOfResourcesWithServerError": 0
}
```

示例：使用启动和监控导入任务 AWS CLI

以下示例说明如何使用启动 AWS Command Line Interface 和监控导入任务。您也可以使用 start-fhir-import-job API。

```
aws healthlake start-fhir-import-job \
--input-data-config S3Uri=s3://amzn-s3-demo-source-bucket/inputFolder/ \
--datastore-id (Datastore ID) \
--data-access-role-arn "arn:aws:iam::012345678910:role/DataAccessRole" \
--job-output-data-config '{"S3Configuration": {"S3Uri":"s3://amzn-s3-demo-logging-bucket/healthlake-output","KmsKeyId":"arn:aws:kms:us-east-1:012345678910:key/d330e7fc-b56c-4216-a250-f4c43ef46e83"}}' \
--region us-east-1
```

导入任务开始时，您将收到以下确认信息。

```
{
 "JobId": "8a4077553e9a485ad889c1a89c7541f0",
 "JobStatus": "SUBMITTED",
 "DatastoreId": "32839038a2f47f17c2fe0f53f0c3a0ba"
}
```

要监视导入任务的状态或了解其配置属性，请使用 describe-fhir-import-jobAPI或 AWS CLI 命令，如以下示例所示。

```
    aws healthlake describe-fhir-import-job \
    --datastore-id (Datastore ID) \
    --job-id c145fbb27b192af392f8ce6e7838e34f \
    --region us-east-1
```

作为回应，您会收到以下信息。

```
{
```

```
    "ImportJobProperties": {
    "InputDataConfig": {
    "S3Uri": "s3://amzn-s3-demo-source-bucket/(Prefix Name)/"
    },
    "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)",
    "JobStatus": "COMPLETED",
    "JobId": "c145fbb27b192af392f8ce6e7838e34f",
    "SubmitTime": 1606272542.161,
    "EndTime": 1606272609.497,
    "DatastoreId": "(Datastore ID)"
    }
}
```

要查看所有导入任务的列表，请使用list-fhir-import-jobsAPI或 AWS CLI 命令，如以下示例所示。您可以添加一个或多个过滤器来限制结果。

```
    aws healthlake list-fhir-import-jobs\
    --datastore-id (Datastore ID) \
    --submitted-before (DATE like 2024-10-13T19:00:00Z)\
    --submitted-after (DATE like 2020-10-13T19:00:00Z )\
    --job-name "FHIR-IMPORT" \
    --job-status SUBMITTED  \
    --max-results (Integer between 1 and 500)
```

作为回应，您会收到以下信息。

```
{
    "ImportJobProperties": {
    "OutputDataConfig": {
    "S3Uri": "s3://(Bucket Name)/(Prefix Name)/",
    "S3Configuration": {
     "S3Uri": "s3://(Bucket Name)/(Prefix Name)/",
     "KmsKeyId" : "(KmsKey Id)"
     },
    },
    "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)",
    "JobStatus": "COMPLETED",
    "JobId": "c145fbb27b192af392f8ce6e7838e34f",
    "JobName" "FHIR-IMPORT",
    "SubmitTime": 1606272542.161,
    "EndTime": 1606272609.497,
```

```
    "DatastoreId": "(Datastore ID)"
    }
}
"NextToken": String
```

从 HealthLake 数据存储中导出文件

创建数据存储并导入数据（或者如果您使用预加载的示例数据）后，您可以将数据导出到 Amazon S3 存储桶。要从数据存储中导出 HealthLake 数据，请使用以下操作。

- 使用 AWS SDKs和的`StartFHIRExportJob`API操作提出导出请求 HealthLake。
  - 此操作仅支持发出系统范围的导出请求。
- 使用`export`语法提出导出请求 HealthLake FHIRRESTAPI。
  - 此操作支持提出系统范围、患者和群组的导出请求。您还可以应用参数来进一步筛选导出请求中的数据。

> **Important**
>
> HealthLake SDK使用`StartFHIRExportJob`API操作的导FHIRRESTAPI出请求和使用`StartFHIRExportJobWithPost`API操作的导出请求有不同的IAM操作。每个IAM操作（使用SDK导出`StartFHIRExportJob`和FHIRRESTAPI导出时使用`StartFHIRExportJobWithPost`）都可以单独处理允许/拒绝权限。如果您希望同时限制两FHIRRESTAPI者SDK兼而有之，请务必拒绝每个IAM操作的权限。

这两个操作都仅支持将您的文件导出到 Amazon S3 (S3) 存储桶。 HealthLake数据存储中的所有文件都导出为以换行符分隔 JSON (`.ndjson`) 的文件，其中每行都包含一个有效的FHIR资源。

这两个操作都需要服务角色。在其中， HealthLake 必须定义为服务主体，并且必须定义要导出文件的 Amazon Simple Storage Service (S3) 存储桶。要了解更多信息，请参阅 为导出任务设置权限。

您可以将导入或导出任务排入队列。这些异步导入或导出任务以FIFO（先入先出）的方式处理。在导入或导出任务进行期间，您可以创建、读取、更新或删除FHIR资源。

要从 HealthLake 数据存储中导出文件，请参阅以下各节。

- 为导出任务设置权限
- 使用 HealthLake 控制台从数据存储中导出文件或 AWS SDKs
- 使用FHIRRESTAPI操作从 HealthLake 数据存储中导出数据

为导出任务设置权限

在从数据存储中导出文件之前，您必须授予访问您在 Amazon S3 中的输出存储桶的 HealthLake 权限。要授予 HealthLake 访问权限，您需要为创建一个IAM服务角色 HealthLake，向该角色添加信任策略以授予 HealthLake 代入角色权限，并向角色附加权限策略，以授予其访问您的 Amazon S3 存储桶的权限。

如果您已经 HealthLake 在中创建了角色，则可以重复使用该角色为导入任务设置权限，并向其授予本主题中列出的导出 Amazon S3 存储桶的额外权限。要了解有关IAM角色和信任策略的更多信息，请参阅IAM策略和权限。

> **Important**
>
> HealthLake SDK使用`StartFHIRExportJob`API操作的导FHIRRESTAPI出请求和使用`StartFHIRExportJobWithPost`API操作的导出请求有不同的IAM操作。每个IAM操作（使用SDK导出`StartFHIRExportJob`和FHIRRESTAPI导出时使用`StartFHIRExportJobWithPost`）都可以单独处理允许/拒绝权限。如果您希望同时限制两FHIRRESTAPI者SDK兼而有之，请务必拒绝每个IAM操作的权限。如果您授予用户完全访问权限 HealthLake，则无需更改IAM用户权限。

设置权限的用户或角色必须具有创建角色、创建策略和将策略附加到角色的权限。以下IAM策略授予这些权限。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Action": ["iam:CreateRole", "iam:CreatePolicy", "iam:AttachRolePolicy"],
    "Effect": "Allow",
    "Resource": "*"
    }, {
    "Action": "iam:PassRole"
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "iam:PassedToService": "healthlake.amazonaws.com"
      }
    }
  }]
```

```
}
```

设置导出权限

1. 如果还没有，请为要从数据存储中导出的数据创建一个目标 Amazon S3 存储桶。Amazon S3 存储桶必须与服务位于同一AWS区域，并且必须为所有选项开启阻止公共访问。要了解更多信息，请参阅使用 Amazon S3 阻止公共访问。还必须使用亚马逊拥有或客户拥有的KMS密钥进行加密。要了解有关使用KMS密钥的更多信息，请参阅 Amazon 密钥管理服务。
2. 如果您尚未创建数据访问服务角色，请使用以下信任策略为 HealthLake 该 HealthLake 服务授予代入该角色的权限。 HealthLake 使用它来写入输出 Amazon S3 存储桶。如果您已经在中创建了一个存储桶为导入任务设置权限，则可以在下一步中重复使用该存储桶并向其授予访问您的 Amazon S3 存储桶的权限。

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Principal": {
            "Service": ["healthlake.amazonaws.com"]
        },
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": {
                "aws:SourceAccount": "your-account-id"
            },
            "ArnEquals": {
                "aws:SourceArn": "arn:aws:healthlake:us-west-2:account:datastore/
fhir/data store ID"
            }
        }
    }]
}
```

3. 向数据访问角色添加权限策略，使其能够访问您的输出 Amazon S3 存储桶。amzn-s3-demo-bucket替换为存储桶的名称。

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Action": [
            "s3:ListBucket",
```

```
            "s3:GetBucketPublicAccessBlock",
            "s3:GetEncryptionConfiguration"
        ],
        "Resource": [
            "arn:aws:s3:::amzn-s3-demo-source-bucket"
        ],
        "Effect": "Allow"
    },
    {
        "Action": [
            "s3:PutObject"
        ],
        "Resource": [
            "arn:aws:s3:::amzn-s3-demo-logging-bucket/*"
        ],
        "Effect": "Allow"
    },
    {
        "Action": [
            "kms:DescribeKey",
            "kms:GenerateDataKey*"
        ],
        "Resource": [
            "arn:aws:kms:us-east-1:012345678910:key/d330e7fc-b56c-4216-a250-
f4c43ef46e83"
        ],
        "Effect": "Allow"
    }]
}
```

使用 HealthLake 控制台从数据存储中导出文件或 AWS SDKs

完成后为导出任务设置权限，您可以将文件从数据存储导出到亚马逊简单存储服务 (Amazon S3) 存储桶。要从数据存储中导出文件，请在中启动导出作业 HealthLake。导出任务以换行符分隔 JSON (.ndjson) 格式从您的数据存储中导出文件，其中每行都包含一个有效的FHIR资源。启动导出任务时，必须指定加密 AWS KMS 密钥。要了解有关创建KMS密钥的更多信息，请参阅《密AWS钥管理服务开发者指南》中的创建密钥。

以下主题介绍如何使用 AWS HealthLake 控制台启动导出任务以及如何使用start-fhir-export-jobAPI操作启动导出任务。 AWS SDKs

主题

从您的数据存储中导出文件（控制台）

要导出文件（控制台），请使用以下步骤。

1. 在与之相同的区域中创建输出 S3 存储桶 HealthLake。
2. 要开始新的导出任务，请识别输出 Amazon S3 存储桶，然后创建或确定要使用的IAM角色。要了解有关IAM角色和信任策略的更多信息，请参阅IAM角色。还要使用密KMS钥加密。要了解有关使用KMS密钥的更多信息，请参阅 Amazon 密钥管理服务。
3. 要查看导出任务的状态，请使用ListFHIRExportJobsAPI操作。

从您的数据存储中导出文件 (AWS SDKs)

要使用从数据存储中导出文件 AWS SDKs，请使用start-fhir-export-job操作。以下代码显示了如何使用SDK适用于 Python 的 (Boto3) 启动导出作业。

```
import boto3

client = boto3.client('healthlake')

response = client.start_fhir_export_job(
    JobName='job name',
    OutputDataConfig={
        'S3Configuration': {
            'S3Uri': 's3://amzn-s3-demo-bucket/output-folder',
            'KmsKeyId': 'arn:aws:kms:us-west-2:account-number:key/AWS KMS key ID'
        }
    },
    DatastoreId='data store ID',
    DataAccessRoleArn='role ARN',
)
print(response['JobStatus'])
```

要获取FHIR导出任务的 ID ARN、、名称、开始时间、结束时间和当前状态，请使用describe-fhir-export-job。list-fhir-export-jobs用于列出所有导出任务及其状态。

以下代码显示了如何使用 for Python (Boto3) 获取特定导出任务SDK的属性。

```
import boto3

client = boto3.client('healthlake')

describe_response = client.describe_fhir_export_job(
    DatastoreId=datastoreId,
    JobId=jobId
)
print(describe_response['ExportJobProperties'])
```

使用FHIRRESTAPI操作从 HealthLake 数据存储中导出数据

完成后为导出任务设置权限，您可以通过FHIRRESTAPI操作从 HealthLake 数据存储中导出数据。要使用提出导出请求 FHIR RESTAPI，您必须拥有具有所需权限的IAM用户、组或角色，在POST请求中进行指定`$export`，并在请求正文中包含请求参数。根据FHIR规范，FHIR服务器必须支持GET请求，并且可以支持POST请求。为了支持其他参数，需要一个主体来开始导出，因此 HealthLake 支持POST请求。

> **Important**
>
> HealthLake 2023 年 6 月 1 日之前创建的数据存储仅支持系统范围导出的FHIRRESTAPI基于导出任务的请求。
>
> HealthLake 2023 年 6 月 1 日之前创建的数据存储不支持使用数据存储端点上的GET请求获取导出状态。

您使用提出的所有导出请求FHIRRESTAPI都将以`ndjson`格式返回并导出到 Amazon S3 存储桶。每个 S3 对象将仅包含一种FHIR资源类型。

您可以根据 AWS 账户配额对导出请求进行排队。要详细了解与之关联的 Service Quota HealthLake s，请参阅AWS HealthLake 终端节点和配额。

HealthLake 支持以下三种类型的批量导出端点请求。

类型	描述	语法
系统导出	从 HealthLake FHIR服务器导出所有数据。	`POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/$export`
所有患者	导出与所有患者相关的所有数据，包括与患者资源类型相关的资源类型。	`POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/Patient/$export`
患者群体	导出与使用群组 ID 指定的一组患者相关的所有数据。	`POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/Group/ID/$export`

开始前的准备工作

使用 for 满足以下要求即可提出导出FHIRRESTAPI请求 HealthLake。

- 您必须已设置具有必要权限的用户、组或角色才能发出导出请求。要了解更多信息，请参阅 授权请求 export。
- 您必须已创建服务角色来授予 HealthLake 访问要将数据导出到的 Amazon S3 存储桶的权限。服务角色还必须指定 HealthLake 为服务主体。有关设置权限的更多信息，请参阅为导出任务设置权限。

授权请求 **export**

要使用成功发出导出请求 FHIR RESTAPI，请使用或 OAuth2 .0 对您的用户、群组或角色进行授权。IAM您还必须具有服务角色。

使用对请求进行授权 IAM

当您提出$export请求时，用户、组或角色必须具有策略中包含的StartFHIRExportJobWithPostDescribeFHIRExportJobWithGet、和CancelFHIRExportJobWithDeleteIAM操作。

> **Important**
>
> HealthLake SDK使用`StartFHIRExportJob`API操作的导FHIRRESTAPI出请求和使用`StartFHIRExportJobWithPost`API操作的导出请求有不同的IAM操作。每个IAM操作（使用SDK导出`StartFHIRExportJob`和FHIRRESTAPI导出时使用`StartFHIRExportJobWithPost`）都可以单独处理允许/拒绝权限。如果您希望同时限制两FHIRRESTAPI者SDK兼而有之，请务必拒绝每个IAM操作的权限。

使用 SMART on FHIR (OAuth2.0) 授权请求

当你在FHIR已启用的 HealthLake 数据存储SMART上`$export`发出请求时，你需要分配相应的范围。要了解有关支持的作用域的更多信息，请参阅HealthLake 数据存储FHIR资源特定范围。

提出`export`请求

本节介绍使用提出导出请求时必须采取的必要步骤FHIRRESTAPI。

为避免意外向您的 AWS 账户收费，我们建议您在不提供`export`语法的情况下通过提出`POST`请求来测试您的请求。

要提出请求，您必须执行以下操作：

1. 在`POST`请求`export`URL中指定支持的终端节点。
2. 指定所需的标题参数。
3. 指定用于定义所需参数的请求正文。

步骤 1：在`POST`请求`export`URL中指定支持的终端节点

HealthLake 支持三种类型的批量导出端点请求。要发出批量导出请求，您必须在三个支持的终端节点之一上发出`POST`基于请求的请求。以下示例演示了如何在请求`export`中指定URL。

- `POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/$export`
- `POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/Patient/$export`
- `POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/Group/ID/$export`

在该POST请求字符串中，您可以使用以下支持的搜索参数。

支持的搜索参数

HealthLake 在批量导出请求中支持以下搜索修饰符。

这些示例包括特殊字符，在提交请求之前必须对其进行编码。

名称	必填？	描述	示例
_outputFormat	否	要生成的请求的批量数据文件的格式。可接受的值为application/fhir+ndjson 、application/ndjson 、ndjson。	
_type	否	要包含在导出任务中的以逗号分隔的FHIR资源类型字符串。我们建议将其包括在内，_type因为在导出所有资源时，这可能会影响成本。	&_type=MedicationStatement,Observation
_since	否	在日期时间戳当天或之后修改的资源类型。如果某个资源类型没有上次更新时间，则会将其包含在您的响应中。	&_since=2024-05-09T00%3A00%3A00Z

步骤 2：指定所需的标题参数

要使用发出导出请求 FHIR RESTAPI，必须指定以下两个标头参数。

- Content-Type：application/fhir+json

- 首选：`respond-async`

接下来，您必须在请求正文中指定所需的元素。

步骤 3：指定用于定义所需参数的请求正文。

导出请求还需要JSON格式化的正文。正文可以包含以下参数。

密钥	必填？	描述	值
DataAccessRoleArn	是	ARN一个 HealthLake 服务角色。使用的服务角色必须指定 HealthLake 为服务主体。	`arn:aws:iam::` **`444455556666`** `:role/`**`your-healthlake-service-role`**
JobName	否	导出请求的名称。	**`your-export-job-name`**
S3Uri	是	OutputDataConfig 钥匙的一部分。将下载导出数据的目标存储桶的 S URI 3。	`s3://DOC-EXAMPLE-DESTINATION-BUCKET/` **`EXPORT-JOB`** `/`
KmsKeyId	是	OutputDataConfig 钥匙的一部分。用于保护 Amazon S3 存储桶的 AWS KMS 密钥之一。ARN	`arn:aws:kms:` **`region-of-bucket:123456789012`** `:key/`**`1234abcd-12ab-34cd-56ef-1234567890ab`**

Example — 使用发出的出口请求的正文 FHIR REST API

要使用提出导出请求 FHIR RESTAPI，必须指定正文，如下所示。

```
{
```

```
  "DataAccessRoleArn": "arn:aws:iam::444455556666:role/your-healthlake-service-role",
  "JobName": "your-export-job",
  "OutputDataConfig": {
    "S3Configuration": {
      "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/EXPORT-JOB",
      "KmsKeyId": "arn:aws:kms:region-of-
bucket:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    }
  }
}
```

请求成功后，您将收到以下回复。

响应标头

```
content-location: https://healthlake.your-region.amazonaws.com/datastore/your-
datastore-id/r4/export/your-export-request-job-id
```

响应正文

```
{
  "datastoreId": "your-data-store-id",
  "jobStatus": "SUBMITTED",
  "jobId": "your-export-request-job-id"
}
```

管理您的导出请求

成功发出导出请求后，您可以使用描述当前导export出请求的状态和取消当前的导出请求export来管理该请求。

当您使用取消导出请求时 RESTAPI，您只需为提交取消请求之前导出的部分数据付费。

以下主题介绍如何获取当前导出请求的状态或取消当前导出请求。

取消导出请求

要取消导出请求，DELETE请提出请求并在请求中提供任务 ID URL。

```
DELETE https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
export/your-export-request-job-id
```

请求成功后，您将收到以下信息。

```
{
  "exportJobProperties": {
    "jobId": "your-original-export-request-job-id",
    "jobStatus": "CANCEL_SUBMITTED",
    "datastoreId": "your-data-store-id"
  }
}
```

当您的请求失败时，您会收到以下信息。

```
{
  "resourceType": "OperationOutcome",
  "issue": [
    {
      "severity": "error",
      "code": "not-supported",
      "diagnostics": "Interaction not supported."
    }
  ]
}
```

描述导出请求

要获取导出请求的状态，GET请使用export和您的**export-request-job-id**。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
export/your-export-request-id
```

JSON响应将包含一个ExportJobProperties对象。它可能包含以下键:值对。

名称	必填？	描述	值
DataAccessRoleArn	否	ARN一个 HealthLake 服务角色。使用的服务角色必须指定 HealthLake 为服务主体。	arn:aws:iam:: **444455556666** :role/**your-healthlake-service-role**

名称	必填？	描述	值
SubmitTime	否	提交导出任务的日期。	`Apr 21, 2023 5:58:02`
EndTime	否	导出任务完成的时间。	`Apr 21, 2023 6:00:08 PM`
JobName	否	导出请求的名称。	**`your-export-job-name`**
JobStatus	否		有效值为： `SUBMITTED | IN_PROGRESS | COMPLETED_WITH_ERRORS | COMPLETED | FAILED`
S3Uri	是	OutputDataConfig对象的一部分。将下载导出数据的目标存储桶的 Amazon S3 URI。	`s3://DOC-EXAMPLE-DESTINATION-BUCKET/` **`EXPORT-JOB /`**
KmsKeyId	是	OutputDataConfig对象的一部分。用于保护 Amazon S3 存储桶的 AWS KMS 密钥之一。ARN	`arn:aws:kms:` **`region-of-bucket:123456789012`** `:key/` **`1234abcd-12ab-34cd-56ef-1234567890ab`**

Example：使用以下方法提出的描述导出请求的正文 FHIR REST API

成功后，您将收到以下JSON响应。

```
{
  "exportJobProperties": {
    "jobId": "your-export-request-id",
    "JobName": "your-export-job",
    "jobStatus": "SUBMITTED",
    "submitTime": "Apr 21, 2023 5:58:02 PM",
    "endTime": "Apr 21, 2023 6:00:08 PM",
    "datastoreId": "your-data-store-id",
    "outputDataConfig": {
      "s3Configuration": {
        "S3Uri": "s3://DOC-EXAMPLE-DESTINATION-BUCKET/EXPORT-JOB",
        "KmsKeyId": "arn:aws:kms:region-of-
bucket:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab""
      }
    },
    "DataAccessRoleArn": "arn:aws:iam::444455556666:role/your-healthlake-service-role",
  }
}
```

删除中的数据存储 HealthLake

删除数据存储是一种异步操作。启动后，状态将更改为 “正在删除”。数据存储将保持 “删除” 状态，直到FHIR数据存储中的所有数据以及必要的底层基础架构也都被移除。

删除数据和基础架构后，您的 HealthLake 数据存储状态将更改为 “已删除”。删除后，只有在七天内使用`DescribeFHIRDataStore`和`ListFHIRDataStores`操作才能获得有关数据存储的详细信息。七天后，已删除的数据存储将不会出现在结果中。

要成功删除数据存储，发出请求的用户、组或角色必须将IAM操作`glue:DeleteDatabase`添加到其IAM策略中。此IAM操作不包含在 AWS 托管策略中`AmazonHealthLakeFullAccess`。

您可以使用 AWS Management Console AWS SDKs、或删除数据存储 AWS CLI。

主题

删除数据存储（控制台）

要使用控制台删除数据存储，请在数据存储页面上选择您的数据存储，然后选择删除。

删除 HealthLake 数据存储

1. 在家中打开 https://console.aws.amazon.com//healthlake/主 HealthLake 机。
2. 打开 “导航” 窗格 ()。
3. 然后，选择数据存储。
4. 在数据存储页面上，选择要删除的数据存储旁边的选项。
5. 然后，选择 “删除”
6. 在对话框中键入**`delete`**以确认您要删除选定的数据存储。
7. 然后选择 Delete(删除)。然后，您的数据存储的状态将从 “活动” 更改为 “正在删除”。

删除数据存储（AWS SDKs和 AWS CLI）

您可以使用下面的代码示例来删除 HealthLake 数据存储。

AWS CLI

以下示例演示了将DeleteFHIRDatastore操作与一起使用 AWS CLI。要运行示例，您必须安装 AWS CLI。

```
aws healthlake delete-fhir-datastore --datastore-id
 'eeb8005725ae22b35b4edbdc68cf2dfd'
```

成功后，您将收到以下JSON响应。

```
{
    "DatastoreProperties": {
  "DatastoreId": "eeb8005725ae22b35b4edbdc68cf2dfd",
        "DatastoreArn": "arn:aws:healthlake:us-west-2:728347309221:datastore/fhir/",
        "DatastoreName": "delete-me",
        "DatastoreStatus": "ACTIVE",
        "CreatedAt": "2022-10-03T10:53:45.020000-07:00",
        "DatastoreTypeVersion": "R4",
        "DatastoreEndpoint": "https://healthlake.us-west-2.amazonaws.com/
datastore/5b6e4cd798289a4ab8dad6c1002dd731/r4/",
        "SseConfiguration": {
            "KmsEncryptionConfig": {
                "CmkType": "AWS_OWNED_KMS_KEY"
            }
        },
        "PreloadDataConfig": {
            "PreloadDataType": "SYNTHEA"
        }
    }
}
```

Python (boto3)

f AWS SDK or Python 支持采用单个参数describe_fhir_datastore的方法DatastoreId。

```
import boto3

#Create a Healthlake client
healthlake_client = boto3.client('healthlake')

#Call the describe_fhir_datastore method
data_store_details =
 healthlake_client.describe_fhir_datastore(DatastoreId='cdf8f1557e57c543bdc627fb8f12b7fd')
```

```
print(data_store_details)
```

成功后，它将返回一个 python 字典。

```
{'DatastoreProperties': {'DatastoreId': 'cdf8f1557e57c543bdc627fb8f12b7fd',
 'DatastoreArn': 'arn:aws:healthlake:us-west-2:728347309221:datastore/fhir/
cdf8f1557e57c543bdc627fb8f12b7fd', 'DatastoreName': '08-24-2022-test-data-
store', 'DatastoreStatus': 'ACTIVE', 'CreatedAt': datetime.datetime(2022,
 8, 23, 22, 12, 14, 359000, tzinfo=tzlocal()), 'DatastoreTypeVersion': 'R4',
 'DatastoreEndpoint': 'https://healthlake.us-west-2.amazonaws.com/datastore/
cdf8f1557e57c543bdc627fb8f12b7fd/r4/', 'SseConfiguration': {'KmsEncryptionConfig':
 {'CmkType': 'AWS_OWNED_KMS_KEY'}}, 'PreloadDataConfig': {'PreloadDataType':
 'SYNTHEA'}}, 'ResponseMetadata': {'RequestId': 'aef4b268-ad4b-4b57-
bc97-2da956356835', 'HTTPStatusCode': 200, 'HTTPHeaders': {'date': 'Wed, 05 Oct
 2022 01:21:44 GMT', 'content-type': 'application/x-amz-json-1.0', 'content-
length': '547', 'connection': 'keep-alive', 'x-amzn-requestid': 'aef4b268-ad4b-4b57-
bc97-2da956356835'}, 'RetryAttempts': 0}}
```

要一次返回有关多个数据存储的详细信息，请使用 `ListFHIRDatastore`

使用`DeleteFHIRDataStore`命令， AWS CLI 如以下示例所示。您也可以使用delete-fhir-datastore API或控制台删除数据存储。删除数据存储会删除该数据存储和底层基础架构中包含的所有FHIR资源版本。根据HIPAA准则，与已删除的数据存储相关的日志将保留在服务帐户中。

```
aws healthlake delete-fhir-datastore
    --datastore-id (Data Store ID)
```

如以下示例JSON响应所示，状态更改`DELETING`为 “”，以确认正在删除数据存储及其内容。

```
{
  "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/
datastore/eeb8005725ae22b35b4edbdc68cf2dfd/r4/",
    "DatastoreArn": "arn:aws:healthlake:us-east-1:(AWS Account ID):datastore/(Datastore
 ID)",
    "DatastoreStatus": "DELETING",
    "DatastoreId": "(Datastore ID)"
}
```

使用与 HealthLake 数据存储的FHIRRESTAPI交互

在中 AWS HealthLake，您可以使用 Fast Healthcare REST API 互操作性FHIR资源 (FHIR) 交互来管理和搜索数据存储中的资源。 FHIRRESTAPI交互用于对数据存储中的资源执行创建、读取、更新和删除 (CRUD) 交互。您还可以使用GET或POSTHTTP请求来形成复杂的搜索字符串，因为 HealthLake支持部分FHIR支持的搜索操作。

出于一致性考虑，将根据 HL7 FHIR R4 FHIR StructureDefinition资源对资源类型进行验证。要查找活动 HealthLake 数据存储的FHIR相关功能，请在中指定的metadata位置发出GET请求URL，如下所示。

```
GET https://healthlake.region.amazonaws.com/datastore/datastore-id/r4/metadata
```

成功后，您将收到 HealthLake 数据存储的200HTTP响应代码和能力声明。有关更多信息，请参阅 HL7FHIRR4 文档*CapabilityStatement*中的。

下表列出了支持的FHIR交互 AWS HealthLake。

FHIR支持的互动 AWS HealthLake

FHIR互动	描述
整个系统的交互	
capabilities	获取系统的能力声明
batch/transaction	在一次交互中更新、创建或删除一组资源
类型级别的互动	
create	使用服务器分配的 ID 创建新资源
search	根据某些筛选条件搜索资源类型
history	检索特定资源类型的更改历史记录
实例级别的交互	
read	读取资源的当前状态

FHIR互动	描述
history	阅读特定资源的变更历史记录
vread	读取资源特定版本的状态
update	按资源的 ID 更新资源（如果资源是新的，则创建它）
delete	删除资源

主题

- 中支持的FHIR资源类型 AWS HealthLake
- 对 HealthLake 数据存储执行创建、读取、更新和删除 (CRUD) 操作
- 使用FHIRRESTAPI操作搜索您的 HealthLake 数据存储
- 阅读FHIR资源历史记录
- 通过 Patient $ FHIR REST API everything 操作获取患者数据
- 使用 $export 从您的 HealthLake 数据存储中导出数据

中支持的FHIR资源类型 AWS HealthLake

下表列出了支持的 FHIR R4 资源类型。 AWS HealthLake有关更多信息，请参阅 HL7FHIRR4 文档中的资源索引。

FHIR支持的 R4 资源类型 HealthLake

帐户	DetectedIssue	发票	从业者
ActivityDefinition	设备	Library	PractitionerRole
AdverseEvent	DeviceDefinition	链接	过程
AllergyIntolerance	DeviceMetric	列出	出处
预约	DeviceUseStatement	位置	问卷
AppointmentResponse	DeviceRequest	度量	QuestionnaireResponse

AuditEvent-参见备注	DiagnosticReport	MeasureReport	RelatedPerson
二元	DocumentManifest	媒体	RequestGroup
BodyStructure	DocumentReference	药物	ResearchStudy
捆绑包-参见备注	EffectEvidenceSynthesis	MedicationAdministration	ResearchSubject
CapabilityStatement	遭遇	MedicationDispense	RiskAssessment
CarePlan	终端节点	MedicationKnowledge	RiskEvidenceSynthesis
CareTeam	EpisodeOfCare	MedicationRequest	计划
ChargeItem	EnrollmentRequest	MedicationStatement	ServiceRequest
ChargeItemDefinition	EnrollmentResponse	MessageHeader	槽位
Claim	ExplanationOfBenefit	MolecularSequence	标本
ClaimResponse	FamilyMemberHistory	NutritionOrder	StructureDefinition
Communication	标记	观察	StructureMap
CommunicationRequest	目标	OperationOutcome	Substance
合成	组	组织	SupplyDelivery
ConceptMap	GuidanceResponse	OrganizationAffiliation	SupplyRequest
状况	HealthcareService	参数	任务
同意	ImagingStudy	病人	ValueSet
合同	免疫接种	PaymentNotice	VisionPrescription
覆盖范围	ImmunizationEvaluation	PaymentReconciliation	VerificationResult

CoverageEligibilityRequest	ImmunizationRecommendation	人员	
CoverageEligibilityResponse	InsurancePlan	PlanDefinition	

⚠ FHIR规格和 HealthLake

- 您不能使用以下资源类型发出GET或POST请求：二进制 OperationOutcome、捆绑包和参数。
- AuditEvent— 可以创建或读取 AuditEvent 资源，但不能对其进行更新或删除。
- 捆绑包 — 有多种 HealthLake 管理捆绑包请求的方法。有关更多详细信息，请参阅使用 Bundle 管理多个FHIR资源。
- VerificationResult— 仅在 2023 年 12 月 9 日之后创建的数据存储支持此资源类型。

对 HealthLake 数据存储执行创建、读取、更新和删除 (CRUD) 操作

尽管在管理数据存储、导入数据和导出数据时使用本机 AWS FHIRHTTP操作，但要使用四个主要操作在 HealthLake 数据存储中创建 (POSTGET)、读取 (PUT)、更新 (DELETE) 和删除 () FHIR 资源。以下主题介绍如何使用FHIRRESTAPI服务对 HealthLake 数据存储执行创建、读取、更新和删除 (CRUD) 操作。必须使用签名版本 4 签名流程对通过HTTP客户端发送的 HealthLake API请求进行身份验证。要了解更多信息，请参阅中的签名版本 4 签名流程AWS 一般参考。

主题

- 使用创建资源 POST
- 使用阅读资源 GET
- 使用更新资源 PUT
- 使用删除资源 DELETE
- 使用 Bundle 管理多个FHIR资源

使用创建资源 POST

您可以使用POST请求在 HealthLake 数据存储中创建新资源。 POST请求不需要您提供id元素。成功201创建资源后， HealthLake 服务器会返回 “已创建” HTTP 状态码。

> **Note**
>
> 当您对DocumentReference资源类型发出POST请求时，不会修改现有的扩展。取而代之的是， AWS HealthLake 将新的扩展与现有扩展程序一起添加到您的数据存储中。有关如何 HealthLake 使用DocumentReference资源类型的自然语言处理 (NLP) 来提取有价值的医疗数据的更多详细信息，请参阅使用基于资源类型的自然语言处理 (NLP) 的自动FHIR DocumentReference 资源生成 AWS HealthLake。

Example 使用**POST**请求创建**Patient**资源。

要创建 HealthLake 数据存储POST请求，请使用数据存储的端点并提供JSON请求正文。要查找数据存储的端点，请在 HealthLake 控制台的 “数据存储” 下查看，或者使用AWS HealthLake API参考资料中的 D escribeFHIRDatastore 操作。

POST Request

```
POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient
```

JSON Request Body

```
{
  "resourceType": "Patient",
  "identifier": [ { "system": "urn:oid:1.2.36.146.595.217.0.1", "value":
 "12345" } ],
  "name": [ {
      "family": "Silva",
      "given": ["Ana", "Carolina"]
  } ],
  "gender": "female",
  "birthDate": "1992-02-10"
}
```

JSON 响应

为了确认患者资源的创建，您将收到 Create 201 d HTTP 状态码和以下JSON响应。

```
{
  "resourceType": "Patient",
  "identifier": [
    {
      "system": "urn:oid:1.2.36.146.595.217.0.1",
      "value": "12345"
    }
  ],
  "name": [
    {
      "family": "Silva",
      "given": [
        "Ana",
        "Carolina"
      ]
    }
  ],
  "gender": "female",
  "birthDate": "1992-02-10",
  "id": "274b408a-1201-4e9f-a621-1df937f1a26d",
  "meta": {
    "lastUpdated": "2022-06-13T23:31:24.427Z"
  }
}
```

使用阅读资源 GET

此示例向您展示如何使用GET请求读取患者FHIR资源。

Example 使用**GET**请求读取特定**Patient**资源。

要创建 HealthLake 数据存储GET请求，请使用数据存储的终端节点。要查找数据存储的端点，请在 HealthLake 控制台的 “数据存储” 下查看，或者使用AWS HealthLake API参考资料中的 DescribeFHIRDatastore 操作。

您还必须包括资源类型**Patient**和有效的标识符**2de04858-ba65-44c1-8af1-f2fe69a977d9**。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient/2de04858-ba65-44c1-8af1-f2fe69a977d9
```

JSON 响应

成功后，您将收到200HTTP状态码和以下JSON响应。

```
{
    "resourceType": "Patient",
    "active": true,
    "name": [
        {
            "use": "official",
            "family": "Doe",
            "given": [
                "Jane"
            ]
        },
        {
            "use": "usual",
            "given": [
                "Jane"
            ]
        }
    ],
    "gender": "female",
    "birthDate": "1966-09-01",
    "meta": {
        "lastUpdated": "2020-11-23T06:24:13.202Z"
    },
    "id": "2de04858-ba65-44c1-8af1-f2fe69a977d9"
}
```

使用更新资源 PUT

以下示例向您展示了PUT如何使用更新患者FHIR资源类型中患者的详细信息。此外，当您对尚未创建的资源PUT发出请求时，它将创建一个初始版本。

如果资源已更新，则您的请求将返回200HTTP状态代码；如果创建了新资源，则将返回201HTTP状态代码。

> **Note**
>
> 当您对DocumentReference资源类型发出PUT请求时，不会修改现有的扩展。取而代之的是， AWS HealthLake 将新的扩展与现有扩展程序一起添加到您的数据存储中。有关如

