Hybrid Committed Spend 1-latest

将 Amazon Web Services (AWS)数据集成到混合提交的成本中

了解如何添加和配置 AWS 集成

Last Updated: 2024-05-18

Hybrid Committed Spend 1-latest 将 Amazon Web Services (AWS)数据集成到混合提交的成本中

了解如何添加和配置 AWS 集成

法律通告

摘要

Amazon Web Services (AWS)集成到混合承诺的花费。

目录

第 1 章 将 AMAZON WEB SERVICES 数据集成到混合承诺的花费中

要将 AWS 帐户添加到混合提交的花费中，您必须配置 AWS 帐户以提供指标，然后将 AWS 帐户添加为混合提交成本用户界面的集成。

注意

在添加集成至混合提交前，您必须拥有具有 Sources Administrator 权利的红帽帐户用户。

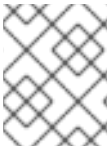
当您将 AWS 帐户添加为集成时，这将创建一个到 AWS 的只读连接，以递增方式收集成本和使用量信息，但不对 AWS 帐户进行任何更改。

在将 AWS 帐户添加到混合提交作为集成前，您必须在 AWS 帐户上配置以下服务，以允许混合提交对指标的访问：

1. 用于 hybrid committed spend 的成本和使用量数据的 S3 存储桶
2. 用于 hybrid committed spend 的 Identity Access Management (IAM) 策略和角色来处理成本和使用数据
3. Red Hat Hybrid Cloud Console 上的混合提交软件集成

当您完成 AWS 控制台中的几个步骤时，以及 hybrid committed spend 用户界面中的一些步骤，使两个应用程序都在 Web 浏览器中打开。

将 AWS 集成添加到 Integrations 页面 的混合提交花费中。



注意

因为非红帽产品和文档可以在不通知的情况下改变，所以配置本指南中提供的第三方集成的说明是常规的，并在发布时正确。有关最新和准确的信息，请参阅 AWS 文档。

1.1. 将 AWS 帐户添加为集成

先决条件

- 在向成本管理添加数据集成前，您必须拥有具有 Sources Administrator 权限的红帽帐户。

流程

1. 在 Red Hat Hybrid Cloud Console 中，点 **Settings Menu** > (**Settings**)。
2. 点 **Integrations**。
3. 在 **Cloud** 选项卡中，点 **Add Integration**。
4. 在 **Select integration type** 步骤中，在 **Add a cloud integration** 向导中选择 **Amazon Web Services**。点击 **Next**。
5. 输入集成的名称，然后点 **Next**。
6. 在 **Select configuration** 步骤中，选择如何连接到 AWS 集成。
    a. 选择 **帐户授权** 以提供 AWS 帐户凭证，并让红帽为您配置和管理您的集成。

b. 选择 **Manual configuration** 以自定义您的集成。您可以在信息发送到混合提交前过滤信息。有关如何过滤数据的步骤，请参考 *第 2 章 在将 Amazon Web Services 数据发送到混合提交的花费前过滤 Amazon Web Services 数据*。点击 **Next**。

7. 在 **Select application** 步骤中，选择 Cost management。点击 **Next**。

8. 如果您选择了帐户授权方法，在 **Review details** 步骤中查看详情并点 **Add**。如果选择了手动配置方法，请继续向导中的下一步并配置 S3 存储桶。

1.2. 创建 AWS S3 存储桶以存储您的成本数据

创建一个 Amazon S3 存储桶，其权限配置为存储计费报告。

流程

1. 登录到 AWS 帐户以开始配置成本和使用情况报告。

2. 在 AWS S3 控制台中，创建一个新的 S3 存储桶或使用现有存储桶。如果要配置新的 S3 存储桶，请接受默认设置。

3. 在 AWS Billing 控制台中，创建一个将传送到 S3 存储桶的成本和使用情况报告。指定以下值并接受任何其他值的默认值：

   - 报告名称：*<rh_cost_report>*（请注意此名称会在以后使用它）
   - 其他报告详情：包含的资源 ID
   - S3 存储桶：*<the S3 bucket you configured previously>*
   - 时间粒度： 每小时
   - 为 Amazon Red Hatshift、Amazon QuickSight （不要启用 Amazon Athena 的报告数据集成）启用报告数据集成
   - 压缩类型：GZIP
   - 报告路径前缀： cost

