RED HAT SUMMIT

Hybrid Cloud: Integrating AWS services with OpenShift

Derek Whatley
Software Engineer, Red Hat

Mandus Momberg
Solutions Architect, Amazon

May 8, 2018

Amazon Web Services & OpenShift

Stronger Together

OPENSHIFT

aws

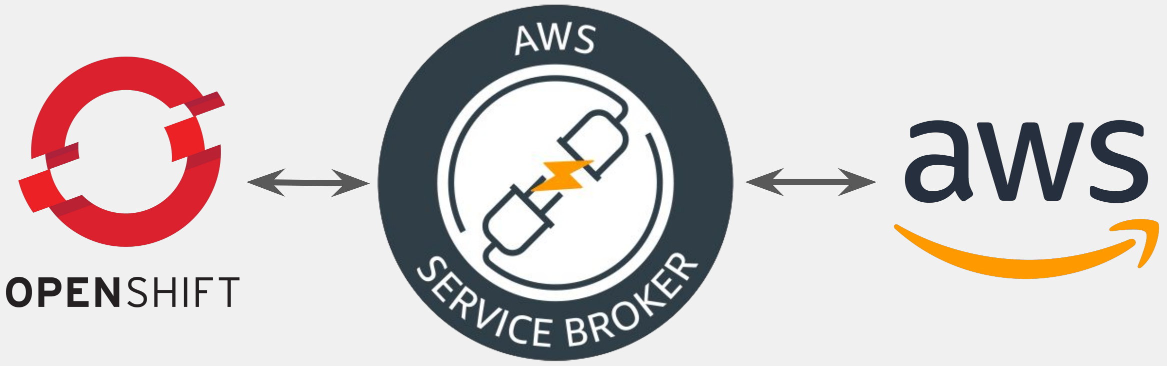
- Container Platform by Red Hat
  - Kubernetes orchestration
  - Powerful web UI
  - Developer centric tools
- Runs in *ANY* environment

- Reliable, scalable cloud computing
- Provides hundreds of web services
- Offerings in 18 geographic regions
- Pay only for what you use

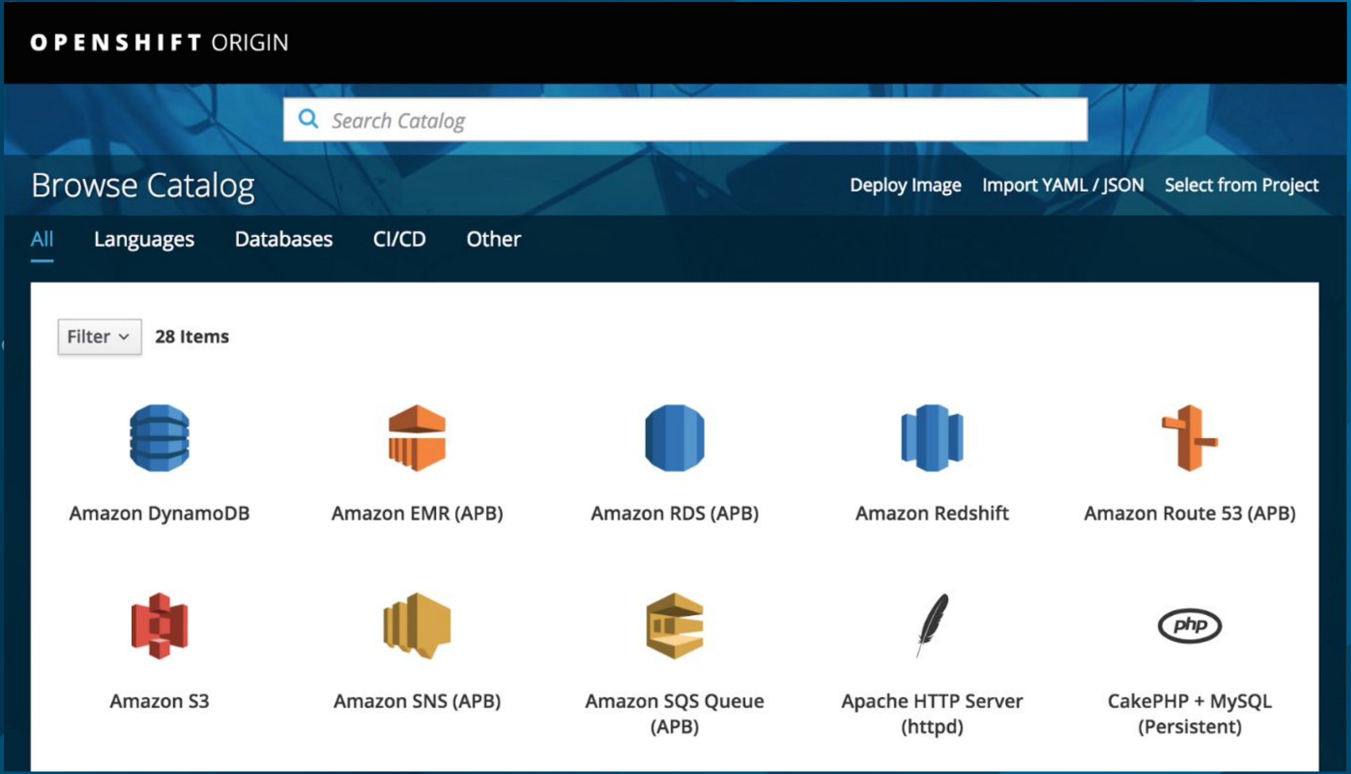
- *Combine the strengths* of OpenShift and AWS
    - Run containerized apps with OpenShift
    - Attach AWS services to your apps as needed
    - Instant scalability and reliability
- Made possible with the AWS Service Broker

Introducing the AWS Service Broker

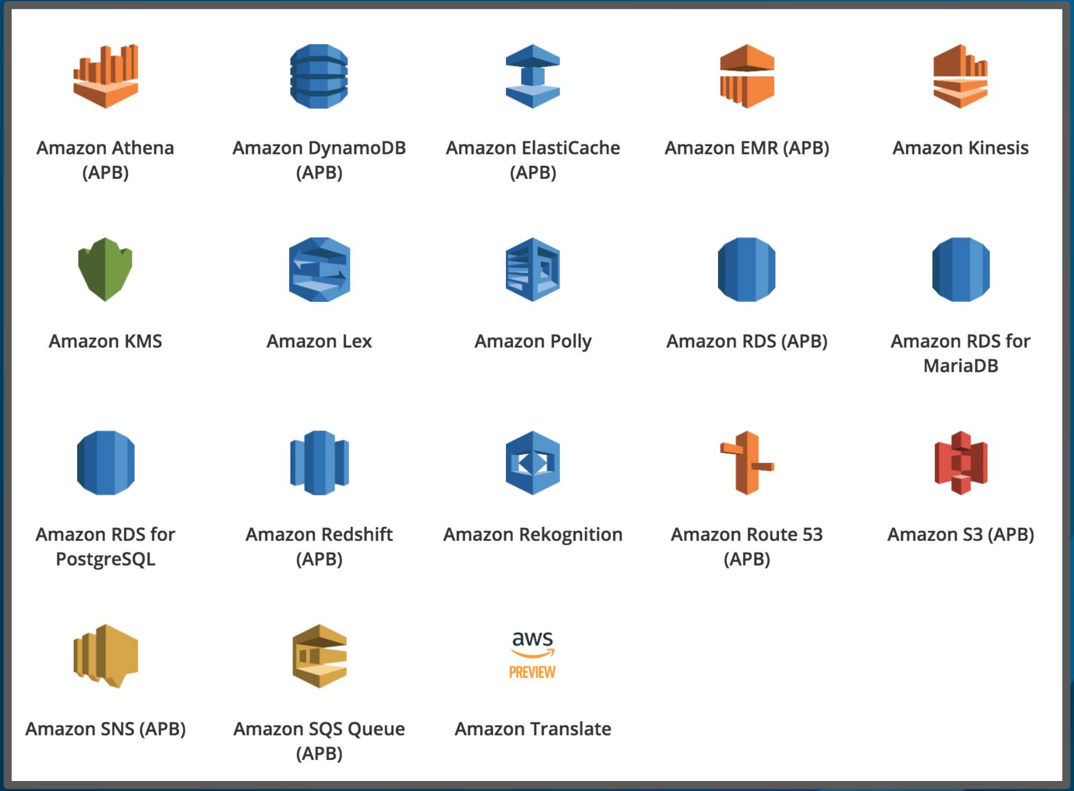
- Collaboration between AWS and Red Hat
- Enables AWS service management from the OpenShift Service Catalog
- Based on open source projects and open standards



AWS in the OpenShift Service Catalog



18 Supported AWS Services



Amazon Athena (APB)

Amazon DynamoDB (APB)

Amazon ElastiCache (APB)

Amazon EMR (APB)

Amazon Kinesis

Amazon KMS

Amazon Lex

Amazon Polly

Amazon RDS (APB)

Amazon RDS for MariaDB

Amazon RDS for PostgreSQL

Amazon Redshift (APB)

Amazon Rekognition

Amazon Route 53 (APB)

Amazon S3 (APB)

Amazon SNS (APB)

Amazon SQS Queue (APB)