何 HealthLake 使用DocumentReference资源类型的自然语言处理 (NLP) 来提取有价值的医疗数据的更多详细信息，请参阅使用基于资源类型的自然语言处理 (NLP) 的自动FHIR DocumentReference 资源生成 AWS HealthLake。

Example 使用**PUT**请求更新**Patient**资源类型

发出PUT请求时，您需要数据存储的终端节点、要更新的资源类型的名称、标识符和JSON请求正文。

如果您使用PUT创建新资源，它将使用提供的标识符来创建新资源。

PUT Request

有效PUT请求的示例结构：

```
PUT https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient/2de04858-ba65-44c1-8af1-f2fe69a977d9
```

JSON Request Body

用于更新指定患者资源的示例JSON正文。

```
{
  "id": "2de04858-ba65-44c1-8af1-f2fe69a977d9",
  "resourceType": "Patient",
  "active": true,
  "name": [
    {
      "use": "official",
      "family": "Doe",
      "given": [
        "Jane"
      ]
    },
    {
      "use": "usual",
      "given": [
        "Jane"
      ]
    }
  ],
  "gender": "female",
```

```
  "birthDate": "1985-12-31"
}
```

JSON 响应

您将收到以下回复JSON以确认更改：

```
{
 "id": "2de04858-ba65-44c1-8af1-f2fe69a977d9",
 "resourceType": "Patient",
 "active": true,
 "name": [{
   "use": "official",
   "family": "Doe",
   "given": [
    "Jane"
   ]
  },
  {
   "use": "usual",
   "given": [
    "Jane"
   ]
  }
 ],
 "gender": "female",
 "birthDate": "1985-12-31",
 "meta": {
  "lastUpdated": "2020-11-23T06:43:45.133Z"
 }
}
```

有条件更新

条件更新允许根据某些标识搜索条件而不是逻辑 ID 更新现有资源。当服务器处理此更新时，它会使用其标准搜索功能对资源类型执行搜索，目标是解析此请求的单个逻辑 ID。

它采取的操作取决于找到的匹配项数量：

- 没有匹配项，请求正文中未提供 ID：服务器创建资源。
- 没有匹配项，已提供 ID 且资源不存在 ID：服务器将交互视为 “更新即创建” 交互。

- 没有匹配项，已提供 ID 且已存在：服务器以409 Conflict错误拒绝更新。
- One Match，未提供资源 ID 或（提供了资源 ID 并且它与找到的资源相匹配）：服务器对匹配的资源执行更新，如上所述，如果资源已更新，服务器将SHALL返回200 OK；
- One Match，提供了资源 ID 但与找到的资源不匹配：服务器返回409 Conflict错误，表明客户端 ID 规范有问题，最好是 OperationOutcome
- 多个匹配项：服务器返回一个412 Precondition Failed错误，表明客户端的标准选择性不够好，最好是 OperationOutcome

Example — 更新名字叫彼得、出生日期为 2000 年 1 月 1 日、电话号码 1234567890 的患者资源：

```
PUT https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient?name=peter&birthdate=2000-01-01&phone=1234567890
```

使用删除资源 DELETE

要删除 HealthLake 数据存储中的资源，必须提出DELETEHTTP请求。

Example 使用**DELETE**请求删除特定**Patient**资源类型。

要创建DELETE请求，请使用数据存储的终端节点。要查找数据存储的端点，请在 HealthLake 控制台的 “数据存储” 下查看，或者使用AWS HealthLake API参考资料中的 D escribeFHIRDatastore 操作。

您还必须包括资源类型和有效的标识符。

```
DELETE https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient/2de04858-ba65-44c1-8af1-f2fe69a977d9
```

HTTP 响应

成功后，您将收到一个204HTTP状态码，确认该资源已不在数据存储中。当删除请求失败时，您将收到 400 系列HTTP状态码，说明DELETE请求失败的原因。

使用 Bundle 管理多个FHIR资源

在 HL7 FHIR R4 规范中，捆绑包只是资源的集合。 HealthLake 支持在FHIRRESTAPI请求中创建 Bundle 资源类型，以及使用捆绑事务在单个FHIRRESTAPI请求中执行多个CRUD操作。在捆绑交易中，您必须像FHIRRESTAPI请求batch中一样指定捆绑包类型。

所有捆绑包请求均由记录 AWS CloudTrail。要了解有关 CloudTrail 与一起使用的更多信息 HealthLake，请参阅使用 AWS CloudTrail记录 AWS HealthLake API 调用。

HL7FHIRR4 资源（外部）

- 要阅读完整的规范，请参阅FHIR文档索引中的资源类型：捆绑包。
- 要阅读有关使用批量交互的信息 FHIR RESTAPI，请参阅FHIR文档索引FHIRRESTAPI中的 Batch 交互。

以下各节介绍如何构建FHIRRESTAPI请求，以便创建新的捆绑包资源或使用捆绑交易单独处理资源。

> ⚠ HealthLake 控制台 AWS CLI、和之间的区别 AWS SDKs
>
> HealthLake 控制台仅支持在FHIRRESTAPI请求中指定捆绑包资源类型的捆绑包类型操作 URL。

使用FHIR捆绑包执行多项CRUD操作

如果您的请求中未指定资源类型URL，则会将该FHIRRESTAPI请求解析为单个数据存储事务。对JSON正文中提供的每个CRUD操作进行评估，并返回特定的HTTP状态码。 HealthLake 支持捆绑包类型`batch`。

要在单个FHIRRESTAPI请求中执行多个CRUD操作，请执行以下操作：

以下列表显示了在捆绑FHIRRESTAPI请求中使用的请求正文中被截断的部分。有关完整的请求正文，请参阅创建涉及多个CRUD操作的捆绑请求。

1. 请勿在POST请求中指定资源类型：

```
POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
```

2. 在请求正文中，将 Bundle 类型指定为 `"type": "batch"`
3. 在请求正文中，从密钥开始为每次CRUD交互指定资源特定的数据。`resource`
4. 每个CRUD操作在请求正文`request`中都指定为，如下所示：

```
{ ...
  "request" : {
    "method" : "HTTP-VERB",
```

```
    "url" : "FHIR-RESOURCE-TYPE-URL"
  }
  ...
}
```

在JSON响应中，您将获得请求中指定的每个CRUD操作的HTTP状态码。

HealthLake 限制捆绑交易

- 要详细了解捆绑包的限制 HealthLake 位置，请参阅AWS HealthLake 终端节点和配额。

以下是包含多个CRUD操作的 Bundle 操作的示例。

Example — 创建涉及多个CRUD操作的 Bundle 请求。

要发出执行多项CRUD操作的FHIRRESTAPI请求，您必须使用您的数据存储终端节点POST发出请求并提供JSON请求正文。

您可以在 HealthLake 控制台的 “数据存储” 下找到数据存储的终端节点，也可以使用AWS HealthLake API参考资料中的 D escribeFHIRDatastore 操作来找到您的数据存储的终端节点。

POST Request

使用数据存储的终端节点POST发出请求。使用下一个选项卡 “JSON请求正文” 来查看请求正文的必需元素。

```
POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
```

JSON Request Body

在请求正文中，您必须提供以下 key: value 对以及有关各个请求的任何其他FHIR特定于资源的数据。CRUD第一个示例显示了一个被截断的JSON请求正文，突出显示了必需的元素。第二个示例显示了完整的JSON请求正文。

```
{
  "resourceType": "Bundle",
  "id": "bundle-batch-operation",
  "meta": {
    "lastUpdated": "2014-08-18T01:43:30Z"
  },
```

```
  "type": "batch", ## Required
  "entry": [
    {
      ## CRUD Transaction - 1
      "resource": {
        "resourceType": "Patient",
        ...
      },
      "request": { ## Required
        "method": "POST",
        "url": "Patient"
      }
    },
    {
      ## CRUD Transaction - 2
      "resource": {
        "resourceType": "Medication",
        ...
      },
      "request": { ## Required
        "method": "POST",
        "url": "Medication"
      }
    }
  ]
}
```

以下是一个完整的示例，展示了如何创建新的Patient和Medication资源类型。

```
{
  "resourceType": "Bundle",
  "id": "bundle-transaction",
  "meta": {
    "lastUpdated": "2014-08-18T01:43:30Z"
  },
  "type": "batch",
  "entry": [
    {
      "resource": {
        "resourceType": "Patient",
        "meta": {
          "lastUpdated": "2022-06-03T17:53:36.724Z"
        },
```

```
        "text": {
          "status": "generated",
          "div": "Some narrative"
        },
        "active": true,
        "name": [
          {
            "use": "official",
            "family": "Jackson",
            "given": [
              "Mateo",
              "James"
            ]
          }
        ],
        "gender": "male",
        "birthDate": "1974-12-25"
      },
      "request": {
        "method": "POST",
        "url": "Patient"
      }
    },
    {
      "resource": {
        "resourceType": "Medication",
        "id": "med0310",
        "contained": [
          {
            "resourceType": "Substance",
            "id": "sub03",
            "code": {
              "coding": [
                {
                  "system": "http://snomed.info/sct",
                  "code": "55452001",
                  "display": "Oxycodone (substance)"
                }
              ]
            }
          }
        ],
        "code": {
          "coding": [
```

```
            {
              "system": "http://snomed.info/sct",
              "code": "430127000",
              "display": "Oral Form Oxycodone (product)"
            }
          ]
        },
        "form": {
          "coding": [
            {
              "system": "http://snomed.info/sct",
              "code": "385055001",
              "display": "Tablet dose form (qualifier value)"
            }
          ]
        },
        "ingredient": [
          {
            "itemReference": {
              "reference": "#sub03"
            },
            "strength": {
              "numerator": {
                "value": 5,
                "system": "http://unitsofmeasure.org",
                "code": "mg"
              },
              "denominator": {
                "value": 1,
                "system": "http://terminology.hl7.org/CodeSystem/v3-
orderableDrugForm",
                "code": "TAB"
              }
            }
          }
        ]
      },
      "request": {
        "method": "POST",
        "url": "Medication"
      }
    }
  ]
```

```
}
```

JSON 响应

要确认创建示例捆绑交易中指定的资源，您将获得包含的每个CRUD操作201的 Created HTTP 状态代码。当CRUD操作失败时，您将获得 400 系列HTTP状态，指示单个请求失败的原因。

```
{
  "resourceType": "Bundle",
  "type": "batch-response",
  "timestamp": "2022-06-15T01:31:34.300+00:00",
  "entry": [
    {
      "response": {
        "status": "201",
        "location": "Patient/fd68ce38-ba30-4459-9eeb-476ad9f4f4ca",
        "lastModified": "2022-06-15T01:31:34.180+00:00"
      }
    },
    {
      "response": {
        "status": "201",
        "location": "Medication/5bf3b8cc-4076-4219-aba1-e2c53d7916f4",
        "lastModified": "2022-06-15T01:31:34.180+00:00"
      }
    }
  ]
}
```

将资源分组为 Bundle 资源类型

要创建新的 Bundle 资源类型，您必须在FHIRRESTAPI请求Bundle中指定并提供包含要组合在一起的资源的有效JSON正文。

在请求中指定 Bundle 后URL，JSON请求正文的内容将按原样保存在您的 HealthLake 数据存储中。因此，不能对单个资源类型执行任何CRUD操作。这种类型的捆绑包会被分配一个新的资源 ID。由于资源按原样保存，因此您无法对以 Bundle 资源类型保存的单个资源进行GET或POST请求。

> **Note**
>
> HL7FHIRR4 规范还支持使用分组、组合和列表对资源进行分组。创建这些资源类型时，不会直接包含各个资源。相反，他们使用Reference元素来指向各个资源。因此，使用这些资源类型可以修改其中包含的各个资源。

要创建Bundle资源类型，您必须在POST请求中指定该资源类型，并提供要包含的资源的JSON枚举。

Example — 使用**POST**请求创建 Bundle 资源

要创建bundle资源，请执行以下操作

1. 按以下方式格式化FHIRRESTAPI请求：

   POST https://healthlake.**your-region**.amazonaws.com/datastore/**your-datastore-id**/r4/**Bundle**

2. 提供 JSON body，指定要组合在一起的资源。此示例将两个患者资源分组。

```
{
  "resourceType": "Bundle",
  "id": "bundle-transaction",
  "meta": {
    "lastUpdated": "2018-03-11T11:22:16Z"
  },
  "type": "document",
  "entry": [
    {
      "resource": {
        "resourceType": "Patient",
        "name": [
          {
            "family": "Smith",
            "given": [
              "Jane"
            ]
          }
        ],
        "gender": "female",
        "address": [
          {
            "line": [
```

```
              "123 Main St."
            ],
            "city": "Anycity",
            "state": "Any State",
            "postalCode": "12345"
          }
        ]
      }
    },
    {
      "resource": {
        "resourceType": "Patient",
        "name": [
          {
            "family": "Jackson",
            "given": [
              "Mateo"
            ]
          }
        ],
        "gender": "male",
        "address": [
          {
            "line": [
              "1234 Main St."
            ],
            "city": "Anycity",
            "state": "Any State",
            "postalCode": "12345"
          }
        ]
      }
    }
  ]
}
```

使用FHIRRESTAPI操作搜索您的 HealthLake 数据存储

HealthLake 支持使用FHIR标准中提供的RESTAPI操作来搜索您的数据存储。在本节中，您将找到有关如何对多种不同资源类型GET进行POST请求的示例。

> **Note**
>
> 对于涉及个人身份信息 (PII) 或 Protected Health Infor PHI mation () 的查询，建议使用POST请求。在POST请求中，PII或作为请求PHI正文的一部分添加并在传输过程中进行加密。

该FHIR规范支持多种搜索参数类型，但仅 HealthLake 支持一个子集。有关更多信息，请参阅支持的搜索参数类型 和支持的高级搜索参数 HealthLake。

使用FHIRRESTAPI操作搜索您的数据存储。

- 支持的搜索参数类型
- 支持的高级搜索参数 HealthLake
  - _include
  - _revinclude
  - _summary
  - _elements
  - _total
  - _sort
  - _count
  - Chaining and Reverse Chaining(_has)
- 支持的搜索修饰符
- 支持的搜索比较器
- 不支持搜索参数 HealthLake
- 使用POST示例进行搜索
- 使用GET示例进行搜索

支持的搜索参数类型

下表显示了中支持的搜索参数类型 HealthLake。

支持的搜索参数类型

搜索参数	描述
_id	资源 ID（不是完整的URL）

搜索参数	描述
_lastUpdated	上次更新日期。服务器可以自行决定边界精度。
_tag	按资源标签搜索。
_个人资料	搜索所有标有个人资料的资源。
_安全	搜索应用于此资源的安全标签。
_来源	搜索资源来源。
_text	搜索资源的叙述。
createdAt	搜索自定义扩展程序-createdAt.

> **Note**
>
> 以下搜索参数仅适用于 2023 年 12 月 9 日之后创建的数据存储：_security、_source、_text、。createdAt

下表显示了如何根据给定资源类型的指定数据类型修改查询字符串的示例。为清楚起见，示例列中的特殊字符未经过编码。要成功查询，请确保查询字符串已正确编码。

搜索参数类型	详细信息	示例

数字	在指定资源中搜索数值。可以观察到重要的数字。
	
	有效位数是按搜索参数值确定的，不包括前导零。
	
	允许使用比较前缀。
		`[parameter]=100`
		
		`[parameter]=1e2`
		
		`[parameter]=lt100`

日期/ DateTime	搜索特定的日期或时间。预期的格式是，`yyyy-mm-ddThh:mm:ss[Z|(+|-)hh:mm]` 但可能有所不同。	`[parameter]=eq2013-01-14`

搜索参数类型	详细信息	示例

	接受以下数据类型：date、dateTimeinstant、Pe和Timing。有关在搜索中使用这些数据类型的更多详细信息，请参阅FHIR文档索引中的日期。
	
	允许使用比较前缀。
		`[parameter]=gt2013-01-14T10:00`
		
		`[parameter]=ne2013-01-14`

String	以区分大小写的方式搜索字符序列。
	
	同时支持HumanName和Address类型。有关更多详细信息，请参阅FHIR文档索引中的Address数据类型条目和数据类型条目。HumanName
	
	使用`:text`修饰符支持高级搜索。
		`[base]/Patient?given=eve`
		
		`[base]/Patient?given:contains=eve`

令牌	搜索 close-to-exact与一串字符的匹配项，通常与一对医疗代码值进行比较。 区分大小写与创建查询时使用的代码系统有关。基于Subsumption的查询可以帮助减少与区分大小写有关的问题。为清楚起见|，尚未编码。	`[parameter]=[system]|[code]` ：这里[system]指的是编码系统，[code]指的是在该特定系统中找到的代码值。 `[parameter]=[code]` ：在这里，您的输入将匹配代码或系统。 `[parameter]=|[code]` ：此处您的输入将与代码匹配，并且系统属性没有标识符。

搜索参数类型	详细信息	示例
复合键	使用修饰符$和,运算在单个资源类型中搜索多个参数。 允许使用比较前缀。	/Patient?language=FR,NL&language=EN Observation?component-code-value-quantity=http://loinc.org|8480-6$lt60 [base]/Group?characteristic-value=gender$mixed
Quantity	以值形式搜索数字、系统和代码。必须输入数字，但系统和代码是可选的。基于数量数据类型。有关更多详细信息，请参阅FHIR文档索引中的数量。 使用以下假设语法 [parameter]=[prefix][number]|[system]|[code]	[base]/Observation?value-quantity=5.4|http://unitsofmeasure.org|mg [base]/Observation?value-quantity=5.4|http://unitsofmeasure.org|mg [base]/Observation?value-quantity=5.4|http://unitsofmeasure.org|mg [base]/Observation?value-quantity=le5.4|http://unitsofmeasure.org|mg
参考	搜索对其他资源的引用。	[base]/Observation?subject=Patient/23 test

搜索参数类型	详细信息	示例
URI	搜索可明确标识特定资源的字符串。	`[base]/ValueSet?url=http://acme.org/fhir/ValueSet/123`
特殊	基于综合医疗NLP扩展进行搜索。	

支持的高级搜索参数 HealthLake

HealthLake 支持以下高级搜索参数。

名称	描述	示例	能力
_includ	用于请求在搜索请求中返回其他资源。它返回目标资源实例引用的资源。	`Encounter?_include=Encounter:subject`	
_revinc de	用于请求在搜索请求中返回其他资源。它返回引用主资源实例的资源。	`Patient?_id=` **`patient-identifier`** `&_revinclude=Encounter:patient`	
_summar	摘要可用于请求资源的子集。	`Patient?_summary=text`	支持以下摘要参数：`_summary=true` 、`_summary=false` 、`_summary=text` 、`_summary=data` 。
_elemen	请求在搜索结果中将一组特定的元素作为资源的一部分返回。	`Patient?_elements=identifie`	

名称	描述	示例	能力
		r,active, link	
_total	返回与搜索参数匹配的资源数量。	Patient?_ total=acc urate	Supp _total=accurate ort，_total=none 。
_sort	使用逗号分隔的列表表示返回的搜索结果的排序顺序。该-前缀可用于逗号分隔列表中的任何排序规则，以指示降序顺序。	Observati on?_sort= status,-date	Support 支持按带有类型的字段进行排序Number, String, Quantity, Token, URI, Reference 。Date仅在 2023 年 12 月 9 日之后创建的数据存储支持排序依据。Support 最多支持 5 个排序规则。
_count	控制搜索包中每页返回多少资源。	Patient?_ count=100	最大页面大小为 100。
chainin	搜索引用资源的元素。将链接搜索.定向到引用资源中的元素。	Diagnosti cReport?s ubject:Pa tient.nam e=peter	
reverse chainin (_has)	根据引用资源的元素搜索资源。	Patient?_ has:Obser vation:pa tient:cod e=1234-5	

_include

_include在搜索查询中使用还允许返回其他指定FHIR资源。用于包括_include向前链接的资源。

Example — 用于**_include**查找被诊断为咳嗽的患者或患者群体

你可以在指定咳嗽诊断代码的Condition资源类型上进行搜索，然后使用_include指定也要返回该诊断subject的代码。在Condition资源类型中，subject指的是患者资源类型或组资源类型。

为清楚起见，示例中的特殊字符未经过编码。要成功查询，请确保查询字符串已正确编码。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Condition?code=49727002&_include=Condition:subject
```

_revinclude

_revinclude在搜索查询中使用还允许返回其他指定FHIR资源。用于包含_revinclude向后链接的资源。

Example — 用于包括**_revinclude**与特定患者关联的相关遭遇和观察资源类型

要进行此搜索，首先要Patient通过在_id搜索参数中指定个人的标识符来定义个人。然后，您可以使用结构Encounter:patient和来指定其他FHIR资源Observation:patient。

为清楚起见，示例中的特殊字符未经过编码。要成功查询，请确保查询字符串已正确编码。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient?_id=patient-
identifier&_revinclude=Encounter:patient&_revinclude=Observation:patient
```

_summary

_summary在搜索查询中使用允许用户请求FHIR资源的子集。它可以包含以下值之一:true, text, data, false. 任何其他值都将被视为无效。返回的资源将在 meta.tag 'SUBSETTED' 中标记，以表示资源不完整。

- true：返回所有在资源基本定义中标记为 “摘要” 的受支持元素。
- text: 仅返回 “文本”、“id”、“meta” 元素，仅返回顶级必填元素。
- data: 返回除'文本'元素之外的所有部分。
- false: 返回资源的所有部分

在单个搜索请求中，_summary=text不能与_include或_revinclude搜索参数组合使用。

Example — 获取数据存储中患者资源的 “文本” 元素。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient?_summary=text
```

_elements

_elements在搜索查询中使用允许请求特定的FHIR资源元素。返回的资源将在 meta.tag 'SUBSETTED' 中标记，以表示资源不完整。

该_elements参数由以逗号分隔的基本元素名称列表组成，例如在资源中根级别定义的元素。只有列出的元素才会被返回。如果_elements参数值包含无效元素，服务器将忽略它们并返回必需元素和有效元素。

_elements不适用于包含的资源（搜索模式为的返回资源include）。

在单个搜索请求中，_elements不能与_summary搜索参数组合使用。

Example — 获取 HealthLake 数据存储中患者资源的 “标识符”、“活动”、“链接” 元素。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient?_elements=identifier,active,link
```

_total

_total在搜索查询中使用将返回与请求的搜索参数相匹配的资源数量。 HealthLake 将返回 of search 响应中匹配资源的总数（搜索模式为Bundle.total的返回资源match）。

_total支持accurate、none参数值。 _total=estimate不支持。任何其他值都将被视为无效。 _total不适用于包含的资源（搜索模式为的返回资源include）。

Example — 获取数据存储中患者资源的总数：

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient?_total=accurate
```

_sort

_sort在搜索查询中使用可按特定顺序排列结果。结果根据以逗号分隔的排序规则列表按优先顺序排序。排序规则应该是有效的搜索参数。任何其他值都将被视为无效。

在单个搜索请求中，您最多可以使用 5 个排序搜索参数。您可以选择使用-前缀来表示降序。默认情况下，服务器将按升序排序。

支持的排序搜索参数类型为:Number, String, Date, Quantity, Token, URI, Reference. 如果搜索参数指的是嵌套的元素，则排序不支持此搜索参数。例如，在资源类型的 “名称” 上搜索患者指的是患者。 HumanName 数据类型的名称元素被视为嵌套。因此，不支持按 “姓名” 对患者资源进行排序。

Example — 在数据存储中获取患者资源，并按出生日期升序对其进行排序：

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient?_sort=birthdate
```

_count

该参数_count被定义为向服务器发出的有关应在单个页面中返回多少资源的指令。

最大页面大小为 100。任何大于 100 的值均无效。 _count=0不支持。

Example — 搜索患者资源并将搜索页面大小设置为 25：

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient?_count=25
```

Chaining and Reverse Chaining(_has)

中的链接和反向链接FHIR提供了一种更高效、更紧凑的方式来获取相互关联的数据，从而减少了对多个单独查询的需求，并使开发人员和用户更方便地检索数据。

如果任何级别的递归返回的结果超过 100 个，则 HealthLake 将返回 4xx，以防止数据存储过载并导致多次分页。

Example — Chaining-获取所有 DiagnosticReport 指向患者姓名为 peter 的患者的内容。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
DiagnosticReport?subject:Patient.name=peter
```

Example — 反向链接-获取患者资源，其中患者资源由至少一个观察点引用，其中观察结果的代码为 1234，其中观察结果指患者搜索参数中的患者资源。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient?_has:Observation:patient:code=1234
```

支持的搜索修饰符

搜索修饰符用于基于字符串的字段。中的所有搜索修饰符都 HealthLake 使用基于布尔值的逻辑。例如，您可以指定:contains较大的字符串字段应包含一个小字符串，以便将其包含在搜索结果中。

支持的搜索修饰符

搜索修饰符	类型
:缺失	除外的所有参数 Composite
:精确	String
:包含	String
:不是	令牌
:文本	令牌
:标识符	参考

支持的搜索比较器

您可以使用搜索比较器来控制搜索中匹配的性质。在搜索数字、日期和数量字段时，您可以使用比较器。下表列出了支持的搜索比较器及其定义。 HealthLake

支持的搜索比较器

搜索比较器	描述
eq	资源中参数的值等于提供的值。
没有	资源中参数的值不等于提供的值。
gt	资源中参数的值大于提供的值。
lt	资源中参数的值小于提供的值。

搜索比较器	描述
ge	资源中参数的值大于或等于提供的值。
le	资源中参数的值小于或等于提供的值。
sa	资源中参数的值从提供的值之后开始。
eb	资源中参数的值在提供的值之前结束。

不支持搜索参数 HealthLake

有关支持的搜索参数的完整列表，请参阅FHIR搜索参数注册表。 HealthLake 支持除表中列出的参数之外的所有搜索参数。

不支持的搜索参数

捆绑包构成	位置-附近
捆绑包标识符	C onsent-source-reference
捆绑消息	合同患者
捆绑包类型	资源内容
捆绑包时间戳	资源查询

使用POST示例进行搜索

您可以通过提出POST请求来搜索 HealthLake 数据存储。您可以在URI或请求正文中提供查询参数，但不能在单个请求中同时使用这两个参数。

本主题中的示例遵循了该最佳实践。

> **Note**
>
> 对于涉及个人身份信息 (PII) 或 Protected Health Infor PHI mation () 的查询，建议使用POST请求。在POST请求中，PII或作为请求PHI正文的一部分添加并在传输过程中进行加密。

使用POST请求正文中的参数发出请求时，请Content-Type: application/x-www-form-urlencoded将其用作标头的一部分。

本主题为您提供了如何使用以下资源类型POST进行搜索的示例。

- 年龄：年龄不是中定义的资源类型FHIR。相反，年龄是作为患者资源类型的一部分捕获的。要根据特定年龄或年龄范围搜索一组患者，请使用the section called “支持的搜索比较器”。有关更多详细信息，请参阅资源类型：FHIR文档索引中的患者。
- 病情：此资源类型存储与临床概念相关的详细信息，例如诊断、情况、临床状况以及已达到令人担忧程度的问题。要了解更多信息，请参阅FHIR文档索引中的资源类型：条件。 HealthLake 根据中找到的文档创建新条件 DocumentReference。在发出POST请求时，默认不包括这些新增内容。要将它们包括在内，您必须在搜索中为条件资源指定有效的标识符。
- DocumentReference:此资源类型受支持。 HealthLake此资源类型支持引用任何类型的文档。要了解更多信息，请参阅FHIR文档索引 DocumentReference中的资源类型：。 HealthLake 还为中找到的文档提供集成的自然语言处理 (NLP) DocumentReference。要了解更多信息，请参阅 使用基于资源类型的自然语言处理 (NLP) 的自动FHIR DocumentReference 资源生成 AWS HealthLake。
- 地点：此资源类型包括附带地点（未经事先指定或授权用于医疗保健的地方）和正式指定的专用地点。有关更多详细信息，请参阅资源类型：FHIR文档索引中的位置。
- 观察：对患者、设备或其他受试者进行的测量和简单断言。 HealthLake 根据资源中找到的文档创建新的观测 DocumentReference 资源。要详细了解如何 HealthLake 创建新资源，请参阅使用基于资源类型的自然语言处理 (NLP) 的自动FHIR DocumentReference 资源生成 AWS HealthLake。在发出POST请求时，默认不包括这些新增内容。要将它们包括在内，您必须在搜索中为观测资源指定有效的标识符。要了解更多信息，请参阅资源类型：FHIR文档索引中的观察。

每个选项卡都显示了如何搜索指定资源类型的示例。它包括一个如何在请求正文中指定请求的示例。

Age

使用以下方法对Patient资源类型发出POST基于搜索的请求。此搜索使用eq搜索比较器来搜索出生于 1997 年的个人。

你必须指定一个请求URL和一个请求正文。以下是一个请求示例URL。

```
POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient/_search
```

要在搜索中指定 1997 年，您需要在请求正文中添加以下元素。

```
birthdate=eq1997
```

JSON 响应

成功后，您将获得200HTTP响应代码和类似的JSON响应。

Condition

使用以下内容对Condition资源类型POST提出请求。此搜索可在您的 HealthLake 数据存储中查找包含医疗代码的位置72892002。

你必须指定一个请求URL和一个请求正文。以下是一个请求示例URL。

```
POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Condition/_search
```

要指定要搜索的医疗代码，请将此JSON元素添加到请求正文中。

```
code=72892002
```

JSON 响应

成功后，您将获得200HTTP响应码。为清楚起见，以下JSON响应已被截断。

```
{
 "resourceType": "Bundle",
 "type": "searchset",
 "entry": [{
   "resource": {
    "resourceType": "Condition",
    "id": "0063326c-6b42-4d13-af2f-1efe0a65f016",
    "meta": {
     "lastUpdated": "2022-08-23T00:22:49.681Z"
    },
    "clinicalStatus": {
     "coding": [{
      "system": "http://terminology.hl7.org/CodeSystem/condition-clinical",
      "code": "resolved"
     }]
    },
    "verificationStatus": {
     "coding": [{
      "system": "http://terminology.hl7.org/CodeSystem/condition-ver-status",
```

```
      "code": "confirmed"
     }]
    },
    "code": {
     "coding": [{
      "system": "http://snomed.info/sct",
      "code": "72892002",
      "display": "Normal pregnancy"
     }],
     "text": "Normal pregnancy"
    },
    "subject": {
     "reference": "Patient/5fc0070a-696a-4855-94a9-175f1c641a33"
    },
    "encounter": {
     "reference": "Encounter/44078ab9-7ac7-4731-9ac8-4b3ff21a7bdb"
    },
    "onsetDateTime": "2019-08-15T01:19:17-07:00",
    "abatementDateTime": "2020-03-26T01:19:17-07:00",
    "recordedDate": "2019-08-15T01:19:17-07:00"
   },
   "search": {
    "mode": "match"
   }
  },
  {
   "resource": {
    "resourceType": "Condition",
    "id": "d00afdb2-1d2c-44fe-9f3b-033c0fe751a3",
    "meta": {
     "lastUpdated": "2022-08-23T00:20:47.100Z"
    },
    "clinicalStatus": {
     "coding": [{
      "system": "http://terminology.hl7.org/CodeSystem/condition-clinical",
      "code": "resolved"
     }]
    },
    "verificationStatus": {
     "coding": [{
      "system": "http://terminology.hl7.org/CodeSystem/condition-ver-status",
      "code": "confirmed"
     }]
    },
```

```
    "code": {
     "coding": [{
      "system": "http://snomed.info/sct",
      "code": "72892002",
      "display": "Normal pregnancy"
     }],
     "text": "Normal pregnancy"
    },
    "subject": {
     "reference": "Patient/d0a5cd1e-8da7-41bd-9b2f-41eef45246e5"
    },
    "encounter": {
     "reference": "Encounter/73758e67-4aaf-4e80-982b-8821f0b6fdfb"
    },
    "onsetDateTime": "2019-06-13T20:37:40-07:00",
    "abatementDateTime": "2020-01-23T19:37:40-08:00",
    "recordedDate": "2019-06-13T20:37:40-07:00"
   },
   "search": {
    "mode": "match"
   }
  }
 ]
}
```

DocumentReference

要在 HealthLake对DocumentReference资源类型POST发出请求时查看集成自然语言处理 (NLP) 的结果，请按以下方式格式化请求。

```
POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
DocumentReference/_search
```

要指定 DocumentReference 要引用的元素，请参阅搜索参数。您将在请求正文中将其指定为 JSON。

```
_lastUpdated=le2021-12-19&infer-icd10cm-entity-text-concept-score;=streptococcal|
0.6&infer-rxnorm-entity-text-concept-score=Amoxicillin|0.8
```

此查询字符串使用多个搜索参数来搜索用于生成综合API医疗结果的 Amazon Comprehend Medical 操作。NLP

Location

使用以下命令对Location资源类型POST提出请求。此搜索会在您的 HealthLake 数据存储中查找地址中包含城市名称波士顿的位置。

您必须指定请求URL和请求正文。以下是一个请求示例URL。

```
POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/Location/_search
```

要在搜索中指定 Boston，请在请求正文中添加以下元素：

```
address=Boston
```

JSON 响应

成功后，您将获得200HTTP响应码。为清楚起见，JSON回复已被截断。

```
{
 "resourceType": "Bundle",
 "type": "searchset",
 "entry": [{
   "resource": {
    "resourceType": "Location",
    "id": "0a6903c7-25c5-4ae4-8354-be88f9c5f2ee",
    "meta": {
     "lastUpdated": "2022-08-23T00:24:24.570Z"
    },
    "status": "active",
    "name": "BRIGHAM AND WOMEN'S HOSPITAL",
    "telecom": [{
     "system": "phone",
     "value": "6177325500"
    }],
    "address": {
     "line": [
      "75 FRANCIS STREET"
     ],
     "city": "BOSTON",
     "state": "MA",
     "postalCode": "02115",
     "country": "US"
    },
```

```
    "position": {
     "longitude": -71.020173,
     "latitude": 42.33196
    },
    "managingOrganization": {
     "reference": "Organization/27379046-608b-32f0-9df7-8c833cf5d11d",
     "display": "BRIGHAM AND WOMEN'S HOSPITAL"
    }
   },
   "search": {
    "mode": "match"
   }
  },

  {
   "resource": {
    "resourceType": "Location",
    "id": "ca5e7f65-4eb5-4bff-9a6f-07bc80acf8d0",
    "meta": {
     "lastUpdated": "2022-08-23T00:20:47.100Z"
    },
    "status": "active",
    "name": "BETH ISRAEL DEACONESS MEDICAL CENTER",
    "telecom": [{
     "system": "phone",
     "value": "6176677000"
    }],
    "address": {
     "line": [
      "330 BROOKLINE AVENUE"
     ],
     "city": "BOSTON",
     "state": "MA",
     "postalCode": "02215",
     "country": "US"
    },
    "position": {
     "longitude": -71.020173,
     "latitude": 42.33196
    },
    "managingOrganization": {
     "reference": "Organization/cb6a50e0-af76-3758-99ad-3200ede03fff",
     "display": "BETH ISRAEL DEACONESS MEDICAL CENTER"
    }
```

```
   },
   "search": {
    "mode": "match"
   }
  }
 ]
}
```

Observation

使用以下内容对Observation资源类型发出POST基于搜索的请求。此搜索使用value-concept搜索参数来查找医疗代码266919005。此状态表明Never smoker。

你必须指定一个请求URL和一个请求正文。以下是一个请求示例URL。

```
POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Observation/_search
```

要指定状态Never smoker，请在正文value-concept=266919005中设置JSON。

```
value-concept=266919005
```

JSON 响应

成功后，您将获得200HTTP响应码。为清楚起见，以下JSON响应已被截断。

```
{
 "resourceType": "Bundle",
 "type": "searchset",
 "link": [{
  "relation": "next",
  "url": "https://healthlake.us-west-2.amazonaws.com/
datastore/3651c6d3c1e81e785adba06b710b52a9/r4/Observation?value-
concept=266919005&=AAMA-
EFRSURBSGlpcGIyN250ZG9WRXVnTTFOdmtxQk9Bb3Y0YjhVcVdUMGV0eVozNmdjQU9nRjRNUUtscjhCZ1NMUG84VGNqN
 }],
 "entry": [{
   "resource": {
    "resourceType": "Observation",
    "id": "000038e0-71c6-4cc0-9c6c-50c8b1c53309",
    "meta": {
     "lastUpdated": "2022-11-03T01:02:38.981Z"
    },
```