注意

有关配置的详情，请参阅 AWS *Billing 和 Cost Management* 文档。

4. 在 **Create storage** 步骤中，在 **Add a cloud integration**向导中粘贴 S3 存储桶的名称，然后选择创建它的区域。点击 **Next**。

5. 在 **Add a cloud integration**向导中，在 **Create cost and usage report**步骤中，点 **Next**。

1.3. 激活 AWS 标签

要使用标签在混合提交的应用程序中组织 AWS 资源，请在 AWS 中激活您的标签，以便自动导入它们。

流程

1. 在 AWS Billing 控制台中：

a. 打开 *Cost Allocation Tags* 部分。

b. 选择您要在混合提交的应用程序中使用的标签，然后点 **Activate**。

2. 如果您的组织正在将系统从 CentOS 7 转换为 RHEL，并使用每小时账单，请在 AWS 控制台的 **Cost Allocation Tags** 部分中激活您系统的 **com_redhat_rhel** 标签。

a. 在标记您要在 AWS 中计量的 RHEL 实例后，选择 **Include RHEL usage**。

3. 在 Red Hat Hybrid Cloud Console **Integrations** 向导中，点 **Next**。

其他资源

有关标记的更多信息，请参阅 为 AWS 资源添加标签。

1.4. 配置 IAM 策略，为成本和用量报告启用最小帐户访问

要在 Web 界面和 API 中提供数据，提交混合的费用必须消耗 AWS 生成的成本和使用情况报告。要只提供对存储的信息的访问，创建一个 IAM 策略和角色来使用混合信息。

流程

1. 在 AWS Identity and Access Management (IAM)控制台中，为之前配置的 S3 存储桶创建一个新的 IAM 策略。

a. 选择 JSON 选项卡并在 JSON 策略文本框中粘贴以下内容：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "s3:Get*",
        "s3:List*"
      ],
        "Resource": [
        "arn:aws:s3:::<your_bucket_name>", (1)
        "arn:aws:s3:::<your_bucket_name>/*"
      ]
    },