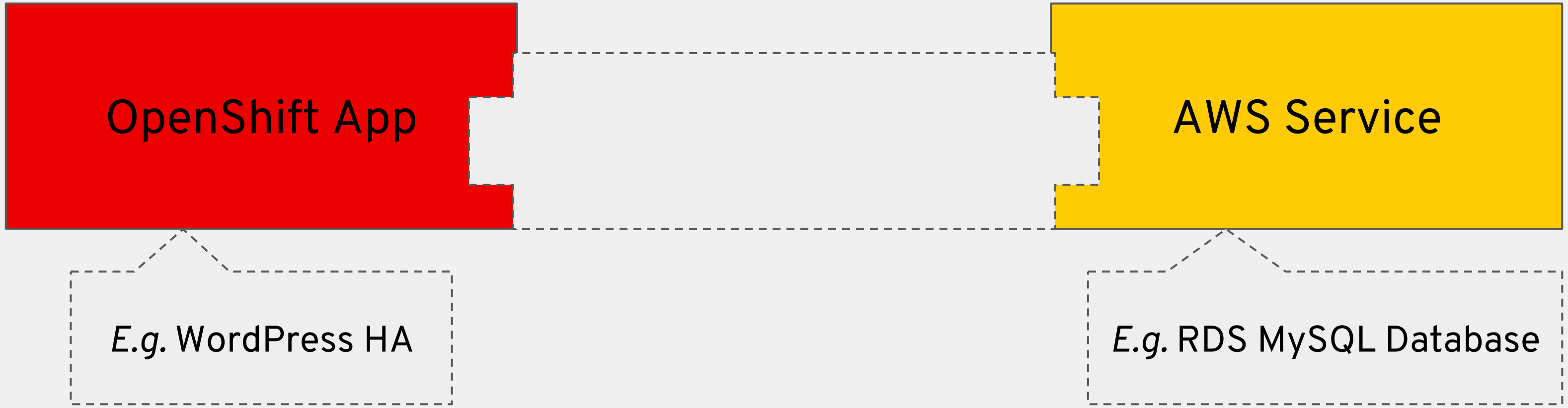
Amazon Translate

Case Study: *A minimal hybrid cloud app*

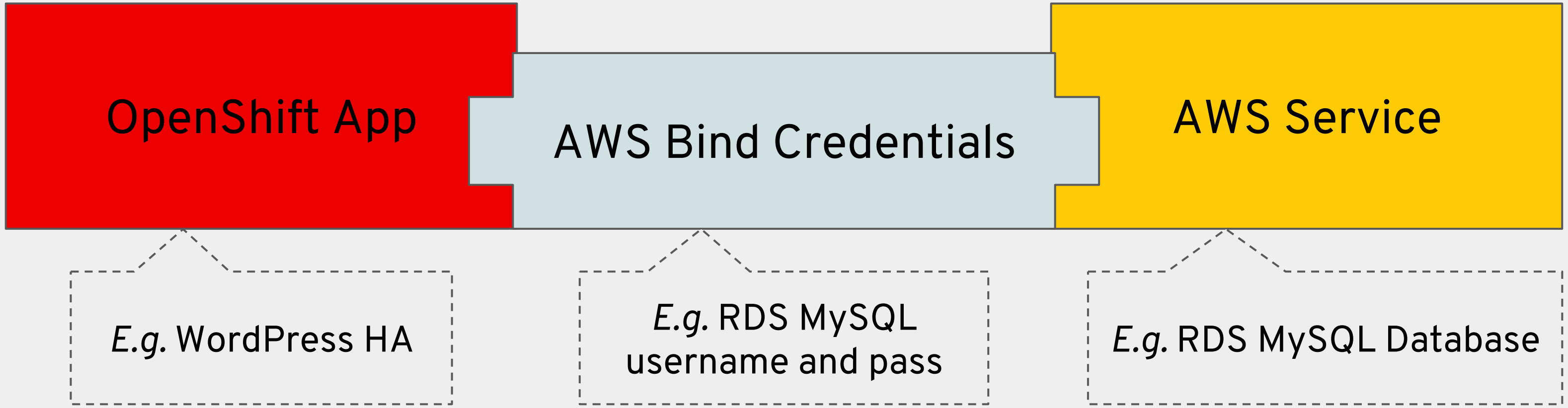
Case Study: A minimal hybrid cloud app

OpenShift App
E.g. WordPress HA

Case Study: A minimal hybrid cloud app



Case Study: A minimal hybrid cloud app



Case Study: A minimal hybrid cloud app

*How you **might** be doing this today*

~ 11 steps, 2 interfaces

not self-service

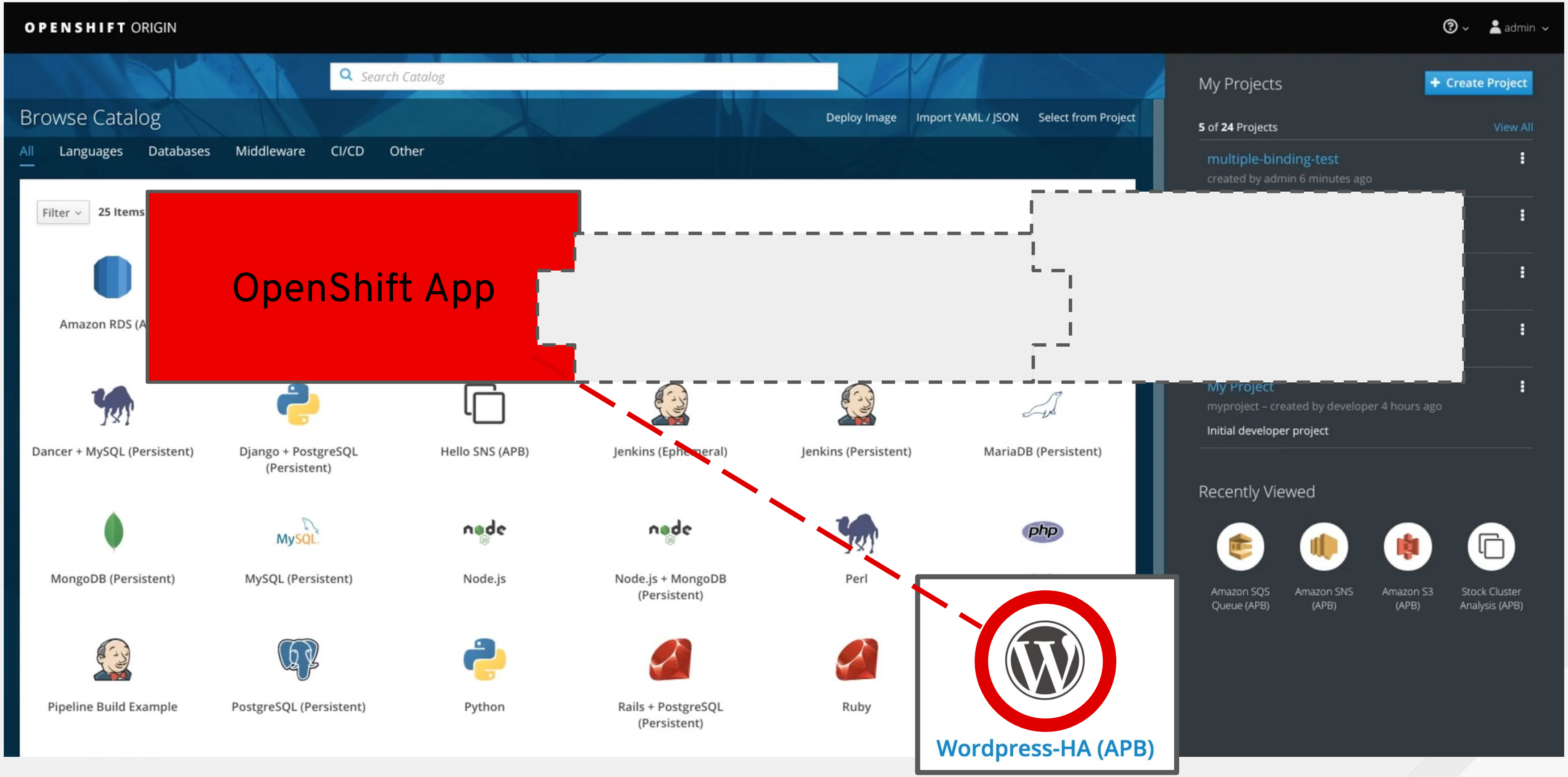
who provides support?

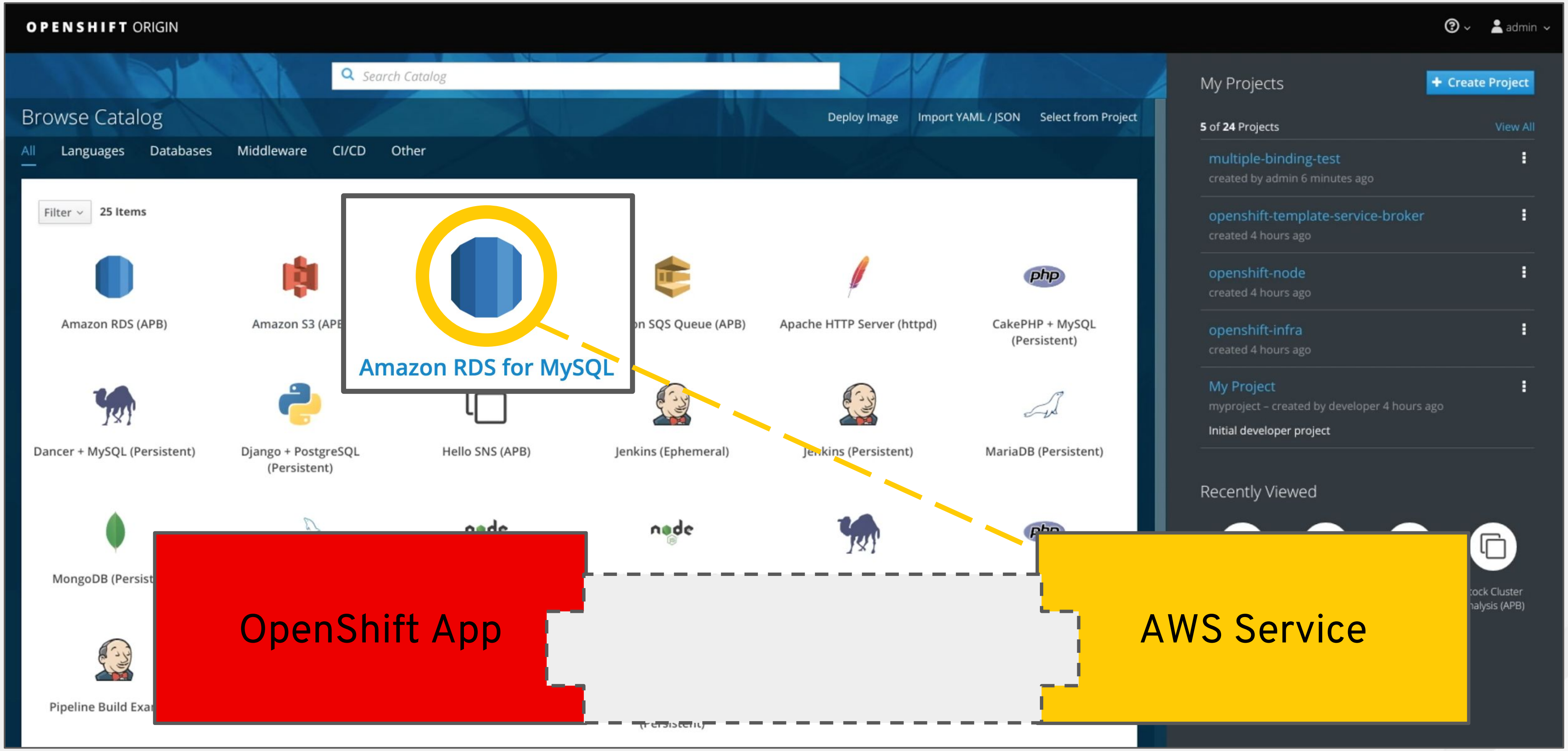
1. Go to OpenShift Service Catalog
    a. Create OpenShift app
2. Open AWS Web Console
    a. Open appropriate service dashboard
        i. Create AWS service
    b. Open IAM service dashboard
        i. Create IAM user
        ii. Find/write a policy giving user access to AWS service
        iii. Create AWS access keys
3. Copy AWS access keys into an OpenShift secret
4. Associate newly created secret with OpenShift app

Case Study: A minimal hybrid cloud app
*How you **should** be doing this today*

- ~ 4 steps, 1 interface
- self-service
- unified support path

1. Go to OpenShift Service Catalog
   a. Create OpenShift app
   b. Create AWS service
2. Attach AWS service binding to OpenShift app