```
    "status": "final",
    "category": [{
     "coding": [{
      "system": "http://terminology.hl7.org/CodeSystem/observation-category",
      "code": "survey",
      "display": "survey"
     }]
    }],
    "code": {
     "coding": [{
      "system": "http://loinc.org",
      "code": "72166-2",
      "display": "Tobacco smoking status NHIS"
     }],
     "text": "Tobacco smoking status NHIS"
    },
    "subject": {
     "reference": "Patient/598c9d7a-0494-448e-a81e-d50e3606e8db"
    },
    "encounter": {
     "reference": "Encounter/86bdee4a-2aa9-474a-b43f-6237cd68e512"
    },
    "effectiveDateTime": "2019-12-11T19:44:57-08:00",
    "issued": "2019-12-11T19:44:57.438-08:00",
    "valueCodeableConcept": {
     "coding": [{
      "system": "http://snomed.info/sct",
      "code": "266919005",
      "display": "Never smoker"
     }],
     "text": "Never smoker"
    }
   },
   "search": {
    "mode": "match"
   }
  },

  {
   "resource": {
    "resourceType": "Observation",
    "id": "0c2f6260-e671-4cfd-ac3d-e75f073fa3cd",
    "meta": {
     "lastUpdated": "2022-11-03T01:05:21.488Z"
```

```
    },
    "status": "final",
    "category": [{
     "coding": [{
      "system": "http://terminology.hl7.org/CodeSystem/observation-category",
      "code": "survey",
      "display": "survey"
     }]
    }],
    "code": {
     "coding": [{
      "system": "http://loinc.org",
      "code": "72166-2",
      "display": "Tobacco smoking status NHIS"
     }],
     "text": "Tobacco smoking status NHIS"
    },
    "subject": {
     "reference": "Patient/89d9a9b7-9720-4881-a2ab-d7907544b26f"
    },
    "encounter": {
     "reference": "Encounter/8ebba7b0-fdfc-4ec1-a9aa-907cccf60925"
    },
    "effectiveDateTime": "2018-11-17T03:59:36-08:00",
    "issued": "2018-11-17T03:59:36.550-08:00",
    "valueCodeableConcept": {
     "coding": [{
      "system": "http://snomed.info/sct",
      "code": "266919005",
      "display": "Never smoker"
     }],
     "text": "Never smoker"
    }
   },
   "search": {
    "mode": "match"
   }
  }
 ]
}
```

使用GET示例进行搜索

您可以通过提出GET请求来搜索 HealthLake 数据存储。 HealthLake 仅支持将查询参数作为请求正文的一部分提供URI，而不支持作为请求正文的一部分提供。

> **Note**
>
> 对于涉及个人身份信息 (PII) 或 Protected Health Infor PHI mation () 的查询，建议使用POST请求。在POST请求中，PII或作为请求PHI正文的一部分添加并在传输过程中进行加密。

本主题提供了如何GET使用中支持的资源类型进行搜索的示例 HealthLake。

- 年龄：年龄不是中定义的资源类型FHIR。相反，年龄是作为患者资源类型的一部分捕获的。要根据特定年龄或年龄范围搜索一组患者，您需要使用the section called “支持的搜索比较器”. 有关更多详细信息，请参阅资源类型：FHIR文档索引中的患者。
- 病情：此资源类型存储与临床概念相关的详细信息，例如诊断、情况、临床状况以及已达到令人担忧程度的问题。要了解更多信息，请参阅FHIR文档索引中的资源类型：条件。 HealthLake 根据中找到的文档创建新条件 DocumentReference。在发出POST请求时，默认不包括这些新增内容。要将它们包括在内，您必须在搜索中为条件资源指定有效的标识符。
- DocumentReference:此资源类型受支持。 HealthLake此资源类型支持引用任何类型的文档。要了解更多信息，请参阅FHIR文档索引 DocumentReference中的资源类型：。 HealthLake 还为中找到的文档提供集成的自然语言处理 (NLP) DocumentReference。要了解更多信息，请参阅 使用基于资源类型的自然语言处理 (NLP) 的自动FHIR DocumentReference 资源生成 AWS HealthLake。
- 地点：此资源类型包括附带地点（未经事先指定或授权用于医疗保健的地方）和正式指定的专用地点。有关更多详细信息，请参阅资源类型：FHIR文档索引中的位置。
- 观察：对患者、设备或其他受试者进行的测量和简单断言。 HealthLake 根据资源中找到的文档创建新的观测 DocumentReference 资源。要了解有关如何 HealthLake 创建新资源的更多信息，请参阅使用基于资源类型的自然语言处理 (NLP) 的自动FHIR DocumentReference 资源生成 AWS HealthLake。在发出POST请求时，默认不包括这些新增内容。要将它们包括在内，您必须在搜索中为观测资源指定有效的标识符。要了解更多信息，请参阅资源类型：FHIR文档索引中的观察。

每个选项卡都显示了如何搜索指定资源类型的示例。它包括一个关于如何在中指定请求的示例URI，以及相关的JSON响应。

Age

使用以下命令对Patient资源类型发出GET基于搜索的请求。此搜索使用eq搜索比较器来搜索出生于 1997 年的个人。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4//
Patient?birthdate=eq1997
```

JSON 响应

成功后，您将获得200HTTP响应码。

Condition

使用以下命令对Condition资源类型GET提出请求。此搜索可在您的 HealthLake 数据存储中查找包含医疗代码的位置72892002。

您必须指定请求URL和请求正文。这是一个请求示例URL。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Condition?code=72892002
```

JSON 响应

成功后，您将获得200HTTP响应码。为清楚起见，以下JSON响应已被截断。

```
{
 "resourceType": "Bundle",
 "type": "searchset",
 "entry": [{
   "resource": {
    "resourceType": "Condition",
    "id": "0063326c-6b42-4d13-af2f-1efe0a65f016",
    "meta": {
     "lastUpdated": "2022-08-23T00:22:49.681Z"
    },
    "clinicalStatus": {
     "coding": [{
      "system": "http://terminology.hl7.org/CodeSystem/condition-clinical",
      "code": "resolved"
     }]
    },
    "verificationStatus": {
     "coding": [{
```

```
      "system": "http://terminology.hl7.org/CodeSystem/condition-ver-status",
      "code": "confirmed"
     }]
    },
    "code": {
     "coding": [{
      "system": "http://snomed.info/sct",
      "code": "72892002",
      "display": "Normal pregnancy"
     }],
     "text": "Normal pregnancy"
    },
    "subject": {
     "reference": "Patient/5fc0070a-696a-4855-94a9-175f1c641a33"
    },
    "encounter": {
     "reference": "Encounter/44078ab9-7ac7-4731-9ac8-4b3ff21a7bdb"
    },
    "onsetDateTime": "2019-08-15T01:19:17-07:00",
    "abatementDateTime": "2020-03-26T01:19:17-07:00",
    "recordedDate": "2019-08-15T01:19:17-07:00"
   },
   "search": {
    "mode": "match"
   }
  },
  {
   "resource": {
    "resourceType": "Condition",
    "id": "d00afdb2-1d2c-44fe-9f3b-033c0fe751a3",
    "meta": {
     "lastUpdated": "2022-08-23T00:20:47.100Z"
    },
    "clinicalStatus": {
     "coding": [{
      "system": "http://terminology.hl7.org/CodeSystem/condition-clinical",
      "code": "resolved"
     }]
    },
    "verificationStatus": {
     "coding": [{
      "system": "http://terminology.hl7.org/CodeSystem/condition-ver-status",
      "code": "confirmed"
     }]
```

```
    },
    "code": {
     "coding": [{
      "system": "http://snomed.info/sct",
      "code": "72892002",
      "display": "Normal pregnancy"
     }],
     "text": "Normal pregnancy"
    },
    "subject": {
     "reference": "Patient/d0a5cd1e-8da7-41bd-9b2f-41eef45246e5"
    },
    "encounter": {
     "reference": "Encounter/73758e67-4aaf-4e80-982b-8821f0b6fdfb"
    },
    "onsetDateTime": "2019-06-13T20:37:40-07:00",
    "abatementDateTime": "2020-01-23T19:37:40-08:00",
    "recordedDate": "2019-06-13T20:37:40-07:00"
   },
   "search": {
    "mode": "match"
   }
  }
 ]
}
```

DocumentationReference

此示例说明如何针对诊断为链球菌且同时服用阿莫西林处方的患者创建DocumentReference资源类型的搜索请求。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
DocumentReference?_lastUpdated=le2021-12-19&infer-icd10cm-entity-text-concept-
score;=streptococcal|0.6&infer-rxnorm-entity-text-concept-score=Amoxicillin|0.8
```

成功后，您将收到以下JSON响应。

```
{
    "resourceType": "Bundle",
    "type": "searchset",
    "entry": [
    {
       "resource": {
```

```
       "resourceType": "DocumentReference",
       "id": "985c3e94-4219-4c79-97a1-c94694525e24",
       "meta": {
          "lastUpdated": "2020-11-23T06:09:10.719Z"
        },
        "extension": [
          {
            "url": "http://healthlake.amazonaws.com/aws-cm/",
            "extension": [
              {
                "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/",
                "extension": [
                  {
                    "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/raw-
response",
                    "valueString": "{Entities: [{Id: 0,Text: otitis media,Category:
 MEDICAL_CONDITION,Type: DX_NAME,Score: 0.9815994,BeginOffset: 151,EndOffset:
 163,Attributes: [],Traits: [{Name: DIAGNOSIS,Score: 0.95042425}],ICD10CMConcepts:
 [{Description: Otitis media, unspecified, unspecified ear,Code: H66.90,Score:
 0.7176407}, {Description: Otitis media, unspecified,Code: H66.9,Score:
 0.6930445}, {Description: Otitis media, unspecified, left ear,Code: H66.92,Score:
 0.688161}, {Description: Otitis media, unspecified, bilateral,Code: H66.93,Score:
 0.6748094}, {Description: Otitis media, unspecified, right ear,Code:
 H66.91,Score: 0.6645618}]}, {Id: 1,Text: streptococcal sore throat,Category:
 MEDICAL_CONDITION,Type: DX_NAME,Score: 0.92208487,BeginOffset: 461,EndOffset:
 486,Attributes: [],Traits: [],ICD10CMConcepts: [{Description: Streptococcal
 pharyngitis,Code: J02.0,Score: 0.55638546}, {Description: Acute streptococcal
 tonsillitis, unspecified,Code: J03.00,Score: 0.53159785}, {Description:
 Streptococcal sepsis, unspecified,Code: A40.9,Score: 0.51865804}, {Description:
 Acute pharyngitis, unspecified,Code: J02.9,Score: 0.45085955}, {Description:
 Streptococcal infection, unspecified site,Code: A49.1,Score: 0.41550553}]},
 {Id: 3,Text: disorder,Category: MEDICAL_CONDITION,Type: DX_NAME,Score:
 0.9191257,BeginOffset: 488,EndOffset: 496,Attributes: [],Traits: [{Name:
 DIAGNOSIS,Score: 0.93372077}],ICD10CMConcepts: [{Description: Parkinson's
 disease,Code: G20,Score: 0.6959145}, {Description: Illness, unspecified,Code:
 R69,Score: 0.68428487}, {Description: Disorder of bone, unspecified,Code:
 M89.9,Score: 0.6542605}, {Description: Unspecified mental disorder due to known
 physiological condition,Code: F09,Score: 0.6240179}, {Description: Mental disorder,
 not otherwise specified,Code: F99,Score: 0.61046}]}],ModelVersion: 0.1.0}"
                  },
                  {
                    "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/
model-version",
                    "valueString": "0.1.0"
```

```
                    },
                    {
                      "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/aws-
cm-icd10-entity",
                      "extension": [
                        {
                          "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/
aws-cm-icd10-entity-id",
                          "valueInteger": 0
                        },
                        {
                          "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/
aws-cm-icd10-entity-text",
                          "valueString": "otitis media"
                        },
                        {
                          "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/
aws-cm-icd10-entity-begin-offset",
                          "valueInteger": 151
                        },
                        {
                          "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/
aws-cm-icd10-entity-end-offset",
                          "valueInteger": 163
                        },
                        {
                          "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/
aws-cm-icd10-entity-score",
                          "valueDecimal": 0.9815994
                        },
                        {
                          "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/
aws-cm-icd10-entity-ConceptList",
                          "extension": [
                            {
                              "url": "http://healthlake.amazonaws.com/aws-cm/infer-
icd10/aws-cm-icd10-entity-Concept",
                              "extension": [
                                {
                                  "url": "http://healthlake.amazonaws.com/aws-cm/
infer-icd10/aws-cm-icd10-entity-Concept-Code",
                                  "valueString": "H66.90"
                                },
                                {
```

```
                                "url": "http://healthlake.amazonaws.com/aws-cm/
infer-icd10/aws-cm-icd10-entity-Concept-Description",
                                "valueString": "Otitis media, unspecified,
 unspecified ear"
                              },
                              {
                                "url": "http://healthlake.amazonaws.com/aws-cm/
infer-icd10/aws-cm-icd10-entity-Concept-Score",
                                "valueDecimal": 0.7176407
                              }
                            ]
                          },
                          {
                            "url": "http://healthlake.amazonaws.com/aws-cm/infer-
icd10/aws-cm-icd10-entity-Concept",
                            "extension": [
                              {
                                "url": "http://healthlake.amazonaws.com/aws-cm/
infer-icd10/aws-cm-icd10-entity-Concept-Code",
                                "valueString": "H66.9"
                              },
                              {
                                "url": "http://healthlake.amazonaws.com/aws-cm/
infer-icd10/aws-cm-icd10-entity-Concept-Description",
                                "valueString": "Otitis media, unspecified"
                              },
                              {
                                "url": "http://healthlake.amazonaws.com/aws-cm/
infer-icd10/aws-cm-icd10-entity-Concept-Score",
                                "valueDecimal": 0.6930445
                              }
                            ]
                          },
                          {
                            "url": "http://healthlake.amazonaws.com/aws-cm/infer-
icd10/aws-cm-icd10-entity-Concept",
                            "extension": [
                              {
                                "url": "http://healthlake.amazonaws.com/aws-cm/
infer-icd10/aws-cm-icd10-entity-Concept-Code",
                                "valueString": "H66.92"
                              }
                            ]
                          }
```

Location

使用以下命令对Location资源类型GET提出请求。此搜索会在您的 HealthLake 数据存储中查找地址中包含城市名称波士顿的位置。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4//
Location?address=boston
```

JSON 响应

成功后，您将获得200HTTP响应码。为清楚起见，JSON回复已被截断。

```
{
    "resourceType": "Bundle",
    "type": "searchset",
    "entry": [
        {
            "resource": {
                "resourceType": "Location",
                "id": "0a6903c7-25c5-4ae4-8354-be88f9c5f2ee",
                "meta": {
                    "lastUpdated": "2022-08-23T00:24:24.570Z"
                },
                "status": "active",
                "name": "BRIGHAM AND WOMEN'S HOSPITAL",
                "telecom": [
                    {
                        "system": "phone",
                        "value": "6177325500"
                    }
                ],
                "address": {
                    "line": [
                        "75 FRANCIS STREET"
                    ],
                    "city": "BOSTON",
                    "state": "MA",
                    "postalCode": "02115",
                    "country": "US"
                },
                "position": {
```

```
                    "longitude": -71.020173,
                    "latitude": 42.33196
                },
                "managingOrganization": {
                    "reference":
 "Organization/27379046-608b-32f0-9df7-8c833cf5d11d",
                    "display": "BRIGHAM AND WOMEN'S HOSPITAL"
                }
            },
            "search": {
                "mode": "match"
            }
        },
        {
            "resource": {
                "resourceType": "Location",
                "id": "3cc3ad99-e0ff-48b4-b277-052abfc41058",
                "meta": {
                    "lastUpdated": "2022-08-23T00:19:37.029Z"
                },
                "status": "active",
                "name": "NEW ENGLAND BAPTIST HOSPITAL",
                "telecom": [
                    {
                        "system": "phone",
                        "value": "6177545800"
                    }
                ],
                "address": {
                    "line": [
                        "125 PARKER HILL AVENUE"
                    ],
                    "city": "BOSTON",
                    "state": "MA",
                    "postalCode": "02120",
                    "country": "US"
                },
                "position": {
                    "longitude": -71.020173,
                    "latitude": 42.33196
                },
                "managingOrganization": {
                    "reference": "Organization/9a7149fa-49fc-3c87-b935-
d29c55808717",
```

```
                    "display": "NEW ENGLAND BAPTIST HOSPITAL"
                }
            },
            "search": {
                "mode": "match"
            }
        },
        {
            "resource": {
                "resourceType": "Location",
                "id": "3f956715-3890-4235-85be-3fba5e3488ee",
                "meta": {
                    "lastUpdated": "2022-08-23T00:23:38.981Z"
                },
                "status": "active",
                "name": "MASSACHUSETTS GENERAL HOSPITAL",
                "telecom": [
                    {
                        "system": "phone",
                        "value": "6177262000"
                    }
                ],
                "address": {
                    "line": [
                        "55 FRUIT STREET"
                    ],
                    "city": "BOSTON",
                    "state": "MA",
                    "postalCode": "02114",
                    "country": "US"
                },
                "position": {
                    "longitude": -71.020173,
                    "latitude": 42.33196
                },
                "managingOrganization": {
                    "reference": "Organization/d78e84ec-30aa-3bba-a33a-
f29a3a454662",
                    "display": "MASSACHUSETTS GENERAL HOSPITAL"
                }
            },
            "search": {
                "mode": "match"
            }
```

```
        },
        {
            "resource": {
                "resourceType": "Location",
                "id": "6cc07b51-7287-443c-b772-c864f7831e13",
                "meta": {
                    "lastUpdated": "2022-08-23T00:21:11.045Z"
                },
                "status": "active",
                "name": "TUFTS MEDICAL CENTER",
                "telecom": [
                    {
                        "system": "phone",
                        "value": "6176365000"
                    }
                ],
                "address": {
                    "line": [
                        "800 WASHINGTON STREET"
                    ],
                    "city": "BOSTON",
                    "state": "MA",
                    "postalCode": "02111",
                    "country": "US"
                },
                "position": {
                    "longitude": -71.020173,
                    "latitude": 42.33196
                },
                "managingOrganization": {
                    "reference": "Organization/b7175ab4-
bde5-3848-891b-579bccb77c7c",
                    "display": "TUFTS MEDICAL CENTER"
                }
            },
            "search": {
                "mode": "match"
            }
        },
        {
            "resource": {
                "resourceType": "Location",
                "id": "8101300f-f685-49e7-b428-43b7855c39ee",
                "meta": {
```

```
                    "lastUpdated": "2022-08-23T00:22:06.474Z"
                },
                "status": "active",
                "name": "BOSTON CHILDREN'S HOSPITAL",
                "telecom": [
                    {
                        "system": "phone",
                        "value": "6177356000"
                    }
                ],
                "address": {
                    "line": [
                        "300 LONGWOOD AVENUE"
                    ],
                    "city": "BOSTON",
                    "state": "MA",
                    "postalCode": "02115",
                    "country": "US"
                },
                "position": {
                    "longitude": -71.020173,
                    "latitude": 42.33196
                },
                "managingOrganization": {
                    "reference": "Organization/d7b11827-25f2-350b-
bcd8-939fc59851b0",
                    "display": "BOSTON CHILDREN'S HOSPITAL"
                }
            },
            "search": {
                "mode": "match"
            }
        },
        {
            "resource": {
                "resourceType": "Location",
                "id": "8b7641d3-6997-48bb-bd60-23e35dfaae9d",
                "meta": {
                    "lastUpdated": "2022-08-23T00:20:47.099Z"
                },
                "status": "active",
                "name": "BRIGHAM AND WOMEN'S FAULKNER HOSPITAL",
                "telecom": [
                    {
```

```
                        "system": "phone",
                        "value": "6179837000"
                    }
                ],
                "address": {
                    "line": [
                        "1153 CENTRE STREET"
                    ],
                    "city": "BOSTON",
                    "state": "MA",
                    "postalCode": "02130",
                    "country": "US"
                },
                "position": {
                    "longitude": -71.020173,
                    "latitude": 42.33196
                },
                "managingOrganization": {
                    "reference": "Organization/d733d4a9-080d-3593-
b910-2366e652b7ea",
                    "display": "BRIGHAM AND WOMEN'S FAULKNER HOSPITAL"
                }
            },
            "search": {
                "mode": "match"
            }
        },
        {
            "resource": {
                "resourceType": "Location",
                "id": "998ef80b-7b58-4dc3-99ac-c440ec9e282d",
                "meta": {
                    "lastUpdated": "2022-08-23T00:21:11.046Z"
                },
                "status": "active",
                "name": "BRIGHAM AND WOMEN'S FAULKNER HOSPITAL",
                "telecom": [
                    {
                        "system": "phone",
                        "value": "6179837000"
                    }
                ],
                "address": {
                    "line": [
```

```
                        "1153 CENTRE STREET"
                    ],
                    "city": "BOSTON",
                    "state": "MA",
                    "postalCode": "02130",
                    "country": "US"
                },
                "position": {
                    "longitude": -71.020173,
                    "latitude": 42.33196
                },
                "managingOrganization": {
                    "reference": "Organization/d733d4a9-080d-3593-
b910-2366e652b7ea",
                    "display": "BRIGHAM AND WOMEN'S FAULKNER HOSPITAL"
                }
            },
            "search": {
                "mode": "match"
            }
        },
        {
            "resource": {
                "resourceType": "Location",
                "id": "c454bed3-7013-4376-81cf-4f49342f1402",
                "meta": {
                    "lastUpdated": "2022-08-23T00:24:24.573Z"
                },
                "status": "active",
                "name": "MASSACHUSETTS GENERAL HOSPITAL",
                "telecom": [
                    {
                        "system": "phone",
                        "value": "6177262000"
                    }
                ],
                "address": {
                    "line": [
                        "55 FRUIT STREET"
                    ],
                    "city": "BOSTON",
                    "state": "MA",
                    "postalCode": "02114",
                    "country": "US"
```

```
                },
                "position": {
                    "longitude": -71.020173,
                    "latitude": 42.33196
                },
                "managingOrganization": {
                    "reference": "Organization/d78e84ec-30aa-3bba-a33a-
f29a3a454662",
                    "display": "MASSACHUSETTS GENERAL HOSPITAL"
                }
            },
            "search": {
                "mode": "match"
            }
        },
        {
            "resource": {
                "resourceType": "Location",
                "id": "ca5e7f65-4eb5-4bff-9a6f-07bc80acf8d0",
                "meta": {
                    "lastUpdated": "2022-08-23T00:20:47.100Z"
                },
                "status": "active",
                "name": "BETH ISRAEL DEACONESS MEDICAL CENTER",
                "telecom": [
                    {
                        "system": "phone",
                        "value": "6176677000"
                    }
                ],
                "address": {
                    "line": [
                        "330 BROOKLINE AVENUE"
                    ],
                    "city": "BOSTON",
                    "state": "MA",
                    "postalCode": "02215",
                    "country": "US"
                },
                "position": {
                    "longitude": -71.020173,
                    "latitude": 42.33196
                },
                "managingOrganization": {
```

```
                    "reference": "Organization/cb6a50e0-
af76-3758-99ad-3200ede03fff",
                    "display": "BETH ISRAEL DEACONESS MEDICAL CENTER"
                }
            },
            "search": {
                "mode": "match"
            }
        }
    ]
}
```

Observation

使用以下命令对Observation资源类型发出GET基于搜索的请求。此搜索使用value-concept搜索参数来查找医疗代码266919005。此状态表明Never smoker。

您必须指定请求URL和查询字符串。这是一个请求示例URL。

```
POST https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Observation?value-concept=266919005
```

要指定状态Never smoker，请value-concept=266919005将其设置为查询字符串。

JSON 响应

成功后，您将获得200HTTP响应码。为清楚起见，以下JSON响应已被截断。

```
{
 "resourceType": "Bundle",
 "type": "searchset",
 "link": [{
  "relation": "next",
  "url": "https://healthlake.us-west-2.amazonaws.com/
datastore/3651c6d3c1e81e785adba06b710b52a9/r4/Observation?value-
concept=266919005&=AAMA-
EFRSURBSGlpcGIyN250ZG9WRXVnTTFOdmtxQk9Bb3Y0YjhVcVdUMGV0eVozNmdjQU9nRjRNUUtscjhCZ1NMUG84VGNqN
 }],
 "entry": [{
   "resource": {
    "resourceType": "Observation",
    "id": "000038e0-71c6-4cc0-9c6c-50c8b1c53309",
    "meta": {
```

```
     "lastUpdated": "2022-11-03T01:02:38.981Z"
    },
    "status": "final",
    "category": [{
     "coding": [{
      "system": "http://terminology.hl7.org/CodeSystem/observation-category",
      "code": "survey",
      "display": "survey"
     }]
    }],
    "code": {
     "coding": [{
      "system": "http://loinc.org",
      "code": "72166-2",
      "display": "Tobacco smoking status NHIS"
     }],
     "text": "Tobacco smoking status NHIS"
    },
    "subject": {
     "reference": "Patient/598c9d7a-0494-448e-a81e-d50e3606e8db"
    },
    "encounter": {
     "reference": "Encounter/86bdee4a-2aa9-474a-b43f-6237cd68e512"
    },
    "effectiveDateTime": "2019-12-11T19:44:57-08:00",
    "issued": "2019-12-11T19:44:57.438-08:00",
    "valueCodeableConcept": {
     "coding": [{
      "system": "http://snomed.info/sct",
      "code": "266919005",
      "display": "Never smoker"
     }],
     "text": "Never smoker"
    }
   },
   "search": {
    "mode": "match"
   }
  },

  {
   "resource": {
    "resourceType": "Observation",
    "id": "0c2f6260-e671-4cfd-ac3d-e75f073fa3cd",
```

```
    "meta": {
     "lastUpdated": "2022-11-03T01:05:21.488Z"
    },
    "status": "final",
    "category": [{
     "coding": [{
      "system": "http://terminology.hl7.org/CodeSystem/observation-category",
      "code": "survey",
      "display": "survey"
     }]
    }],
    "code": {
     "coding": [{
      "system": "http://loinc.org",
      "code": "72166-2",
      "display": "Tobacco smoking status NHIS"
     }],
     "text": "Tobacco smoking status NHIS"
    },
    "subject": {
     "reference": "Patient/89d9a9b7-9720-4881-a2ab-d7907544b26f"
    },
    "encounter": {
     "reference": "Encounter/8ebba7b0-fdfc-4ec1-a9aa-907cccf60925"
    },
    "effectiveDateTime": "2018-11-17T03:59:36-08:00",
    "issued": "2018-11-17T03:59:36.550-08:00",
    "valueCodeableConcept": {
     "coding": [{
      "system": "http://snomed.info/sct",
      "code": "266919005",
      "display": "Never smoker"
     }],
     "text": "Never smoker"
    }
   },
   "search": {
    "mode": "match"
   }
  }
 ]
}
```

阅读FHIR资源历史记录

交FHIRhistory互会检索 HealthLake 数据存储中特定FHIR资源的历史记录。使用此交互，您可以确定FHIR资源内容如何随时间变化。与审计日志配合使用时，查看修改前后的资源状态也很有用。

> **Note**
>
> FHIR在 202 history 4 年 10 月 25 日之后创建的所有 HealthLake 数据存储中，默认启用资源。如果您的数据存储是在此日期之前创建的，则可以提交支持请求以启用FHIRhistory交互。使用创建案例AWS Support Center Console。要创建您的案例，请登录您的 AWS 账户 并选择创建案例。

交history互是使用HTTP GET命令执行的。交FHIR互createupdate、和delete会生成要保存的资源的历史版本。 HealthLake 支持交FHIRhistory互的以下搜索参数。

HealthLake 支持的FHIR**history**互动搜索参数

搜索参数	描述
_count : integer	一页上搜索结果的最大数量。服务器将返回请求的数量或数据存储默认允许的最大搜索结果数，以较低者为准。
_since : instant	仅包括在给定时刻或之后创建的资源版本。
_at : date(Time)	仅包括在日期时间值中指定的时间段内某个时刻处于最新状态的资源版本。有关更多信息，请参阅HL7FHIRRESTfulAPI文档*date*中的。

以下示例为中的FHIRPatient资源每页返回 100 个历史搜索结果 HealthLake。要查看整个URL路径，请滚动到 “复制” 按钮。其形式URL是：

```
GET https://healthlake.region.amazonaws.com/datastore/datastore-id/r4/Patient/id/
_history?_count=100
```

历史交互的返回内容包含在FHIR资源中 Bundle，类型设置为history。它包含指定的版本历史记录，按最旧版本排序，并包含已删除的资源。有关history交互的更多信息，请参阅HL7FHIRRESTfulAPI文档history中的。

> **Note**
>
> 您可以选择不使用特history定的FHIR资源类型。要选择退出，请使用创建案例AWS Support Center Console。要创建您的案例，请登录您的 AWS 账户 并选择创建案例。

阅读特定版本的资源历史记录 FHIR

该FHIRvread交互对 HealthLake 数据存储中的资源执行特定版本的读取。使用此交互，您可以像过去特定时间一样查看FHIR资源的内容。

HealthLake 声明它支持对每个支持的资源CapabilityStatement.rest.resource.versioning进行版本控制。所有资源上的所有 HealthLake 数据存储都包含 Resource.meta.versionId (vid)。

启用FHIRhistory交互后（默认情况下，对于在 2024 年 10 月 25 日之后创建的数据存储，或者通过请求创建较旧的数据存储），Bundle响应会将vid作为其中的一部分。location在以下示例中，vid显示为数字1。要查看完整示例，请参阅 bundle /bundle-response 示例 ()。JSON

```
"response" : {
    "status" : "201 Created",
    "location" : "Patient/12423/_history/1",
    ...}
```

交vread互是使用HTTP GET命令执行的。以下vread交互返回一个实例，其中包含为FHIRPatient资源指定的内容的资源元数据版本vid。要查看以下示例中的整个URL路径，请在“复制”按钮上滚动。其形式URL是：

```
GET https://healthlake.region.amazonaws.com/datastore/datastore-id/r4/Patient/id/
_history/vid
```

> **Note**
>
> 如果您在读取FHIR资源vread时不使用history交互，则 HealthLake 始终返回资源元数据的最新版本。

有关vread交互的更多信息，请参阅 HL7FHIRRestful API 文档*vread*中的。

通过 Patient $ FHIR REST API everything 操作获取患者数据

Patient $everything 操作用于查询FHIR患者资源以及与该患者相关的任何其他资源。此操作可用于为患者提供访问其全部记录的权限，也可以让提供者执行与患者相关的批量数据下载。 HealthLake 支持特定患者 ID 的所有内容。

> **Note**
>
> 2024 年 2 月 27 日之后创建的数据存储目前支持 Patient $everything 操作。

获取与患者相关的所有资源

patient $everything 是一项可以调用的RESTAPI操作，如以下示例所示。

GET Request

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient/patient-id/$everything
```

> **Note**
>
> 响应中的资源按资源类型和资源 ID 排序。
> 响应总是用 bundle.Total 填充。

病人 $所有参数

HealthLake 支持以下查询参数

参数	详细信息
开启	获取指定开始日期之后的所有患者数据。
end	在指定结束日期之前获取所有患者数据。

参数	详细信息
since	在指定日期之后更新所有患者数据。
_type	获取特定资源类型的患者数据。
_count	获取患者数据并指定页面大小。

Example -获取指定开始日期之后的所有患者数据

patient $every `start` thing 只能使用过滤器来查询特定日期之后的数据。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient/patient-id/$everything?start=2024-03-15T00:00:00.000Z
```

Example -在指定结束日期之前获取所有患者数据

patient $every end thing 只能使用过滤器来查询特定日期之前的数据。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient/patient-id/$everything?end=2024-03-15T00:00:00.000Z
```

Example -在指定日期之后更新所有患者数据

patient $everything 可以使用`since`过滤器仅查询在特定日期之后更新的数据。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient/patient-id/$everything?since=2024-03-15T00:00:00.000Z
```

Example -获取特定资源类型的患者数据

patient $everything 可以使用`_type`过滤器来指定要包含在响应中的特定资源类型。可以在逗号分隔的列表中指定多种资源类型。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient/patient-id/$everything?_type=Observation,Condition
```