    {
      "Sid": "VisualEditor1",
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket",
        "cur:DescribeReportDefinitions"
      ],
      "Resource": "*"
    }
  ]
}
```

1 将两个位置的 < **your_bucket_name** > 替换为您之前配置的 Amazon S3 存储桶的名称。

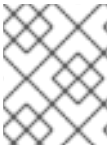
2. 在 AWS IAM 控制台中，创建一个新的 IAM 角色：
   a. 选择 **Another AWS 帐户** 作为可信实体的类型。
   b. 输入 *589173575009* 作为帐户 ID，以提供对 AWS 帐户成本数据的混合提交应用程序。
   c. 附加您刚才配置的 IAM 策略。
   d. 输入角色名称和描述。
3. 在 AWS IAM 控制台中，在 **Roles** 部分中为您刚才创建的角色打开概述屏幕。
   a. 复制 Role ARN，它是一个以 **arn:aws** 开头的字符串：
4. 在 Red Hat Hybrid Cloud Console **Add a cloud integration**向导中，粘贴您的角色 ARN 并点 **Next**。
5. 查看详情并点 **Finish** 将 AWS 帐户添加到成本管理。

成本管理将开始从 AWS 帐户和任何链接的 AWS 帐户收集成本和使用数据。

在 成本管理 仪表板中显示前，数据可能需要几天时间来填充。

第 2 章 在将 AMAZON WEB SERVICES 数据发送到混合提交的花费前过滤 AMAZON WEB SERVICES 数据

在 AWS 中配置函数脚本，以复制成本导出和对象存储存储桶，该存储桶可以混合使用它们访问并过滤您的数据，以与红帽共享您的账单数据子集。只有在您的机构有第三方数据限制时才建议使用这个选项。



注意

在添加数据集成以混合投入前，您必须拥有具有 Sources Administrator 权限的红帽帐户用户。

要将 AWS 帐户配置为混合提交的数据集成，您必须完成以下任务：

- 创建一个 AWS S3 存储桶来存储您的成本数据。
- 创建一个 AWS S3 存储桶来报告您过滤的 hybrid committed spend 数据。
- 为您的成本数据存储桶配置 IAM 角色。
- 将 AWS 集成添加到 Red Hat Hybrid Cloud Console 。
- 为 AWS Athena 配置 IAM 角色。
- 启用 Athena。
- 为 Athena 创建 Lambda 任务，将过滤的数据导出到 S3 存储桶。

当您完成 AWS 控制台中的几个步骤时，以及 hybrid committed spend 用户界面中的一些步骤，使两个应用程序都在 Web 浏览器中打开。

2.1. 将 AWS 帐户添加为集成

添加 AWS 集成，以便混合提交的应用程序可以处理 AWS 帐户中的成本和使用情况报告。您可以通过提供 AWS 帐户凭证来自动添加 AWS 集成，或者配置成本管理来过滤发送到红帽的数据。

先决条件

- 在向成本管理添加数据集成前，您必须拥有具有 Sources Administrator 权限的红帽帐户。

流程

1. 在 Red Hat Hybrid Cloud Console 中，点 **Settings Menu** > (**Settings**)。
2. 点 **Integrations**。
3. 在 **Cloud** 选项卡中，点 **Add Integration**。
4. 在 **Select integration type** 步骤中，在 **Add a cloud integration** 向导中选择 **Amazon Web Services**。点击 **Next**。
5. 输入集成的名称，然后点 **Next**。
    a. 选择 **Manual configuration** 以自定义您的集成。例如，您可以在将信息发送到混合提交成本前过滤信息。点击 **Next**。

6. 在 **Select application** 步骤中，选择 Cost management。点击 **Next**。

2.2. 创建 AWS S3 存储桶以存储您的成本数据

创建一个 Amazon S3 存储桶，其权限配置为存储计费报告。

流程

1. 登录到 AWS 帐户以开始配置成本和使用情况报告。
2. 在 AWS S3 控制台中，创建一个新的 S3 存储桶或使用现有存储桶。如果要配置新的 S3 存储桶，请接受默认设置。
3. 在 AWS Billing 控制台中，创建一个将传送到 S3 存储桶的成本和使用情况报告。指定以下值并接受任何其他值的默认值：
   - 报告名称：*<rh_cost_report>*（请注意此名称会在以后使用它）
   - 其他报告详情：包含的资源 ID
   - S3 存储桶：*<the S3 bucket you configured previously>*
   - 时间粒度： 每小时
   - 为 Amazon Red Hatshift、Amazon QuickSight （不要启用 Amazon Athena 的报告数据集成）启用报告数据集成
   - 压缩类型：GZIP
   - 报告路径前缀： cost

   注意

   有关配置的详情，请参阅 AWS *Billing 和 Cost Management* 文档。
4. 在 **Create storage** 步骤中，在 **Add a cloud integration**向导中粘贴 S3 存储桶的名称，然后选择创建它的区域。点击 **Next**。
5. 在 **Add a cloud integration**向导中，在 **Create cost and usage report**步骤中，点 **Next**。

2.3. 为过滤的数据报告创建成本和使用情况报告

在 AWS 中创建成本和使用情况报告(CUR)。CUR 将传送到您的 S3 存储桶。您将设置 Athena 和 Lambda 功能，以过滤此过程稍后的数据。

流程

1. 登录到 AWS 帐户。
2. 在 AWS S3 控制台中，创建一个要传送到 S3 存储桶的成本和使用情况报告。
3. 输入报告名称。保存此名称。您稍后会用到它。
4. 点 include resource ID。

5. 点击 **Next**。
6. 从 Configure S3 Bucket，单击 Configure。创建存储桶并应用默认策略。
7. 点击 **Save**。
8. 指定以下值并接受任何其他值的默认值：
   - 报告路径前缀：HCS-Athena
   - 时间粒度： 每小时
   - 为 Amazon Athena 启用报告数据集成
9. 点 **Next**
10. 验证信息并点 **Create report**。
11. 在 **Create storage** 步骤中，在 **Add a cloud integration**向导中粘贴 S3 存储桶的名称，再选择它在 中创建的区域，然后点 **Next**。
12. 在 **Add a cloud integration**向导中的 **Create Cost and usage report**步骤中，选择 **I want to custom the CUR to Cost Management** 并点 **Next**。

2.4. 激活 AWS 标签

要使用标签在混合提交的应用程序中组织 AWS 资源，请在 AWS 中激活您的标签，以便自动导入它们。

流程

1. 在 AWS Billing 控制台中：
   a. 打开 *Cost Allocation Tags* 部分。
   b. 选择您要在混合提交的应用程序中使用的标签，然后点 **Activate**。
2. 如果您的组织正在将系统从 CentOS 7 转换为 RHEL，并使用每小时账单，请在 AWS 控制台的 **Cost Allocation Tags** 部分中激活您系统的 **com_redhat_rhel** 标签。
   a. 在标记您要在 AWS 中计量的 RHEL 实例后，选择 **Include RHEL usage**。
3. 在 Red Hat Hybrid Cloud Console **Integrations** 向导中，点 **Next**。

其他资源

有关标记的更多信息，请参阅 为 AWS 资源添加标签。

2.5. 为成本和使用量启用最小帐户访问

对于提供数据的混合成本，它必须消耗 AWS 生成的成本和使用情况报告。对于具有最少访问权限的混合提交来获取这些数据，请为 hybrid committed spend 的混合提交创建 IAM 策略和角色。此配置仅提供对存储的信息的访问。

流程

1. 在 AWS Identity and Access Management (IAM)控制台中，为您配置的 S3 存储桶创建一个新的 IAM 策略。

   a. 选择 JSON 选项卡并在 **JSON 策略中** 粘贴以下内容：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "s3:*"
      ],
        "Resource": [
        "arn:aws:s3:::<your_bucket_name>",
        "arn:aws:s3:::<your_bucket_name>/*"
      ]
    },

    {
      "Sid": "VisualEditor1",
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket",
        "cur:DescribeReportDefinitions"
      ],
      "Resource": "*"
    }
  ]
}
```