OPENSHIFT ORIGIN
admin
Search Catalog
Browse Catalog
Deploy Image
Import YAML / JSON
Select from Project
All
Languages
Databases
Middleware
CI/CD
Other
Filter
25 Items
OpenShift App
Amazon RDS (A
Dancer + MySQL (Persistent)
Django + PostgreSQL (Persistent)
Hello SNS (APB)
Jenkins (Ephemeral)
Jenkins (Persistent)
MariaDB (Persistent)
MongoDB (Persistent)
MySQL (Persistent)
Node.js
Node.js + MongoDB (Persistent)
Perl
Pipeline Build Example
PostgreSQL (Persistent)
Python
Rails + PostgreSQL (Persistent)
Ruby
Wordpress-HA (APB)
My Projects
Create Project
5 of 24 Projects
View All
multiple-binding-test
created by admin 6 minutes ago
My Project
myproject – created by developer 4 hours ago
Initial developer project
Recently Viewed
Amazon SQS Queue (APB)
Amazon SNS (APB)
Amazon S3 (APB)
Stock Cluster Analysis (APB)


OPENSHIFT ORIGIN

admin

Search Catalog

Browse Catalog

Deploy Image   Import YAML / JSON   Select from Project

All   Languages   Databases   Middleware   CI/CD   Other

Filter   25 Items

Amazon RDS (APB)
Amazon S3 (APB)
Amazon SQS Queue (APB)
Apache HTTP Server (httpd)
CakePHP + MySQL (Persistent)
Dancer + MySQL (Persistent)
Django + PostgreSQL (Persistent)
Hello SNS (APB)
Jenkins (Ephemeral)
Jenkins (Persistent)
MariaDB (Persistent)
MongoDB (Persist
Pipeline Build Exa

My Projects   Create Project

5 of 24 Projects   View All

multiple-binding-test
created by admin 6 minutes ago

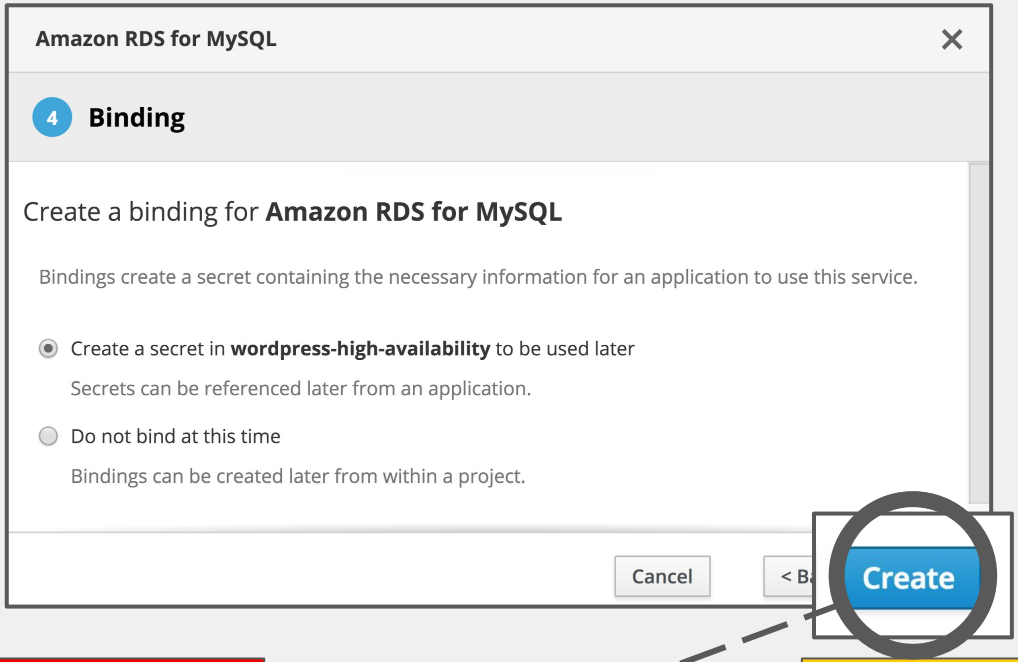
openshift-template-service-broker
created 4 hours ago

openshift-node
created 4 hours ago

openshift-infra
created 4 hours ago

My Project
myproject – created by developer 4 hours ago
Initial developer project

Recently Viewed



Amazon RDS for MySQL

4 Binding

Create a binding for **Amazon RDS for MySQL**

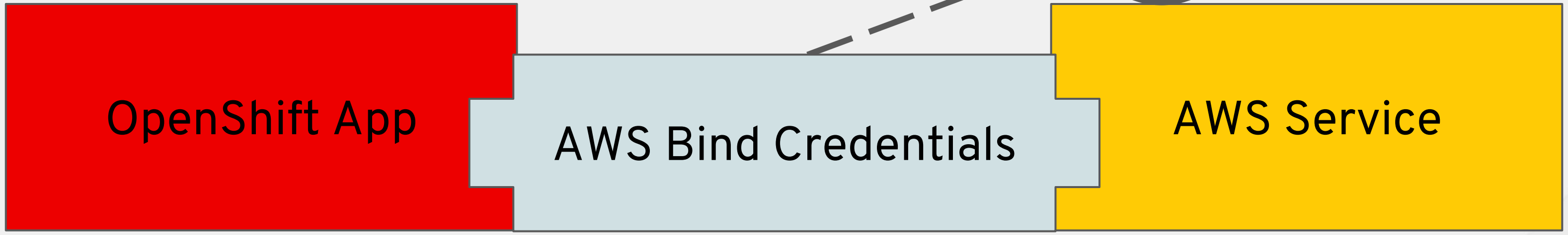
Bindings create a secret containing the necessary information for an application to use this service.

[✓ Create a secret in **wordpress-high-availability** to be used later]

Secrets can be referenced later from an application.

Do not bind at this time

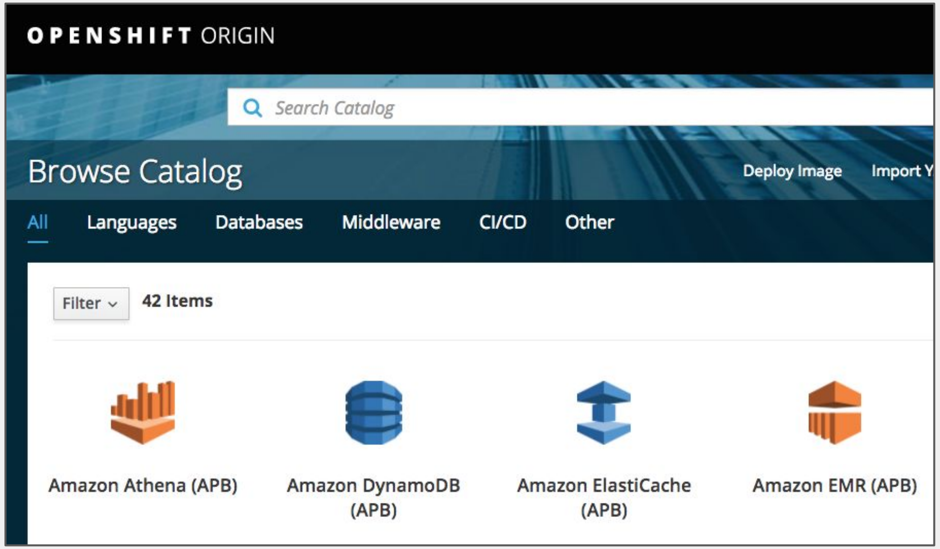
Bindings can be created later from within a project.

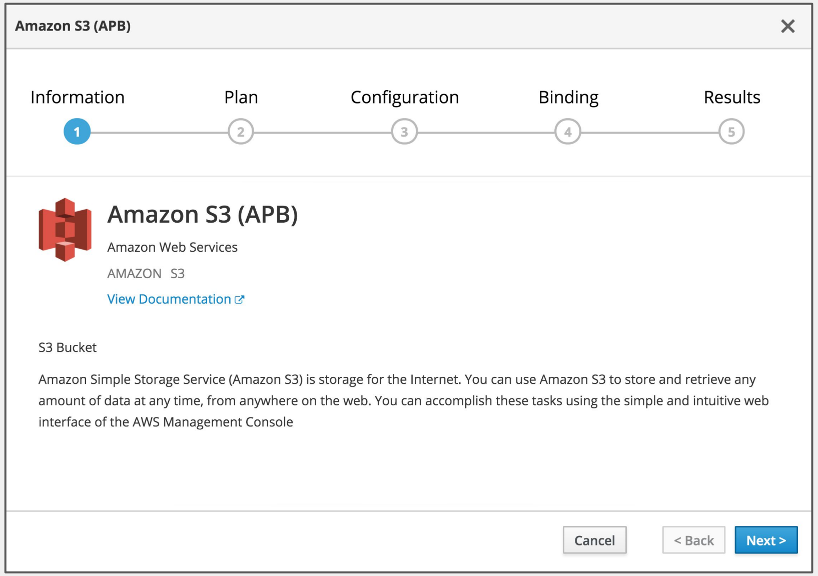


What is a *Service Broker*?

Service Catalog (*kubernetes-incubator/service-catalog*)

Where Services are Published





Amazon S3 (APB)

Amazon Web Services

AMAZON S3

View Documentation

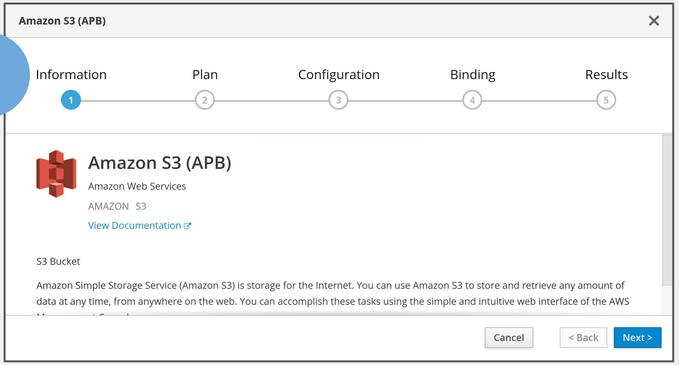
S3 Bucket

Amazon Simple Storage Service (Amazon S3) is storage for the Internet. You can use Amazon S3 to store and retrieve any amount of data at any time, from anywhere on the web. You can accomplish these tasks using the simple and intuitive web interface of the AWS Management Console

- **Provision and manage services from a central interface**
- **Guides users through service creation flow**
- **OpenShift has a *Service Catalog* component (shown above)**

Service Catalog (kubernetes-incubator/service-catalog)

1


Amazon S3 (APB)