Example -获取患者数据并指定页面大小

病人 $everything 都可以使用_count来设置页面大小。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient/patient-id/$everything?_count=15
```

病人 $所有内容start和属性 end

HealthLake 支持以下资源属性作为开始和结束查询参数。

资源	资源元素
帐户	账户。 servicePeriod.start
AdverseEvent	AdverseEvent. 日期
AllergyIntolerance	AllergyIntolerance.recordedDate
预约	预约. 开始
AppointmentResponse	AppointmentResponse.start
AuditEvent	AuditEvent.period.start
基本	basic.created
BodyStructure	不_ DATE
CarePlan	CarePlan.period.start
CareTeam	CareTeam.period.start
ChargeItem	ChargeItem。 occurrenceDateTime ， ChargeItem。 occurrencePeriod.start ， ChargeItem。 occurrenceTiming. 事件

资源	资源元素
Claim	索赔。 billablePeriod.start
ClaimResponse	ClaimResponse。已创建
ClinicalImpression	ClinicalImpression. 日期
Communication	通信. 已发送
CommunicationRequest	CommunicationRequest。 occurrenceDateTime ， CommunicationRequest。 occurrencePeriod.start
合成	作文. 日期
状况	状况。 recordedDate
同意	同意。 dateTime
覆盖范围	Coverage.period.Start
CoverageEligibilityRequest	CoverageEligibilityRequest。已创建
CoverageEligibilityResponse	CoverageEligibilityResponse。已创建
DetectedIssue	DetectedIssue. 已识别
DeviceRequest	DeviceRequest.authoredOn

资源	资源元素
DeviceUseStatement	DeviceUseStatement.recordedOn
DiagnosticReport	DiagnosticReport。有效
DocumentManifest	DocumentManifest。已创建
DocumentReference	DocumentReference.context.period.start
遭遇	Encounter.period.Start
EnrollmentRequest	EnrollmentRequest。已创建
EpisodeOfCare	EpisodeOfCare.period.start
ExplanationOfBenefit	ExplanationOfBenefit。 billablePeriod.start
FamilyMemberHistory	不_ DATE
标记	flag.period.start
目标	目标。 statusDate
组	不_ DATE
ImagingStudy	ImagingStudy。已开始
免疫接种	免疫接种。已记录

资源	资源元素
ImmunizationEvaluation	ImmunizationEvaluation. 日期
ImmunizationRecommendation	ImmunizationRecommendation. 日期
发票	发票日期
列出	List.date
MeasureReport	MeasureReport.period.start
媒体	Media. 已发布
MedicationAdministration	MedicationAdministration。有效
MedicationDispense	MedicationDispense.whenPrepared
MedicationRequest	MedicationRequest.authoredOn
MedicationStatement	MedicationStatement.dateAsserted
MolecularSequence	不_ DATE
NutritionOrder	NutritionOrder.dateTime
观察	观察。有效

资源	资源元素
病人	不_ DATE
人员	不_ DATE
过程	程序. 已执行
出处	出处。 occurredPeriod.start，Provenance。 occurredDateTime
QuestionnaireResponse	QuestionnaireResponse. 创作
RelatedPerson	不_ DATE
RequestGroup	RequestGroup.authoredOn
ResearchSubject	ResearchSubject. 句点
RiskAssessment	RiskAssessment。 occurrenceDateTime， RiskAssessment。 occurrencePeriod.start
计划	日程安排。 planningHorizon
ServiceRequest	ServiceRequest.authoredOn
标本	标本。 receivedTime
SupplyDelivery	SupplyDelivery。 occurrenceDateTime， SupplyDelivery。 occurrencePeriod.start， SupplyDelivery。 occurrenceTiming. 事件
SupplyRequest	SupplyRequest.authoredOn

资源	资源元素
VisionPrescription	VisionPrescription.dateWritten

使用 $export 从您的 HealthLake 数据存储中导出数据

使用FHIRRESTAPI指定`$export`作为请求的一部分提出导出POST请求，并在请求正文中包含请求参数。根据FHIR规范，FHIR服务器必须支持GET请求，并且可以支持POST请求。为了支持其他参数，需要一个主体来开始导出，因此 HealthLake 支持POST请求。

> **Important**
>
> HealthLake 2023 年 6 月 1 日之前创建的数据存储仅支持系统范围导出的FHIRRESTAPI基于导出任务的请求。
> HealthLake 2023 年 6 月 1 日之前创建的数据存储不支持使用数据存储端点上的GET请求来获取导出状态。

您使用提出的所有导出请求FHIRRESTAPI都将以`ndjson`格式返回并导出到 Amazon S3 存储桶。每个 S3 对象将仅包含一种FHIR资源类型。

您一次可以为每个 AWS 账户提出一个导出请求。要详细了解与之关联的 Service Quota HealthLake s，请参阅AWS HealthLake 终端节点和配额。

要了解有关使用提出导出请求的更多信息 FHIR RESTAPI，请参阅使用FHIRRESTAPI操作从 HealthLake 数据存储中导出数据。

在 Amazon Athena SQL 中使用查询 AWS HealthLake 数据存储

创建 HealthLake 数据存储时，高度嵌套FHIR的数据结构会被提取到 Amazon Athena 中，并自动转换为可使用查询的 Iceberg 表。SQL使用La AWS ke Formation对授予对这一新资源的访问权限进行管理。在 Athena 中，每种FHIR资源类型都表示为一个单独的表。

> **Important**
>
> 对于在 2022 年 11 月 14 日之前创建的数据存储，您必须将现有数据存储迁移到新的数据存储才能使用它进行查询SQL。有关帮助信息，请参阅 迁移现有数据存储以使用 Amazon Athena。

> **Note**
>
> 2023 年 2 月 20 日之后，默认情况下， HealthLake 数据存储不使用集成的自然语言处理 (NLP)。如果您有兴趣在数据存储上启用此功能，请参阅 “故障排除” 一章如何开启 HealthLake 集成的自然语言处理功能？中的。

要创建 HealthLake 数据存储，必须向 HealthLake 管理员IAM用户或角色添加其他IAM策略和服务角色。有关设置权限的更多信息，请参阅设置开始使用的权限 AWS HealthLake。

HealthLake 数据存储以 Iceberg 表的形式存储到 Athena 中。要详细了解 Iceberg 表在 Athena 中的工作原理，请参阅《Athena 用户指南》中的 “使用冰山表”。

HealthLake 支持在 Athena 中READ操作您的 HealthLake 数据存储数据存储。要详细了解如何使用这些操作创建、读取、更新和删除 (CRUD) FHIR REST API 操作，请参阅使用与 HealthLake 数据存储的FHIRRESTAPI交互，详细了解CRUD操作如何影响 Athena 中的数据。

本章中的主题介绍如何将您的 HealthLake 数据存储连接到 Athena，如何SQL使用它进行查询，以及如何将结果 AWS 与其他服务连接起来以供进一步分析。

目录

  - Athena 入门
- 使用查询您的 HealthLake 数据存储 SQL
- 具有复杂筛选功能的SQL查询示例

将您的数据存储连接到亚马逊 Athena

> **Important**
>
> 2022 年 11 月 14 日之后，访问IAM要求 HealthLake 发生了变化。要在 Athena 中创建数据存储并授予对它们的访问权限，您必须`AWSLakeFormationDataAdmin`将托管策略添加到IAM您的用户、群组或角色中。您可以使用该`AWSLakeFormationDataAdmin`策略创建数据湖管理员并授予对 Athena 中数据存储的访问权限。

本主题概述了创建 Athena 用户、群组或角色并授予他们访问FHIR数据存储中 HealthLake 资源的权限的必要步骤。

- 向用户、群组或角色授予对 HealthLake 数据存储的访问权限（AWS Lake Formation 控制台）
- 设置 Athena 账号

向用户、群组或角色授予对 HealthLake 数据存储的访问权限（AWS Lake Formation 控制台）

> **角色： HealthLake 管理员**
>
> HealthLake 管理员角色是 Lake Formation 中的数据 AWS 湖管理员。他们授予对 Lake Formation 中 HealthLake 数据存储的访问权限。

对于创建的每个数据存储，La AWS ke Formation 控制台中都有两个条目可见。一个条目是资源链接。资源链接名称始终以斜体显示。每个资源链接都显示其链接的共享资源的名称和所有者。对于所有 HealthLake 数据存储，共享资源所有者是 HealthLake 服务帐号。另一个条目是 HealthLake 服务帐户中的 HealthLake 数据存储。此过程中的步骤使用作为资源链接的数据存储。

要了解有关资源链接的更多信息，请参阅 Lake For mation 开发者指南中的资源链接在 Lak AWS e Formation 中的工作原理。

要使用户、群组或角色能够在 Athena 中查询数据，必须授予对资源数据库的 “描述” 权限。然后，您必须在表格上授予选择和描述权限。

STEP1：授予对 HealthLake 数据存储资源链接数据库的DESCRIBE权限

1. 打开 AWS Lake Formation 控制台：https://console.aws.amazon.com/lakeformation/
2. 在主导航栏中，选择数据库。
3. 在数据库页面上，选择斜体数据存储名称旁边的单选按钮。
4. 选择操作 (▾)。
5. 选择授权。
6. 在授予数据权限页面的委托人下，选择IAM用户或角色。
7. 在 “IAM用户或角色” 下，使用向下箭头 (▾)，或者在 Athena 中搜索您希望能够对其进行查询的IAM用户、角色或群组。
8. 在 LF-Tags 或目录资源卡下，选择命名数据目录资源选项。
9. 在数据库下，使用向下箭头 (▾) 选择要共享访问权限 HealthLake 的数据存储数据库。
10. 在资源链接权限卡的资源链接权限下，选择描述。

成功授予后，将显示授予权限成功横幅。要查看您刚刚授予的权限，请选择数据湖权限。在表格中找到用户、组和角色。在 “权限” 列下，您将看到列出的 “描述”。

现在，您必须使用 Grant on target 对数据库中的所有表授予选择和描述权限。

STEP2：授予对 HealthLake 数据存储资源链接中所有表的访问权限

1. 打开 AWS Lake Formation 控制台：https://console.aws.amazon.com/lakeformation/
2. 在主导航栏中，选择数据库。
3. 在数据库页面上，选择斜体数据存储名称旁边的单选按钮。
4. 选择操作 (▾)。
5. 选择向目标授予。
6. 在授予数据权限页面的委托人下，选择IAM用户或角色。
7. 在IAM用户或角色下，使用向下箭头 (▾) 或搜索您希望能够在 Athena 中进行查询的IAM用户、群组或角色。
8. 在 LF-Tags 或目录资源卡下，选择命名数据目录资源选项。
9. 在数据库下，使用向下箭头 (▾) 选择要授予访问权限 HealthLake 的数据存储数据库。

10. 在 “表” 下，选择 “所有表” 以与 HealthLake 用户共享所有表。
11. 在 “表权限” 卡的 “表格权限” 下，选择 “描述并选择”。
12. 选择授权。

选择授予后，将出现 “授予权限” 成功横幅。现在，指定的用户可以在 Athena 中的 HealthLake 数据存储上进行查询。

Athena 入门

> **HealthLake 用户**
>
> HealthLake 用户将使用 Athena 控制台 AWS CLI、 AWS SDKs或来查询 HealthLake 管理员与其共享的数据存储。 HealthLake

要使用 Athena 查询数据存储，必须执行以下三项操作。

- 通过 Lake Formation 向IAM用户或角色授予对 HealthLake 数据存储的访问权限。要了解更多信息，请参阅 向用户、群组或角色授予对 HealthLake 数据存储的访问权限（AWS Lake Formation 控制台）。
- 为您的 HealthLake 数据存储创建一个工作组。
- 指定一个 Amazon S3 存储桶来存储您的查询结果。

要开始使用 Athena，请将和 A FullAccess AWS mazonS3 托管策略添加到AmazonAthenaFullAccess您的用户、群组或角色中。使用 AWS 托管策略是开始使用新服务的好方法。请记住，AWS 托管策略可能不会为您的特定使用场景授予最低权限许可，因为它们可供所有 AWS 客户使用。使用IAM策略设置权限时，仅授予执行任务所需的权限。要了解有关IAM和应用最低权限的更多信息，请参阅《用户指南》中的应用最低权限权限。IAM

> **Important**
>
> 要查询 Athena 中的 HealthLake 数据存储，必须使用 A thena 引擎版本 3。

工作组是资源，因此您可以使用IAM基于策略来控制对特定工作组的访问权限。要了解更多信息，请参阅《Athena 用户指南》中的使用工作组控制查询访问权限和费用。

要了解有关设置工作组的更多信息，请参阅https://docs.aws.amazon.com/athena/latest/ug/workgroups-procedure.html《Athena 用户指南》。

> **Note**
>
> 您的 Amazon S3 存储桶所在的区域和 Athena 控制台必须匹配。

在运行查询之前，必须指定 Amazon S3 中的查询结果存储桶位置，或者您必须使用已指定存储桶且其配置覆盖客户端设置的工作组。对于运行的每个查询，将自动保存输出文件。

有关在 Athena 控制台中指定查询结果位置的更多详情，请参阅 Amazon Athena 用户指南中的使用 Athena 控制台指定查询结果位置。

要查看如何在 Athena 中查询 HealthLake 数据存储的示例，请参阅。使用查询您的 HealthLake 数据存储 SQL

使用查询您的 HealthLake 数据存储 SQL

> **Note**
>
> 2023 年 2 月 20 日之后，默认情况下， HealthLake 数据存储不使用集成的自然语言处理 (NLP)。如果您有兴趣在数据存储上启用此功能，请参阅 “故障排除” 一章如何开启 HealthLake 集成的自然语言处理功能？中的。
> 本主题中的所有示例都使用使用 Synthea 创建的虚构数据。要了解有关创建预加载了 Synthea 数据的数据存储的更多信息，请参阅。在中创建数据存储 AWS HealthLake

当您将 HealthLake 数据存储导入 Athena 时，数据存储中的每种资源类型都将转换为表。 HealthLake 这些表可以单独查询，也可以使用SQL基于基础的查询成组查询。由于数据存储结构的原因，您的数据将作为多种不同的数据类型导入 Athena。要详细了解如何创建可以访问这些数据类型的SQL查询，请参阅 Amazon Athen a 用户指南中的查询具有复杂类型和嵌套结构的数组。

对于资源类型中的每个元素，FHIR规范都定义了一个基数。元素的基数定义了该元素可以出现的次数的下限和上限。在构造SQL查询时，必须考虑到这一点。例如，让我们看看 “资源类型：患者” 中的一些元素。

- 元素：名称FHIR规范将基数设置为。`0..*`

该元素被捕获为数组。

```
[{
 id = null,
 extension = null,
 use = official,
 _use = null,
 text = null,
 _text = null,
 family = Wolf938,
 _family = null,
 given = [Noel608],
 _given = null,
 prefix = null,
 _prefix = null,
 suffix = null,
 _suffix = null,
 period = null
}]
```

在 Athena 中，要查看资源类型是如何被摄取的，请在表格和视图下搜索该资源类型。要访问此数组中的元素，可以使用点表示法。这是一个访问given和值的简单示例family。

```
SELECT
    name[1].given as FirstName,
    name[1].family as LastName
FROM Patient
```

- 元素：FHIR规范 MaritalStatus将基数设置为。0..1

  此元素被捕获为JSON。

```
{
 id = null,
 extension = null,
 coding = [
  {
   id = null,
   extension = null,
   system = http: //terminology.hl7.org/CodeSystem/v3-MaritalStatus,
    _system = null,
```

```
    version = null,
    _version = null,
    code = S,
    _code = null,
    display = Never Married,
    _display = null,
    userSelected = null,
    _userSelected = null
   }

 ],
 text = Never Married,
 _text = null
}
```

在 Athena 中，要查看资源类型是如何被摄取的，请在表格和视图下搜索该资源类型。要访问中的键值对JSON，可以使用点表示法。因为它不是数组，所以不需要数组索引。以下是一个可以访问值的简单示例`text`。

```
SELECT
    maritalstatus.text as MaritalStatus
FROM Patient
```

要了解有关访问和搜索的更多信息JSON，请参阅《Athena 用户指南》JSON中的查询。

Athena 数据操纵语言 DML () 查询语句基于 Trino。Athena并不支持Trino的所有功能，并且存在显著差异。要了解更多信息，请参阅 Amazon Athen a 用户指南中的DML查询、函数和运算符。

此外，Athena 支持您在创建数据存储查询时可能会遇到的 HealthLake 多种数据类型。要了解有关 Athena 中数据类型的更多信息，请参阅亚马逊 Athena 用户指南中的亚马逊 Athena 中的数据类型。

要详细了解在 Athena 中如何SQL进行查询，SQL请参阅亚马逊 Athena 用户指南中有关亚马逊 Athena 的参考资料。

每个选项卡都显示了如何使用 Athena 搜索指定资源类型和关联元素的示例。

Element: Extension

该元素`extension`用于在数据存储中创建自定义字段。

此示例向您展示如何访问`Patient`资源类型中`extension`元素的功能。

将 HealthLake 数据存储导入 Athena 时，对资源类型的元素的解析会有所不同。由于 element is 的结构变量，因此无法在架构中对其进行完全指定。为了处理这种可变性，数组中的元素作为字符串传递。

在的表描述中Patient，您可以看到extension描述为的元素array<string>，这意味着您可以使用索引值访问数组的元素。但是，要访问字符串的元素，必须使用json_extract。

以下是患者表中extension元素的单个条目。

```
[{
  "valueString": "Kerry175 Cummerata161",
  "url": "http://hl7.org/fhir/StructureDefinition/patient-mothersMaidenName"
 },
 {
  "valueAddress": {
   "country": "DE",
   "city": "Hamburg",
   "state": "Hamburg"
  },
  "url": "http://hl7.org/fhir/StructureDefinition/patient-birthPlace"
 },
 {
  "valueDecimal": 0.0,
  "url": "http://synthetichealth.github.io/synthea/disability-adjusted-life-years"
 },
 {
  "valueDecimal": 5.0,
  "url": "http://synthetichealth.github.io/synthea/quality-adjusted-life-years"
 }
]
```

尽管这是有效的JSON，但 Athena 仍将其视为字符串。

此SQL查询示例演示如何创建包含patient-mothersMaidenName和patient-birthPlace元素的表。要访问这些元素，你需要使用不同的数组索引和 json_extract.

```
SELECT
    extension[1],
    json_extract(extension[1], '$.valueString') AS MothersMaidenName,
    extension[2],
    json_extract(extension[2], '$.valueAddress.city') AS birthPlace
FROM patient
```

要详细了解涉及的查询JSON，请参阅 A mazon Athena 用户指南JSON中的从中提取数据。

Element: birthDate (Age)

年龄不是患者资源类型的要素FHIR。以下是根据年龄进行筛选的两个搜索示例。

因为年龄不是一个元素，所以我们使用birthDate进行SQL查询。要查看元素是如何被引入的FHIR，请在 “表和视图” 下搜索表名。你可以看到它的类型是字符串。

示例 1：计算年龄值

在此示例SQL查询中，我们使用内置SQL工具，current_dateyear并提取这些组件。然后，我们减去它们以返回患者的实际年龄，列名为age。

```
SELECT
 (year(current_date) - year(date(birthdate))) as age
FROM patient
```

示例 2：筛选出生之前2019-01-01和现在出生的患者male。

该SQL查询向您展示了如何使用CAST函数将birthDate元素转换为类型DATE，以及如何根据WHERE子句中的两个条件进行筛选。由于默认情况下，该元素是作为字符串类型提取的，CAST因此我们必须将其作为类型DATE。然后，您可以使用<运算符将其与其他日期进行比较,2019-01-01. 通过使用AND，您可以向子WHERE句添加第二个条件。

```
SELECT birthdate
FROM patient
-- we convert birthdate (varchar) to date  > cast that as date too
WHERE CAST(birthdate AS DATE) < CAST('2019-01-01' AS DATE) AND gender = 'male'
```

Resource type: Location

此示例显示了在 “位置” 资源类型中搜索城市名称为 Attleboro 的地点。

```
SELECT *
FROM Location
WHERE address.city='ATTLEBORO'
LIMIT 10;
```

Element: Age

```
SELECT birthdate
FROM patient
```

```
-- we convert birthdate (varchar) to date  > cast that as date too
WHERE CAST(birthdate AS DATE) < CAST('2019-01-01' AS DATE) AND gender = 'male'
```

Resource type: Condition

资源类型条件存储与已上升到令人担忧程度的问题相关的诊断数据。 HealthLake的集成医学自然语言处理 (NLP) 会根据Condition资源类型中的详细信息生成新 DocumentReference 资源。生成新资源时， HealthLake 将标签附加SYSTEM_GENERATED到meta元素。此示例SQL查询演示了如何搜索条件表并返回已删除SYSTEM_GENERATED结果的结果。

要了解有关集成自然语言处理 (NLP) HealthLake 的更多信息，请参阅使用基于资源类型的自然语言处理 (NLP) 的自动FHIR DocumentReference 资源生成 AWS HealthLake。

```
SELECT *
FROM condition
WHERE meta.tag[1] is NULL
```

您也可以在指定的字符串元素中进行搜索以进一步筛选查询。该modifierextension元素包含有关使用哪个DocumentReference资源生成一组条件的详细信息。同样，您必须使用json_extract来访问作为字符串引入 Athena 的嵌套JSON元素。

此示例SQL查询演示了如何搜索根据特定内容生成的所有内容DocumentReference。Condition用于CAST将JSON元素设置为字符串，以便可以LIKE用来比较。

```
SELECT
    meta.tag[1].display as SystemGenerated,
    json_extract(modifierextension[4], '$.valueReference.reference') as
 DocumentReference
FROM condition
WHERE meta.tag[1].display = 'SYSTEM_GENERATED'

AND CAST(json_extract(modifierextension[4], '$.valueReference.reference') as
 VARCHAR) LIKE '%DocumentReference/67aa0278-8111-40d0-8adc-43055eb9d18d%'
```

Resource type: Observation

资源类型 “观察” 存储有关患者、设备或其他受试者的测量结果和简单断言。 HealthLake的集成自然语言处理 (NLP) 会根据Observation资源中的详细信息生成新DocumentReference资源。此示例SQL查询包括WHERE meta.tag[1] is NULL注释掉的内容，这意味着包含了SYSTEM_GENERATED结果。

```
SELECT valueCodeableConcept.coding[1].code
FROM Observation
WHERE  valueCodeableConcept.coding[1].code = '266919005'
-- WHERE meta.tag[1] is NULL
```

此列是作为导入的`struct`。因此，您可以使用点表示法访问其中的元素。

Resource type: MedicationStatement

MedicationStatement 是一种FHIR资源类型，可用于存储有关患者已服用、正在服用或将来将要服用的药物的详细信息。 HealthLake的集成医学自然语言处理 (NLP) 会根据 MedicationStatement 资源类型中的文档生成新 DocumentReference 资源。生成新资源时， HealthLake 将标签附加SYSTEM_GENERATED到meta元素。此示例SQL查询演示了如何创建一个查询，该查询使用患者标识符根据单个患者进行筛选，并查找已由 “s integrated” 添加 HealthLake的资源NLP。

```
SELECT *
FROM medicationstatement
WHERE meta.tag[1].display = 'SYSTEM_GENERATED' AND subject.reference =
 'Patient/0679b7b7-937d-488a-b48d-6315b8e7003b';
```

要了解有关综合医疗 HealthLake的更多信息NLP，请参阅使用基于资源类型的自然语言处理 (NLP) 的自动FHIR DocumentReference 资源生成 AWS HealthLake。

具有复杂筛选功能的SQL查询示例

> **Note**
>
> 2023 年 2 月 20 日之后，默认情况下， HealthLake 数据存储不使用集成的自然语言处理 (NLP)。如果您有兴趣在数据存储上启用此功能，请参阅 “故障排除” 一章如何开启 HealthLake 集成的自然语言处理功能？中的。

本主题中的示例包括使用复杂SQL筛选 HealthLake的与 Athena 集成的查询。

Example 创建基于人口统计数据的筛选标准

在创建患者群组时，确定正确的患者人口统计数据非常重要。此示例查询演示了如何使用 Trino 点符号和json_extract筛选数据存储中的 HealthLake 数据。

```
SELECT
```

```
    id
    , CONCAT(name[1].family, ' ', name[1].given[1]) as name
    , (year(current_date) - year(date(birthdate))) as age
    , gender as gender
    , json_extract(extension[1], '$.valueString') as MothersMaidenName
    , json_extract(extension[2], '$.valueAddress.city') as birthPlace
    , maritalstatus.coding[1].display as maritalstatus
    , address[1].line[1] as addressline
    , address[1].city as city
    , address[1].district as district
    , address[1].state as state
    , address[1].postalcode as postalcode
    , address[1].country as country
    , json_extract(address[1].extension[1], '$.extension[0].valueDecimal') as latitude
    , json_extract(address[1].extension[1], '$.extension[1].valueDecimal') as longitude
    , telecom[1].value as telNumber
    , deceasedboolean as deceasedIndicator
    , deceaseddatetime
FROM database.patient;
```

使用 Athena 控制台，您可以进一步排序和下载结果。

Example 为患者及其相关病症创建过滤器

此示例查询演示了如何查找和排序在 HealthLake 数据存储中发现的患者的所有相关病症。

```
SELECT
 patient.id as patientId
    , condition.id  as conditionId
    , CONCAT(name[1].family, ' ', name[1].given[1]) as name
    , condition.meta.tag[1].display
    , json_extract(condition.modifierextension[1], '$.valueDecimal') AS confidenceScore
    , category[1].coding[1].code as categoryCode
    , category[1].coding[1].display as categoryDescription
    , code.coding[1].code as diagnosisCode
    , code.coding[1].display as diagnosisDescription
    , onsetdatetime
    , severity.coding[1].code as severityCode
    , severity.coding[1].display as severityDescription
    , verificationstatus.coding[1].display as verificationStatus
    , clinicalstatus.coding[1].display as clinicalStatus
    , encounter.reference as encounterId
    , encounter.type as encountertype
FROM database.patient, condition
```

```
WHERE CONCAT('Patient/', patient.id) = condition.subject.reference
ORDER BY name;
```

您可以使用 Athena 控制台对这些结果进行进一步排序或下载以进行进一步分析。

Example 为患者及其相关观察结果创建过滤器

此示例查询演示了如何对 HealthLake 数据存储中发现的患者的所有相关观察结果进行查找和排序。

```
SELECT
 patient.id as patientId
    , observation.id as observationId
    , CONCAT(name[1].family, ' ', name[1].given[1]) as name
    , meta.tag[1].display
    , json_extract(modifierextension[1], '$.valueDecimal') AS confidenceScore
    , status
    , category[1].coding[1].code as categoryCode
    , category[1].coding[1].display as categoryDescription
    , code.coding[1].code as observationCode
    , code.coding[1].display as observationDescription
    , effectivedatetime
    , CASE
  WHEN valuequantity.value IS NOT NULL THEN CONCAT(CAST(valuequantity.value AS
 VARCHAR),' ',valuequantity.unit)
       WHEN valueCodeableConcept.coding [ 1 ].code IS NOT NULL THEN
 CAST(valueCodeableConcept.coding [ 1 ].code AS VARCHAR)
       WHEN valuestring IS NOT NULL THEN CAST(valuestring AS VARCHAR)
       WHEN valueboolean IS NOT NULL THEN CAST(valueboolean AS VARCHAR)
       WHEN valueinteger IS NOT NULL THEN CAST(valueinteger AS VARCHAR)
       WHEN valueratio IS NOT NULL THEN CONCAT(CAST(valueratio.numerator.value AS
 VARCHAR),'/',CAST(valueratio.denominator.value AS VARCHAR))
       WHEN valuerange IS NOT NULL THEN CONCAT(CAST(valuerange.low.value AS
 VARCHAR),'-',CAST(valuerange.high.value AS VARCHAR))
       WHEN valueSampledData IS NOT NULL THEN CAST(valueSampledData.data AS VARCHAR)
       WHEN valueTime IS NOT NULL THEN CAST(valueTime AS VARCHAR)
       WHEN valueDateTime IS NOT NULL THEN CAST(valueDateTime AS VARCHAR)
       WHEN valuePeriod IS NOT NULL THEN valuePeriod.start
       WHEN component[1] IS NOT NULL THEN CONCAT(CAST(component[2].valuequantity.value
 AS VARCHAR),' ',CAST(component[2].valuequantity.unit AS VARCHAR),
 '/', CAST(component[1].valuequantity.value AS VARCHAR),'
 ',CAST(component[1].valuequantity.unit AS VARCHAR))
    END AS observationvalue
 , encounter.reference as encounterId
    , encounter.type as encountertype
```

```
FROM database.patient, observation
WHERE CONCAT('Patient/', patient.id) = observation.subject.reference
ORDER BY name;
```

Example 为患者及其相关程序创建筛选条件

将手术与患者联系起来是医疗保健的一个重要方面。此SQL查询演示了如何在 Athena 中使用患者和手术资源类型来实现此目的。此SQL查询将返回在您的 HealthLake 数据存储中找到的所有患者及其相关手术。

```
SELECT
 patient.id  as patientId
 , PROCEDURE.id as procedureId
 , CONCAT(name[1].family, ' ', name[1].given[1]) as name
 , status
 , category.coding[1].code as categoryCode
 , category.coding[1].display as categoryDescription
 , code.coding[1].code as procedureCode
 , code.coding[1].display as procedureDescription
 , performeddatetime
 , performer[1]
 , encounter.reference as encounterId
 , encounter.type as encountertype
FROM database.patient, procedure
WHERE CONCAT('Patient/', patient.id) = procedure.subject.reference
ORDER BY name;
```

现在，您可以使用 Athena 控制台下载结果以供进一步分析，也可以对其进行排序以更好地了解结果。

Example 为患者及其相关处方创建筛选条件

查看患者正在服用的药物的最新清单很重要。使用 Athena，您可以编写SQL同时使用数据存储中的 HealthLake 患者 MedicationRequest 和资源类型的查询。

此SQL查询将患者和导入到 Athena MedicationRequest 中的表格连接在一起。它还使用点符号将处方组织到各自的条目中。

```
SELECT
 patient.id  as patientId
 , medicationrequest.id  as medicationrequestid
 , CONCAT(name[1].family, ' ', name[1].given[1]) as name
 , status
 , statusreason.coding[1].code as categoryCode
```

```
 , statusreason.coding[1].display as categoryDescription
 , category[1].coding[1].code as categoryCode
 , category[1].coding[1].display as categoryDescription
 , priority
 , donotperform
 , encounter.reference as encounterId
 , encounter.type as encountertype
 , medicationcodeableconcept.coding[1].code as medicationCode
 , medicationcodeableconcept.coding[1].display as medicationDescription
 , dosageinstruction[1].text as dosage
FROM database.patient, medicationrequest
WHERE CONCAT('Patient/', patient.id ) = medicationrequest.subject.reference
ORDER BY name
```

您可以使用 Athena 控制台对结果进行排序或下载以进行进一步分析。

Example 查看 MedicationStatement 资源类型中发现的药物

示例查询向您展示了如何使用整理JSON导入到 Athena 的嵌套内容。SQL该查询使用meta元素来指示何时通过 HealthLake集成的自然语言处理 (NLP) 添加了药物。要了解有关与 Amazon Compre HealthLake hend Medical 集成的更多信息，请参阅。使用基于资源类型的自然语言处理 (NLP) 的自动 FHIR DocumentReference 资源生成 AWS HealthLake它还json_extract用于在JSON字符串数组中搜索数据。

```
SELECT
 medicationcodeableconcept.coding[1].code as medicationCode
 , medicationcodeableconcept.coding[1].display as medicationDescription
 , meta.tag[1].display
 , json_extract(modifierextension[1], '$.valueDecimal') AS confidenceScore
FROM medicationstatement;
```

您可以使用 Athena 控制台下载这些结果或对其进行排序。

Example 筛选特定疾病类型

该示例显示了如何找到一组年龄在18至75岁之间、被诊断出患有糖尿病的患者。

```
SELECT patient.id as patientId,
 condition.id as conditionId,
 CONCAT(name [ 1 ].family, ' ', name [ 1 ].given [ 1 ]) as name,
 (year(current_date) - year(date(birthdate))) AS age,
 CASE
  WHEN condition.encounter.reference IS NOT NULL THEN condition.encounter.reference
```

```
 WHEN observation.encounter.reference IS NOT NULL THEN observation.encounter.reference
END as encounterId,
CASE
 WHEN condition.encounter.type IS NOT NULL THEN observation.encounter.type
 WHEN observation.encounter.type IS NOT NULL THEN observation.encounter.type
END AS encountertype,
condition.code.coding [ 1 ].code as diagnosisCode,
condition.code.coding [ 1 ].display as diagnosisDescription,
observation.category [ 1 ].coding [ 1 ].code as categoryCode,
observation.category [ 1 ].coding [ 1 ].display as categoryDescription,
observation.code.coding [ 1 ].code as observationCode,
observation.code.coding [ 1 ].display as observationDescription,
effectivedatetime AS observationDateTime,
CASE
     WHEN valuequantity.value IS NOT NULL THEN CONCAT(CAST(valuequantity.value AS
VARCHAR),' ',valuequantity.unit)
     WHEN valueCodeableConcept.coding [ 1 ].code IS NOT NULL THEN
CAST(valueCodeableConcept.coding [ 1 ].code AS VARCHAR)
     WHEN valuestring IS NOT NULL THEN CAST(valuestring AS VARCHAR)
     WHEN valueboolean IS NOT NULL THEN CAST(valueboolean AS VARCHAR)
     WHEN valueinteger IS NOT NULL THEN CAST(valueinteger AS VARCHAR)
     WHEN valueratio IS NOT NULL THEN CONCAT(CAST(valueratio.numerator.value AS
VARCHAR),'/',CAST(valueratio.denominator.value AS VARCHAR))
     WHEN valuerange IS NOT NULL THEN CONCAT(CAST(valuerange.low.value AS
VARCHAR),'-',CAST(valuerange.high.value AS VARCHAR))
     WHEN valueSampledData IS NOT NULL THEN CAST(valueSampledData.data AS VARCHAR)
     WHEN valueTime IS NOT NULL THEN CAST(valueTime AS VARCHAR)
     WHEN valueDateTime IS NOT NULL THEN CAST(valueDateTime AS VARCHAR)
     WHEN valuePeriod IS NOT NULL THEN valuePeriod.start
     WHEN component[1] IS NOT NULL THEN CONCAT(CAST(component[2].valuequantity.value
AS VARCHAR),' ',CAST(component[2].valuequantity.unit AS VARCHAR),
'/', CAST(component[1].valuequantity.value AS VARCHAR),'
',CAST(component[1].valuequantity.unit AS VARCHAR))
   END AS observationvalue,
CASE
 WHEN condition.meta.tag [ 1 ].display = 'SYSTEM GENERATED' THEN 'YES'
 WHEN condition.meta.tag [ 1 ].display IS NULL THEN 'NO'
 WHEN observation.meta.tag [ 1 ].display = 'SYSTEM GENERATED' THEN 'YES'
 WHEN observation.meta.tag [ 1 ].display IS NULL THEN 'NO'
  END AS IsSystemGenerated,
 CAST(
   json_extract(
     condition.modifierextension [ 1 ],
     '$.valueDecimal'
```

```
    ) AS int
  ) AS confidenceScore
FROM database.patient,
 database.condition,
 database.observation
WHERE CONCAT('Patient/', patient.id) = condition.subject.reference
 AND CONCAT('Patient/', patient.id) = observation.subject.reference
   AND (year(current_date) - year(date(birthdate))) >= 18
   AND (year(current_date) - year(date(birthdate))) <= 75
   AND condition.code.coding [ 1 ].display like ('%diabetes%');
```

现在，您可以使用 Athena 控制台对结果进行排序或下载以进行进一步分析。

AWS HealthLake 和接口VPC端点 (AWS PrivateLink)

您可以通过创建接口VPC终端节点在您的VPC和 AWS HealthLake 之间建立私有连接。接口VPC端点由AWS PrivateLink一种可用于私密访问的技术提供支持 HealthLake；APIs无需互联网网关、NAT设备、VPN连接或 AWS Direct Connect 连接。您中的实例VPC不需要公有 IP 地址即可与 HealthLake; 通信APIs。您VPC和 HealthLake; 之间的流量不会离开 Amazon 网络。

每个接口端点均由子网中的一个或多个弹性网络接口表示。

有关更多信息，请参阅 Amazon VPC 用户指南中的接口VPC终端节点 (AWS PrivateLink)。

HealthLake VPC端点注意事项

在为设置接口VPC终端节点之前 HealthLake，请务必查看《亚马逊VPC用户指南》中的接口终端节点属性和限制。

HealthLake 支持从您调用其所有API操作VPC。

为以下对象创建接口VPC端点： HealthLake

您可以使用 Amazon VPC 控制台或 AWS Command Line Interface (AWS CLI) 为 HealthLake; 服务创建VPC终端节点。有关更多信息，请参阅 Amazon VPC 用户指南中的创建接口终端节点。

使用以下服务名称为 HealthLake; 创建VPC终端节点：

- com.amazonaws。 *region*.healthl

如果您DNS为终端节点开启私有功能，则可以使用该终端节点的默认DNS名称向 HealthLake该区域API发出请求。例如，*healthlake.us-east-1.amazonaws.com*。

有关更多信息，请参阅 Amazon VPC 用户指南中的通过接口终端节点访问服务。

为创建VPC终端节点策略 HealthLake

您可以将终端节点策略附加到控制访问权限的VPC终端节点 HealthLake。该策略指定以下信息：

- 可执行操作的主体。

- 可执行的操作。
- 可对其执行操作的资源。

有关更多信息，请参阅 Amazon VPC 用户指南中的使用VPC终端节点控制对服务的访问。

示例： HealthLake 操作的VPC端点策略

以下是的终端节点策略示例 HealthLake。当连接到终端节点时，此策略将授予所有资源的所有委托人访问该 HealthLake`CreateFHIRDatastore`操作的权限。

```
{
   "Statement":[
      {
         "Principal":"*",
         "Effect":"Allow",
         "Action":[
            "healthlake:create-fhir-datastore"
         ],
         "Resource":"*"
      }
   ]
}
```

在中标记资源 AWS HealthLake

您可以将自己的元数据以标签的形式分配给 AWS 资源。每个标签都是由用户定义的键和值组成的标签。标签可帮助您管理、识别、组织、搜索和筛选资源。

本主题介绍常用的标记类别和策略，以帮助您实施一致且有效的标记策略。以下各节假设对AWS资源、标记、详细账单以及AWS身份和访问管理 (IAM) 有基本的了解。

每个标签具有两个部分：

- 标签密钥（例如 CostCenter，“环境” 或 “项目”）。标签键区分大小写。
- 标签值（例如，111122223333 或 Production）。与标签键一样，标签值区分大小写。

您可使用标签，按用途、所有者、环境或其他标准对资源进行分类。有关更多信息，请参阅 AWS 标记策略。

您可以从每个资源的服务控制台、服务或，一次为一个资源添加、更改或移除标签AWSCLI。API

要启用标记，请确保 TagResources 已获得授权。您可以 TagResources 通过附加IAM策略进行授权，如下例所示。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "healthlake:CreateFHIRDatastore",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "healthlake:TagResource",
            "Resource": "*"
        }
    ]
}
```

重要提示

AWS HealthLake 根据责任AWS共担模式政策保护客户数据。这意味着所有客户数据在过渡和静态时都经过加密。但是，并非所有客户为数据存储或基于作业的操作输入的名称都经过加密。它们不应包含个人身份信息或 Protected Health 信息。有关更多信息，请参阅 “ AWS HealthLake 安全” 一章。

最佳实践

在为 AWS 资源创建标记策略时，请遵循最佳实践：

- 请勿在标签中存储个人身份信息 (PII)、Personal Health 信息 (PHI) 或其他敏感信息。
- 对标签使用标准化的区分大小写格式，并跨所有资源类型一致地应用该格式。
- 考虑支持多种用途的标签准则，如管理资源访问控制、成本跟踪、自动化和组织。
- 使用自动化工具来帮助管理资源标签。 AWSResource Gro ups 和 Resource Groups 标签API允许对标签进行编程控制，从而可以自动管理、搜索和筛选标签和资源。
- 使用更多标签时，标记会更有效。
- 标签可以根据用户需求的变化进行编辑或修改，但是要更新访问控制标签，您还必须更新引用这些标签的策略以控制对资源的访问。

标记要求

标签具有以下要求：

- 密钥不能以 aws: 为前缀。
- 每个标签集中的各个键必须是独一无二的。
- 键的长度必须介于 1 到 128 个允许的字符之间。
- 值的长度必须介于 0 到 256 个允许的字符之间。
- 每个标签集中的值不需要是唯一的。
- 可以用作键和值的字符包括 Unicode 字符、数字、空格及以下符号：_ . : / = + - @。
- 键和值区分大小写。

向数据存储添加标签

向数据存储添加标签可以帮助您识别和组织AWS资源并管理对资源的访问权限。首先，向数据存储添加一个或多个标签（键值对）。每个用户最多可以使用五十个标签。在键和值字段中可以使用的字符也有限制。