   (1) 将 **两个位置中的 <your_bucket_name >** 替换为您为存储过滤数据的 Amazon S3 存储桶的名称。

   b. 为策略提供名称并完成策略的创建。保持 AWS IAM 控制台打开。下一步需要它。

   c. 在 Red Hat Hybrid Cloud Console **Add a cloud integration**向导中，点 **Next**。

2. 在 AWS IAM 控制台中，创建一个新的 IAM 角色：

   a. 对于可信实体的类型，选择 **AWS 帐户**。

   b. 输入 *589173575009* 作为帐户 ID，以提供对 AWS 帐户成本数据的混合提交应用程序。

   c. 附加您刚才配置的 IAM 策略。

   d. 输入角色名称和描述，并完成角色创建。

   e. 在 Red Hat Hybrid Cloud Console **Add a cloud integration**向导中，点 **Next**。

3. 在 AWS IAM 控制台中，从 Roles 中为您刚才创建的角色打开概述屏幕，并复制角色 ARN。它是以 **arn:aws:** 开头的字符串。

4. 在 Red Hat Hybrid Cloud Console **Add a cloud integration**向导中，输入 **Enter ARN** 页面的 ARN，然后点 **Next**。

5. 查看云集成的详细信息，然后点 **Add**。

后续步骤

返回到 AWS，通过配置 Athena 和 Lambda 来过滤您的报告来自定义 AWS 成本和使用情况报告。

2.6. 为 ATHENA 启用帐户访问

要在 Web 界面和 API 中提供数据，请为要使用的 hybrid committed spend 创建一个 IAM 策略和角色。此配置提供对存储的信息以及任何其他信息的访问。

流程

1. 在 AWS Identity and Access Management (IAM)控制台中，为您要配置的 Athena Lambda 功能创建一个 IAM 策略。
   a. 选择 JSON 选项卡并在 JSON 策略文本框中粘贴以下内容：

```
{
 "Version": "2012-10-17",
 "Statement": [
    {
        "Effect": "Allow",
        "Action": [
            "athena:*"
        ],
        "Resource": [
            "*"
        ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "glue:CreateDatabase",
            "glue:DeleteDatabase",
            "glue:GetDatabase",
            "glue:GetDatabases",
            "glue:UpdateDatabase",
            "glue:CreateTable",
            "glue:DeleteTable",
            "glue:BatchDeleteTable",
            "glue:UpdateTable",
            "glue:GetTable",
            "glue:GetTables",
            "glue:BatchCreatePartition",
            "glue:CreatePartition",
            "glue:DeletePartition",
            "glue:BatchDeletePartition",
            "glue:UpdatePartition",
            "glue:GetPartition",
            "glue:GetPartitions",
            "glue:BatchGetPartition"
        ],
        "Resource": [
            "*"
        ]
```

```
    },
    {
        "Effect": "Allow",
        "Action": [
            "s3:GetBucketLocation",
            "s3:GetObject",
            "s3:ListBucket",
            "s3:ListBucketMultipartUploads",
            "s3:ListMultipartUploadParts",
            "s3:AbortMultipartUpload",
            "s3:CreateBucket",
            "s3:PutObject",
            "s3:PutBucketPublicAccessBlock"
        ],
        "Resource": [
            "arn:aws:s3:::CHANGE-ME*" (1)
        ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "s3:GetObject",
            "s3:ListBucket"
        ],
        "Resource": [
            "arn:aws:s3:::CHANGE-ME*" (2)
        ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "s3:ListBucket",
            "s3:GetBucketLocation",
            "s3:ListAllMyBuckets"
        ],
        "Resource": [
            "*"
        ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "sns:ListTopics",
            "sns:GetTopicAttributes"
        ],
        "Resource": [
            "*"
        ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "cloudwatch:PutMetricAlarm",
            "cloudwatch:DescribeAlarms",
            "cloudwatch:DeleteAlarms",
            "cloudwatch:GetMetricData"
```

```
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "lakeformation:GetDataAccess"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:*"
            ],
            "Resource": "*"
        }
    ]
}
```