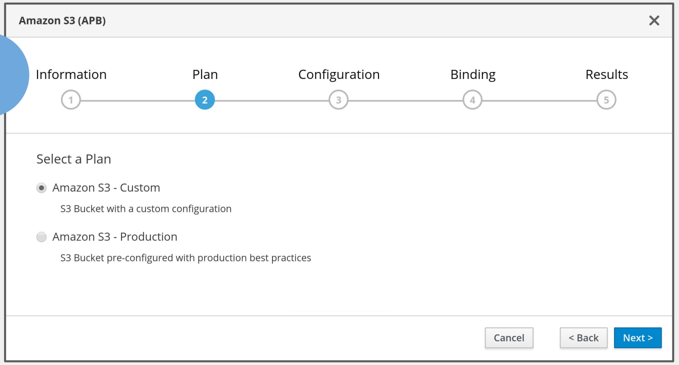
Information ① — Plan ② — Configuration ③ — Binding ④ — Results ⑤

Amazon S3 (APB)
Amazon Web Services
AMAZON S3
View Documentation

S3 Bucket

Amazon Simple Storage Service (Amazon S3) is storage for the Internet. You can use Amazon S3 to store and retrieve any amount of data at any time, from anywhere on the web. You can accomplish these tasks using the simple and intuitive web interface of the AWS

2


Amazon S3 (APB)

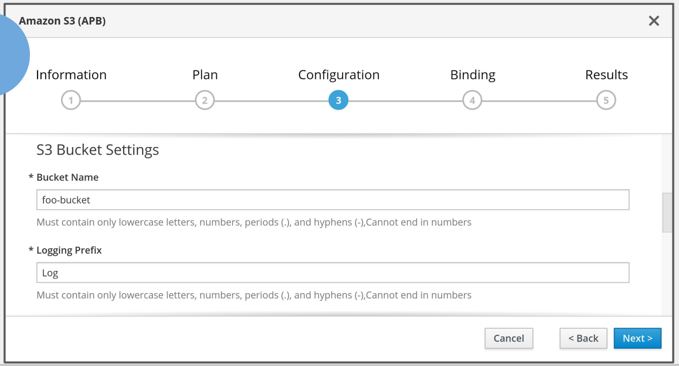
Information ① — Plan ② — Configuration ③ — Binding ④ — Results ⑤

Select a Plan

[✓ Amazon S3 - Custom]
S3 Bucket with a custom configuration

Amazon S3 - Production
S3 Bucket pre-configured with production best practices

3


Amazon S3 (APB)

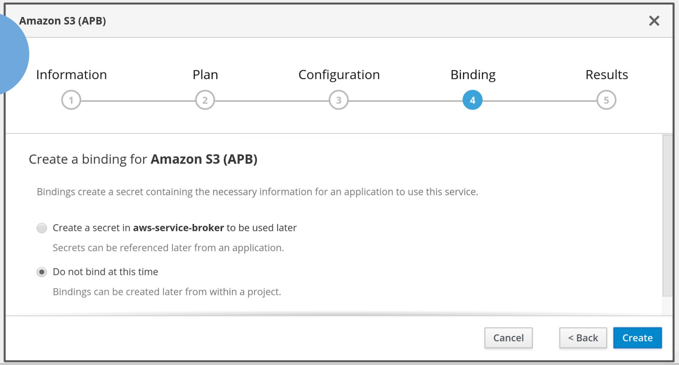
Information ① — Plan ② — Configuration ③ — Binding ④ — Results ⑤

S3 Bucket Settings

* Bucket Name
foo-bucket
Must contain only lowercase letters, numbers, periods (.), and hyphens (-),Cannot end in numbers

* Logging Prefix
Log
Must contain only lowercase letters, numbers, periods (.), and hyphens (-),Cannot end in numbers

4


Amazon S3 (APB)

Information ① — Plan ② — Configuration ③ — Binding ④ — Results ⑤

Create a binding for **Amazon S3 (APB)**

Bindings create a secret containing the necessary information for an application to use this service.

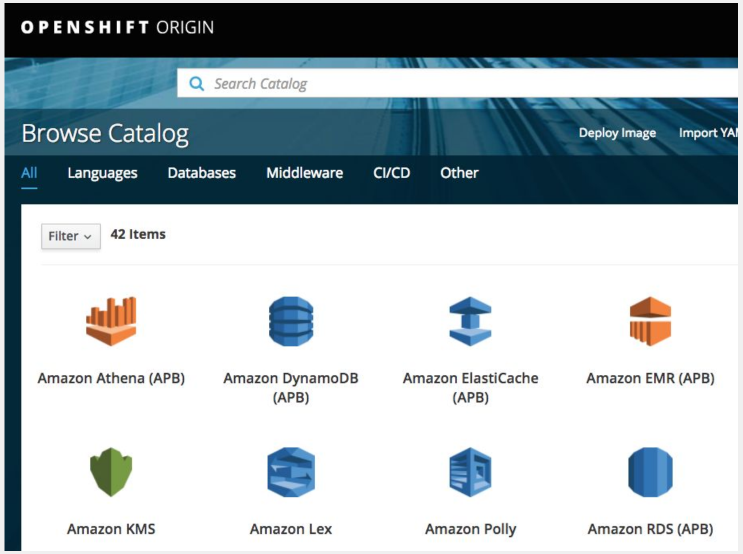
Create a secret in **aws-service-broker** to be used later
Secrets can be referenced later from an application.

[✓ Do not bind at this time]
Bindings can be created later from within a project.

What is a Service Broker?

Service Catalog and Brokers

Expose and Provision Services


OPENSHIFT ORIGIN

Search Catalog

Browse Catalog

Deploy Image

Import YA

All · Languages · Databases · Middleware · CI/CD · Other

Filter

42 Items

Amazon Athena (APB)

Amazon DynamoDB (APB)

Amazon ElastiCache (APB)

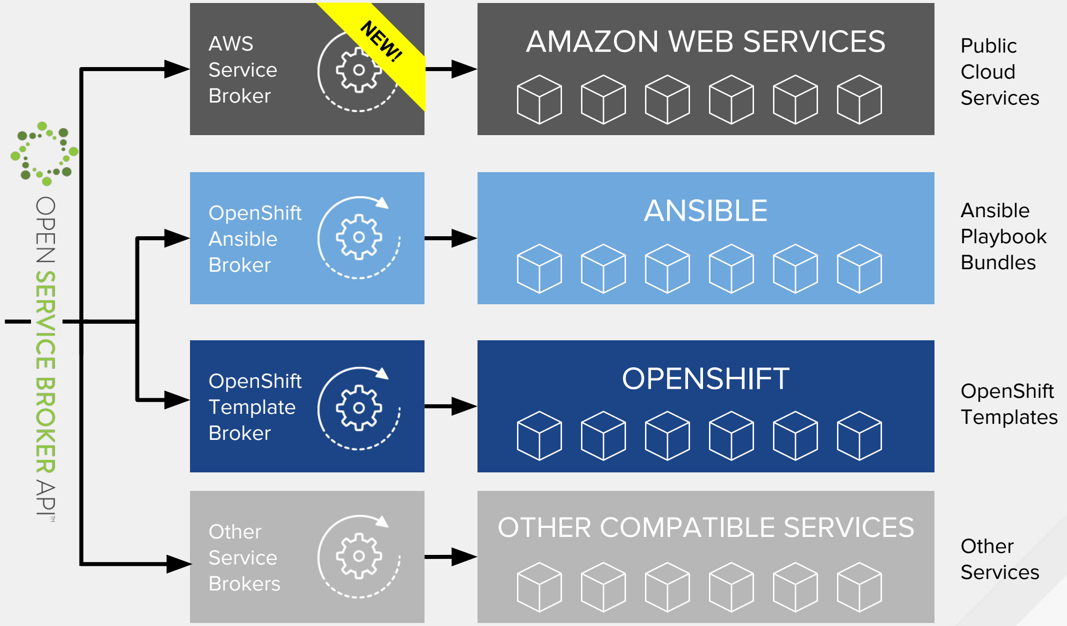
Amazon EMR (APB)

Amazon KMS

Amazon Lex

Amazon Polly

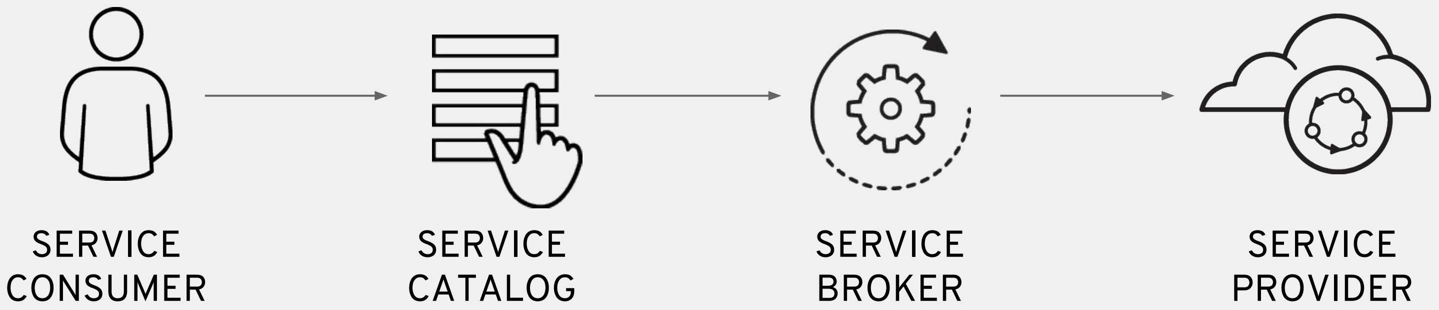
Amazon RDS (APB)