获得标签后，您可以根据这些标签创建IAM策略来管理对数据存储的访问权限。您可以使用 HealthLake 控制台或 AWS CLI 向数据存储添加标签。为存储库添加标签会影响对该存储库的访问。在向数据存储添加标签之前，请务必查看任何可能使用标签来控制对资源（例如数据存储）的访问权限的IAM策略。

按照以下步骤使用 AWS CLI 向 HealthLake 数据存储添加标签。要在创建数据存储时向其添加标签，请参阅在中创建数据存储 AWS HealthLake。

在终端或命令行上，运行 tag-resource 命令，指定要在其中添加标签的数据存储的 Amazon 资源名称 (ARN) 以及要添加的标签的键和值。您可以向数据存储中添加多个标签。在键和值字段中可以使用的字符也有限制，如中所列。标记要求例如，要在创建数据存储时向其添加标签，可以在中使用以下命令 AWS CLI。数据存储的名称是 Test_Data_Store，添加的两个带键的标签是 key1 和 key2，其值分别为 value1 和 value2 :

```
aws healthlake create-fhir-datastore --datastore-type-version R4 --preload-data-config
 PreloadDataType="SYNTHEA" --datastore-name "Test_Data_Store" --tags '[{"Key": "key1",
 "Value": "value1"}, {"Key": "key2", "Value": "value2"}]' --region us-east-1
```

要向现有数据存储添加标签，可以运行以下示例命令：

```
aws healthlake tag-resource --resource-arn "arn:aws:healthlake:us-
east-1:691207106566:datastore/fhir/0725c83f4307f263e16fd56b6d8ebdbe" --tags '[{"Key":
 "key1", "Value": "value1"}]' --region us-east-1
```

如果成功，此命令将不返回任何响应。

列出数据存储的标签

按照以下步骤 AWS CLI 使用查看 HealthLake 数据存储的AWS标签列表。如果尚未添加标签，则返回的列表为空。

在终端或命令行中，运行 list-tags-for-resource命令，如以下示例所示。

```
aws healthlake-test list-tags-for-resource --resource-arn "arn:aws:healthlake:us-
east-1:674914422125:datastore/fhir/0725c83f4307f263e16fd56b6d8ebdbe" --region us-
east-1
```

```
{
    "tags": {
        "key": "value",
        "key1": "value1"
    }
}
```

从数据存储中移除标签

您可以移除一个或多个与数据存储关联的标签。删除标签不会从与该标签关联的其他 AWS 资源中删除该标签。

在终端或命令行中，运行 untag-resource 命令，指定要移除标签的数据存储的 Amazon 资源名称 (ARN) 和要删除的标签的标签密钥。

```
aws healthlake untag-resource --resource-arn "arn:aws:healthlake:us-
east-1:674914422125:datastore/fhir/b91723d65c6fdeb1d26543a49d2ed1fa" --tag-keys
 '["key1"]' --region us-east-1
```

如果成功，则此命令不会返回响应。要验证与数据存储关联的标签，请运行 list-tags-for-resource命令。

监控 HealthLake

监控是维护和其他AWS解决方案的可靠性、可用性和性能的重要组成部分。 HealthLake AWS提供以下监控工具 HealthLake，供您监视、报告问题并在适当时自动采取措施：

- 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，创建自定义仪表板，并设置警报，以便在指定指标达到特定阈值时通知您或采取行动。例如，您可以 CloudWatch 跟踪您的 Amazon EC2 实例的CPU使用情况或其他指标，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。
- AWS CloudTrail捕获您的账户或代表您的 AWS 账户API拨打的电话和相关事件。然后它将日志文件传送到您指定的 Amazon S3 存储桶。您可以识别哪些用户和账户拨打了电话 AWS、这些呼叫的源 IP 地址以及呼叫发生的时间。有关更多信息，请参阅 用户指南。AWS CloudTrail

主题

- HealthLake 使用 Amazon 进行监控 CloudWatch

HealthLake 使用 Amazon 进行监控 CloudWatch

您可以使用 HealthLake 进行监控 CloudWatch，它收集原始数据并将其处理为可读的近乎实时的指标。这些统计数据会保存 15 个月，从而使您能够使用历史信息，并能够更好地了解您的 Web 应用程序或服务的执行情况。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。

将报告所有指标 HealthLake APIs，包括以下指标。

- 数据存储管理 APIs —C reateFHIRDatastore、D eleteFHIRDatastore、D escribeFHIRDatastore、L istFHIRDatastores
- 导入和导出 APIs —S tartFHIRImport Job、L istFHIRImport Job、D escribeFHIRImport Job、S Jo tartFHIRExport b、L Job、D istFHIRExport escribeFHIRExport Job
- HTTPREST客户和资源管理 APIs — CreateResource、 DeleteResource、 GetCapabilities、 ReadResource、 SearchAll、 SearchWithGet、 SearchWithPost、 UpdateResource。
- 标记 APIs — ListTagsForResource、、 TagResource UntagResource

以下各表列出 HealthLake 的指标和维度。

报告了以下指标。每个都以用户指定数据范围的频率计数表示。

指标

指标	描述
调用计数	呼叫的次数APIs。可以为账户或指定的数据存储报告此问题。 单位：计数 有效统计数据：Sum、Count 维度：操作、数据存储 ID、数据存储类型
成功请求	成功API请求的数量。 单位：计数 有效统计数据：Sum、Average 维度：操作、数据存储、数据存储类型
用户错误	由于用户错误而失败的请求数。 单位：计数 有效统计数据：Sum、Average 维度：操作、数据存储 ID、数据存储类型
服务器错误	由于服务器错误而失败的请求数。 单位：计数 有效统计数据：Sum、Average 维度：操作、数据存储 ID、数据存储类型
限制的请求	已被限制的请求数。此指标不包含在用户或服务器错误计数中。 单位：计数

指标	描述
	有效统计数据：Sum、Average 维度：操作、数据存储 ID、数据存储类型
延迟	处理用户请求所用的时间（以毫秒为单位）。 单位：毫秒 有效统计数据：Minimum、Maximum、Sum、Average 维度：操作、数据存储 ID、数据存储类型

报告了以下维度。

尺寸

Dimensions	描述
操作	使用了哪个API操作
DataStore身份证	API请求中包含的数据存储
DataStoreType	数据存储的类型（目前仅支持 FHIR R4）

您可以使用AWS管理 HealthLake 控制台 AWS CLI、或 CloudWatch API。您可以 CloudWatch API通过其中一个 Amazon AWS 软件开发套件 (SDKs) 或 CloudWatch API工具来使用。 HealthLake 控制台根据来自的原始数据显示图表 CloudWatch API。

您必须具有相应的 CloudWatch 权限才能 HealthLake 进行监控 CloudWatch。有关更多信息，请参阅《亚马逊 CloudWatch 用户指南》 CloudWatch 中的 Amazon 身份验证和访问控制。

查看 HealthLake 指标

查看指标（CloudWatch 控制台）

1. 登录 AWS 管理控制台，打开 CloudWatch 控制台。
2. 选择 “指标”，选择 “所有指标”，然后选择 AWS/HealthLake。

3. 选择维度、指标名称，然后选择 添加到图表。
4. 选择日期范围的值。所选日期范围的指标计数将显示在该图表中。

使用创建警报 CloudWatch

CloudWatch 警报在指定时间段内监视单个指标，并执行一项或多项操作：向亚马逊简单通知服务 (AmazonSNS) 主题或 Auto Scaling 策略发送通知。一个或多个操作基于指标在您指定的多个时间段内相对于给定阈值的值。 CloudWatch 还可以在警报状态发生变化时向您发送 Amazon SNS 消息。

CloudWatch 警报仅在状态发生变化并且持续到您指定的时间段内时才会调用操作。

查看指标（CloudWatch 控制台）

1. 登录 AWS 管理控制台，打开 CloudWatch 控制台。
2. 依次选择 Alarms 和 Create Alarm。
3. 选择 AWS/HealthLake，然后选择一个指标。
4. 对于 Time Range，请选择要监控的时间范围，然后选择 Next。
5. 输入名称和描述。
6. 对于 Whenever，选择 >=，然后键入一个最大值。
7. 如果 CloudWatch 要在达到警报状态时发送电子邮件，请在 “操作” 部分的 “无论何时出现此警报”，选择 “状态” ALARM。在 “发送通知至” 中，选择一个邮件列表或选择 “新建列表” 并创建新的邮件列表。
8. 预览警报预览部分中的警报。如果对警报满意，请选择 Create Alarm (创建警报)。

继续FHIR与SMART集成 AWS HealthLake

FHIR启用 HealthLake 数据存储上的可替代医疗应用程序和可重复使用的技术 (SMART) 允许SMARTFHIR合规应用程序访问存储在数据存储中的 HealthLake 数据。 HealthLake 通过使用第三方授权服务器对请求进行身份验证和授权，以及在中设置其他资源来访问数据。 AWS

要在 HealthLake 数据存储中FHIR使用 SMART on，您必须在 C reateFHIRDatastore API 请求中提供以下内容。

- 将AuthorizationStrategy等于设置为SMART_ON_FHIR_V1。
- 将设置为IdpLambdaArn 等于 AWS Lambda 您创建的，以便使用授权服务器管理令牌解码。ARN
- 定义授权服务器中指定的元数据元素。这些元数据元素在发现文档中返回。要了解更多信息，请参阅正在获取已SMARTFHIR启用 HealthLake 数据存储的发现文档。
- 可选：FineGrainedAuthorizationEnabled如果您已在授权服务器上设置了细粒度授权，则启用此选项。

您可以使用 AWS Command Line Interface (AWS CLI) 或通过 AWS 支持的数据存储来创建FHIR已启用的数据存储SDKs。SMART不支持使用 HealthLake 控制台SMART创建FHIR已启用的 HealthLake 数据存储。要了解更多信息，请参阅 创建FHIR已SMART启用的数据存储。

要在请求中规定这些参数，您需要在其他 AWS 服务（ AWS Secrets Manager 和 AWS Lambda ）中设置资源，创建新的IAM服务角色并设置FHIR符合要求SMART的授权服务器。使用 “设置实现不FHIR合规的数据存储所需的资源” 一SMART节，了解有关设置所需资源的更多信息，并查看FHIR应用程序如何与 HealthLake之交互的SMART高级概述。

这意味着， AWS Identity and Access Management 您不是通过管理用户凭证，而是使用不FHIR合规的SMART授权服务器来管理用户凭证。

HealthLake SMART在 FHIR 1.0 上支持。要了解有关此框架的更多信息，请参阅SMART应用程序启动框架实施指南版本 1.0。

要使用 SMART on 对数据存储请求进行授权和身份验证FHIR， HealthLake 支持使用：

- OpenID (AuthN) 集成：用于验证该人或客户端应用程序是否是他们声称的身份（或什么）。
- OAuth2.0 (AuthZ) 集成：用于授权经过身份验证的请求也可以读取或写入 HealthLake 数据存储中的哪些FHIR资源。这是由授权服务器中设置的范围定义的

目录

SMARTon 的身份验证要求 FHIR

要访问FHIR HealthLake 数据存储中的FHIR资源，客户端应用程序必须由OAuth兼容 2.0 的授权服务器进行授权，并在请求中出示 OAuth Bearer 令牌。SMART FHIR REST API要通过众所周知的统一资源标识符在 Discovery Document HealthLake SMART 上使用 “FHIR发现文档” 来查找授权服务器的端点。要了解有关此过程的更多信息，请参阅正在获取已SMARTFHIR启用 HealthLake 数据存储的发现文档。

SMART在FHIR HealthLake 数据存储上创建时，必须在 C reateFHIRDatastore 请求的`metadata`元素中定义授权服务器的端点和令牌端点。要了解定义`metadata`元素的更多信息，请参阅创建FHIR已SMART启用的 HealthLake 数据存储。

使用授权服务器端点，客户端应用程序将使用授权服务对用户进行身份验证。授权和身份验证后，授权服务会生成 JSON Web 令牌 (JWT) 并传递给客户端应用程序。此令牌包含允许客户端应用程序使用的FHIR资源范围，这反过来又限制了用户能够访问的数据。或者，如果提供了启动范围，则响应中将包含这些详细信息。要了解有关支持的开SMART启FHIR作用域的更多信息 HealthLake，请参阅SMART在FHIROAuth作用域上受支持 HealthLake。

使用授权服务器JWT授予的，客户端应用程序对FHIR已启用的 HealthLake 数据存储SMART进行FHIRRESTAPI调用。要验证和解码JWT，您需要创建一个 Lambda 函数。 HealthLake 在收到请求时代表您调用此 Lambda 函数。FHIR REST API要查看起始 Lambda 函数的示例，请参阅。使用 AWS Lambda FHIR已启用的 HealthLake 数据存储SMART进行令牌验证

创建FHIR未启用的 HealthLake 数据存储所需的授权服务器元素 SMART

在 C reateFHIRDatastore 请求中，您需要提供授权端点和令牌端点作为`IdentityProviderConfiguration`对象中`metadata`元素的一部分。授权端点和令牌端点都是必需的。要查看在 C reateFHIRDatastore 请求中如何指定这一点的示例，请参阅创建FHIR已SMART启用的 HealthLake 数据存储。

需要声明才能在FHIR未启用的 HealthLake 数据存储SMART上完成FHIRRESTAPI请求

您的 AWS Lambda 函数必须包含以下声明才能成为对FHIR未启用的 HealthLake 数据存储SMART的有效FHIRRESTAPI请求。

- nbf:（Not Before）索赔 — “nbf”（不在此之前）索赔指明了受理索赔之前的时间。JWT MUST NOT“nbf” 索赔的处理要求是 “nbf” 索赔date/time MUST be after or equal to the not-before date/time 中所列的当前索赔。我们提供的示例 Lambda 函数`iat`从服务器响应转换为。`nbf`
- `exp`:（到期时间）索赔 — “过期时间”（到期时间）索赔确定了到期时间，在此时间或之后JWT不得接受处理。
- `isAuthorized`: 布尔值设置为`True`。表示请求已在授权服务器上获得授权。
- aud:（受众）声明 — “aud”（受众）索赔用于标识其目标收件人。JWT这必须是FHIR已SMART启用的 HealthLake 数据存储端点。

- `scope`：这必须是至少一个与FHIR资源相关的作用域。此范围是在您的授权服务器上定义的。要详细了解所接受的FHIR资源相关作用域 HealthLake，请参阅HealthLake 数据存储FHIR资源特定范围。

SMART在FHIROAuth作用域上受支持 HealthLake

HealthLake 使用 OAuth 2.0 作为授权协议。在授权服务器上使用此协议可以定义客户端应用程序也可以拥有读取和/或写入权限 HealthLake 的数据存储中的哪些FHIR资源。

SMARTon f FHIR ramework 定义了一组可以向授权服务器请求的作用域。要查看 SMART on FHIR framework 中的作用域定义，请参阅SMART《HL7FHIR资源指南》中的 “FHIR作用域”。

例如，仅允许患者查看实验室结果或查看其联系方式的客户端应用程序只能获得（通过请求）FHIRREST请求*read*范围的授权。要将它们定义为作用域，您需要提供一个如下所示的字符串`patient/Observation.read`。这将允许客户端应用程序以只读方式请求对`Patient`资源类型的访问权限。`Observation`

独立发布范围

HealthLake 支持独立启动模式范围`launch/patient`。

在独立启动模式下，客户端应用程序请求访问患者的临床数据，因为客户端应用程序不知道用户和患者。因此，客户端应用程序的授权请求明确要求返回患者范围。成功进行身份验证后，授权服务器会发出包含请求的启动患者范围的访问令牌。所需的患者环境与访问令牌一起在授权服务器的响应中提供。

支持的启动模式范围

范围	描述
`launch/patient`	OAuth2.0 授权请求中的一个参数，要求在授权响应中返回患者数据。

HealthLake 数据存储FHIR资源特定范围

HealthLake 定义了三个级别的作用域。

- 特定于患者的范围允许访问有关单个患者的特定数据。在启动上下文中指定了哪位患者。
- 用户级范围授予对用户可以访问的特定数据的访问权限。
- 系统级作用域授予对数据存储中所有FHIR资源的读/写访问权限。 HealthLake

下表显示了构造支持的FHIR资源相关作用域的语法 HealthLake。一般格式如下：

```
( 'patient' | 'user' | 'system' ) '/' ( fhir-resource | '*' ) '.' ( 'read' | 'write' |
 '*' )
```

HealthLake 数据存储上支持的授权范围

作用域语法	示例：作用域	结果
`patient/(fhir-resource | '*').('read' | 'write' | '*')`	`patient/AllergyIntolerance.*`	客户端应用程序将具有对过敏症的读/写权限。
`user/(fhir-resource | '*').('read' | 'write' | '*')`	`user/Observation.read`	客户端应用程序将具有对所有记录的观测值的读取权限。
`system/('read' | 'write' | *)`	`system/*.*`	客户端应用程序将具有对所有数据的读/写访问权限。

使用 AWS Lambda FHIR已启用的 HealthLake 数据存储SMART进行令牌验证

创建FHIR未SMART启用的 HealthLake 数据存储时，需要在`CreateFHIRDatastore`请求中提供该ARN AWS Lambda 函数的。Lambda 函数ARN是使用参数在`IdentityProviderConfiguration`对象中指定的。`IdpLambdaArn`

您必须先创建 Lambda 函数，然后才能创建FHIR已SMART启用的 HealthLake 数据存储。一旦创建了数据存储，就ARN无法更改 Lambda。要查看ARN您在创建数据存储时指定的 Lambda，请使用操作。`DescribeFHIRDatastore` API

要在FHIR已启用的 HealthLake 数据存储上成功FHIRREST请求请求，您的 Lambda 函数需要执行以下操作：SMART

- Lambda 函数必须在不到 1 秒的时间内向 HealthLake 数据存储端点返回响应。
- 对客户端应用程序发送的RESTAPI请求的授权标头中提供的访问令牌进行解码。
- 分配具有足够权限的IAM服务角色来执行FHIRRESTAPI请求。

- 完成申请需要以下索赔。FHIR REST API要了解更多信息，请参阅 必需的索赔。
  - nbf
  - exp
  - isAuthorized
  - aud
  - scope

使用 Lambda 时，除了您的 Lambda 函数外，您还需要创建执行角色和基于资源的策略。Lambda 函数的执行IAM角色是一个向函数授予访问运行时所需AWS服务和资源的权限的角色。您提供的基于资源的策略必须 HealthLake 允许代表您调用您的函数。

本主题中的各节描述了来自客户端应用程序的示例请求和解码后的响应、创建 Lamb AWS da 函数所需的步骤以及如何创建可以假设的基于资源的策略 HealthLake 。

- 第 1 部分：创建 Lambda 函数
- 第 2 部分：创建 AWS Lambda 函数使用的 HealthLake 服务角色
- 第 3 部分：更新 Lambda 函数的执行角色
- 第 4 部分：向您的 Lambda 函数添加资源策略
- 第 5 部分：为您的 Lambda 函数配置并发性

创建 AWS Lambda 函数

本主题中创建的 Lambda 函数在 HealthLake 收到对FHIR未启用的 HealthLake 数据存储的SMART请求时触发。来自客户端应用程序的请求包含RESTAPI调用和包含访问令牌的授权标头。

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Authorization: Bearer i8hweunweunweofiwweoijewiwe
```

本主题中的示例 Lambda 函数 AWS Secrets Manager 用于掩盖与授权服务器相关的证书。我们强烈建议不要直接在 Lambda 函数中提供授权服务器登录详细信息。

Example 验证包含授权持有者令牌的FHIRREST请求

Lambda 函数示例向您展示了如何验证发送到FHIR已启用 HealthLake 数据存储SMART的FHIRREST请求。要查看有关如何实现此 Lambda 函数的 step-by-steps说明，请参阅。使用创建 Lambda 函数 AWS Management Console

如果FHIRRESTAPI请求不包含有效的数据存储终端节点、访问令牌和REST操作，则 Lambda 函数将失败。要了解有关所需授权服务器元素的更多信息，请参阅必需的索赔。

```
import base64
import boto3
import logging
import json
import os
from urllib import request, parse

logger = logging.getLogger()
logger.setLevel(logging.INFO)

## Uses Secrets manager to gain access to the access key ID and secret access key for
 the authorization server
client = boto3.client('secretsmanager', region_name="region-of-datastore")
response = client.get_secret_value(SecretId='name-specified-by-customer-in-
secretsmanager')
secret = json.loads(response['SecretString'])
client_id = secret['client_id']
client_secret = secret['client_secret']


unencoded_auth = f'{client_id}:{client_secret}'
headers = {
  'Authorization': f'Basic {base64.b64encode(unencoded_auth.encode()).decode()}',
  'Content-Type': 'application/x-www-form-urlencoded'
}

auth_endpoint = os.environ['auth-server-base-url'] # Base URL of the Authorization
 server
user_role_arn = os.environ['iam-role-arn'] # The IAM role client application will use
 to complete the HTTP request on the datastore

def lambda_handler(event, context):
    if 'datastoreEndpoint' not in event or 'operationName' not in event or
 'bearerToken' not in event:
    return {}

    datastore_endpoint = event['datastoreEndpoint']
    operation_name = event['operationName']
    bearer_token = event['bearerToken']
```

```
    logger.info('Datastore Endpoint [{}], Operation Name:
 [{}]'.format(datastore_endpoint, operation_name))

    ## To validate the token
    auth_response = auth_with_provider(bearer_token)
    logger.info('Auth response: [{}]'.format(auth_response))
    auth_payload = json.loads(auth_response)
    ## Required parameters needed to be sent to the datastore endpoint for the HTTP
 request to go through
    auth_payload["isAuthorized"] = bool(auth_payload["active"])
    auth_payload["nbf"] = auth_payload["iat"]
    return {"authPayload": auth_payload, "iamRoleARN": user_role_arn}

## access the server
def auth_with_provider(token):
    data = {'token': token, 'token_type_hint': 'access_token'}
    req = request.Request(url=auth_endpoint + '/v1/introspect',
 data=parse.urlencode(data).encode(), headers=headers)
    with request.urlopen(req) as resp:
    return resp.read().decode()
```

使用创建 Lambda 函数 AWS Management Console

此过程假设您已经创建了在处理FHIR未启用的 HealthLake 数据存储上的FHIRRESTAPI请求时 HealthLake 要代入SMART的服务角色。如果您尚未创建服务角色，则仍然可以创建 Lambda 函数。您需要先添加 of 服务角色，ARNLambda 函数才会起作用。要了解有关创建服务角色并在 Lambda 函数中指定该角色的更多信息，请参阅 创建用于解码的 AWS Lambda 函数的 HealthLake 服务角色 JWT

创建 Lambda 函数 ()AWS Management Console

1. 打开 Lamba 控制台的 Functions page（函数页面）。
2. 选择 Create function（创建函数）。
3. 选择从头开始编写。
4. 在 “基本信息” 下输入函数名称。在运行时下，选择基于 python 的运行时。
5. 在 Execution Role（执行角色）中，选择 Create a new role with basic Lambda permissions（创建具有基本 Lambda 权限的新角色）。

   Lambda 创建了一个执行角色，该角色向该函数授予将日志上传到亚马逊的权限。CloudWatchLambda 函数在您调用函数时担任执行角色，并使用执行角色为创建证书。 AWS SDK

6. 选择代码选项卡，然后添加示例 Lambda 函数。

   如果您尚未为 Lambda 函数创建要使用的服务角色，则需要先创建该角色，然后示例 Lambda 函数才能运行。要了解有关为 Lambda 函数创建服务角色的更多信息，请参阅。[创建用于解码的 AWS Lambda 函数的 HealthLake 服务角色 JWT]

```
import base64
import boto3
import logging
import json
import os
from urllib import request, parse

logger = logging.getLogger()
logger.setLevel(logging.INFO)

## Uses Secrets manager to gain access to the access key ID and secret access key
 for the authorization server
client = boto3.client('secretsmanager', region_name="region-of-datastore")
response = client.get_secret_value(SecretId='name-specified-by-customer-in-
secretsmanager')
secret = json.loads(response['SecretString'])
client_id = secret['client_id']
client_secret = secret['client_secret']


unencoded_auth = f'{client_id}:{client_secret}'
headers = {
  'Authorization': f'Basic {base64.b64encode(unencoded_auth.encode()).decode()}',
  'Content-Type': 'application/x-www-form-urlencoded'
}

auth_endpoint = os.environ['auth-server-base-url'] # Base URL of the Authorization
 server
user_role_arn = os.environ['iam-role-arn'] # The IAM role client application will
 use to complete the HTTP request on the datastore

def lambda_handler(event, context):
    if 'datastoreEndpoint' not in event or 'operationName' not in event or
 'bearerToken' not in event:
    return {}

    datastore_endpoint = event['datastoreEndpoint']
```

```
    operation_name = event['operationName']
    bearer_token = event['bearerToken']
    logger.info('Datastore Endpoint [{}], Operation Name:
 [{}]'.format(datastore_endpoint, operation_name))

    ## To validate the token
    auth_response = auth_with_provider(bearer_token)
    logger.info('Auth response: [{}]'.format(auth_response))
    auth_payload = json.loads(auth_response)
    ## Required parameters needed to be sent to the datastore endpoint for the HTTP
 request to go through
    auth_payload["isAuthorized"] = bool(auth_payload["active"])
    auth_payload["nbf"] = auth_payload["iat"]
    return {"authPayload": auth_payload, "iamRoleARN": user_role_arn}

## Access the server
def auth_with_provider(token):
    data = {'token': token, 'token_type_hint': 'access_token'}
    req = request.Request(url=auth_endpoint + '/v1/introspect',
 data=parse.urlencode(data).encode(), headers=headers)
    with request.urlopen(req) as resp:
    return resp.read().decode()
```

修改 Lambda 函数的执行角色

创建 Lambda 函数后，您需要更新执行角色以包含调用 Secrets Manager 所需的权限。在 Secrets Manager 中，你创建的每个密钥都有一个ARN。要应用最低权限，执行角色只能访问 Lambda 函数执行所需的资源。

您可以通过在控制台中搜索或在 Lambda IAM 控制台中选择配置来修改 Lambda 函数的执行角色。要了解有关管理 Lambda 函数执行角色的更多信息，请参阅。Lambda 执行角色

Example 授予访问权限的 Lambda 函数执行角色 **`GetSecretValue`**

将IAM操作`GetSecretValue`添加到执行角色可授予示例 Lambda 函数运行所需的权限。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "secretsmanager:GetSecretValue",
```

```
      "Resource": "arn:aws:secretsmanager:your-region:your-aws-account-
id:secret:secret-name-DKodTA"
    }
  ]
}
```

此时，您已经创建了一个 Lambda 函数，该函数可用于验证在FHIR已启用 HealthLake 数据存储的FHIRREST请求中提供的访问令牌。SMART

创建用于解码的 AWS Lambda 函数的 HealthLake 服务角色 JWT

角色：IAM管理员

可以添加或删除IAM策略以及创建新IAM身份的用户。

服务角色

服务角色是由一项服务代入、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》 AWS 服务中的创建角色以向委派权限。

在 JSON Web 令牌 (JWT) 被解码后，Lambda 还需要返回IAM一个角色的授权。ARN此角色必须具有执行请求所需的权限，否则RESTAPI请求将因权限不足而失败。

使用自定义策略设置时IAM，最好授予所需的最低权限。要了解更多信息，请参阅《IAM用户指南》中的应用最低权限权限。

创建要在授权 Lambda 函数中指定的 HealthLake 服务角色需要两个步骤。

- 首先，你需要创建IAM策略。该策略必须指定对您在授权服务器中为其提供范围的FHIR资源的访问权限。
- 其次，你需要创建服务角色。创建角色时，您可以指定信任关系并附加您在第一步中创建的策略。信任关系指定 HealthLake 为服务主体。您需要在此步骤中指定 HealthLake 数据存储ARN和 AWS 帐户 ID。

创建新IAM政策

您在授权服务器中定义的范围决定了经过身份验证的用户可以访问 HealthLake 数据存储中的哪些FHIR资源。

可以对您创建的IAM策略进行定制，使其与您定义的范围相匹配。

可以在IAM策略声明的Action元素中定义以下操作。您可以为表Action中的每一个定义一个Resource types。 HealthLake 在数据存储中，唯一支持的资源类型可以在IAM权限策略语句的Resource元素中定义。

单个FHIR资源不是可以定义为IAM权限策略元素的资源。

操作定义为 HealthLake

操作	描述	访问级别	资源类型（必填）
CreateResource	向创建资源授予权限	写入	数据存储：arn: aws: healthlakeARN:: datastore /fhir/ **your-region 111122223333 your-datastore-id**
DeleteResource	授予删除资源的权限	写入	数据存储：arn: aws: healthlakeARN:: datastore /fhir/ **your-region 111122223333 your-datastore-id**
ReadResource	授予读取资源的权限	读取	数据存储：arn: aws: healthlakeARN:: datastore /fhir/ **your-region 111122223333 your-datastore-id**
SearchWithGet	授予使用GET方法搜索资源的权限	读取	数据存储：arn: aws: healthlakeARN:: datastore /fhir/ **your-region 111122223333 your-datastore-id**
SearchWithPost	授予使用POST方法搜索资源的权限	读取	数据存储：arn: aws: healthlakeARN:: datastore /fhir/ **your-region 111122223333 your-datastore-id**
StartFHIRExportJobWithPost	授予开始FHIR导出任务的权限 GET	写入	数据存储：arn: aws: healthlakeARN:: datastore /fhir/ **your-region 111122223333 your-datastore-id**

操作	描述	访问级别	资源类型（必填）
UpdateResource	授予更新资源的权限	写入	数据存储：arn: aws: healthlakeARN:: datastore /fhir/ **your-region 111122223333 your-datastore-id**

要开始使用，你可以使用AmazonHealthLakeFullAccess。此策略将允许对数据存储中找到的所有FHIR资源进行读取、写入、搜索和导出。要授予对数据存储的只读权限，请使用AmazonHealthLakeReadOnlyAccess。

要了解有关使用 AWS Management Console、 AWS CLI或创建自定义策略的更多信息 IAMSDKs，请参阅IAM用户指南中的创建IAM策略。

为 HealthLake （IAM控制台）创建服务角色

使用此过程创建服务角色。创建服务时，还需要指定IAM策略。

为 HealthLake （IAM控制台）创建服务角色

1. 登录 AWS Management Console 并打开IAM控制台，网址为https://console.aws.amazon.com/iam/。
2. 在 IAM 控制台的导航窗格中，选择 Roles。
3. 然后，选择创建角色。
4. 在选择信任实体页面上，选择自定义信任策略。
5. 接下来，在 “自定义信任策略” 下更新示例策略，如下所示。**your-account-id**替换为您的账号，然后添加要在导入或导出任务中使用的数据存储。ARN

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Principal": {
            "Service": "healthlake.amazonaws.com"
```

```
        },
        "Condition": {
          "StringEquals": {
            "aws:SourceAccount": "your-account-id"
            },
            "ArnEquals": {
              "aws:SourceArn": "arn:aws:healthlake:your-region:your-account-
id:datastore/fhir/your-datastore-id"
            }
          }
        }
  ]
}
```

6. 然后选择下一步。
7. 在添加权限页面上，选择您希望 HealthLake 服务采用的策略。要查找您的策略，请在权限策略下进行搜索。
8. 然后，选择附加策略。
9. 然后在 “名称、查看和创建” 页面的 “角色名称” 下输入名称。
10. （可选）然后在 “描述” 下，为您的角色添加简短描述。
11. 如果可能，输入有助于识别该角色的作用的角色名称或角色名称后缀。角色名称在您的角色中必须是唯一的 AWS 账户。名称不区分大小写。例如，您无法同时创建名为 **PRODROLE** 和 **prodrole** 的角色。由于多个单位可能引用该角色，角色创建完毕后无法编辑角色名称。
12. 查看角色详细信息，然后选择创建角色。

要了解如何在示例 Lambda 函数ARN中指定角色，请参阅。创建 AWS Lambda 函数

Lambda 执行角色

Lambda 函数的执行IAM角色是一个向函数授予访问 AWS 服务和资源的权限的角色。此页面提供有关如何创建、查看和管理 Lambda 函数执行角色的信息。

默认情况下，当您使用创建新的 Lambda 函数时，Lambda 会创建具有最低权限的执行角色。 AWS Management Console要管理在执行角色中授予的权限，请参阅 Lambda 开发人员指南中的在IAM控制台中创建执行角色。

本主题中提供的示例 Lambda 函数使用 Secrets Manager 来掩盖授权服务器的证书。

与您创建的任何IAM角色一样，遵循最低权限的最佳实践非常重要。在开发阶段，您有时可能会授予超出所需权限的权限。在生产环境中发布函数之前，最佳实践是调整策略，使其仅包含所需权限。有关更多信息，请参阅《用户指南》中的应用最低权限。 IAM

HealthLake 允许触发你的 Lambda 函数

因此， HealthLake 可以代表您调用 Lambda 函数，您必须执行以下操作：

- 您需要设置`IdpLambdaArn`等于您要在请求ARN中调用的 Lambda 函数的值。 HealthLake `CreateFHIRDatastore`
- 您需要一个基于资源的策略， HealthLake 允许您代表您调用 Lambda 函数。

在FHIR已启用的 HealthLake 数据存储SMART上 HealthLake 收到FHIRRESTAPI请求时，它需要权限才能代表您调用创建数据存储时指定的 Lambda 函数。要授予 HealthLake 访问权限，您将使用基于资源的策略。要详细了解如何为 Lambda 函数创建基于资源的策略，请参阅开发者指南中的允许 AWS 服务调用 Lambda 函数。AWS Lambda

为您的 Lambda 函数配置并发性

> **Important**
>
> HealthLake 要求您的 Lambda 函数的最大运行时间必须小于一秒（ 1000 毫秒）。如果您的 Lambda 函数超过了运行时间限制，则会出现异常。TimeOut

为避免出现此异常，我们建议配置预配置的并发性。通过在调用增加之前分配预置并发，您可以确保所有请求都由延迟较低的初始化实例来提供。要了解有关配置预配置并发的更多信息，请参阅 Lambda 开发人员指南中的配置预配置并发

要查看您的 Lambda 函数当前的平均运行时间，请使用 Lambda 控制台上您的 Lambda 函数的 “监控” 页面。默认情况下，Lambda 控制台提供持续时间图表，显示您的函数代码处理事件所花费的平均时间、最小时间和最大时间。要了解有关监控 Lambda 函数的更多信息，请参阅 Lambda 开发人员指南中的 Lambda 控制台中的监控函数。

如果您已经为 Lambda 函数配置了并发并想要对其进行监控，请参阅 Lambda 开发者指南中的监控并发性。

创建FHIR已SMART启用的 HealthLake 数据存储

要将 SMART on fr FHIR amework 与一起使用 HealthLake，请使用 C reateFHIRDatastore 请求中指定的`IdentityProviderConfiguration`参数创建 HealthLake 数据存储。在`IdentityProviderConfiguration`参数中，您可以指定以下信息：

- 将AuthorizationStrategy等于设置为`SMART_ON_FHIR_V1`。
- 将设置为IdpLambdaArn等于 AWS Lambda 您创建的，以便使用授权服务器管理令牌解码。ARN
- 将授权服务器中指定的元数据元素定义为一个JSON块。这些元数据元素在发现文档中返回。
- 可选：启用FineGrainedAuthorizationEnabled。指定`True`使用由提供的细粒度授权 HealthLake

您可以使用 AWS Command Line Interface (AWS CLI) 或通过 AWS 支持的数据存储来创建FHIR已启用的数据存储SDKs。SMART不支持使用 HealthLake 控制台SMART创建FHIR已启用的 HealthLake 数据存储。

使用创建FHIR已SMART启用的 HealthLake 数据存储 AWS CLI

您可以使用以下代码示例，使用SMART在FHIR已启用的 HealthLake 数据存储上进行创建 AWS CLI。创建FHIR已SMART启用的 HealthLake 数据存储时，必须指定`identity-provider-configuration`参数。

在`identity-provider-configuration`参数中，您可以选择通过将设置为`FineGrainedAuthorizationEnabled`等于来启用细粒度授权。`True`要了解有关细粒度授权的更多信息，请参阅对FHIR已启用的 HealthLake 数据存储使用细粒度授权 SMART。以下示例包含一个用于表示换行符或作为转义字符的特殊字符。\这是为了清楚起见。

```
aws healthlake create-fhir-datastore \
  --region us-east-1 \
  --datastore-name "your-data-store-name" \
  --datastore-type-version R4 \
  --preload-data-config PreloadDataType="SYNTHEA" \
  --sse-configuration '{ "KmsEncryptionConfig": { \
    "CmkType": "customer-managed-kms-key1",
    "KmsKeyId": "arn:aws:kms:us-east-1:your-account-id:key/your-key-id" } }' \
  --identity-provider-configuration  \
      '{"AuthorizationStrategy": "SMART_ON_FHIR_V1", \
      "FineGrainedAuthorizationEnabled": boolean-false-by-default, \
      "IdpLambdaArn": "arn:aws:lambda:your-region:your-account-id:function:your-lambda-
name" \
```