1 2 将两个位置中的 **CHANGE-ME*** 替换为 Athena Lambda 功能的 IAM 策略的 ARN。

b. 为策略提供名称并完成策略的创建。保持 AWS IAM 控制台打开，因为您需要它才能继续下一步。

2. 在 AWS IAM 控制台中，创建一个新的 IAM 角色：

a. 对于可信实体的类型，选择 **AWS 服务**。

b. 选择 Lambda。

c. 附加您刚才配置的 IAM 策略。

d. 输入角色名称和描述，并完成角色创建。

2.6.1. 为报告生成配置 Athena

配置 Athena 为混合提交成本提供经过过滤的成本和使用情况报告。配置 Athena 为成本管理提供经过过滤的成本和使用情况报告。

以下配置仅提供对额外存储信息的访问。它不提供对任何其他操作的访问：

流程

1. 在 AWS S3 控制台中，导航到您创建的过滤存储桶并下载 **crawler-cfn.yml** 文件。
2. 从 AWS 控制台中的 **Cloudformation** 中，创建一个新堆栈。
3. 将 **Template** 选择为 **Ready**。
4. 上传您之前下载的 **crawler-cfn.yml** 文件。这应该会立即加载。

5. 点击 **Next**。
6. 输入名称并点 **Next**。
7. 点 **I 确认 AWS Cloudformation 可能会创建 IAM 资源**，然后点 **Submit**。

2.6.2. 为 Athena 创建 Lambda 功能

您必须创建一个 Lambda 功能来查询成本和使用情况报告。此 Lambda 功能查询红帽相关费用的成本和使用情况报告，并产生您过滤的费用报告。

流程

1. 在 AWS 控制台中进入到 Lambda，然后点 **Create function**。
2. 点 **Author from scratch**。
3. 输入您的功能的名称。
4. 在 Runtime 下拉菜单中选择 python 3.7。
5. 选择 x86_64 作为 Architecture。
6. 在 Permissions 下，选择您创建的 Athena 角色。
7. 点 **Create function**。
8. 将以下代码粘贴到功能中：

```
import boto3
import uuid
import json
from datetime import datetime

now = datetime.now()
year = now.strftime("%Y")
month = now.strftime("%m")
day = now.strftime("%d")

# Vars to Change!
integration_uuid = <your_integration_uuid>                    # integration_uuid
bucket = <your_S3_Bucket_Name>                    # Bucket created for query results
database = 'athenacurcfn_athena_cost_and_usage'          # Database to execute athena
queries
output=f's3://{bucket}/{year}/{month}/{day}/{uuid.uuid4()}' # Output location for query results

# Athena query
query = f"SELECT * FROM {database}.koku_athena WHERE ((bill_billing_entity = 'AWS
Marketplace' AND line_item_legal_entity like '%Red Hat%') OR (line_item_legal_entity like
'%Amazon Web Services%' AND line_item_line_item_description like '%Red Hat%') OR
(line_item_legal_entity like '%Amazon Web Services%' AND line_item_line_item_description
like '%RHEL%') OR (line_item_legal_entity like '%AWS%' AND
line_item_line_item_description like '%Red Hat%') OR (line_item_legal_entity like '%AWS%'
AND line_item_line_item_description like '%RHEL%') OR (line_item_legal_entity like
'%AWS%' AND product_product_name like '%Red Hat%') OR (line_item_legal_entity like
'%Amazon Web Services%' AND product_product_name like '%Red Hat%')) AND year =
```

```
'{year}' AND month = '{month}'"

def lambda_handler(event, context):
    # Initiate Boto3 athena Client
    athena_client = boto3.client('athena')

    # Trigger athena query
    response = athena_client.start_query_execution(
        QueryString=query,
        QueryExecutionContext={
            'Database': database
        },
        ResultConfiguration={
            'OutputLocation': output
        }
    )

    # Save query execution to s3 object
    s3 = boto3.client('s3')
    json_object = {"integration_uuid": integration_uuid, "bill_year": year, "bill_month": month,
"query_execution_id": response.get("QueryExecutionId"), "result_prefix": output}
    s3.put_object(
        Body=json.dumps(json_object),
        Bucket=bucket,
        Key='query-data.json'
    )

    return json_object
```