OPENSHIFT SERVICE CATALOG **SERVICE BROKERS**

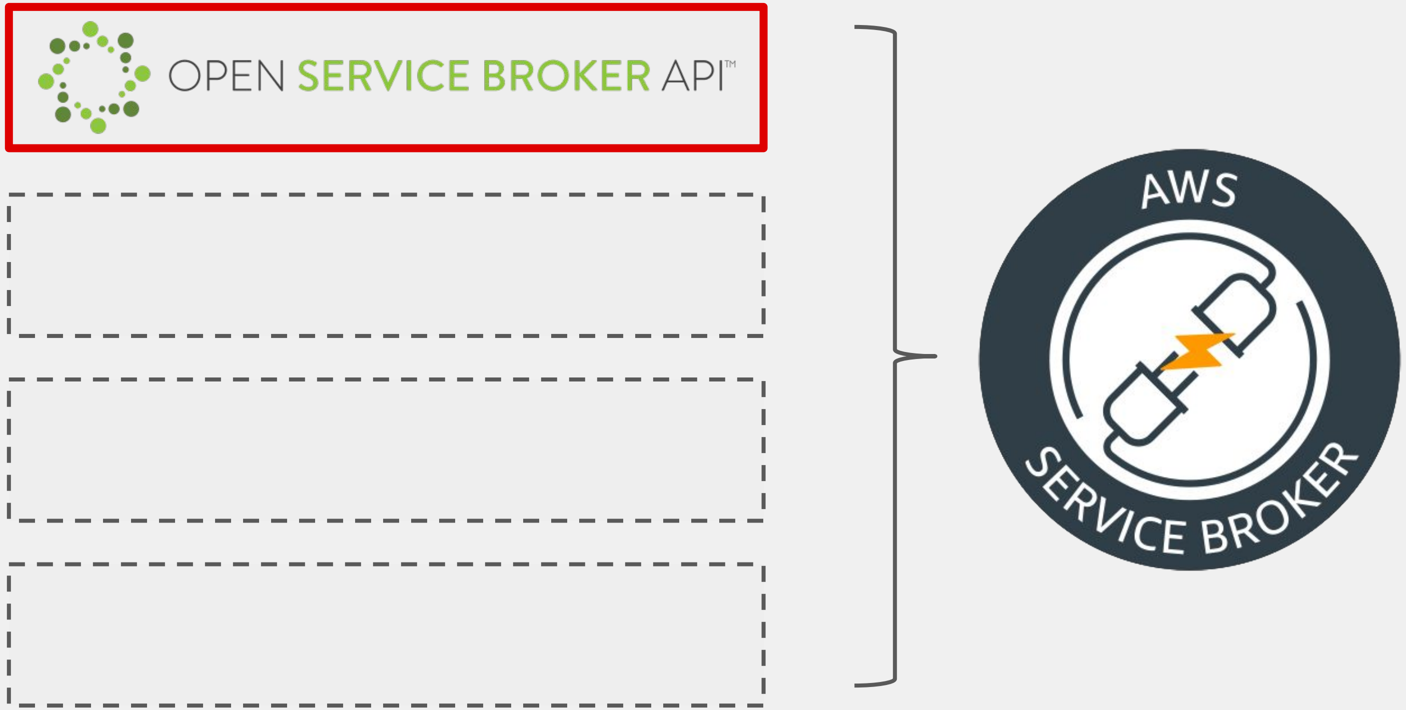
Service Catalog + AWS Broker Demo

The Service Catalog + Broker Workflow

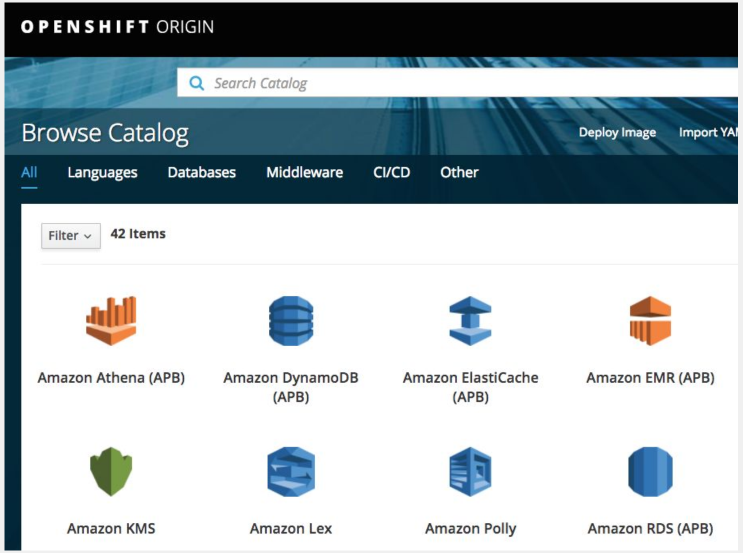


How we *built* the AWS Broker

AWS Broker: How It's Made



Open Service Broker API in action



OPENSHIFT ORIGIN

Search Catalog

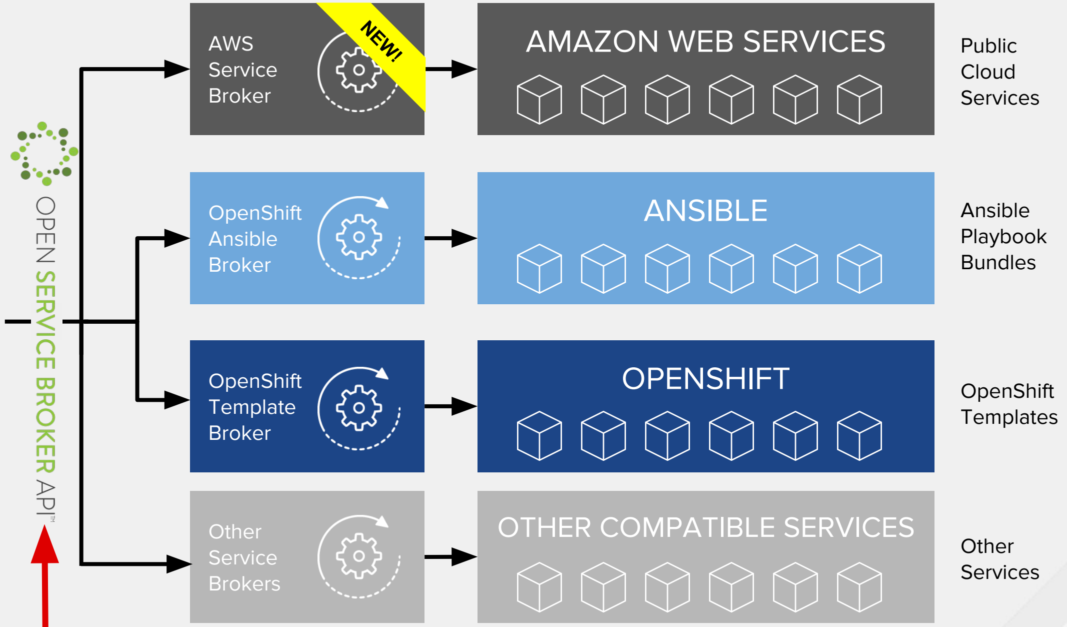
Browse Catalog

Deploy Image | Import YA

All | Languages | Databases | Middleware | CI/CD | Other

Filter | 42 Items

Amazon Athena (APB)	Amazon DynamoDB (APB)	Amazon ElastiCache (APB)	Amazon EMR (APB)
Amazon KMS	Amazon Lex	Amazon Polly	Amazon RDS (APB)



OPENSHIFT SERVICE CATALOG

SERVICE BROKERS

Interface between Service Marketplace and Brokers

Open Service Broker API

Defines an HTTP interface between service marketplaces and service brokers

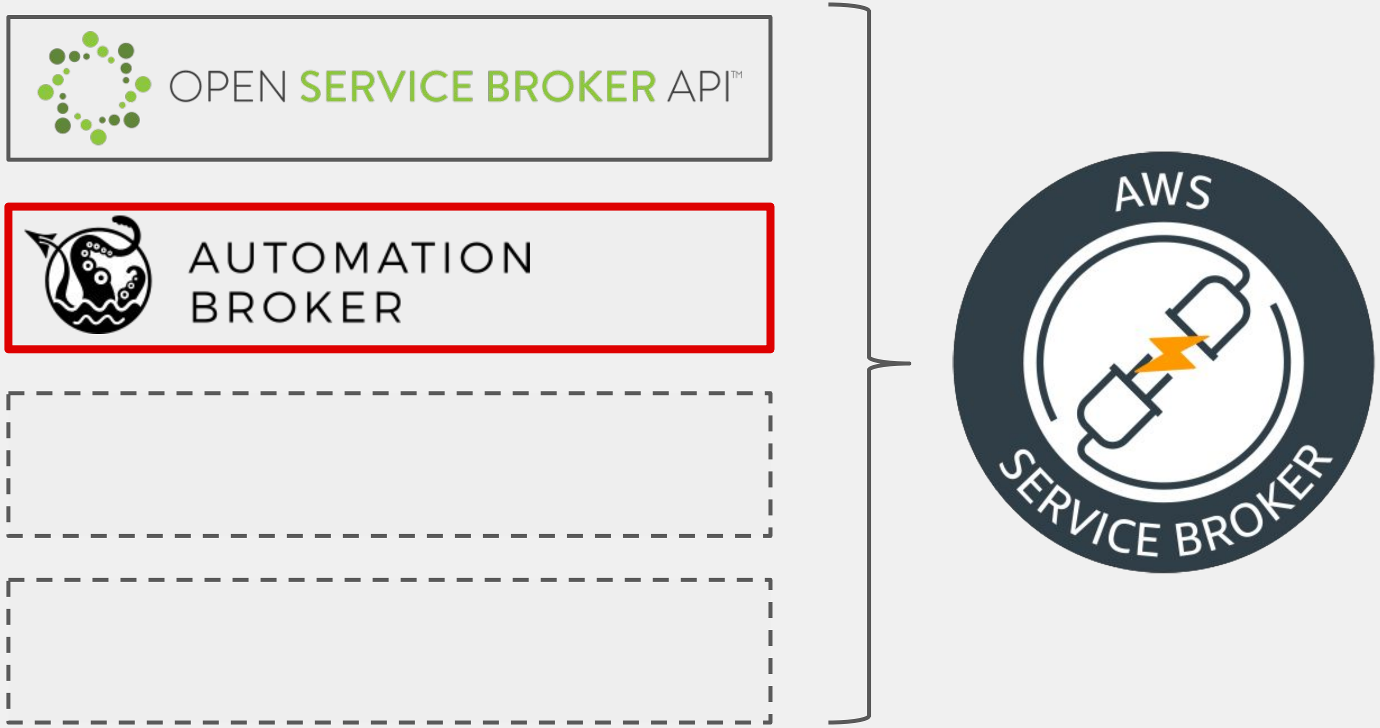
Background

- Aims to standardize how services are consumed on cloud platforms
- Product of a multi-vendor working group formed in September 2016

- ***Service Brokers*** implement the API
  - AWS Service Broker
  - OpenShift Ansible Broker

- ***Service Marketplaces*** give users access Service Broker offerings.
  - OpenShift Service Catalog

AWS Broker: How It's Made

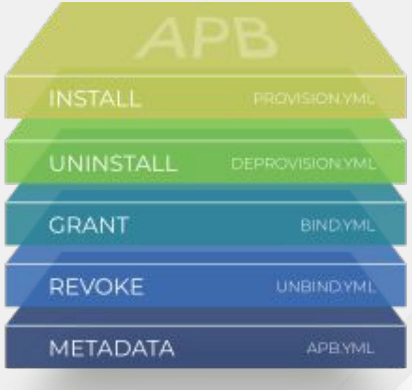


Automation Broker

- Open source project **powering the AWS Broker**
- Uses Ansible Playbook Bundles (APBs) to provision apps and services

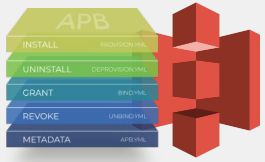
Ansible Playbook Bundle (APB)

- Collection of named Ansible playbooks
- **Each playbook handles a service management action**
  - `provision.yml   -` *install*
  - `deprovision.yml -` *uninstall*
- Containerized with an embedded Ansible runtime