```
      "Metadata": "{\"issuer\":\"https://ehr.example.com\",\"jwks_uri\":\"https://
ehr.example.com/.well-known/jwks.json\",\"authorization_endpoint\":\"https://
ehr.example.com/auth/authorize\",\"token_endpoint\":\"https://ehr.token.com/auth/
token\",\"token_endpoint_auth_methods_supported\":[\"client_secret_basic\",\"foo\"],
\"grant_types_supported\":[\"client_credential\",\"foo\"],\"registration_endpoint\":
\"https://ehr.example.com/auth/register\",\"scopes_supported\":[\"openId\",\"profile\",
\"launch\"],\"response_types_supported\":[\"code\"],\"management_endpoint\":\"https://
ehr.example.com/user/manage\",\"introspection_endpoint\":\"https://ehr.example.com/
user/introspect\",\"revocation_endpoint\":\"https://ehr.example.com/user/revoke\",
\"code_challenge_methods_supported\":[\"S256\"],\"capabilities\":[\"launch-ehr\",\"sso-
openid-connect\",\"client-public\"]}"}'
```

成功后，您将收到以下JSON响应：

```
{
  "DatastoreArn": "arn:aws:healthlake:your-region:111122223333:datastore/fhir/your-
datastore-id",
  "DatastoreEndpoint": "https://healthlake.your-region.amazonaws.com/datastore/your-
datastore-id/r4/",
  "DatastoreId": "your-data-store-id",
  "DatastoreStatus": "data-store-creation-status"
}
```

对FHIR已启用的 HealthLake 数据存储使用细粒度授权 SMART

仅凭@@ 作用域并不能为你提供必要的具体信息，说明请求者有权在数据存储中访问哪些数据。在授予对已FHIR启用的 HealthLake 数据存储的访问权限时，使用细粒度授权可以提高特异性。SMART 要使用细粒度授权，请在 C reateFHIRDatastore 请求的`IdentityProviderConfiguration`参数`True`中设置`FineGrainedAuthorizationEnabled`等于。

如果您启用了细粒度授权，则您的授权服务器会返回一个`fhirUser`范围`id_token`以及访问令牌。这允许客户端应用程序检索有关用户的信息。客户端应用程序应将`fhirUser`声明视为URI代表当前用户的FHIR资源。这可以是 `Patient`、`Practitioner` 或 `RelatedPerson`。授权服务器的响应还包括一个`user/`范围，该范围定义了用户可以访问哪些数据。这使用为与FHIR资源特定作用域相关的作用域定义的语法：

```
user/(fhir-resource | '*').('read' | 'write' | '*')
```

以下是如何使用细粒度授权来进一步指定与数据访问相关的FHIR资源类型的示例。

- 何时fhirUser是Practitioner，细粒度的授权决定了用户可以访问的患者集合。只有患者以fhirUser全科医生的fhirUser身份提及的患者才允许进入。

```
Patient.generalPractitioner : [{Reference(Practitioner)}]
```

- 何时fhirUser为Patient或，请求中提RelatedPerson及的患者与请求中提及的患者不同fhirUser，细粒度的授权决定了所请求患fhirUser者的访问权限。如果请求的Patient资源中指定了关系，则允许访问。

```
Patient.link.other : {Reference(Patient|RelatedPerson)}
```

正在获取已SMARTFHIR启用 HealthLake 数据存储的发现文档

要使客户端应用程序成功发出FHIRREST请求，它需要收集 HealthLake 数据存储中定义的授权要求。此请求无需授权（持有者令牌）即可成功。

为此，GET请发出请求并附加/.well-known/smart-configuration到数据存储的端点

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/.well-known/smart-configuration
```

这将以 JSON blob 形式返回 HealthLake 数据存储的发现文档。在其中，您将找到authorization_endpoint和token_endpoint以及 HealthLake 数据存储中定义的规格和功能。

```
{
  "authorization_endpoint": "https://oidc.example.com/authorize",
  "token_endpoint": "https://oidc.example.com/oauth/token",
  "capabilities": [
    "launch-ehr",
    "client-public"
  ]
}
```

URLs成功启动客户端应用程序所必需的

- 授权端点：授权客户端应用程序或用户URL所需的端点。
- 令牌端点：客户端应用程序用于与其通信的授权服务器的端点。

在SMART已启用的 HealthLake 数据存储上FHIRRESTAPI发出请求

您可以在启用了FHIR开启 HealthLake 的数据存储SMART上发出FHIRRESTAPI请求。以下示例显示了来自客户端应用程序的请求，其中包含授权标头，以及 Lambda 应如何解码响应。JWT在客户端应用程序请求获得授权和身份验证后，它必须收到来自授权服务器的持有者令牌。在FHIR启用了开启 HealthLake 的数据存储上发送FHIRRESTAPI请求时，请在授权标头中使用不记名令牌。SMART

```
GET https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/
Patient/[ID]
Authorization: Bearer auth-server-provided-bearer-token
```

由于在授权标头中找到了不记名令牌且未检测到 AWS IAM身份，因此会 HealthLake 调用创建FHIR启用 HealthLake 数据存储时指定的 Lambda 函数。SMART当您的 Lambda 函数成功解码令牌后，以下是发送到的示例响应。 HealthLake

```
{
  "authPayload": {
    "iss": "https://authorization-server-endpoint/oauth2/token", # The issuer
 identifier of the authorization server
    "aud": "https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/
r4/", # Required, data store endpoint
    "iat": 1677115637,  # Identifies the time at which the token was issued
    "nbf": 1677115637,  # Required, the earliest time the JWT would be valid
    "exp": 1997877061,  # Required, the time at which the JWT is no longer valid
    "isAuthorized": "true",  # Required, boolean indicating the request has been
 authorized
    "uid": "100101",  # Unique identifier returned by the auth server
    "scope": "system/*.*" # Required, the scope of the request
  },
  "iamRoleARN": "iam-role-arn" #Required, IAM role to complete the request
}
```

设置实现不合FHIR规的数据存储所需的资源 SMART

本主题介绍您需要在 AWS 账户外部配置的资源 HealthLake、创建FHIR未SMART启用的 HealthLake 数据存储，以及FHIR客户端应用程序如何与授权服务器和 HealthLake 数据存储进行交互。SMART

此工作流程中的步骤定义了如何SMART处理FHIR请求以及成功处理请求所需的资源的基本步骤。

在 SMART “FHIR应请求” 流程中，三个应用程序协同工作：

- 最终用户：通常，患者或临床医生SMART在FHIR应用程序上使用第三方访问数据存储中的 HealthLake 数据。
- SMARTon A FHIR pplication（称为客户端应用程序）：想要访问在数据存储中找到 HealthLake 的数据的应用程序。
- 授权服务器：符合 OpenID Connect 标准的服务器，能够对用户进行身份验证并颁发访问令牌。
- HealthLake 数据存储：一种FHIR已SMART启用的 HealthLake 数据存储，它使用 Lambda 函数来响应提供不记名令牌的FHIRREST请求。

要使这些应用程序协同工作，您需要创建以下资源。

我们建议您在设置授权服务器、在授权服务器SMART上定义必要的作用域并创建处理令牌自省的 AWS Lambda 函数之后创建FHIR启用 HealthLake 数据存储。

1. 设置授权服务器端点-授权服务器

要使用 SMART on fr FHIR amework，你需要设置一个可以验证在数据存储上发出的FHIRREST请求的第三方授权服务器。要了解有关设置可与之配合使用的授权服务器端点的更多信息 HealthLake，请参阅SMARTon 的身份验证要求 FHIR。

2. 定义范围以控制谁可以访问授权服务器上 HealthLake 数据存储中的哪些数据-授权服务器

SMARTon fr FHIR amew OAuth ork 使用作用域来确定经过身份验证的请求可以访问哪些FHIR资源以及访问的范围。定义作用域是一种针对最低权限进行设计的方法。要详细了解由 SMART on framewor FHIR k 定义并由其支持的范围， HealthLake 请参阅SMART在FHIROAuth作用域上受支持 HealthLake。

3. 设置一个能够执行代币内省的 AWS Lambda 功能——你的账户 AWS

客户端应用程序在FHIR未启用的数据存储SMART上发送的FHIRREST请求将包含 JSON Web 令牌 (JWT)。要详细了解如何设置能够对其进行解码和验证的 Lambda 函数，请参阅解码 a。JWT

4. 创建FHIR已SMART启用的 HealthLake 数据存储 — 您的 AWS 账户

要创建FHIR HealthLake 数据SMART存储，您需要提供一个`IdentityProviderConfiguration`。要了解有关 C reateFHIRDatastore 请求中必需`IdentityProviderConfiguration`参数的更多信息，请参阅创建FHIR已SMART启用的 HealthLake 数据存储。

客户端应用程序如何启动并从FHIR启用后的数据存储中请求 HealthLake 数据 SMART

本节说明客户端应用程序如何在 SMART on FHIR 上下文中启动，以及如何在 HealthLake 数据存储上成功发出FHIRREST请求。

1. 客户端应用程序向众所周知的统一资源标识符**GET**发出请求

SMART已启用的客户端应用程序需要发出GET请求以查找 HealthLake 数据存储的授权端点。这是通过众所周知的统一资源标识符 (URI) 请求完成的。要了解有关此内容的更多信息，请参阅获取FHIR已启用 HealthLake 数据存储的发现文档。SMART

2. 请求访问权限和范围

客户端应用程序使用授权服务器的授权端点，以便用户可以登录。此过程对用户进行身份验证。作用域用于定义客户端应用程序可以访问 HealthLake 数据存储中的哪些FHIR资源。要了解有关定义范围的更多信息，请参阅SMART在FHIROAuth作用域上受支持 HealthLake。

3. 访问令牌

现在，用户已通过身份验证，客户端应用程序将收到来自授权服务器的JWT访问令牌。此令牌是在客户端应用程序向发送FHIRREST请求时提供的 HealthLake。要了解有关如何使用 Lambda 函数解码的更多信息，请参阅。JWT 执行令牌验证

4. 在FHIR已启用的 HealthLake 数据存储SMART上FHIRREST发出请求

现在，客户端应用程序可以使用授权服务器提供的访问令牌向 HealthLake 数据存储端点发送FHIRREST请求。要查看示例FHIRREST请求，请参阅在SMART已启用的 HealthLake 数据存储上FHIRRESTAPI发出请求。

5. 验证JWT访问令牌

要验证FHIRREST请求中发送的访问令牌，请使用 Lambda 函数。要了解如何创建可执行令牌内省的 Lambda 函数，请参阅。创建 AWS Lambda 函数

使用基于资源类型的自然语言处理 (NLP) 的自动FHIR DocumentReference 资源生成 AWS HealthLake

> **Note**
>
> 2023 年 2 月 20 日之后，默认情况下， HealthLake 数据存储不使用集成的自然语言处理 (NLP)。如果您有兴趣在数据存储上启用此功能，请参阅 “故障排除” 一章如何开启 HealthLake 集成的自然语言处理功能？中的。
>
> 如果您开启了 Amazon Comprehend Medical NLP 的集成版，那么当您创建DocumentReference或更新资源时，您的账户将产生费用。 AWS 如需了解更多详情，请参阅AWS HealthLake 定价。
>
> Amazon Comprehend Medical 不在亚太地区（孟买）上市。 HealthLake 在亚太地区（孟买）地区创建的数据存储不支持集成的自然语言处理 (NLP)。

HealthLake 使用 Amazon Comprehend Medical 自动为您提供集成的自然语言处理 (NLP)，用于处理存储在资源类型中的数据的非结构化数据。DocumentReference为此， HealthLake 请致电亚马逊 Comprehend M DetectEntities-V2 edical 和运营部门。InferICD10-CM InferRxNorm API结果将作为扩展自动附加到DocumentReference资源中。当 Amazon Comprehend M API edical 操作检测到、DIAGNOSIS和的特征SIGN时SYMPTOM，会自动生成资源Linkage类型。新的条件和观测资源由标识为SIGNSYMPTOM、或特征的实体创建DIAGNOSIS，它们通过此链接资源链接到源文档。

对于集成版生成的资源NLP，您可以提出GET请求，但不支持搜索这些新资源。

要详细了解如何使用与 Athena HealthLake 的集成来搜索这些扩展，请参阅。使用查询您的 HealthLake 数据存储 SQL

目录

亚马逊 Comprehend Medical 是如何与之整合的 HealthLake

HealthLake 使用 Amazon Comprehend Medical 推断在`DocumentReference`资源类型中找到的数据。Amazon Comprehend Medic API al 的*`DetectEntities-V2`*运营*`InferICD10-CM`*，*`InferRxNorm`*并将疾病检测为特征。每项操作都提供了不同的见解。

> ⚠ 语言支持
>
> Amazon Comprehend M API edical 的运营仅检测英语文本中的医疗实体。

- DetectEntities-V2：检查各种医疗实体的临床文本，并返回有关它们的特定信息，例如实体类别、位置和置信度分数。
- 推断 ICD1 0-CM：以实体形式检测患者记录中的医疗状况，并将这些实体与经世界卫生组织授权的国家卫生统计中心 ICD -10-CM知识库中的标准化概念标识符关联起来（ ）。CDC WHO
- InferRxNorm: 将药物检测为患者记录中列出的实体，并将其与美国国家医学图书馆 RxNorm 数据库中的标准化概念标识符关联起来。

每个API操作的支持特征是`SIGNSYMPTOM`、和`DIAGNOSIS`。如果检测到特征，则会将它们作为FHIR兼容的扩展添加到 HealthLake 数据存储中的不同位置。

添加扩展程序的位置。

- `DocumentReference`：Amazon Comprehend Medic API al 操作的结果将作为`extension`一个添加到资源类型中的每个文档中。 DocumentReference 扩展的结果分为两组。你可以根据它们在结果中找到它们URL。
  - `http://healthlake.amazonaws.com/system-generated-resources/`
    - 这些是由创建或添加的资源类型 HealthLake。
  - `http://healthlake.amazonaws.com/aws-cm/`
    - 将 Amazon Comprehend Medic API al 操作的原始输出添加到您的数据存储中。 HealthLake
- `Linkage`：此资源类型要么是由于集成而添加的，要么是创建的NLP。对特定项的`GET`请求`Linkage`会返回链接资源的列表。要确定是否由添加`Linkage`了 HealthLake，请查找添加的`"tag": [{"display": "SYSTEM_GENERATED"}]`键值对。要了解有关 Linkage FHIR 规范的更多信息，请参阅FHIR文档索引中的资源类型：链接。
- FHIR由亚马逊 Comprehend Medical 操作生成的资源类型。API

- Observation: 当特征为或时，添加了 Amazon Comprehend M API edical DetectEntities 操作的结果-V2 和 ICD1 Infer 0-CM。SIGN SYMPTOM
- Condition: 有亚马逊 Comprehend Medic API al DetectEntities 运营的结果——V2 和 ICD1 Infer 0-CM 当特征存在时。DIAGNOSIS
- MedicationStatement: 还有亚马逊 Comprehend Medic API al 的运营结果。 InferRxNorm

与FHIRRESTAPI运营集成

默认情况下，Amazon Comprehend Medic API al 操作检测到的特征在提出请求时不会返回。GET

要查看这些资源类型的集成NLP操作的结果，必须指定已知ID。

- Linkage
- Observation
- Condition
- MedicationStatement

只有在已知指定ID内容包含 Amazon Comprehend Medical NLP 操作结果的GET请求时，才能获得 DocumentReference 资源类型之外的整合操作结果。API

Amazon Comprehend Medical 运营如何整API合到的示例 HealthLake

示例 1：将患者记录导入 HealthLake 数据存储

以下是基于患者与医疗专业人员的接触的临床记录示例。

> **合成数据**
>
> 此示例中的文本是合成内容，不包含个人健康信息 (PHI)。

```
1991-08-31

# Chief Complaint
- Headache
- Sinus Pain
- Nasal Congestion
- Sore Throat
```

```
- Pain with Bright Lights
- Nasal Discharge
- Cough


# History of Present Illness
Jerónimo599
 is a 4 month-old non-hispanic white male.

# Social History
 Patient has never smoked.


Patient comes from a middle socioeconomic background.

Patient currently has Aetna.

# Allergies
No Known Allergies.

# Medications
No Active Medications.

# Assessment and Plan
Patient is presenting with bee venom (substance), mold (organism), house dust
 mite (organism), animal dander (substance), grass pollen (substance), tree pollen
 (substance), lisinopril, sulfamethoxazole / trimethoprim, fish (substance).

## Plan

The patient was prescribed the following medications:
- astemizole 10 mg oral tablet
- nda020800 0.3 ml epinephrine 1 mg/ml auto-injector
The patient was placed on a careplan:
- self-care interventions (procedure)
```

提醒一下，这些信息在资源中以 base64 格式编码。 DocumentReference 当本文档被收录 HealthLake 并且 Amazon Comp API rehend Medical 操作完成后，要查看结果，您可以从请求资源类型GET开始。DocumentReference

```
 GET https://https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/
r4/eeb8005725ae22b35b4edbdc68cf2dfd/r4/DocumentReference
```

当 Amazon Comprehend Medic API al 操作成功后，请在链接到以下内容的链接中查找这些键值对 `extension "url": "http://healthlake.amazonaws.com/aws-cm/"`

```
{
  "url": "http://healthlake.amazonaws.com/aws-cm/status/",
  "valueString": "SUCCESS"
 },
 {
  "url": "http://healthlake.amazonaws.com/aws-cm/message/",
  "valueString": "The Amazon HealthLake integrated medical NLP operation was
 successful."
 }
```

以下选项卡显示了如何根据资源类型在 HealthLake 数据存储中报告摄取的医疗记录。

DocumentReference

要查看单个DocumentReference资源类型的结果，请在提供特定资源的地方GET发出请求。id

```
GET https://https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/eeb8005725ae22b35b4edbdc68cf2dfd/r4/DocumentReference/0e938f03-da7f-4178-acd8-eea9586c46ed
```

成功后，您将获得一个200HTTP响应代码和以下响JSON应（为清楚起见，该响应已被截断）。

这是http://healthlake.amazonaws.com/system-generated-resources/部分。你可以看到已经添加Linkage/*e366d29f-2c22-4c19-866e-09603937935a*了一个新的。您还可以查看在哪些地方向特定Observation和Condition资源类型添加 HealthLake 了基于推断的结果。

要查看这些资源类型是如何修改的，请选择相关选项卡。

```
{
  "extension": [
   {
    "url": "http://healthlake.amazonaws.com/linkage",
    "valueReference": {
     "reference": "Linkage/e366d29f-2c22-4c19-866e-09603937935a"
    }
   },
   {
    "url": "http://healthlake.amazonaws.com/nlp-entity",
```

```
    "valueReference": {
     "reference": "Observation/c6e0a3ff-7a17-4d8b-bfd0-d02d7da090c5"
    }
   },
   {
    "url": "http://healthlake.amazonaws.com/nlp-entity",
    "valueReference": {
     "reference": "Condition/0854e1f3-894d-448e-a8d9-3af5b9902baf"
    }
   }
  ],
  "url": "http://healthlake.amazonaws.com/system-generated-resources/"
 }
```

Linkage

要查看单个Linkage资源类型的结果，请在提供特定资源的地方GET发出请求。ID

```
GET https://https://healthlake.your-region.amazonaws.com/
datastore/your-datastore-id/r4/eeb8005725ae22b35b4edbdc68cf2dfd/r4/
Linkage/e366d29f-2c22-4c19-866e-09603937935a
```

成功后，你会得到一个200HTTP响应代码，以及以下被截断JSON的响应。

响应包含item元素。其中，键值对"type": "source"表示用于修改的特定DocumentReference条目，Condition并Observations列在"type": "alternate"键值对下。

您还可以看到*meta*元素和相应的键值对*"tag": [{"display": "SYSTEM_GENERATED"}]*，表示这些资源是由创建的。 HealthLake

```
{
  "resourceType": "Linkage",
  "id": "e366d29f-2c22-4c19-866e-09603937935a",
  "active": true,
  "item":
  [
   {
    "type": "alternate",
    "resource": {
     "reference": "Observation/c6e0a3ff-7a17-4d8b-bfd0-d02d7da090c5",
     "type": "Observation"
```

```
    }
   },
   {
    "type": "alternate",
    "resource": {
     "reference": "Condition/9d5c1ef6-f822-4faf-b55f-7c70f2a4aa8d",
     "type": "Condition"
    }
   },
   {
    "type": "source",
    "resource": {
     "reference": "DocumentReference/0e938f03-da7f-4178-acd8-eea9586c46ed",
     "type": "DocumentReference"
    }
   }
  ],
  "meta": {
   "lastUpdated": "2022-10-21T19:38:31.327Z",
   "tag": [{
    "display": "SYSTEM_GENERATED"
   }]
  }
 }
```

Resource type: Observation

要查看单个Observation资源类型的结果，请在提供特定资源的地方GET发出请求。ID

```
GET https://https://healthlake.your-region.amazonaws.com/
datastore/your-datastore-id/r4/eeb8005725ae22b35b4edbdc68cf2dfd/r4/
Observation/e366d29f-2c22-4c19-866e-09603937935a
```

Amazon Comprehend Medic API al 的运营结果已修改为以下内容：、和。code meta modifierExtension

code

类型为的元素CodeableConcept。要了解更多信息，请参阅FHIR文档索引CodeableConcept中的。

HealthLake 附加以下三个键值对。

- "system": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/": 其中URL指的是一项特定的亚马逊 Comprehend Medical 业务。API在本例中，推断 ICD1 0CM。
- "code": "A52.06": 标识疾病控制中心知识库中概念的 ICD -10-CM代码在哪里A52.06。
- "display": "Other syphilitic heart involvement": 本体中"Other syphilitic heart involvement"对 ICD -10-CM代码的详细描述在哪里。

以下被截断的JSON响应仅包含元素。code

```
"code": {
  "coding":
  [
    {
   "system": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/",
   "code": "A52.06",
   "display": "Other syphilitic heart involvement"
    }
  ],
  "text": "Other syphilitic heart involvement"
 }
```

要了解模型对分配的 ICD -10-CM 代码正确性的置信度，请使用元素。modifierExtension

meta

该meta元素包含的元数据表明该code元素是否包含由 Amazon Comprehend Medical 操作添加的详细信息。API

以下被截断的JSON响应仅包含元素。meta

```
"meta": {
  "lastUpdated": "2022-10-21T19:38:30.879Z",
  "tag": [{
   "display": "SYSTEM_GENERATED"
  }]
 }
```

modifierExtension

该modifierExtension元素包含有关code元素中已分配代码的可信度等级的更多详细信息。它还具有键值对，这些键值对提供了指向 DocumentReference 用于生成结果的原始值和相关的 Linkage 资源类型的链接。

对于添加的每个coding元素，您都会看到一个entity-score和一个entity-Concept-Score被添加到中modifierExtension。对于键值对中的每个值，您会看到一个分数。因为entity-score，该分数是 Amazon Comprehend Medical 对检测准确性的可信度。因为entity-Concept-Score，该分数是Amazon Comprehend Medical对该实体与-10厘米概念准确关联的信心程度。ICD

以下被截断的JSON响应仅包含元素。modifierExtension

```
"modifierExtension": [{
   "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/aws-cm-icd10-entity-score",
   "valueDecimal": 0.45005733
  },
  {
   "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/aws-cm-icd10-entity-Concept-Score",
   "valueDecimal": 0.1111792
  },
  {
   "url": "http://healthlake.amazonaws.com/system-generated-linkage",
   "valueReference": {
    "reference": "Linkage/e366d29f-2c22-4c19-866e-09603937935a"
   }
  },
  {
   "url": "http://healthlake.amazonaws.com/source-document-reference",
   "valueReference": {
    "reference": "DocumentReference/0e938f03-da7f-4178-acd8-eea9586c46ed"
   }
  }
 ]
```

完整JSON回应

```
{
  "subject": {
   "reference": "Patient/0679b7b7-937d-488a-b48d-6315b8e7003b"
  },
  "resourceType": "Observation",
  "status": "unknown",
  "code": {
   "coding": [{
```

```
    "system": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/",
    "code": "A52.06",
    "display": "Other syphilitic heart involvement"
   }],
   "text": "Other syphilitic heart involvement"
  },
  "meta": {
   "lastUpdated": "2022-10-21T19:38:30.879Z",
   "tag": [{
    "display": "SYSTEM_GENERATED"
   }]
  },
  "modifierExtension": [{
    "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/aws-cm-icd10-entity-
score",
    "valueDecimal": 0.45005733
   },
   {
    "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/aws-cm-icd10-entity-
Concept-Score",
    "valueDecimal": 0.1111792
   },
   {
    "url": "http://healthlake.amazonaws.com/system-generated-linkage",
    "valueReference": {
     "reference": "Linkage/e366d29f-2c22-4c19-866e-09603937935a"
    }
   },
   {
    "url": "http://healthlake.amazonaws.com/source-document-reference",
    "valueReference": {
     "reference": "DocumentReference/0e938f03-da7f-4178-acd8-eea9586c46ed"
    }
   }
  ],
  "id": "7e88c7c5-21a5-4dd7-8fc2-a02474fba583"
 }
```


Condition

要查看单个Condition资源类型的结果，请在提供特定资源的地方GET发出请求。ID

```
GET https://https://healthlake.your-region.amazonaws.com/datastore/your-
datastore-id/r4/eeb8005725ae22b35b4edbdc68cf2dfd/r4/Condition/b06d343d-
ddb8-4f36-82cb-853fcd434dfd
```

Amazon Comprehend Medic API al 的运营结果已修改为以下内容：、和。code meta modifierExtension

code

类型为的元素CodeableConcept。要了解更多信息，请参阅FHIR文档索引CodeableConcept中的。

HealthLake 附加以下三个键值对。

- "system": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/": 其中URL指的是一项特定的亚马逊 Comprehend Medical 业务。API在本例中，推断 ICD1 0CM。
- "code": "I70.0": 标识疾病控制中心知识库中概念的 ICD -10-CM代码在哪里A52.06。
- "display": "Atherosclerosis of aorta": 本体中"Other syphilitic heart involvement"对 ICD -10-CM代码的详细描述在哪里。

以下被截断的JSON响应仅包含元素。code

```
"code": {
  "coding":
  [
    {
   "system": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/",
   "code": "I70.0",
   "display": "Atherosclerosis of aorta"
    }
  ],
  "text": "Atherosclerosis of aorta"
 }
```

要了解模型对分配的 ICD -10-CM 代码正确性的置信度，请使用元素。modifierExtension

meta

该meta元素包含的元数据表明该code元素是否包含由 Amazon Comprehend Medical 操作添加的详细信息。API

以下被截断的JSON响应仅包含元素。meta

```
"meta": {
  "lastUpdated": "2022-10-21T19:38:30.877Z",
  "tag": [{
   "display": "SYSTEM_GENERATED"
  }]
 }
```

modifierExtension

该modifierExtension元素包含有关code元素中已分配代码的可信度等级的更多详细信息。它还具有键值对，这些键值对提供了指向 DocumentReference 用于生成结果的原始值和相关的 Linkage 资源类型的链接。

对于添加的每个coding元素，您都会看到一个entity-score和一个entity-Concept-Score被添加到中modifierExtension。对于键值对中的每个值，您会看到一个分数。因为entity-score，该分数是 Amazon Comprehend Medical 对检测准确性的可信度。因为entity-Concept-Score，该分数是Amazon Comprehend Medical对该实体与-10厘米概念准确关联的信心程度。ICD

以下被截断的JSON响应仅包含元素。modifierExtension

```
"modifierExtension": [{
   "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/aws-cm-icd10-entity-
score",
   "valueDecimal": 0.94417894
  },
  {
   "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/aws-cm-icd10-entity-
Concept-Score",
   "valueDecimal": 0.8458298
  },
  {
   "url": "http://healthlake.amazonaws.com/system-generated-linkage",
   "valueReference": {
    "reference": "Linkage/e366d29f-2c22-4c19-866e-09603937935a"
   }
  },
  {
   "url": "http://healthlake.amazonaws.com/source-document-reference",
   "valueReference": {
```

```
    "reference": "DocumentReference/0e938f03-da7f-4178-acd8-eea9586c46ed"
   }
  }
 ]
```

完整JSON回应

```
{
  "subject": {
   "reference": "Patient/0679b7b7-937d-488a-b48d-6315b8e7003b"
  },
  "resourceType": "Condition",
  "code": {
   "coding": [{
    "system": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/",
    "code": "I70.0",
    "display": "Atherosclerosis of aorta"
   }],
   "text": "Atherosclerosis of aorta"
  },
  "meta": {
   "lastUpdated": "2022-10-21T19:38:30.877Z",
   "tag": [{
    "display": "SYSTEM_GENERATED"
   }]
  },
  "modifierExtension": [{
    "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/aws-cm-icd10-entity-
score",
    "valueDecimal": 0.94417894
   },
   {
    "url": "http://healthlake.amazonaws.com/aws-cm/infer-icd10/aws-cm-icd10-entity-
Concept-Score",
    "valueDecimal": 0.8458298
   },
   {
    "url": "http://healthlake.amazonaws.com/system-generated-linkage",
    "valueReference": {
     "reference": "Linkage/e366d29f-2c22-4c19-866e-09603937935a"
    }
   },
   {
```

```
    "url": "http://healthlake.amazonaws.com/source-document-reference",
    "valueReference": {
     "reference": "DocumentReference/0e938f03-da7f-4178-acd8-eea9586c46ed"
    }
   }
  ],
  "id": "b06d343d-ddb8-4f36-82cb-853fcd434dfd"
 }
```

示例 2：**DocumentReference**包含 MedicationStatement 资源类型的 A

以下是根据患者与医疗专业人员的接触而编写的临床记录示例。

> ⚠ 合成数据
>
> 此示例中的文本是合成内容，不包含个人健康信息 (PHI)。

```
Tom is not prescribed Advil
```

以下选项卡显示了如何根据资源类型在 HealthLake 数据存储中报告摄取的医疗记录。

DocumentReference

要查看单个DocumentReference资源类型的结果，请在提供特定资源的地方GET发出请求。ID

```
GET https://https://healthlake.your-region.amazonaws.com/datastore/your-datastore-id/r4/eeb8005725ae22b35b4edbdc68cf2dfd/r4/DocumentReference/c549125d-a218-421f-b8bf-23614c5e796c
```

成功后，您将获得200HTTP响应代码和以下截断JSON的响应。

键值对表示其中的资源类型是由 Amazon Comprehen extension d Medical API 操作添加的。"url": "http://healthlake.amazonaws.com/system-generated-resources/"您可以看到新的Linkage资源类型和多个MedicationStatement资源。

```
"extension": [{
   "extension": [{
     "url": "http://healthlake.amazonaws.com/linkage",
     "valueReference": {
      "reference": "Linkage/394bb244-177b-4409-8657-26b20ed56dd7"
```

```
     }
    },
    {
     "url": "http://healthlake.amazonaws.com/nlp-entity",
     "valueReference": {
      "reference": "MedicationStatement/cbf6af10-b0b9-451c-bdde-99611e3498a8"
     }
    },
    {
     "url": "http://healthlake.amazonaws.com/nlp-entity",
     "valueReference": {
      "reference": "MedicationStatement/9a89b0d3-6681-45ca-9926-27951edce5c7"
     }
    },
    {
     "url": "http://healthlake.amazonaws.com/nlp-entity",
     "valueReference": {
      "reference": "MedicationStatement/4a01f6c8-5f3a-4122-80ab-405312f96aa2"
     }
    },
    {
     "url": "http://healthlake.amazonaws.com/nlp-entity",
     "valueReference": {
      "reference": "MedicationStatement/fbfb77d8-70cf-4579-b4c0-d6fe3c01656b"
     }
    },
    {
     "url": "http://healthlake.amazonaws.com/nlp-entity",
     "valueReference": {
      "reference": "MedicationStatement/1340c9ce-9c48-4bf9-9b2f-d0ab027f5e0b"
     }
    }
   ],
   "url": "http://healthlake.amazonaws.com/system-generated-resources/"
  }
```

Linkage

要查看单个Linkage资源类型的结果，请在提供特定资源的地方GET发出请求。ID

```
GET https://https://healthlake.your-region.amazonaws.com/
datastore/your-datastore-id/r4/eeb8005725ae22b35b4edbdc68cf2dfd/r4/
Linkage/394bb244-177b-4409-8657-26b20ed56dd7
```

成功后，您将获得200HTTP响应代码和以下JSON响应。

响应包含item元素。其中，键值对"type": "source"表示用于修改MedicationStatement资源DocumentReference类型的特定条目。

您还可以看到meta元素和相应的键值对"tag": [{"display": "SYSTEM_GENERATED"}]，表示这些资源是由创建的。 HealthLake


```
{
  "resourceType": "Linkage",
  "id": "394bb244-177b-4409-8657-26b20ed56dd7",
  "active": true,
  "item": [{
    "type": "alternate",
    "resource": {
     "reference": "MedicationStatement/cbf6af10-b0b9-451c-bdde-99611e3498a8",
     "type": "MedicationStatement"
    }
   },
   {
    "type": "alternate",
    "resource": {
     "reference": "MedicationStatement/9a89b0d3-6681-45ca-9926-27951edce5c7",
     "type": "MedicationStatement"
    }
   },
   {
    "type": "alternate",
    "resource": {
     "reference": "MedicationStatement/4a01f6c8-5f3a-4122-80ab-405312f96aa2",
     "type": "MedicationStatement"
    }
   },
   {
    "type": "alternate",
    "resource": {
     "reference": "MedicationStatement/fbfb77d8-70cf-4579-b4c0-d6fe3c01656b",
     "type": "MedicationStatement"
    }
   },
   {
    "type": "alternate",
    "resource": {
```

```
     "reference": "MedicationStatement/1340c9ce-9c48-4bf9-9b2f-d0ab027f5e0b",
     "type": "MedicationStatement"
    }
   },
   {
    "type": "source",
    "resource": {
     "reference": "DocumentReference/c549125d-a218-421f-b8bf-23614c5e796c",
     "type": "DocumentReference"
    }
   }
  ],
  "meta": {
   "lastUpdated": "2022-10-24T20:05:03.501Z",
   "tag": [{
    "display": "SYSTEM_GENERATED"
   }]
  }
 }
```

MedicationStatement

要查看单个MedicationStatement资源类型的结果，请在提供特定资源的地方GET发出请求。ID

```
GET https://https://healthlake.your-region.amazonaws.com/
datastore/your-datastore-id/r4/eeb8005725ae22b35b4edbdc68cf2dfd/r4/
MedicationStatement/9a89b0d3-6681-45ca-9926-27951edce5c7
```

MedicationStatement 资源类型是查找 Amazon Comprehend Medical 操作 InferRxNorm API结果的地方。将结果修改为以下要素：medicationCodeableConceptmeta、和modifierExtension。

medicationCodeableConcept

类型为的元素CodeableConcept。要了解更多信息，请参阅FHIR文档索引CodeableConcept中的。

HealthLake 附加以下三个键值对。

- "system": ""http://healthlake.amazonaws.com/aws-cm/infer-rxnorm/: 其中URL指的是一项特定的亚马逊 Comprehend Medical 业务。API在这种情况下， InferRxNorm.
- "code": "731533": 概 RxNorm 念 ID 在731533哪里，也称为 Rx CUI。

- "display": "ibuprofen 200 MG Oral Capsule [Advil]": RxNorm 概念ibuprofen 200 MG Oral Capsule [Advil]的描述在哪里。

以下被截断的JSON响应仅包含元素。MedicationStatement

```
"medicationCodeableConcept": {
  "coding": [
   {
    "system": "http://healthlake.amazonaws.com/aws-cm/infer-rxnorm/",
    "code": "731533",
    "display": "ibuprofen 200 MG Oral Capsule [Advil]"
   }
  ]
 }
```

meta

该meta元素包含的元数据表明该code元素是否包含由 Amazon Comprehend Medical 操作添加的详细信息。API

以下被截断的JSON响应仅包含元素。meta

```
"meta": {
  "lastUpdated": "2022-10-24T20:05:02.800Z",
  "tag": [
   {
    "display": "SYSTEM_GENERATED"
   }
  ]
 }
```

modifierExtension

该modifierExtension元素包含键值对，这些键值对提供了指向 DocumentReference 用于生成结果的原始元素的链接以及相关的 Linkage 资源类型。

```
"modifierExtension": [
   {
    "url": "http://healthlake.amazonaws.com/system-generated-linkage",
    "valueReference": {
```

```
      "reference": "Linkage/394bb244-177b-4409-8657-26b20ed56dd7"
     }
    },
    {
     "url": "http://healthlake.amazonaws.com/source-document-reference",
     "valueReference": {
      "reference": "DocumentReference/c549125d-a218-421f-b8bf-23614c5e796c"
     }
    }
]
```