将 **<your_integration_uuid>** 替换为您在 console.redhat.com 上创建的集成中的 UUID。将 **<your_S3_Bucket_Name>** 替换为您为存储报告创建的 S3 存储桶的名称。

9. 点 **Deploy** 来测试该功能。

2.6.3. 创建 Lambda 功能来发布报告文件

您必须创建一个 Lambda 功能，将报告文件发布到您创建的 S3 存储桶。

流程

1. 在 AWS 控制台中进入到 Lambda，然后点 **Create function**。
2. 点 Author from scratch
3. 输入您的功能的名称
4. 在 Runtime 下拉菜单中选择 python 3.7。
5. 选择 x86_64 作为 Architecture。
6. 在 Permissions 下，选择您创建的 Athena 角色。
7. 点 **Create function**。
8. 将以下代码粘贴到功能中：

```
import boto3
import json
import requests
from botocore.exceptions import ClientError


def get_credentials(secret_name, region_name):
    session = boto3.session.Session()
    client = session.client(
        service_name='secretsmanager',
        region_name=region_name
    )
    try:
        get_secret_value_response = client.get_secret_value(
            SecretId=secret_name
        )
    except ClientError as e:
        raise e
    secret = get_secret_value_response['SecretString']
    return secret

secret_name = "CHANGEME"
region_name = "us-east-1"
secret = get_credentials(secret_name, region_name)
json_creds = json.loads(secret)

USER = json_creds.get("<your_username>")      # console.redhat.com Username
PASS = json_creds.get("<your_password>")      # console.redhat.com  Password
bucket = "<your_S3_Bucket_Name>"              # Bucket for athena query results


def lambda_handler(event, context):
    # Initiate Boto3 s3 and fetch query file
    s3_resource = boto3.resource('s3')
    json_content = json.loads(s3_resource.Object(bucket, 'query-data.json').get()
['Body'].read().decode('utf-8'))

    # Initiate Boto3 athena Client and attempt to fetch athena results
    athena_client = boto3.client('athena')
    try:
        athena_results =
athena_client.get_query_execution(QueryExecutionId=json_content["query_execution_id"])
    except Exception as e:
        return f"Error fetching athena query results: {e} \n Consider increasing the time between
running and fetching results"

    reports_list = []
    prefix = json_content["result_prefix"].split(f'{bucket}/')[-1]

    # Initiate Boto3 s3 client
    s3_client = boto3.client('s3')
    result_data = s3_client.list_objects(Bucket=bucket, Prefix=prefix)
    for item in result_data.get("Contents"):
        if item.get("Key").endswith(".csv"):
            reports_list.append(item.get("Key"))
```

```
    # Post results to console.redhat.com API
    url = "https://console.redhat.com/api/cost-management/v1/ingress/reports/"
    json_data = {"source": json_content["integration_uuid"], "reports_list": reports_list,
"bill_year": json_content["bill_year"], "bill_month": json_content["bill_month"]}
    resp = requests.post(url, json=json_data, auth=(USER, PASS))

    return resp
```

将 **<your_username>** 替换为您的 console.redhat.com 的用户名。将 **<your_password>** 替换为您的 console.redhat.com 的密码。将 **<your_S3_Bucket_Name** > 替换为您为存储报告的 S3 存储桶的名称。

9. 点 **Deploy** 来测试该功能。

对红帽文档提供反馈

如果您发现了错误，或者对如何改进这些指南有建议，请在 成本管理 JIRA 板 中创建一个问题并添加 **Documentation** 标签。

非常感谢您的反馈意见！