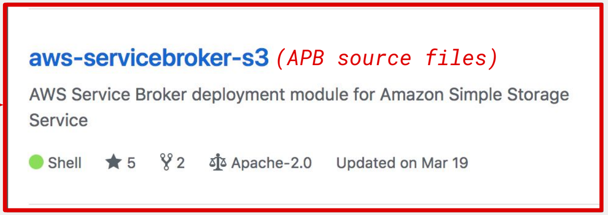


Ansible Playbook Bundles (APBs) for AWS

- Each AWS service has a corresponding Ansible Playbook Bundle on GitHub
- These APBs contain Ansible Playbooks for managing AWS services with CloudFormation



S3 Management Playbooks

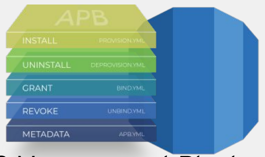


aws-servicebroker-s3 *(APB source files)*

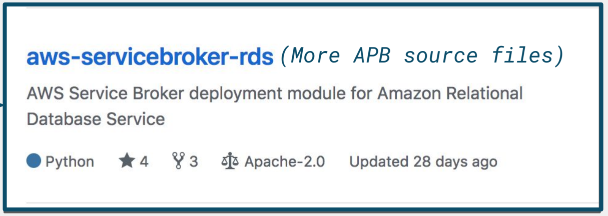
AWS Service Broker deployment module for Amazon Simple Storage Service

● Shell ★ 5 2 Apache-2.0 Updated on Mar 19

```
- name: Launch S3
  cloudformation:
   stack_name: "apb-s3-{{ stack_suffix }}"
   state: "present"
   region: "{{ region }}"
```



RDS Management Playbooks



aws-servicebroker-rds *(More APB source files)*

AWS Service Broker deployment module for Amazon Relational Database Service

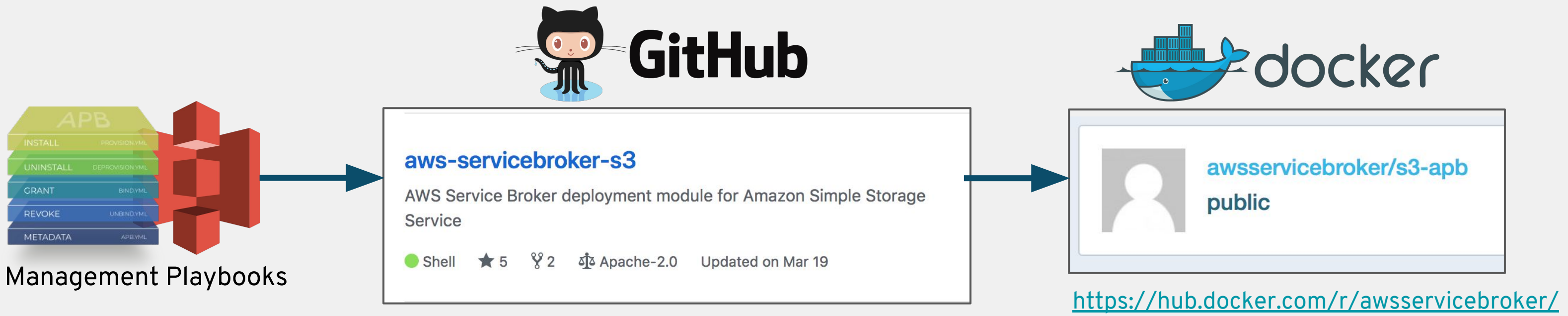
● Python ★ 4 3 Apache-2.0 Updated 28 days ago

```
- name: Launch RDS MySQL cluster
  cloudformation:
    stack_name: "apb-rds-{{ stack_suffix }}"
    state: "present"
    region: "{{ region }}"
```

https://github.com/awslabs?q=aws-servicebroker

Ansible Playbook Bundles (APBs) for AWS

- Containers are built from AWS Labs GitHub repos and published on Docker Hub
- AWS Broker gets list of available AWS services from `docker.io/awsservicebroker`



S3 Management Playbooks

aws-servicebroker-s3

AWS Service Broker deployment module for Amazon Simple Storage Service

Shell ★ 5 2 Apache-2.0 Updated on Mar 19

awsservicebroker/s3-apb

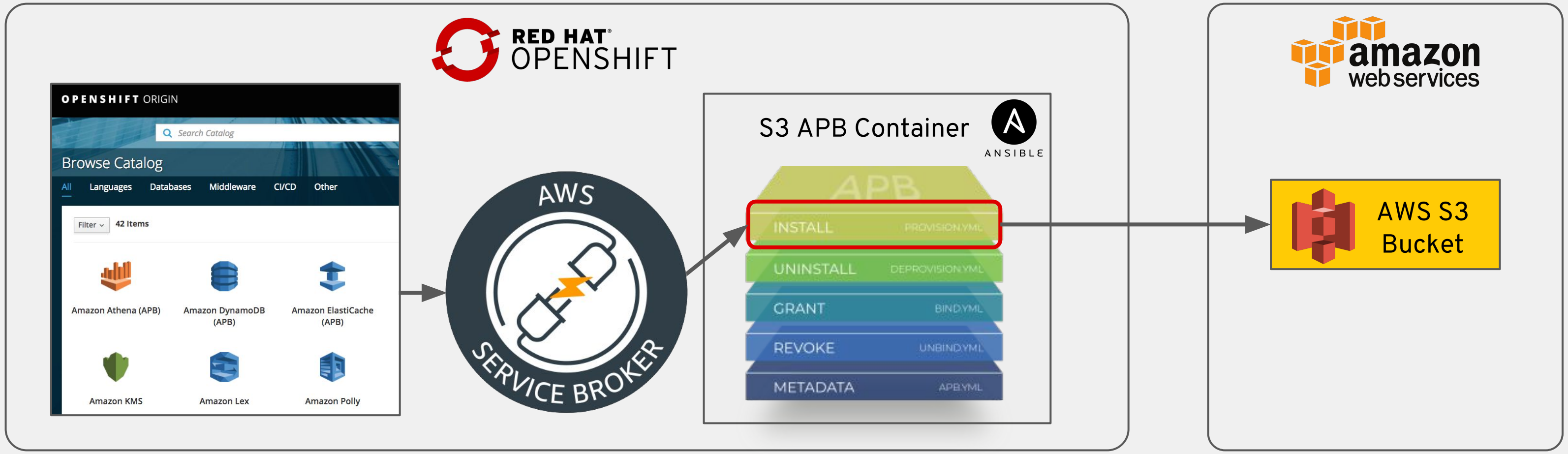
public

https://github.com/awslabs?q=aws-servicebroker

https://hub.docker.com/r/awsservicebroker/

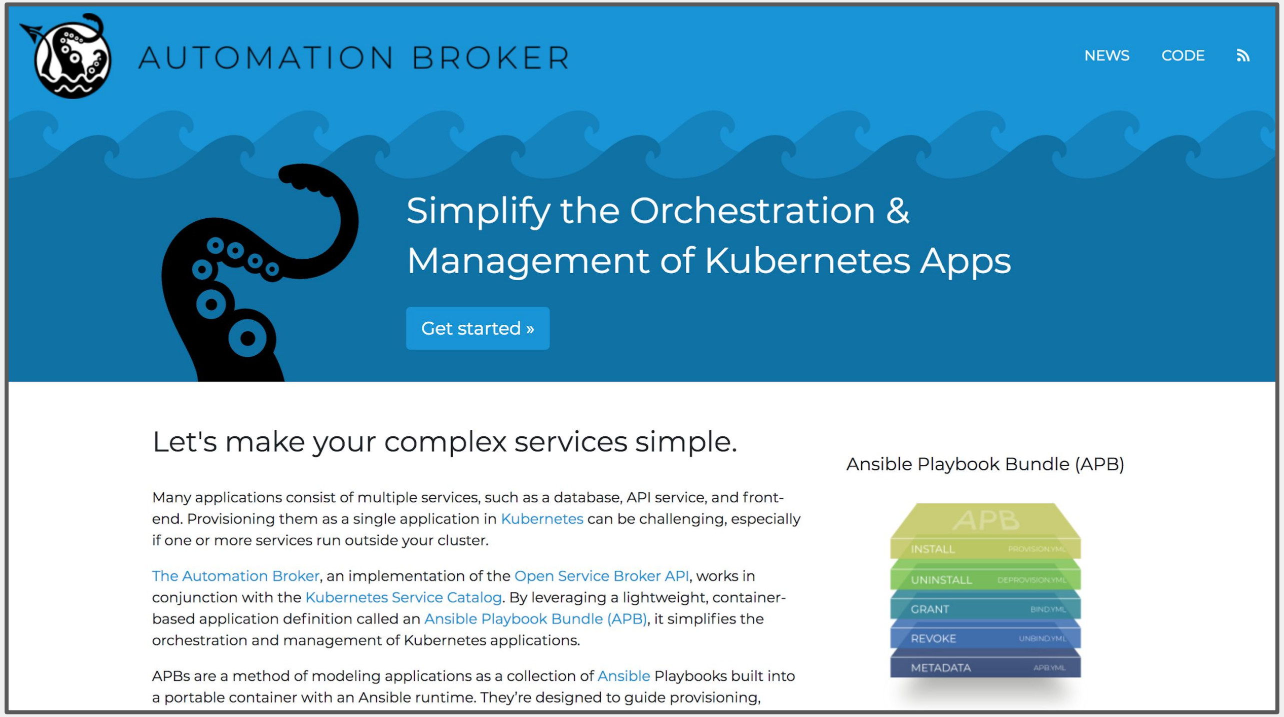
Ansible Playbook Bundles (APBs) for AWS

- When an OpenShift user requests an AWS service, an APB container runs to complete the task



User requests new S3 bucket, request sent to AWS Broker	AWS Broker starts S3 APB container	S3 APB container runs Ansible Playbook `provision.yml`	AWS S3 Bucket is created

automationbroker.io



Simplify the Orchestration & Management of Kubernetes Apps

Let's make your complex services simple.

Many applications consist of multiple services, such as a database, API service, and front-end. Provisioning them as a single application in Kubernetes can be challenging, especially if one or more services run outside your cluster.