搜索参数

下表列出了综合医疗NLP的可搜索属性。

搜索参数

搜索参数	查找以下项的匹配项
detectEntities-实体类别	CM 扩展中 DetectEntities 子扩展中的实体类别 AWS
detectEntities-实体文本	CM 扩展中 DetectEntities 子扩展中的实体文本 AWS
detectEntities-实体类型	CM 扩展中 DetectEntities 子扩展中的实体类型 AWS
detectEntities-实体分数	CM 扩展中 DetectEntities 子扩展中的实体分数 AWS
infer-icd10 cm-entity-text	CM 扩展中推断 ICD1 0CM 子扩展中的实体文本 AWS
infer-icd10 cm-entity-score	CM 扩展中推断 ICD1 0CM 子扩展中的实体分数 AWS
infer-icd10 cm-entity-concept-code	CM 扩展中的 Infer ICD1 0CM 子扩展中的实体概念代码 AWS
infer-icd10 cm-entity-concept-description	CM 扩展中的 Infer ICD1 0CM 子扩展中的实体概念描述 AWS
infer-icd10 cm-entity-concept-score	CM 扩展中的 Infer ICD1 0CM 子扩展中的实体概念分数 AWS
infer-rxnorm-entity-score	CM 扩展中 InferRxNorm 子扩展中的实体分数 AWS

搜索参数	查找以下项的匹配项
infer-rxnorm-entity-text	CM 扩展中 InferRxNorm 子扩展中的实体文本 AWS
infer-rxnorm-entity-concept-代码	CM 扩展中 InferRxNorm 子扩展中的实体概念代码 AWS
infer-rxnorm-entity-concept-描述	CM 扩展中 InferRxNorm 子扩展中的实体概念描述 AWS
infer-rxnorm-entity-concept-分数	CM 扩展中 InferRxNorm 子扩展中的实体概念分数 AWS

要匹配`EntityText`和`EntityCategory`属于同一实体的条件，请 HealthLake 提供特殊搜索。下表描述了中支持的特殊搜索参数 HealthLake。

搜索参数

搜索参数	返回的匹配项
detectEntities-entity-text-category	如果 DetectEntities 子扩展中至少有一个实体同时与和匹配。 entityText entityCategory
detectEntities-entity-type-score	如果 DetectEntities 子扩展中至少有一个实体同时与和匹配。 entityType entityScore
detectEntities-entity-text-score	如果 DetectEntities 子扩展中至少有一个实体同时与和匹配。 entityText entityScore
detectEntities-entity-text-type	如果 DetectEntities 子扩展中至少有一个实体同时与和匹配。 entityText entityType
detectEntities-entity-category-score	如果至少有一个实体同时与 entityCategory 和匹配 entityScore。
infer-icd10-cod cm-entity-text-concept e	如果 Infer ICD1 0CM 子扩展中至少有一个实体与匹配， entityText 并且该实体中至少有一个与代码相匹配 conceptCode 的实体。

搜索参数	返回的匹配项
infer-icd10-scor cm-entity-text-concept e	如果 Infer ICD1 0CM 子扩展中至少有一个实体与匹配， entityText 并且该实体中至少有一个与分数相匹配 conceptScore 的实体。
infer-icd10-concept-scor cm-entity-concept-description e	如果 Infer ICD1 0CM 子扩展中的实体中至少有一个概念与概念描述和. conceptScore
infer-rxnorm-entity-text-概念代码	如果 InferRxNorm 子扩展中至少有一个与匹配的实体， entityText 并且该实体中至少有一个 conceptCode 与代码相匹配的实体。
infer-rxnorm-entity-text-概念分数	如果 InferRxNorm 子扩展中至少有一个实体与匹配， entityText 并且该实体中至少有一个 conceptScore 与分数相匹配的实体。
infer-rxnorm-entity-concept-description-concept-score	如果 InferRxNorm 子扩展中的实体中至少有一个概念与概念描述和. conceptScore

安全性 AWS HealthLake

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：

- 云的安全性 AWS 负责保护在 AWS 云中运行 AWS 服务的基础架构。 AWS 还为您提供可以安全使用的服务。作为AWS 合规计划合规计划合规计划合的一部分，第三方审计师定期测试和验证我们安全的有效性。要了解适用的合规计划 HealthLake，请参阅按合规计划划分的范围内的AWSAWS服务按合规计划。
- 云端安全-您的责任由您使用的 AWS 服务决定。您还需要对其他因素负责，包括您的数据的敏感性、您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 HealthLake。以下主题向您介绍如何进行配置 HealthLake 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服务来帮助您监控和保护您的 HealthLake 资源。

主题

- 中的数据保护 AWS HealthLake
- 在 for 处REST加密 AWS HealthLake
- 正在对以下对象进行加密 AWS HealthLake
- 的身份和访问管理 AWS HealthLake
- 使用 AWS CloudTrail记录 AWS HealthLake API 调用
- 的合规性验证 AWS HealthLake
- 韧性在 AWS HealthLake
- AWS HealthLake 中的基础架构安全性
- AWS HealthLake 中的安全最佳实践

中的数据保护 AWS HealthLake

分 AWS 担责任模型适用于中的数据保护 AWS HealthLake。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FAQ。 有关欧洲数据保护的信息，请参阅责任AWS 共担模型和AWS安全GDPR博客上的博客文章。

出于数据保护目的，我们建议您保护 AWS 账户 凭据并使用 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 AWS 资源通信。我们需要 TLS 1.2，建议使用 TLS 1.3。
- 使用API进行设置和用户活动记录 AWS CloudTrail。有关使用 CloudTrail 跟踪捕获 AWS 活动的信息，请参阅《AWS CloudTrail 用户指南》中的使用跟 CloudTrail 踪。
- 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 通过命令行界面或访问时需要 FIPS 140-3 经过验证的加密模块API，请使用端点。FIPS有关可用FIPS端点的更多信息，请参阅联邦信息处理标准 (FIPS) 140-3。

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 HealthLake 或以其他 AWS 服务 方式使用控制台时API、 AWS CLI、或 AWS SDKs。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URL向外部服务器提供，我们强烈建议您不要在中包含凭据信息，URL以验证您对该服务器的请求。

在 for 处REST加密 AWS HealthLake

HealthLake 默认提供加密，以使用服务拥有的AWS密钥管理服务 (AWSKMS) 密钥保护敏感的静态客户数据。还支持客户管理的KMS密钥，并且是从数据存储中导入和导出文件所必需的。要了解有关客户管理KMS密钥的更多信息，请参阅 Amazon 密钥管理服务。在创建数据存储时，客户可以选择AWS自有KMS密KMS钥或客户管理的密钥。创建数据存储后，无法更改加密配置。如果数据存储使用的是AWS自有KMS密钥，则该密钥将表示为， AWS_OWNED_KMS_KEY 并且您将看不到用于静态加密的特定密钥。

AWS拥有的KMS密钥

HealthLake 默认使用这些密钥自动加密潜在的敏感信息，例如个人身份信息或静态私人健康信息 (PHI) 数据。 AWS拥有的KMS密钥不会存储在您的账户中。它们是AWS拥有和管理的KMS密钥集合的一部

分，可在多个AWS账户中使用。 AWS服务可以使用AWS自有KMS密钥来保护您的数据。您无法查看、管理、使用AWS自有KMS密钥或审核其使用情况。但是无需执行任何工作或更改任何计划即可保护用于加密数据的密钥。

如果您使用AWS自有KMS密钥，则无需支付月费或使用费，也不会计入您账户的AWSKMS配额。有关更多信息，请参阅AWS自有密钥。

客户托管 KMS 密钥

HealthLake 支持使用您创建、拥有和管理的对称客户托管KMS密钥，在现有的AWS自有加密基础上添加第二层加密。由于您可以完全控制这层加密，因此可以执行以下任务：

- 制定和维护关键政策、IAM政策和补助金
- 轮换密钥加密材料
- 启用和禁用密钥政策
- 添加标签
- 创建密钥别名
- 安排密钥删除

您还可以使用 CloudTrail 来跟踪代表您 HealthLake 发送 AWS KMS 的请求。需 AWS KMS 支付额外费用。有关更多信息，请参阅客户拥有的密钥。

创建客户托管密钥

您可以使用管理控制台创建对称客户托管密钥，或者。AWS AWS KMS APIs

按照《密钥管理服务开发人员指南》中创建对称客户托管AWS密钥的步骤进行操作。

密钥策略控制对客户托管密钥的访问。每个客户托管式密钥必须只有一个密钥政策，其中包含确定谁可以使用密钥以及如何使用密钥的声明。创建客户托管式密钥时，可以指定密钥政策。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权限。

要将客户托管密钥用于您的 HealthLake 资源，必须在密钥策略中允许 kms: CreateGrant 操作。这会向客户托管密钥添加授权，该密钥控制对指定KMS密钥的访问权限，从而允许用户访问所需的 kms: grant 操作。 HealthLake 有关更多信息，请参阅使用授权。

要将客户托管KMS密钥 HealthLake 用于您的资源，密钥策略中必须允许以下API操作：

- kms：向特定的客户托管KMS密钥CreateGrant 添加授权，该密钥允许访问授权操作。

- km DescribeKey s：提供验证密钥所需的客户托管密钥详细信息。这是所有操作所必需的。
- kms：GenerateDataKey 为所有写入操作提供对静态加密资源的访问权限。
- KMS: Decrypt 提供对加密资源的读取或搜索操作的访问权限。

以下是一个策略声明示例，允许用户创建由 AWS HealthLake 该密钥加密的数据存储并与之交互：

```
"Statement": [
    {
        "Sid": "Allow access to create data stores and do CRUD/search in AWS
 HealthLake",
        "Effect": "Allow",
        "Principal": {
            "AWS": "arn:aws:iam::111122223333:HealthLakeFullAccessRole"
        },
        "Action": [
            "kms:DescribeKey",
            "kms:CreateGrant",
            "kms:GenerateDataKey",
            "kms:Decrypt"
        ],
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "kms:ViaService": "healthlake.amazonaws.com",
                "kms:CallerAccount": "111122223333"
            }
        }
    }
]
```

使用客户托管KMS密钥所需的IAM权限

使用客户托管密钥创建启用 AWS KMS 加密的数据存储时，KMS密钥策略和创建 HealthLake 数据存储的用户或角色的IAM策略都需要权限。

您可以使用 k ms: ViaService 条件密钥将KMS密钥的使用限制为仅来自 HealthLake的请求。

有关密钥策略的更多信息，请参阅《密AWS钥管理服务开发人员指南》中的启用IAM策略。

创建存储库的IAM用户、IAM角色或AWS账户必须具有 kms:CreateGrant、kms: GenerateDataKey 和 kms:DescribeKey 权限以及必要的 HealthLake权限。

如何在中 HealthLake 使用补助金 AWS KMS

HealthLake 需要获得授权才能使用您的客户托管KMS密钥。当您创建使用客户托管KMS密钥加密的数据存储时， HealthLake 会通过向发送CreateGrant请求来代表您创建授权AWSKMS。中的授权AWSKMS用于授予对客户账户中KMS密钥的 HealthLake 访问权限。

代表您 HealthLake 创建的赠款不应被撤销或撤销。如果您撤销或取消授予您账户中AWSKMS密钥使用 HealthLake 权限的授权，则 HealthLake 无法访问这些数据、加密推送到数据存储的新FHIR资源或在提取时对其进行解密。当您撤销或撤销的授予时 HealthLake，更改会立即生效。要撤消访问权限，应删除数据存储而不是撤消授权。删除数据存储后， HealthLake 将代表您停用授权。

监控 HealthLake 的加密密钥

使用客户托管KMS密钥时，您可以使用 CloudTrail 来跟踪 AWS KMS 代表您 HealthLake 发送的请求。日志中的日志条目在 userAgent 字段中显示 health CloudTrail lake.amazonaws.com，以明确区分由发出的请求。 HealthLake

以下示例是 CreateGrant、 GenerateDataKey、Decrypt 和 DescribeKey 监视 AWS KMS 操作 CloudTrail 的事件，这些操作被调用 HealthLake 以访问由您的客户托管密钥加密的数据。

以下内容显示了 CreateGrant 如何使用允许 HealthLake 访问客户提供的KMS密钥，从而 HealthLake 允许使用该KMS密钥对所有静态客户数据进行加密。

用户无需创建自己的授权。 HealthLake 通过向发送 CreateGrant 请求来代表您创建授权AWSKMS。中的授权 AWS KMS 用于授予对客户账户中 AWS KMS 密钥的 HealthLake 访问权限。

```
        {
"eventVersion": "1.08",
"userIdentity": {
    "type": "AssumedRole",
    "principalId": "EXAMPLEROLE:Sampleuser01",
    "arn": "arn:aws:sts::111122223333:assumed-role/Sampleuser01,
    "accountId": "111122223333",
    "accessKeyId": "EXAMPLEKEYID",
    "sessionContext": {
        "sessionIssuer": {
            "type": "Role",
            "principalId": "EXAMPLEROLE",
```

```
                "arn": "arn:aws:iam::111122223333:role/Sampleuser01",
                "accountId": "111122223333",
                "userName": "Sampleuser01"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2021-06-30T19:33:37Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "healthlake.amazonaws.com"
    },
    "eventTime": "2021-06-30T20:31:15Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateGrant",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "healthlake.amazonaws.com",
    "userAgent": "healthlake.amazonaws.com",
    "requestParameters": {
        "operations": [
            "CreateGrant",
            "Decrypt",
            "DescribeKey",
            "Encrypt",
            "GenerateDataKey",
            "GenerateDataKeyWithoutPlaintext",
            "ReEncryptFrom",
            "ReEncryptTo",
            "RetireGrant"
        ],
        "granteePrincipal": "healthlake.us-east-1.amazonaws.com",
        "keyId": "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE_KEY_ARN",
        "retiringPrincipal": "healthlake.us-east-1.amazonaws.com"
    },
    "responseElements": {
        "grantId": "EXAMPLE_ID_01"
    },
    "requestID": "EXAMPLE_ID_02",
    "eventID": "EXAMPLE_ID_03",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
```

```
            "ARN": "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE_KEY_ARN"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

以下示例说明如何使用 GenerateDataKey 来确保用户在存储数据之前拥有加密数据的必要权限。

```
         {
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "EXAMPLEUSER",
        "arn": "arn:aws:sts::111122223333:assumed-role/Sampleuser01",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLEKEYID",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "EXAMPLEROLE",
                "arn": "arn:aws:iam::111122223333:role/Sampleuser01",
                "accountId": "111122223333",
                "userName": "Sampleuser01"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2021-06-30T21:17:06Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "healthlake.amazonaws.com"
    },
    "eventTime": "2021-06-30T21:17:37Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKey",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "healthlake.amazonaws.com",
    "userAgent": "healthlake.amazonaws.com",
```

```
    "requestParameters": {
        "keySpec": "AES_256",
        "keyId": "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE_KEY_ARN"
    },
    "responseElements": null,
    "requestID": "EXAMPLE_ID_01",
    "eventID": "EXAMPLE_ID_02",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE_KEY_ARN"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

以下示例显示了如何 HealthLake 调用 Decrypt 操作以使用存储的加密数据密钥来访问加密数据。

```
         {
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "EXAMPLEUSER",
        "arn": "arn:aws:sts::111122223333:assumed-role/Sampleuser01",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLEKEYID",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "EXAMPLEROLE",
                "arn": "arn:aws:iam::111122223333:role/Sampleuser01",
                "accountId": "111122223333",
                "userName": "Sampleuser01"
            },
            "webIdFederationData": {},
            "attributes": {
```

```
                "creationDate": "2021-06-30T21:17:06Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "healthlake.amazonaws.com"
    },
    "eventTime": "2021-06-30T21:21:59Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "healthlake.amazonaws.com",
    "userAgent": "healthlake.amazonaws.com",
    "requestParameters": {
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "keyId": "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE_KEY_ARN"
    },
    "responseElements": null,
    "requestID": "EXAMPLE_ID_01",
    "eventID": "EXAMPLE_ID_02",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE_KEY_ARN"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

以下示例显示了如何 HealthLake 使用该 DescribeKey 操作来验证 AWS KMS 客户拥有的 AWS KMS 密钥是否处于可用状态，以及如何帮助用户对其无法运行进行故障排除。

```
          {
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "EXAMPLEUSER",
```

```
        "arn": "arn:aws:sts::111122223333:assumed-role/Sampleuser01",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLEKEYID",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "EXAMPLEROLE",
                "arn": "arn:aws:iam::111122223333:role/Sampleuser01",
                "accountId": "111122223333",
                "userName": "Sampleuser01"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2021-07-01T18:36:14Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "healthlake.amazonaws.com"
    },
    "eventTime": "2021-07-01T18:36:36Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DescribeKey",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "healthlake.amazonaws.com",
    "userAgent": "healthlake.amazonaws.com",
    "requestParameters": {
        "keyId": "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE_KEY_ARN"
    },
    "responseElements": null,
    "requestID": "EXAMPLE_ID_01",
    "eventID": "EXAMPLE_ID_02",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE_KEY_ARN"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

了解更多

以下资源提供了有关静态数据加密的更多信息。

有关AWS密钥管理服务基本概念的更多信息，请参阅 AWS KMS 文档。

有关安全最佳实践的更多信息， AWS KMS 请参阅文档。

正在对以下对象进行加密 AWS HealthLake

AWS HealthLake 使用 TLS 1.2 对通过公共端点和后端服务传输的数据进行加密。

的身份和访问管理 AWS HealthLake

AWS Identity and Access Management (IAM) AWS 服务 可以帮助管理员安全地控制对 AWS 资源的访问权限。 IAM管理员控制谁可以通过身份验证（登录）和授权（拥有权限）使用 HealthLake 资源。IAM无需支付额外费用即可使用。 AWS 服务

主题

- 受众
- 使用身份进行身份验证
- 使用策略管理访问
- AWS HealthLake 如何使用 IAM
- 基于身份的策略示例 AWS HealthLake
- AWS 的托管策略 AWS HealthLake
- 对 AWS HealthLake 身份和访问进行故障排除

受众

你使用 AWS Identity and Access Management (IAM) 的方式会有所不同，具体取决于你所做的工作 HealthLake。

服务用户-如果您使用该 HealthLake 服务完成工作，则您的管理员会为您提供所需的凭证和权限。当你使用更多 HealthLake 功能来完成工作时，你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 HealthLake 中的特征，请参阅 对 AWS HealthLake 身份和访问进行故障排除。

服务管理员-如果您负责公司的 HealthLake 资源，则可能拥有完全访问权限 HealthLake。您的工作是确定您的服务用户应访问哪些 HealthLake 功能和资源。然后，您必须向 IAM 管理员提交请求，这样才能更改您的服务用户的权限。查看此页面的信息，了解 IAM 的基本概念。要详细了解贵公司如何IAM与配合使用 HealthLake，请参阅AWS HealthLake 如何使用 IAM。

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用于管理访问权限的策略HealthLake。要查看可在中使用的 HealthLake 基于身份的策略示例IAM，请参阅。基于身份的策略示例 AWS HealthLake

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 AWS 账户根用户、IAM用户身份或通过担任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身份中心）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。在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。当你使用联合访问 AWS 时，你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API请求AWS 签名版本 4。

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，请参阅用户指南中的多因素身份验证和AWS IAM Identity Center 用户指南IAM中的AWS 多因素身份验证。IAM

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。

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 C enter 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 Ident IAM ity 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身份中心的信息，请参阅什么是IAM身份中心？ 在《AWS IAM Identity Center 用户指南》中。

IAM 用户和组

IAM用户是您内部 AWS 账户 对个人或应用程序具有特定权限的身份。在可能的情况下，我们建议使用临时证书，而不是创建拥有密码和访问密钥等长期凭证的IAM用户。但是，如果您有需要IAM用户长期凭证的特定用例，我们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 是指定一个 IAM 用户集合的身份。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，您可以拥有一个名为的群组，IAMAdmins并授予该群组管理IAM资源的权限。

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，请参阅用户指南中的IAMIAM用户用例。

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，但未与特定人员关联。要在中临时扮IAM演角色 AWS Management Console，可以从用户切换到IAM角色（控制台）。您可以通过调用 AWS CLI 或 AWS API操作或使用自定义操作来代入角色URL。有关使用角色的方法的更多信息，请参阅《IAM用户指南》中的代入角色的方法。

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，请创建一个角色，并为该角色定义权限。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。有关用于联合身份验证的角

色的信息，请参阅《IAM用户指南》中的为第三方身份提供商（联合）创建角色。如果您使用 IAM Identity Center，则需要配置权限集。为了控制您的身份在进行身份验证后可以访问的内容，Ident IAM ity Center 会将权限集关联到中的IAM角色。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 用户或角色可以代入一个IAM角色，为特定任务临时获得不同的权限。
- 跨账户存取 - 您可以使用 IAM 角色允许其他账户中的某个人（可信任主体）访问您账户中的资源。角色是授予跨账户存取权限的主要方式。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IAM中的跨账户资源访问权限。
- 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，当您在服务中拨打电话时，该服务通常会在 Amazon 中运行应用程序EC2或在 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。当你使用某些服务时，你可能会执行一个操作，然后在不同的服务中启动另一个操作。 FAS使用调用委托人的权限 AWS 服务以及 AWS 服务 向下游服务发出请求的请求。 FAS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。有关提出FAS请求时的政策详情，请参阅转发访问会话。
  - 服务角色-服务IAM角色是服务代替您执行操作的角色。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》 AWS 服务中的创建角色以向委派权限。
  - 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 Amazon 上运行的应用程序 EC2 — 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。要为EC2实例分配 AWS 角色并使其可供其所有应用程序使用，您需要创建一个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角色向在 Amazon EC2 实例上运行的应用程序授予权限。

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策略文档结构和内容的更多信息，请参阅《IAM用户指南》中的JSON策略概述。

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 策略，用户可以代入这些角色。

IAM 策略定义操作的权限，无论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、或获取角色信息 AWS API。

基于身份的策略

基于身份的策略是可以附加到身份（例如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用户指南中的使用客户托管策略定义自定义IAM权限。

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。要了解如何在托管策略或内联策略之间进行选择，请参阅《IAM用户指南》中的在托管策略和内联策略之间进行选择。

基于资源的策略

基于资源的JSON策略是您附加到资源的策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 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IAM中使用 AWS 托管策略。

访问控制列表 (ACLs)

访问控制列表 (ACLs) 控制哪些委托人（账户成员、用户或角色）有权访问资源。 ACLs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 AWS WAF、和亚马逊VPC就是支持的服务示例ACLs。要了解更多信息ACLs，请参阅《亚马逊简单存储服务开发者指南》中的访问控制列表 (ACL) 概述。

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-权限边界是一项高级功能，您可以在其中设置基于身份的策略可以向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 (SCPs)-SCPs 是为中的组织或组织单位 (OU) 指定最大权限的JSON策略 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。如果您启用组织中的所有功能，则可以将服务控制策略 (SCPs) 应用于您的任何或所有帐户。对成员账户中的实体（包括每个实体）的权限进行了SCP限制 AWS 账户根用户。有关 Organization SCPs s 和的更多信息，请参阅《AWS Organizations 用户指南》中的服务控制策略。
- 资源控制策略 (RCPs) — RCPs 这些JSON策略可用于设置账户中资源的最大可用权限，而无需更新附加到您拥有的每项资源的IAM策略。这会RCP限制成员账户中资源的权限，并可能影响身份（包括身份）的有效权限 AWS 账户根用户，无论这些身份是否属于您的组织。有关 Organizations 的更多信息RCPs，包括 AWS 服务 该支持的列表RCPs，请参阅《AWS Organizations 用户指南》中的资源控制策略 (RCPs)。
- 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。

AWS HealthLake 如何使用 IAM

在使用管理IAM访问权限之前 HealthLake，请先了解哪些IAM功能可供使用 HealthLake。

IAM你可以使用的功能 AWS HealthLake

IAM 功能	HealthLake 支持
基于身份的策略	是
基于资源的策略	否
策略操作	是

IAM 功能	HealthLake 支持
策略资源	是
策略条件键	是
ACLs	否
ABAC（策略中的标签）	是
临时凭证	是
主体权限	是
服务角色	是
服务相关角色	否

要全面了解大多数IAM功能 HealthLake 以及其他 AWS 服务是如何使用的，请参阅《IAM用户指南》IAM中与之配合使用的AWS 服务。

基于身份的策略 AWS HealthLake

支持基于身份的策略：是

基于身份的策略是可以附加到身份（例如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用户指南中的使用客户托管策略定义自定义IAM权限。

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。要了解可以在JSON策略中使用的所有元素，请参阅IAM用户指南中的IAMJSON策略元素参考。

基于身份的策略示例 AWS HealthLake

要查看 HealthLake 基于身份的策略的示例，请参阅。基于身份的策略示例 AWS HealthLake

内部基于资源的政策 AWS HealthLake

支持基于资源的策略：否

基于资源的JSON策略是您附加到资源的策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 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》IAM中的跨账户资源访问权限。

的政策行动 AWS HealthLake

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`Action`元素描述了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。也有一些例外，例如没有匹配API操作的仅限权限的操作。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 HealthLake 操作列表，请参阅《服务授权参考》 AWS HealthLake中定义的操作。

正在执行的策略操作在操作前 HealthLake 使用以下前缀：

```
healthlake
```

要在单个语句中指定多项操作，请使用逗号将它们隔开。

```
"Action": [
      "healthlake:action1",
      "healthlake:action2"
         ]
```

要查看 HealthLake 基于身份的策略的示例，请参阅。基于身份的策略示例 AWS HealthLake

的政策资源 AWS HealthLake

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策略元素指定要应用操作的一个或多个对象。语句必须包含 `Resource` 或 `NotResource` 元素。最佳做法是，使用资源的 Amazon 资源名称 (ARN) 来指定资源。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 HealthLake 资源类型及其列表ARNs，请参阅《服务授权参考》 AWS HealthLake中定义的资源。要了解可用于指定每ARN种资源的操作，请参阅由定义的操作 AWS HealthLake。

要查看 HealthLake 基于身份的策略的示例，请参阅。基于身份的策略示例 AWS HealthLake

的策略条件密钥 AWS HealthLake

支持特定于服务的策略条件键：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，仅当 IAM 用户使用其 IAM 用户名进行标记时，您才可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 策略元素：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 HealthLake 条件密钥列表，请参阅《服务授权参考》 AWS HealthLake中的条件密钥。要了解可用于使用条件键的操作和资源，请参阅由定义的操作 AWS HealthLake。

要查看 HealthLake 基于身份的策略的示例，请参阅。基于身份的策略示例 AWS HealthLake

中的访问控制列表 (ACLs) AWS HealthLake

支持ACLs：否

访问控制列表 (ACLs) 控制哪些委托人（账户成员、用户或角色）有权访问资源。 ACLs与基于资源的策略类似，尽管它们不使用JSON策略文档格式。

基于属性的访问控制 () ABAC AWS HealthLake

支持ABAC（策略中的标签）：是

基于属性的访问控制 (ABAC) 是一种基于属性定义权限的授权策略。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。为实体和资源添加标签是的第一步。ABAC然后，您可以设计ABAC策略，允许在委托人的标签与他们尝试访问的资源上的标签匹配时进行操作。

ABAC在快速增长的环境中很有用，也有助于解决策略管理变得繁琐的情况。

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。

有关更多信息ABAC，请参阅《IAM用户指南》中的使用ABAC授权定义权限。要查看包含设置步骤的教程ABAC，请参阅IAM用户指南中的使用基于属性的访问控制 (ABAC)。

将临时证书与 AWS HealthLake

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。有关其他信息，包括哪些 AWS 服务 适用于临时证书 AWS 服务 ，请参阅《IAM用户指南》IAM中的 “适用于临时证书”。

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《用户指南》中的从IAM用户切换到IAM角色（控制台）。

您可以使用 AWS CLI 或手动创建临时证书 AWS API。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，请参阅中的临时安全证书IAM。

的跨服务主体权限 AWS HealthLake

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。当你使用某些服务时，你可能会执行一个操作，然后在不同的服务中启动另一个操作。 FAS使用调用委托人的权限 AWS 服务以及 AWS 服务 向下游服务发出请求的请求。 FAS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。有关提出FAS请求时的政策详情，请参阅转发访问会话。

AWS HealthLake 的服务角色

支持服务角色：是

服务角色是由一项服务代入、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》 AWS 服务中的创建角色以向委派权限。

有关服务角色和完全访问权限所需的内联策略的信息 AWS HealthLake，请参阅设置开始使用的权限 AWS HealthLake。

> **Warning**
>
> 更改服务角色的权限可能会中断 HealthLake 功能。只有在 HealthLake 提供操作指导时才编辑服务角色。

的服务相关角色 AWS HealthLake

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。

有关创建或管理服务相关角色的详细信息，请参阅使用 IAM 的AWS 服务。在表中查找服务相关角色列中包含 Yes 的服务。选择是链接以查看该服务的服务相关角色文档。

基于身份的策略示例 AWS HealthLake

默认情况下，用户和角色没有创建或修改 HealthLake 资源的权限。他们也无法使用 AWS Management Console、 AWS Command Line Interface (AWS CLI) 或来执行任务 AWS API。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 策略，用户可以代入这些角色。

要了解如何使用这些示例策略文档创建IAM基于身份的JSON策略，请参阅IAM用户指南中的创建IAM策略（控制台）。

有关由 HealthLake定义的操作和资源类型（包括每种资源类型的格式）的详细信息，请参阅《服务授权参考》 AWS HealthLake中的操作、资源和条件密钥。ARNs

主题

- 策略最佳实践
- 使用 AWS HealthLake 控制台
- 访问中的 AWS HealthLake 数据存储 Amazon Athena
- 允许用户查看他们自己的权限

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 HealthLake 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 托管式策略 或 工作职能的AWS 托管式策略。
- 应用最低权限许可 – 在使用 IAM 策略设置权限时，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 策略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，您可以编写一个策略条件来指定所有请求都必须使用发送SSL。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IAMJSON策略元素：条件。
- 使用 A IAM ccess Analyzer 验证您的IAM策略以确保权限的安全性和功能性 — A IAM ccess Analyzer 会验证新的和现有的策略，以便策略符合IAM策略语言 (JSON) 和IAM最佳实践。IAMAccess Analyzer 提供了 100 多项策略检查和可行的建议，可帮助您制定安全和实用的策略。有关更多信息，请参阅《IAM用户指南》中的使用 A IAM ccess Analyzer 验证策略。
- 需要多重身份验证 (MFA)-如果您的场景需要IAM用户或 root 用户 AWS 账户，请打开MFA以提高安全性。要要求MFA何时调用API操作，请在策略中添加MFA条件。有关更多信息，请参阅《IAM用户指南》MFA中的使用进行安全API访问。

有关 IAM 中最佳实践的更多信息，请参阅《IAM 用户指南》中的 IAM 中的安全最佳实践。

使用 AWS HealthLake 控制台

要访问 AWS HealthLake 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 HealthLake 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。

对于仅拨打 AWS CLI 或电话的用户，您无需为其设置最低控制台权限 AWS API。相反，只允许访问与他们尝试执行的API操作相匹配的操作。

要获得完全访问权限 HealthLake，请将以下策略附加到IAM用户或角色：`AmazonHealthLakeFullAccess`和`AWSLakeFormationDataAdmin`。您还需要附加作为服务角色的 HealthLake 内联策略。服务角色是由一项服务代入、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》 AWS 服务中的创建角色以向委派权限。有关创建所需服务角色的内联策略的信息，请参阅设置开始使用的权限 AWS HealthLake。您还必须使用 AWS Lake Formation 控制台或CLI将您的 HealthLake 管理员指定为 AWS Lake Formation 数据湖管理员。有关更多信息，请参阅 设置开始使用的权限 AWS HealthLake。

访问中的 AWS HealthLake 数据存储 Amazon Athena

如果要为用户和角色提供对中 HealthLake 数据存储的访问权限 Amazon Athena，请将以下IAM策略附加到该角色或用户：`AmazonAthenaFullAccess`和`AmazonS3FullAccess`。 `Select`并且还需要对由管理的表`Describe`具有权限 AWS Lake Formation。 AWS Lake Formation 表权限由 AWS Lake Formation 管理员在 AWS Lake Formation 控制台中或通过CLI。有关更多信息，请参阅 设置开始使用的权限 AWS HealthLake

允许用户查看他们自己的权限

此示例显示您可以如何创建策略，以便允许 IAM 用户查看附加到其用户身份的内联和托管策略。此策略包括在控制台上或使用或以编程方式完成此操作的 AWS CLI 权限。 AWS API

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS 的托管策略 AWS HealthLake

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托管式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 当新服务启动或现有服务 AWS 服务 有新API操作可用时，最有可能更新 AWS 托管策略。

有关更多信息，请参阅 IAM IAM 用户指南中的 AWS 托管式策略。

AWS 托管策略： AmazonHealthLakeFullAccess

该AmazonHealthLakeFullAccess策略提供对的完全访问权限 HealthLake。将此策略附加到其用户或角色后，用户 HealthLake 就可以使用来访问、查询、导入和导出中的数据 HealthLake。要在中执行许多常见操作 HealthLake，必须向用户或角色添加其他策略。有关更多信息，请参阅HealthLake 操作设置开始使用的权限 AWS HealthLake和权限。

您可以将 AmazonHealthLakeFullAccess 策略附加到 IAM 身份。

此策略授予的*administrative and contributor*权限允许用户和角色查询、搜索 HealthLake、导入和导出，还可以代表具有这些权限的用户和角色执行操作。 HealthLake

权限详细信息

本政策包括以下声明。

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Action": [
    "healthlake:*",
    "s3:ListAllMyBuckets",
```

```
    "s3:ListBucket",
    "s3:GetBucketLocation",
    "iam:ListRoles"
   ],
   "Resource": "*",
   "Effect": "Allow"
  },
  {
   "Effect": "Allow",
   "Action": "iam:PassRole",
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "iam:PassedToService": "healthlake.amazonaws.com"
    }
   }
  }
 ]
}
```

AWS 托管策略： AmazonHealthLakeReadOnlyAccess

AmazonHealthLakeReadOnlyAccess策略授予对其他 AWS 服务中 HealthLake 及相关资源的只读访问权限和权限。将此策略应用于您希望授予其查询和查看 HealthLake 数据存储的能力，但不允许其创建或更改数据的用户。

您可以将 AmazonHealthLakeReadOnlyAccess 策略附加到 IAM 身份。

此策略授予允许用户和角色进行查询的*read-only*权限 HealthLake。

权限详细信息

本政策包括以下声明。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
```

```
                "healthlake:ListFHIRDatastores",
                "healthlake:DescribeFHIRDatastore",
                "healthlake:DescribeFHIRImportJob",
                "healthlake:DescribeFHIRExportJob",
                "healthlake:GetCapabilities",
                "healthlake:ReadResource",
                "healthlake:SearchWithGet",
                "healthlake:SearchWithPost",
                "healthlake:SearchEverything"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

HealthLake 操作和权限

下表列出了中的典型操作 HealthLake 以及执行这些操作所需的权限。

HealthLake 操作	所需的权限
在中创建数据存储 HealthLake	`AmazonHealthLakeFullAccess` 、`AmazonLakeFormationDataAdmin` 、内联策略和 AWS Lake Formation 管理员权限由 AWS Lake Formation
删除中的数据存储 HealthLake	`AmazonHealthLakeFullAccess` 、`AmazonLakeFormationDataAdmin` 、内联策略和 AWS Lake Formation 管理员权限由 AWS Lake Formation
在中列出、搜索或查询数据存储 HealthLake	`AmazonHealthLakeReadOnlyAccess`
使用查询数据存储 Amazon Athena	`AmazonAthenaFullAccess` `AmazonS3FullAccess` 、 AWS Lake Formation `Select`和对由管理的表的`Describe`权限 AWS Lake Formation
从中导入数据 HealthLake	请参阅 为导入任务设置权限。

HealthLake 操作	所需的权限
从中导出数据 HealthLake	请参阅 从您的数据存储中导出文件 (AWS SDKs)。

HealthLake AWS 托管策略的更新

查看自该服务开始跟踪这些更改之时 HealthLake 起的 AWS 托管策略更新的详细信息。要获得有关此页面变更的自动提醒，请订阅 “ HealthLake 文档历史记录” 页面上的订阅RSS源。

更改	描述	日期
AmazonHealthLakeFullAccess	AmazonHealthLakeFullAccess 需要策略才能允许完全访问 HealthLake。	2022年11月14日
AmazonHealthLakeReadOnlyAccess	AmazonHealthLakeReadOnlyAccess 对的只读访问权限需要策略 HealthLake。	2022年11月14日
HealthLake 已开始跟踪更改	HealthLake 开始跟踪其 AWS 托管策略的更改。	2022年11月14日

对 AWS HealthLake 身份和访问进行故障排除

使用以下信息来帮助您诊断和修复在使用 HealthLake 和时可能遇到的常见问题IAM。

主题

- 我无权在以下位置执行操作 AWS HealthLake
- 我无权执行 iam：PassRole
- 我想允许 AWS 账户之外的人访问我的 AWS HealthLake 资源

我无权在以下位置执行操作 AWS HealthLake

如果 AWS Management Console 告诉您您无权执行某项操作，则必须联系管理员寻求帮助。管理员是指提供用户名和密码的人员。

当mateojacksonIAM用户尝试使用控制台查看虚构*my-example-widget*资源的详细信息但没有虚构权限时，就会出现以下示例错误。healthlake:*GetWidget*

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 healthlake:GetWidget on resource: my-example-widget
```