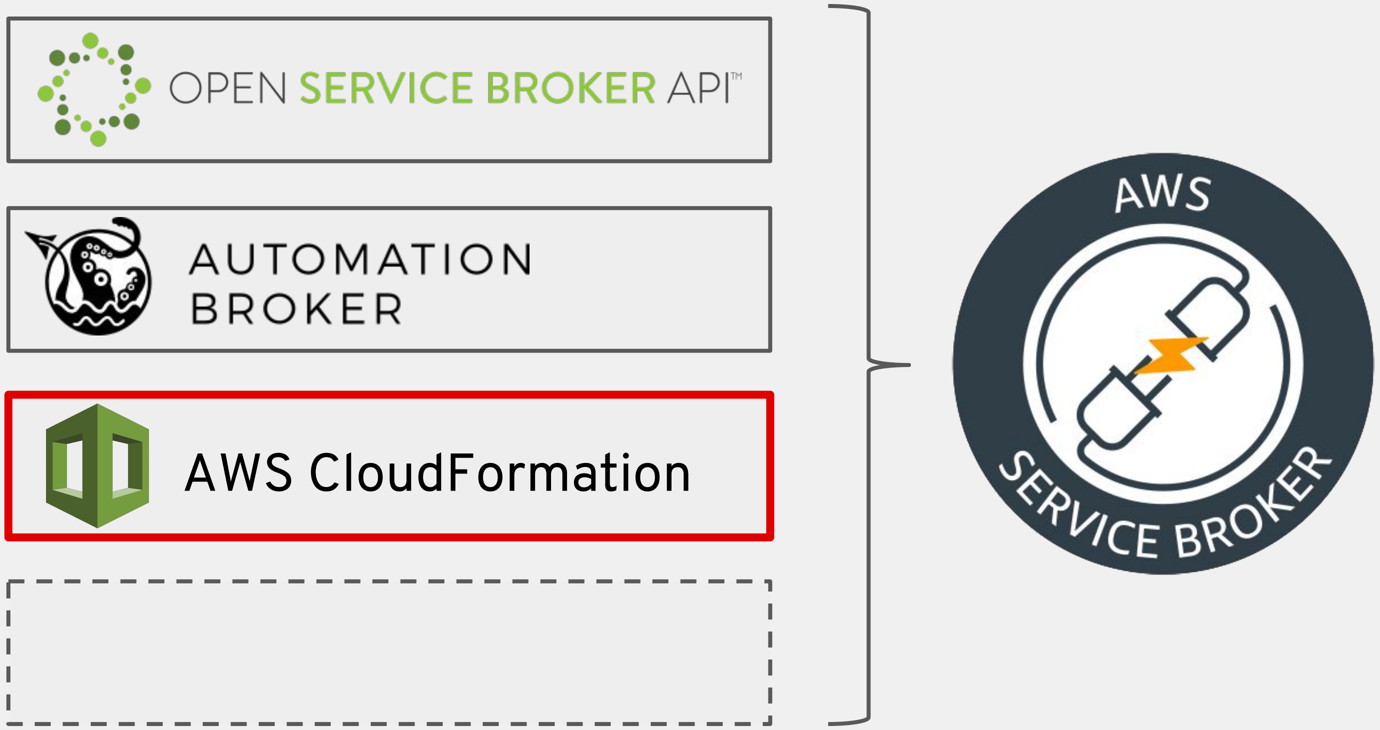
The Automation Broker, an implementation of the Open Service Broker API, works in conjunction with the Kubernetes Service Catalog. By leveraging a lightweight, container-based application definition called an Ansible Playbook Bundle (APB), it simplifies the orchestration and management of Kubernetes applications.

APBs are a method of modeling applications as a collection of Ansible Playbooks built into a portable container with an Ansible runtime. They're designed to guide provisioning,

Ansible Playbook Bundle (APB)

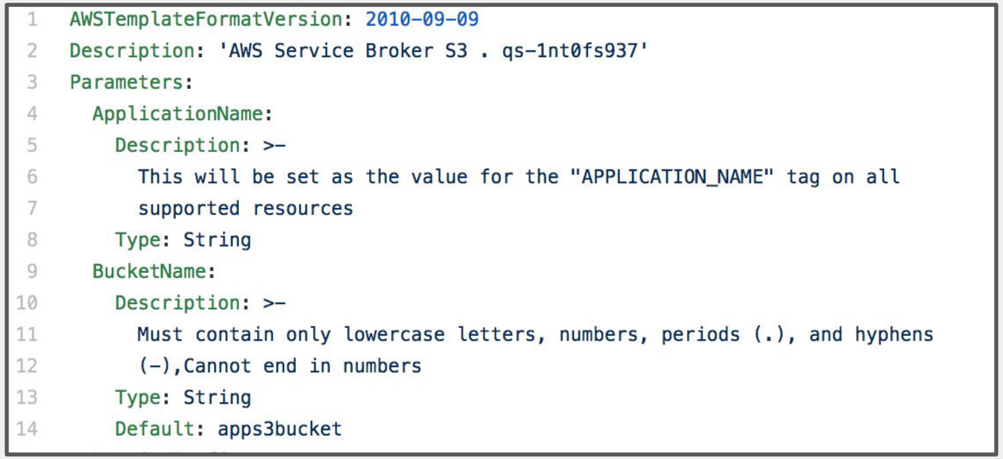
twitter.com/autom8broker/

AWS Broker: How It's Made



AWS CloudFormation

- *Infrastructure as code* for AWS, written in JSON or YAML
- Used by the AWS Broker to create best-practice adherent AWS services

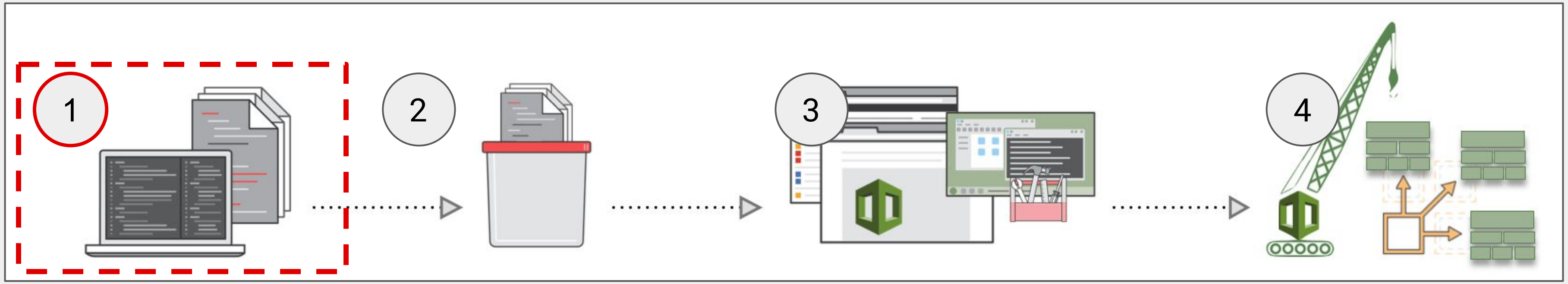


```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: 'AWS Service Broker S3 . qs-1nt0fs937'
Parameters:
  ApplicationName:
    Description: >-
      This will be set as the value for the "APPLICATION_NAME" tag on all
      supported resources
    Type: String
  BucketName:
    Description: >-
      Must contain only lowercase letters, numbers, periods (.), and hyphens
      (-),Cannot end in numbers
    Type: String
    Default: apps3bucket
```

https://github.com/awslabs/aws-servicebroker-s3/blob/master/roles/provision-s3-apb-openshift/files/S3Bucket.yml

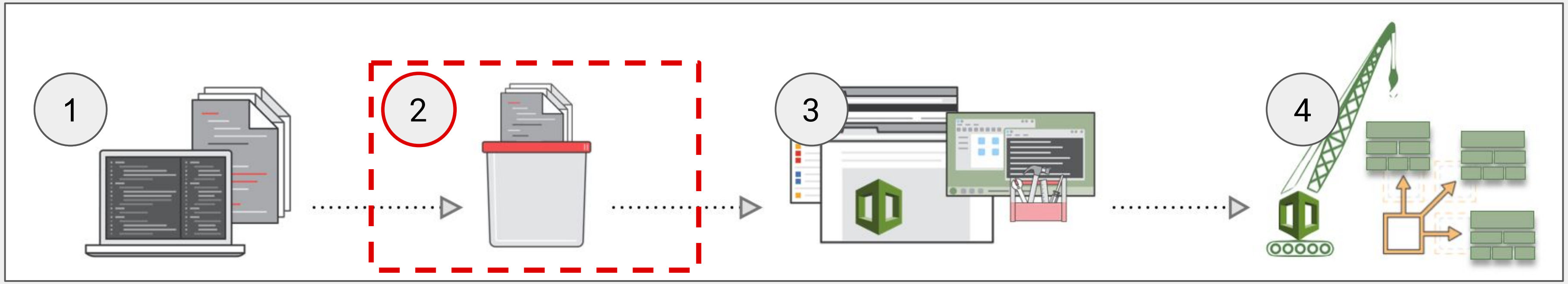
AWS CloudFormation

→ Step 1: AWS experts write CloudFormation templates
Step 2: CloudFormation templates are packaged into APBs
Step 3: User requests an AWS Service
Step 4: APB container runs, uses Ansible Playbooks + CloudFormation to create service



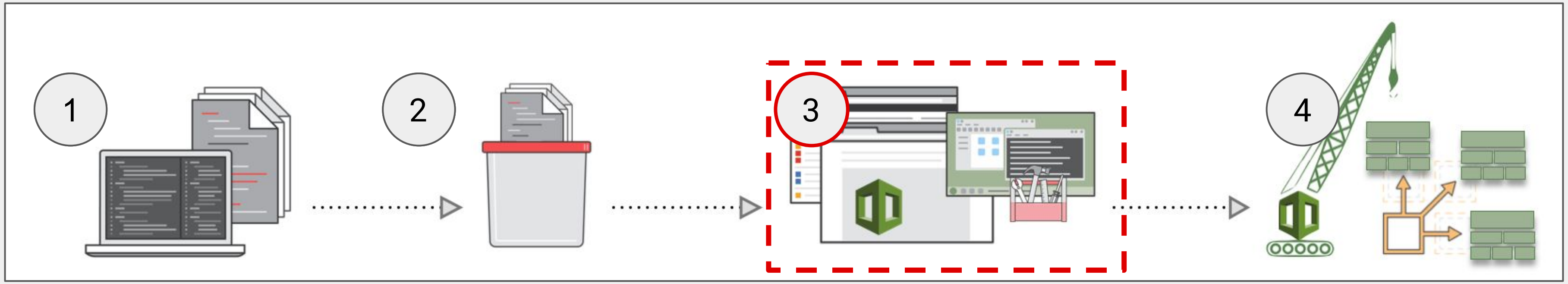
AWS CloudFormation

```
Step 1: AWS experts write CloudFormation templates
```
→ **Step 2: CloudFormation templates are packaged into APBs**
```
Step 3: User requests an AWS Service
Step 4: APB container runs, uses Ansible Playbooks + CloudFormation to create service
```



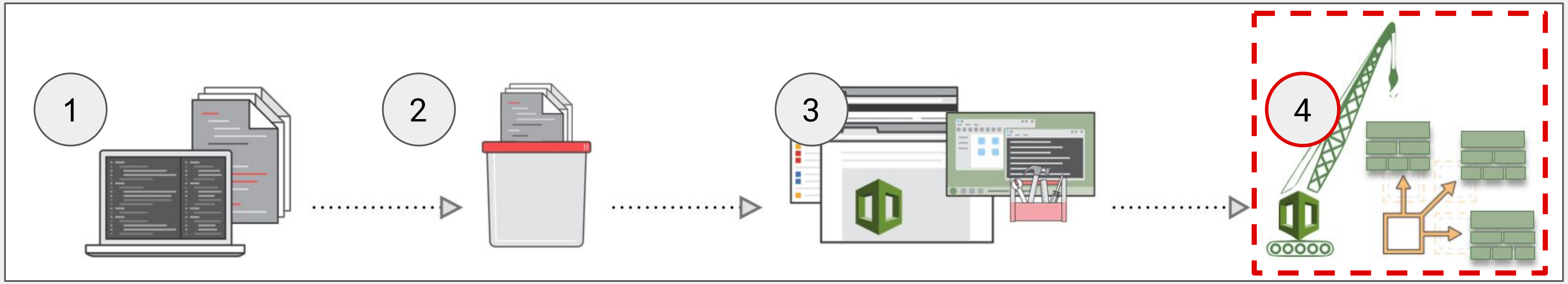
AWS CloudFormation

```
Step 1: AWS experts write CloudFormation templates
Step 2: CloudFormation templates are packaged into APBs
Step 3: User requests an AWS Service
Step 4: APB container runs, uses Ansible Playbooks + CloudFormation to create service
```