在这种情况下，Mateo 请求他的管理员更新其策略，以允许他使用 healthlake:*GetWidget* 操作访问 *my-example-widget* 资源。

我无权执行 iam：PassRole

如果您收到一个错误，表明您无权执行 iam:PassRole 操作，则必须更新策略以允许您将角色传递给 HealthLake。

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 marymajor 的 IAM 用户尝试使用控制台在 HealthLake 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 AWS 账户之外的人访问我的 AWS HealthLake 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 (ACLs) 的服务，您可以使用这些策略向人们授予访问您的资源的权限。

要了解更多信息，请参阅以下内容：

- 要了解是否 HealthLake 支持这些功能，请参阅AWS HealthLake 如何使用 IAM。
- 要了解如何提供对您拥有的资源的访问权限，请参阅《IAM用户指南》中的 AWS 账户 向其他IAM用户提供访问权限。 AWS 账户
- 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

- 要了解如何通过身份联合验证提供访问权限，请参阅《IAM 用户指南》中的为经过外部身份验证的用户（身份联合验证）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》IAM中的跨账户资源访问权限。

使用 AWS CloudTrail记录 AWS HealthLake API 调用

AWS HealthLake 与 AWS CloudTrail一项服务集成，该服务提供用户、角色或 AWS 服务在中执行的操作的记录 HealthLake。 CloudTrail 将所有API呼叫捕获 HealthLake 为事件。捕获的调用包括来自 HealthLake 控制台的调用和对 HealthLake APIAPI操作的代码调用。如果您创建了跟踪，则可以允许将 CloudTrail事件持续传输到 Amazon S3 存储桶，包括的事件 HealthLake。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，您可以确定向哪个请求发出 HealthLake、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。

AWS HealthLake CloudTrail 中的信息

CloudTrail 在您创建 AWS 账户时已在您的账户上启用。当活动发生在中时 HealthLake，该活动会与其他 AWS 服务 CloudTrail 事件一起记录在事件历史记录中。您可以在 AWS 账户中查看、搜索和下载最新事件。有关更多信息，请参阅使用事件历史记录查看 CloudTrail 事件。

要持续记录您 AWS 账户中的事件，包括的事件 HealthLake，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时，此跟踪应用于所有 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：

- 创建跟踪概述
- CloudTrail 支持的服务和集成
- 为以下各项配置亚马逊SNS通知 CloudTrail
- 接收来自多个区域的 CloudTrail 日志文件和接收来自多个账户的 CloudTrail日志文件

所有 HealthLake 操作均由记录 CloudTrail 并记录在《HealthLake API参考资料》和本开发人员指南中，用于使用执行的操作FHIRRESTAPI。例如，对以下操作的调用会在 CloudTrail 日志文件中生成条目：

- DescribeFHIRImportJob
- DescribeFHIRExportJob
- StartFHIRImportJob
- ListFHIRImportJobs
- StartFHIRExportJob
- ListFHIRExportJobs
- CreateFHIRDatastore
- ListFHIRDatastores
- DeleteFHIRDatastore
- DescribeFHIRDatastore
- UpdateResource
- CreateResource
- DeleteResource
- ReadResource
- GetCapabilities
- SearchWithGet
- SearchWithPost

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 root 还是 AWS Identity and Access Management (IAM) 用户凭据发出。
- 请求是使用角色还是联合用户的临时安全凭证发出的。
- 请求是否由其他 AWS 服务发出。

有关更多信息，请参阅CloudTrail userIdentity元素。

了解 AWS HealthLake 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。

以下示例显示了演示该CreateFHIRDatastore操作的 CloudTrail 日志条目。


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROA2B3ZHOADD2OJ4AHJX:git
 full_access_iam_role580074395690222150",
        "arn": "arn:aws:sts::691207106566:assumed-role/
colossusfrontend_full_access_iam_role/_iam_role580074395690222150",
        "accountId": "AccountID",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROA2B3ZHOADD2OJ4AHJX",
                "arn": "arn:aws:iam::691207106566:role/full_access_iam_role",
                "accountId": "AccountID",
                "userName": "full_access_iam_role"
            },
            "webIdFederationData": {

            },
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2020-11-20T00:08:15Z"
            }
        }
    },
    "eventTime": "2020-11-20T00:08:16Z",
    "eventSource": "healthlake.amazonaws.com",
    "eventName": "CreateFHIRDatastore",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "3.213.247.1",
    "userAgent": "Coral/Netty4",
    "requestParameters": {
        "datastoreName":
 "testCreateFHIRDatastore_GBYAZFCLLBLSUTOYYFQZRLBLQJNFOYQVRPZBOJAIIUAHICAEAGIWLNVQEYAMSXVWMBLXC
        "datastoreTypeVersion": "R4",
        "clientToken": "d737ffe0-14dd-44cc-9f0a-fdf59b26c66b"
    },
    "responseElements": {
        "datastoreId": "datastoreID",
```

```
        "datastoreArn": "arn:aws:healthlake:us-
east-1:691207106566:datastore/55576c487ff4975262b10d1d65eb4509",
        "datastoreStatus": "CREATING",
        "datastoreEndpoint": "datastore_endpoint/"
    },
    "requestID": "68e62bdd-d2d4-44c1-af69-e6f055a69f99",
    "eventID": "7ef483dc-5dca-469e-823a-7d9e3a7fe924",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "691207106566"
}
```

的合规性验证 AWS HealthLake

AWS HealthLake 作为多个合规计划的一部分，第三方审计师对安全性和 AWS 合规性进行评估。为 HealthLake 此，包括HIPAA。

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：

- Security Compliance & Governance：这些解决方案实施指南讨论了架构考虑因素，并提供了部署安全性和合规性功能的步骤。
- 在 Amazon Web Services 上进行HIPAA安全与合规架构 — 本白皮书描述了各公司如何使用 AWS 来创建HIPAA符合条件的应用程序。

> **Note**
>
> 并非所有 AWS 服务 人都有HIPAA资格。有关更多信息，请参阅《HIPAA符合条件的服务参考》。

- AWS 合规资源AWS — 此工作簿和指南集可能适用于您所在的行业和所在地区。

- 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、行业准则和法规。
- AWS Security Hub— 这 AWS 服务 提供了您内部安全状态的全面视图 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 可以帮助您满足各种合规性要求 PCIDSS，例如满足某些合规性框架规定的入侵检测要求。
- 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 AWS HealthLake

AWS 全球基础设施是围绕 AWS 区域和可用区构建的。 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相连。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。

除了 AWS 全球基础架构外，还 HealthLake 提供多项功能来帮助支持您的数据弹性和备份需求。

AWS HealthLake 中的基础架构安全性

作为一项托管服务， AWS HealthLake 受到《Amazon Web Services：安全流程概述》白皮书中描述的 AWS 全球网络安全程序的保护。

您可以使用 AWS 已发布的API呼叫 HealthLake 通过网络进行访问。客户端必须支持传输层安全 (TLS) 1.0 或更高版本。我们建议使用 TLS 1.2 或更高版本。客户端还必须支持具有完全向前保密性的密码套件 ()，例如 Ephemeral Diffie-Hellman (PFS) 或 Elliptic Curve Ephemeral Diffie-Hellman ()。DHE ECDHE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。

AWS HealthLake 中的安全最佳实践

AWS HealthLake 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。

- 实施最低权限访问。
- 只要有可能，请使用 Customer-Managed-Keys (CMKs) 加密您的数据。要了解更多信息CMKs，请参阅 Amazon 密钥管理服务。
- 在数据存储中查询PHI或PII在数据存储中进行查询GET时POST，请使用 “搜索方式”，而不是 “搜索方式”。
- 限制对敏感和重要的审计功能的访问。
- 通过更新或批量导入创建资源时APIs，请勿在任何可见字段PHI或PII逻辑 FHIR ID (LID) 中使用或，包括数据存储和作业的名称。
- 发送创建、读取、更新、删除或搜索请求时，不要在HTTP标题PHI中使用。
- 启用 AWS CloudTrail 该选项可以审计 AWS HealthLake 使用情况并确保没有意外活动。
- 查看安全使用 Amazon S3 存储桶的最佳实践。要了解更多信息，请参阅 Amazon S3 用户指南中的安全最佳实践。

AWS HealthLake 终端节点和配额

以下各节包含有关 AWS HealthLake 配额和终端节点的信息。对于可调限额，您可以使用服务限额控制台请求增加限额。有关更多信息，请参阅服务限额用户指南 中的请求增加限额。

服务端点

该表显示了给定区域中可用的 HealthLake 服务终端节点。

区域名称	区域	端点	协议
美国东部（俄亥俄州）	us-east-2	healthlake.us-east-2.amazonaws.com healthlake-fips.us-east-2.amazonaws.com	HTTPS HTTPS
美国东部（弗吉尼亚州北部）	us-east-1	healthlake.us-east-1.amazonaws.com healthlake-fips.us-east-1.amazonaws.com	HTTPS HTTPS
美国西部（俄勒冈州）	us-west-2	healthlake.us-west-2.amazonaws.com healthlake-fips.us-west-2.amazonaws.com	HTTPS HTTPS
亚太地区（孟买）	ap-south-1	healthlake.ap-south-1.amazonaws.com	HTTPS
亚太地区（悉尼）	ap-southeast-2	healthlake.ap-southeast-2.amazonaws.com	HTTPS
欧洲地区（伦敦）	eu-west-2	healthlake.eu-west-2.amazonaws.com	HTTPS

的服务配额 HealthLake

以下是的默认配额 HealthLake。

名称	默认值	可调整	描述
医疗记录中的字符数	每个受支持的区域：1 万个	否	DocumentReference 资源类型（POST/PUTrequests）中单个医疗记录中的最大字符数。
并发 S tartFHIRImport Job 作业数	每个受支持的区域：1 个	否	最大并发 S tartFHIRImport Job 作业数。
concurrentStartFHIRExportJob工作岗位数量	每个受支持的区域：1 个	否	最大并发 S tartFHIRExport Job 作业数。
每个账户的数据存储数量	每个受支持的区域：10 个	是	每个账户的默认最大活动数据存储数量。
S tartFHIRImport Job 中的文件数	每个受支持的区域：1 万个	否	S Jo tartFHIRImport b 中的最大文件数。
每个捆绑包的资源数量	每个受支持的区域：160 个	否	捆绑包请求中允许的最大资源数量。
每个账户的捆绑包请求速率	每个受支持的区域：20 个	是	每个账户每秒可以发出的最大POST捆绑请求数。
每个数据存储的捆绑包请求速率	每个受支持的区域：10 个	是	每个数据存储每秒可以发出的最大 POST Bundle 请求数。2023 年 8 月 21 日之前创建的数据存储将限制为每秒 1 个请求。
DELETE每个账户使用的 C ancelFHIR Export Job 请求率	每个受支持的区域：1 个	否	每个账户每分钟可以发出DELETE的 C ancelFHIR Export Job 请求的最大数量。

名称	默认值	可调整	描述
每个账户 C reateFHIRDatastore 请求的比率	每个受支持的区域：1 个	否	每个账户每分钟可以发出的最大 C reateFHIR Datastore 请求数。
每个账户的 DELETE 请求速率	每个支持的区域：2000 个	是	每个账户每秒可以发出的最大DELETE请求数。
每个数据存储的DELETE请求率	每个受支持的区域：1,000 个	是	每个数据存储每秒可以发出的最大DELETE请求数。2023 年 8 月 21 日之前创建的数据存储将限制为每秒 100 个请求。
每个账户 D eleteFHIRDatastore 请求的比率	每个受支持的区域：1 个	否	每个账户每分钟可以发出的最大 D eleteFHIR Datastore 请求数。
每个账户 D escribeFHIRDatastore 请求的比率	每个受支持的区域：10 个	否	每个账户每秒可以发出的最大 D escribeFH IRDatastore 请求数。
每个账户的 D escribeFHIRExport Job 请求率	每个受支持的区域：10 个	否	每个账户每秒可以发出的最大 D escribeFH IRExport Job 请求数。
GET每个账户使用的 D escribeFH IRExport Job 请求率	每个受支持的区域：10 个	否	每个账户每秒可使用GET的最大 D escribeFH IRExport Job 请求数。
每个账户的 D escribeFHIRImport Job 请求率	每个受支持的区域：10 个	否	每个账户每秒可以发出的最大 D escribeFH IRImport Job 请求数。

名称	默认值	可调整	描述
每个账户的 Discovery 请求速率	每个受支持的区域：10 个	否	每个账户每分钟可发出的最大 Discovery 请求数量。
每个账户的 GET 请求速率	每个受支持的区域：6000 个	是	每个账户每秒可以发出的最大GET请求数。
每个数据存储的GET请求率	每个受支持的区域：3000 个	是	每个数据存储每秒可以发出的最大GET请求数。2023 年 8 月 21 日之前创建的数据存储将限制为每秒 100 个请求。
每个账户的 GetCapabilities 请求率	每个受支持的区域：10 个	否	每个账户每秒可以发出的最大 GetCapabilities 请求数。
每个账户 L 次istFHIRDatastores 请求的比率	每个受支持的区域：10 个	否	每个账户每秒可以发出的最大 L istFHIRDatastores 请求数。
每个账户的 L istFHIRExport Jobs 请求率	每个受支持的区域：10 个	否	每个账户每秒可以发出的最大 L istFHIRExport Jobs 请求数。
每个账户的 L istFHIRImport Jobs 请求率	每个受支持的区域：10 个	否	每个账户每秒可以发出的最大 L istFHIRImport Jobs 请求数。
每个账户的 ListTagsforResource 请求率	每个受支持的区域：10 个	否	每个账户每秒可以发出的最大 ListTagsforResource 请求数。
每个账户的 POST 请求速率	每个支持的区域：2000 个	是	每个账户每秒可以发出的最大POST请求数。

名称	默认值	可调整	描述
每个数据存储的POST请求率	每个受支持的区域：1,000 个	是	每个数据存储每秒可以发出的最大POST请求数。2023 年 8 月 21 日之前创建的数据存储将限制为每秒 100 个请求。
每个账户的 PUT 请求速率	每个支持的区域：2000 个	是	每个账户每秒可以发出的最大PUT请求数。
每个数据存储的PUT请求率	每个受支持的区域：1,000 个	是	每个数据存储每秒可以发出的最大PUT请求数。2023 年 8 月 21 日之前创建的数据存储将限制为每秒 100 个请求。
每个账户的 S tartFHIRExport Job 请求率	每个受支持的区域：1 个	否	每个账户每分钟可以发出的最大 S tartFHIRExport Job 请求数。
POST每个账户使用的 S tartFHIRExport Job 请求率	每个受支持的区域：1 个	否	每个账户每分钟PO ST可以发出的最大 S tartFHIRExport Job 请求数。
每个账户的 S tartFHIRImport Job 请求率	每个受支持的区域：1 个	否	每个账户每分钟可以发出的最大 S tartFHIRImport Job 请求数。
每个账户的 TagResource 请求率	每个受支持的区域：10 个	否	您每秒可以发出的最大 TagResource 请求数。
每个账户的 UntagResource 请求率	每个受支持的区域：10 个	否	每个账户每秒可以发出的最大 UntagResource 请求数。

名称	默认值	可调整	描述
GET每个账户的搜索请求使用率	每个受支持的区域：200 个	是	每个账户每秒GET可使用的最大搜索请求数。
GET每个数据存储使用的搜索请求率	每个受支持的区域：100 个	是	每个数据存储每秒GET可使用的最大搜索请求数。
POST每个账户的搜索请求使用率	每个受支持的区域：200 个	是	您每秒可使用POST的最大搜索请求数。
POST每个数据存储使用的搜索请求率	每个受支持的区域：100 个	是	每个数据存储每秒POST可使用的最大搜索请求数。
单个导入文件的大小	每个受支持的区域：5 GB	否	S tartFHIRImport Job 中包含的单个文件的最大大小（以 GB 为单位）。
总导入任务大小	每个支持区域：500 GB	否	导入任务中包含的所有文件的最大大小（以 GB 为单位）。

问题排查

以下文档可以帮助您解决在使用时可能遇到的问题 AWS HealthLake。

主题

为什么我无法创建 HealthLake 数据存储？

2022 年 11 月 14 日，HealthLake 更新了创建新数据存储所需的IAM权限。如果您尚未更新附加到访问权限的用户或角色的策略，HealthLake 则会出现以下错误。

```
AccessDeniedException: Insufficient Lake Formation permission(s): Required Database on
 Catalog
```

要查看创建数据存储的更新IAM策略要求，请参阅 AWS 托管策略： AmazonHealthLakeFullAccess。有关如何将这些策略添加到您的IAM用户或角色的 step-by-step说明，请参阅设置开始使用的权限 AWS HealthLake。

要创建数据存储，您还需要使用对称的客户拥有或亚马逊KMS拥有的密钥。确保您的IAM策略中有正确的权限。要了解更多信息 AWS KMS，请参阅AWS Key Management Service《AWS Key Management Service 开发人员指南》。

已超过每个账户允许的数据存储数量

HealthLake 每个账户的配额为 10 个数据存储。要了解如何申请增加配额，请访问 Su AWS pport Center。

如何为创建授权 FHIR RESTfulAPIs？

用户应使用签名版本 4 签名流程为通过HTTP客户端发送的 HealthLake API请求添加身份验证。要了解更多信息，请参阅签名版本 4 的签名流程。

要使用适用于 AWS SDK Python 的 sigv4 授权，请创建一个类似于以下示例的脚本。

```
import boto3
import requests
import json
from requests_auth_aws_sigv4 import AWSSigV4

# Set the input arguments
data_store_endpoint = 'https://healthlake.us-east-1.amazonaws.com/datastore/<datastore
 id>/r4//'
resource_path = "Patient"
requestBody = {"resourceType": "Patient", "active": True, "name": [{"use":
 "official","family": "Dow","given": ["Jen"]},{"use": "usual","given":
 ["Jen"]}],"gender": "female","birthDate": "1966-09-01"}
region = 'us-east-1'

#Frame the resource endpoint
resource_endpoint = data_store_endpoint+resource_path
session = boto3.session.Session(region_name=region)
client = session.client("healthlake")
```

```
# Frame authorization
auth = AWSSigV4("healthlake", session=session)

# Calling data store FHIR endpoint using SigV4 auth

r = requests.post(resource_endpoint, json=requestBody, auth=auth, )
print(r.json())
```

有关使用 AWS SDK Python 的 sigv4 授权的其他信息可以在 Boto 3 凭据主题中找到。

我的数据不是 FHIR R4 格式——我还能使用 HealthLake吗？

只有 FHIR R4 格式的数据可以导入到 HealthLake 数据存储中。有关提供产品以帮助用户转换数据的合作伙伴列表，请参阅AWS HealthLake 合作伙伴。

为什么我在使用客户托管KMS密钥加密的数据存储时会收到 AccessDenied 错误？FHIRRESTfulAPIs

用户或角色需要拥有客户托管密钥和IAM策略的权限才能访问数据存储。用户必须具有使用客户托管密钥所需的IAM权限。如果用户撤销或取消了授予使用客户托管KMS密钥 HealthLake权限的授权，则 HealthLake 会返回 AccessDenied错误。

HealthLake 必须拥有访问客户数据、加密导入到数据存储的新FHIR资源以及在收到请求时对FHIR资源进行解密的权限。

要了解更多信息，请参阅密钥访问疑难解答。

为什么我的导入失败了？

成功的导入任务将生成一个包含输出 inputFileName .ndjson 文件的文件夹，但是单个记录可能无法导入。发生这种情况时，将生成第二个FAILURE文件夹，其中包含导入失败的记录清单。访问清单文件的任务输出位置是 JobProperties。 JobOutputDataConfig.s3Configuration.S3Uri。

此清单文件包含有关任务输出的详细信息，例如所有成功响应的位置 (successOutput. successOutputS3Uri)，所有失败响应的位置 (. failureOutput failureOutputS3Uri) 和其他工作指标。清单文件的内容可以通过编程方式进行解析。以下示例清单文件列出了输入和输出 Amazon S3 存储桶，以及有关已扫描的资源数量和成功导入的资源数量的信息。

```
            {
            "inputDataConfig": {
            "s3Uri": "s3://amzn-s3-demo-source-bucket/healthlake-input/invalidInput/"
            },
            "outputDataConfig": {
            "s3Uri": "s3://amzn-s3-demo-logging-
bucket/32839038a2f47f17c2fe0f53f0c3a0ba-FHIR_IMPORT-19dd7bb7bcc8ee12a09bf6d322744a3d/",
            "encryptionKeyID": "arn:aws:kms:us-west-2:123456789012:key/
fbbbfee3-20b3-42a5-a99d-c48c655ed545"
            },
            "successOutput": {
            "successOutputS3Uri": "s3://amzn-s3-demo-logging-
bucket/32839038a2f47f17c2fe0f53f0c3a0ba-FHIR_IMPORT-19dd7bb7bcc8ee12a09bf6d322744a3d/
SUCCESS/"
            },
            "failureOutput": {
            "failureOutputS3Uri": "s3://amzn-s3-demo-logging-
bucket/32839038a2f47f17c2fe0f53f0c3a0ba-FHIR_IMPORT-19dd7bb7bcc8ee12a09bf6d322744a3d/
FAILURE/"
            },
            "numberOfScannedFiles": 1,
            "numberOfFilesImported": 1,
            "sizeOfScannedFilesInMB": 0.023627,
            "sizeOfDataImportedSuccessfullyInMB": 0.011232,
            "numberOfResourcesScanned": 9,
            "numberOfResourcesImportedSuccessfully": 4,
            "numberOfResourcesWithCustomerError": 5,
            "numberOfResourcesWithServerError": 0
            }
```

要分析导入任务失败的原因，请使用 D escribeFHIRImport Job API 来分析 JobProperties。建议采取以下措施：

- 如果状态为FAILED且存在消息，则失败与作业参数有关，例如输入数据大小或输入文件数量超出 HealthLake 配额。
- 如果导入任务状态为 COMPLETED _ WITH _ERRORS，请查看清单文件 Manifest.json，了解有关哪些文件未成功导入的信息。
- 如果导入任务状态为FAILED，但消息不存在，请前往任务输出位置访问清单文件 Manifest.json。

对于每个输入文件，都有失败输出文件，其中包含任何未能导入的资源的输入文件名。响应包含与输入数据位置相对应的行号 (lineId)、FHIR响应对象 (UpdateResourceResponse) 和响应的状态码 (statusCode)。

示例输出文件如下所示：


```
{"lineId":3, UpdateResourceResponse:{"jsonBlob":
{"resourceType":"OperationOutcome","issue":
[{"severity":"error","code":"processing","diagnostics":"1 validation error detected:
 Value 'Patient123' at 'resourceType' failed to satisfy constraint: Member must satisfy
 regular expression pattern: [A-Za-z]{1,256}"}]}, "statusCode":400}
{"lineId":5, UpdateResourceResponse:{"jsonBlob":
{"resourceType":"OperationOutcome","issue":
[{"severity":"error","code":"processing","diagnostics":"This property must be an
 simple value, not a com.google.gson.JsonArray","location":["/EffectEvidenceSynthesis/
name"]},{"severity":"error","code":"processing","diagnostics":"Unrecognised
 property '@telecom'","location":["/EffectEvidenceSynthesis"]},
{"severity":"error","code":"processing","diagnostics":"Unrecognised
 property '@gender'","location":["/EffectEvidenceSynthesis"]},
{"severity":"error","code":"processing","diagnostics":"Unrecognised
 property '@birthDate'","location":["/EffectEvidenceSynthesis"]},
{"severity":"error","code":"processing","diagnostics":"Unrecognised
 property '@address'","location":["/EffectEvidenceSynthesis"]},
{"severity":"error","code":"processing","diagnostics":"Unrecognised
 property '@maritalStatus'","location":["/EffectEvidenceSynthesis"]},
{"severity":"error","code":"processing","diagnostics":"Unrecognised
 property '@multipleBirthBoolean'","location":["/EffectEvidenceSynthesis"]},
{"severity":"error","code":"processing","diagnostics":"Unrecognised
 property '@communication'","location":["/EffectEvidenceSynthesis"]},
{"severity":"warning","code":"processing","diagnostics":"Name should be usable as an
 identifier for the module by machine processing applications such as code generation
 [name.matches('[A-Z]([A-Za-z0-9_]){0,254}')]","location":["EffectEvidenceSynthesis"]},
{"severity":"error","code":"processing","diagnostics":"Profile http://hl7.org/fhir/
StructureDefinition/EffectEvidenceSynthesis, Element 'EffectEvidenceSynthesis.status':
 minimum required = 1, but only found 0","location":["EffectEvidenceSynthesis"]},
{"severity":"error","code":"processing","diagnostics":"Profile
 http://hl7.org/fhir/StructureDefinition/EffectEvidenceSynthesis,
 Element 'EffectEvidenceSynthesis.population': minimum required
 = 1, but only found 0","location":["EffectEvidenceSynthesis"]},
{"severity":"error","code":"processing","diagnostics":"Profile
 http://hl7.org/fhir/StructureDefinition/EffectEvidenceSynthesis,
 Element 'EffectEvidenceSynthesis.exposure': minimum required =
```

```
 1, but only found 0","location":["EffectEvidenceSynthesis"]},
{"severity":"error","code":"processing","diagnostics":"Profile http://
hl7.org/fhir/StructureDefinition/EffectEvidenceSynthesis, Element
 'EffectEvidenceSynthesis.exposureAlternative': minimum required
 = 1, but only found 0","location":["EffectEvidenceSynthesis"]},
{"severity":"error","code":"processing","diagnostics":"Profile http://hl7.org/fhir/
StructureDefinition/EffectEvidenceSynthesis, Element 'EffectEvidenceSynthesis.outcome':
 minimum required = 1, but only found 0","location":["EffectEvidenceSynthesis"]},
{"severity":"information","code":"processing","diagnostics":"Unknown
 extension http://synthetichealth.github.io/synthea/disability-adjusted-
life-years","location":["EffectEvidenceSynthesis.extension[3]"]},
{"severity":"information","code":"processing","diagnostics":"Unknown extension
 http://synthetichealth.github.io/synthea/quality-adjusted-life-years","location":
["EffectEvidenceSynthesis.extension[4]"]}]}, "statusCode":400}
{"lineId":7, UpdateResourceResponse:{"jsonBlob":
{"resourceType":"OperationOutcome","issue":
[{"severity":"error","code":"processing","diagnostics":"2 validation errors detected:
 Value at 'resourceId' failed to satisfy constraint: Member must satisfy regular
 expression pattern: [A-Za-z0-9-.]{1,64}; Value at 'resourceId' failed to satisfy
 constraint: Member must have length greater than or equal to 1"}]}, "statusCode":400}
{"lineId":9, UpdateResourceResponse:{"jsonBlob":
{"resourceType":"OperationOutcome","issue":
[{"severity":"error","code":"processing","diagnostics":"Missing required id field in
 resource json"}]}, "statusCode":400}
{"lineId":15, UpdateResourceResponse:{"jsonBlob":
{"resourceType":"OperationOutcome","issue":
[{"severity":"error","code":"processing","diagnostics":"Invalid JSON found in input
 file"}]}, "statusCode":400}
```

该示例显示输入文件中相应输入行的第 3、4、7、9、15 行出现故障。对于其中的每一行，解释如下：

- 在第 3 行，响应解释了输入文件第 3 行中 resourceType 提供的内容无效。
- 在第 5 行，响应说明输入文件的第 5 行存在FHIR验证错误。
- 在第 7 行，响应解释说，作为输入 resourceId 提供的存在验证问题。
- 在第 9 行，响应说明输入文件必须包含有效的资源 ID。
- 在第 15 行，输入文件的响应是该文件的JSON格式无效。

如何找到无法处理的 DocumentReference资源？

如果 DocumentReference 资源无效， HealthLake 将提供表示验证错误的扩展名，而不是综合医疗NLP输出。为了查找在NLP处理过程中导致验证错误的 DocumentReference 资源，客户可以使用带有搜索键cm-decoration-status和搜索值 HealthLake的搜索功能 VALIDATION_ ERROR。此搜索将列出导致验证错误的所有 DocumentReference 资源，以及描述错误性质的错误消息。那些存在验证错误的 DocumentReference 资源中扩展字段的结构将类似于以下示例。

```
"extension": [
          {
              "extension": [
                  {
                      "url": "http://healthlake.amazonaws.com/aws-cm/status/",
                      "valueString": "VALIDATION_ERROR"
                  },
                  {
                      "url": "http://healthlake.amazonaws.com/aws-cm/message/",
                      "valueString": "Resource led to too many nested objects after NLP
 operation processed the document. 10937 nested objects exceeds the limit of 10000."
                  }
              ],
              "url": "http://healthlake.amazonaws.com/aws-cm/"
          }
    ]
```

如果NLP装饰创建的嵌套对象超过 10,000 个，ERROR也会出现 VALIDATION_。发生这种情况时，必须先将文档拆分成较小的文档，然后再进行处理。

迁移现有数据存储以使用 Amazon Athena

2022 年 11 月 14 日之前创建的数据存储可以正常运行，但无法在 Athena 中使用进行查询。SQL要使用 Athena 查询先前存在的数据存储，必须先将其迁移到新的数据存储。

将数据迁移到新的数据存储

1. 创建新的数据存储。
2. 将数据从先前存在的存储桶导出到 Amazon S3 存储桶。
3. 将数据从 Amazon S3 存储桶导入到新的数据存储中。

将数据导出到 Amazon S3 存储桶需要支付额外费用。额外费用取决于您导出的数据的大小。

将 Athena 中的搜索结果连接到其他服务 AWS

与其他服务共享来自 Athena 的搜索结果时，您可能会遇到问题。 AWS

当你`json_extract[1]`作为SQL搜索查询的一部分使用时，可能会出现问题。

要修复此问题，必须更新到CATVAR。

在尝试创建保存结果、表格（静态）或视图（动态）时，您可能会遇到此问题。

将数据导入新数据存储后，Athena 控制台无法正常工作

将数据导入新的数据存储后，数据可能无法立即使用。这是为了留出时间将数据提取到冰山表中。请稍后再试。

为什么我在添加新的数据湖管理员PutDataLakeSettings 时会出现 Lake Formation 权限错误：lakeformation： ？

如果您的IAM用户或角色包含`AWSLakeFormationDataAdmin` AWS 托管策略，则无法添加新的数据湖管理员。您将收到一条包含以下内容的错误：

```
User arn:aws:sts::111122223333:assumed-role/lakeformation-admin-user is not authorized
 to perform: lakeformation:PutDataLakeSettings on resource: arn:aws:lakeformation:us-
east-2:111122223333:catalog:111122223333 with an explicit deny in an identity-based
 policy
```

需要使用 AWS 托管策略`AdministratorAccess`才能添加IAM用户或角色作为 AWS Lake Formation 数据湖管理员。如果您的IAM用户或角色也包含`AWSLakeFormationDataAdmin`该操作，则操作将失败。`AWSLakeFormationDataAdmin` AWS 托管策略包含对 La AWS ke Formation API 操作的明确拒绝。`PutDataLakeSetting`

即使管理员拥有AWS使用`AdministratorAccess`AWS托管策略的完全访问权限，也可能受到该`AWSLakeFormationDataAdmin`策略的限制。

如何开启 HealthLake集成的自然语言处理功能？

自 2023 年 2 月 20 日起， HealthLake 数据存储的默认行为发生了变化。

当前数据存储：所有当前 HealthLake 的数据存储都将停止在 base64 编码DocumentReference的资源上使用自然语言处理 (NLP)。这意味着不会使用分析新DocumentReference资源NLP，也不会根据资源类型中的文本生成任何新DocumentReference资源。对于现有DocumentReference资源，通过生成的数据和资源将NLP保留，但在 2023 年 2 月 20 日之后不会更新。

新数据存储：2023 年 2 月 20 日之后创建 HealthLake 的数据存储将不会对 base64 编码DocumentReference的资源执行自然语言处理 (NLP)。

要启用此功能，您必须使用创建案例AWS Support Center Console。要创建您的案例，请登录您的 AWS 账户，然后选择创建案例。要了解有关创建案例和案例管理的更多信息，请参阅《Support 用户指南》中的创建支持案例和案例管理。

我的数据存储状态未从 “正在创建” 中改变

如果您尝试创建新的 HealthLake 数据存储，但您的数据存储状态未从 “正在创建” 发生变化，则需要更新 Athena 才能使用。 AWS Glue Data Catalog

要了解更多信息，请参阅亚马逊 Athena 用户指南 step-by-step中的升级到 AWS Glue 数据目录。

成功升级后 AWS Glue Data Catalog，您现在可以创建数据存储了。

要删除旧的数据存储，请先使用创建案例AWS Support Center Console。要创建您的案例，请登录您的 AWS 账户，然后选择创建案例。要了解更多信息，请参阅Support 用户指南中的创建支持案例和案例管理。

我的SDK数据存储创建状态返回异常或未知状态

如果您的SDK列表数据存储或描述数据存储API调用返回异常或未知数据存储状态，请将您的列表更新到最新版本。

我对一个 10MB 文档的FHIRPOSTAPI操作出现 HealthLake 了 413Request Entity Too Large 错误。

AWS HealthLake 同步创建和更新API限制为 5MB，以避免延迟和超时增加。

您可以使用 Binary（批量导入）， ResourceType 使用二进制文件提取最大 164MB 的大型文档。API

AWS HealthLake 开发者指南的文档历史记录

下表描述了各 AWS HealthLake 版本的文档更改。

- API版本：最新
- 最新文档更新：2024 年 10 月 25 日

变更	说明	日期
HealthLake 现在支持FHIRhistory和vread互动	HealthLake 现在支持用于检索特定资源历史记录的交FHIRhistory互以及用于执行特定版本的资源读取的vread交互。	2024 年 10 月 25 日
HealthLake 现在支持新的FHIR搜索参数、扩展名和资源类型。	HealthLake 现在支持新的FHIR搜索参数、扩展名和资源类型。	2023 年 12 月 9 日
HealthLake 现在支持 SMART on FHIR 框架	HealthLake 现在支持SMART在FHIR已启用的 HealthLake 数据存储上创建。	2023 年 5 月 31 日
HealthLake 现在支持配置文件验证	HealthLake 现在支持FHIR配置文件验证。	2023 年 5 月 31 日
HealthLake 现在支持 export	HealthLake 现在支持使用FHIRRESTAPI操作导出文件export。	2023 年 5 月 31 日
亚太地区（孟买）区域	AWS HealthLake 现已在亚太地区（孟买）区域推出。	2023 年 4 月 4 日
集成自然语言处理已关闭	HealthLake 自 2023 年 2 月 20 日起，已在所有数据存储上关闭集成自然语言处理 (NLP)。	2023 年 2 月 20 日

HealthLake 与亚马逊 Athena 集成	现在，您可以使用 Athena 来查询 2022 年 11 月 14 日之后创建的数据存储。	2022 年 11 月 14 日
导入任务总量增加了	现在，S tartFHIRImport Job 请求中所有文件的最大总大小为 500 GB。	2022 年 10 月 3 日
捆绑包支持	HealthLake 现在支持 Bundle 资源类型，用于摄取多个资源。	2022 年 8 月 5 日
更新了中CRUD操作的配额 HealthLake	HealthLake 现在支持更高的CRUD请求限制。	2022 年 7 月 14 日
包括支持	HealthLake 现在支持_include数据存储查询。	2022 年 7 月 14 日
AWS HealthLake 现已正式上市	HealthLake 现已正式上市。	2020 年 7 月 30 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。