AWS CloudFormation

Step 1: AWS experts write CloudFormation templates
Step 2: CloudFormation templates are packaged into APBs
Step 3: User requests an AWS Service
→ **Step 4: APB container runs, uses Ansible Playbooks + CloudFormation to create service**



AWS Broker: How It's Made

OPEN SERVICE BROKER API™

AUTOMATION BROKER

AWS CloudFormation

AWS IAM

AWS SERVICE BROKER

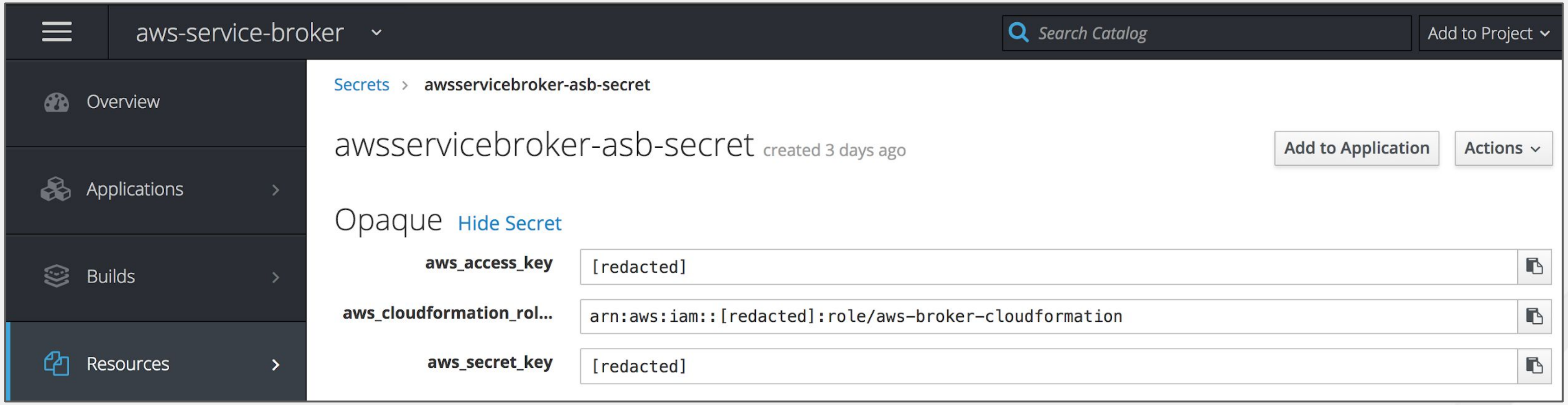
AWS IAM (Identity and Access Management)

- Enables *fine-grained* *access control* to AWS resources
- Used by AWS Broker to keep your master AWS access keys safe

IAM - Automatic Access Key Management

- OpenShift admin sets master AWS keys and IAM role to use for service management
  - Stored as a secret in the *aws-service-broker* namespace (see image below)

- OpenShift users can create AWS services without knowing master AWS keys
  - Users only have access to *scoped AWS keys*



awsservicebroker-asb-secret created 3 days ago

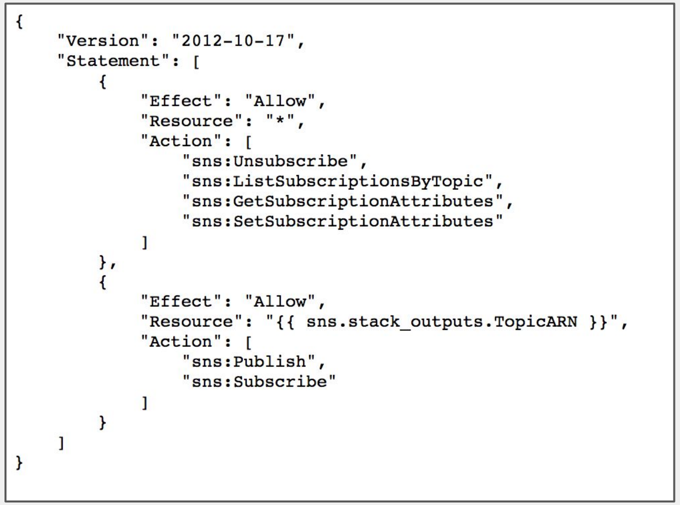
Opaque

aws_access_key	[redacted]
aws_cloudformation_rol...	arn:aws:iam::[redacted]:role/aws-broker-cloudformation
aws_secret_key	[redacted]

IAM - Scoped AWS Keys

- What are scoped AWS keys?
  - Gives app permission to talk to an AWS service
  - Limited in scope of permissions granted

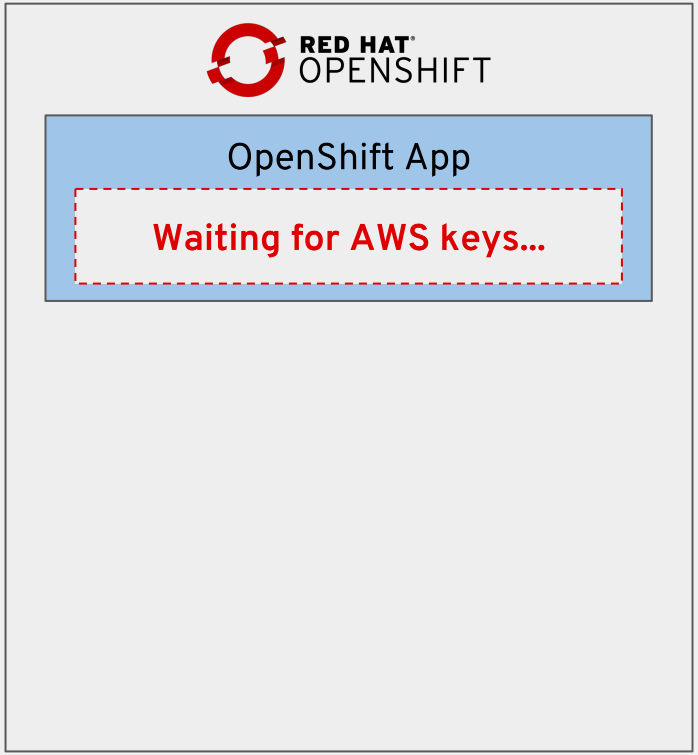
- Benefits of scoped AWS access keys
  - If bad actor gets keys, damage is limited
  - Enables self-service for regular users



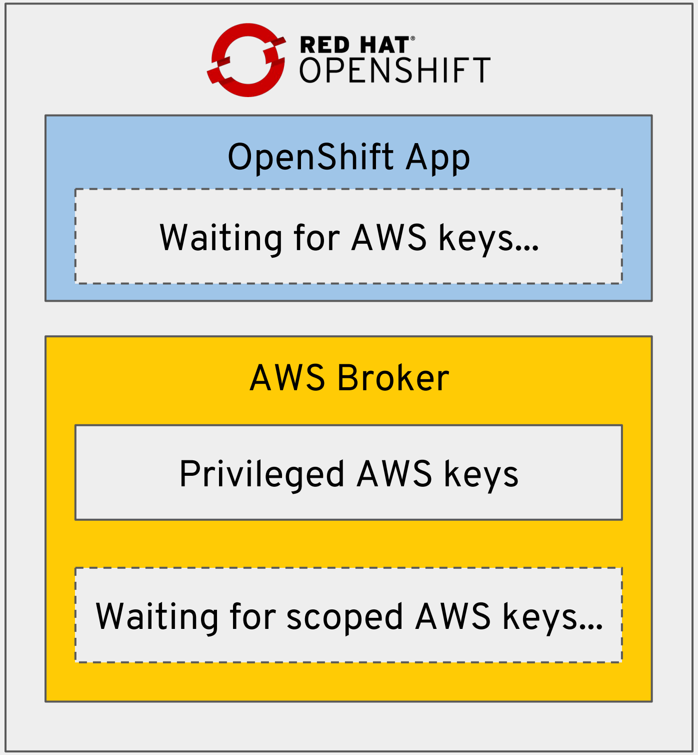
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": "*",
            "Action": [
                "sns:Unsubscribe",
                "sns:ListSubscriptionsByTopic",
                "sns:GetSubscriptionAttributes",
                "sns:SetSubscriptionAttributes"
            ]
        },
        {
            "Effect": "Allow",
            "Resource": "{{ sns.stack_outputs.TopicARN }}",
            "Action": [
                "sns:Publish",
                "sns:Subscribe"
            ]
        }
    ]
}
```

Sample IAM policy granting access to an SNS topic

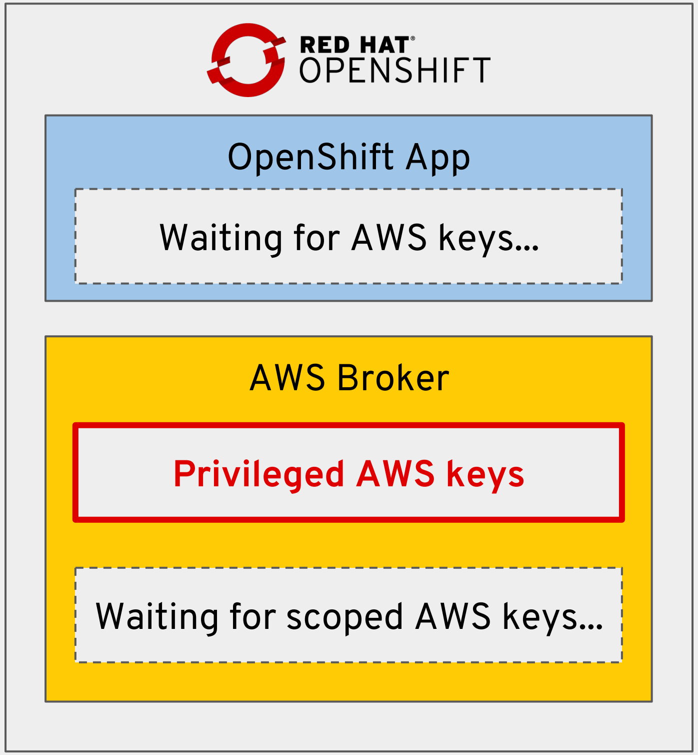
IAM - Creating Scoped AWS Keys



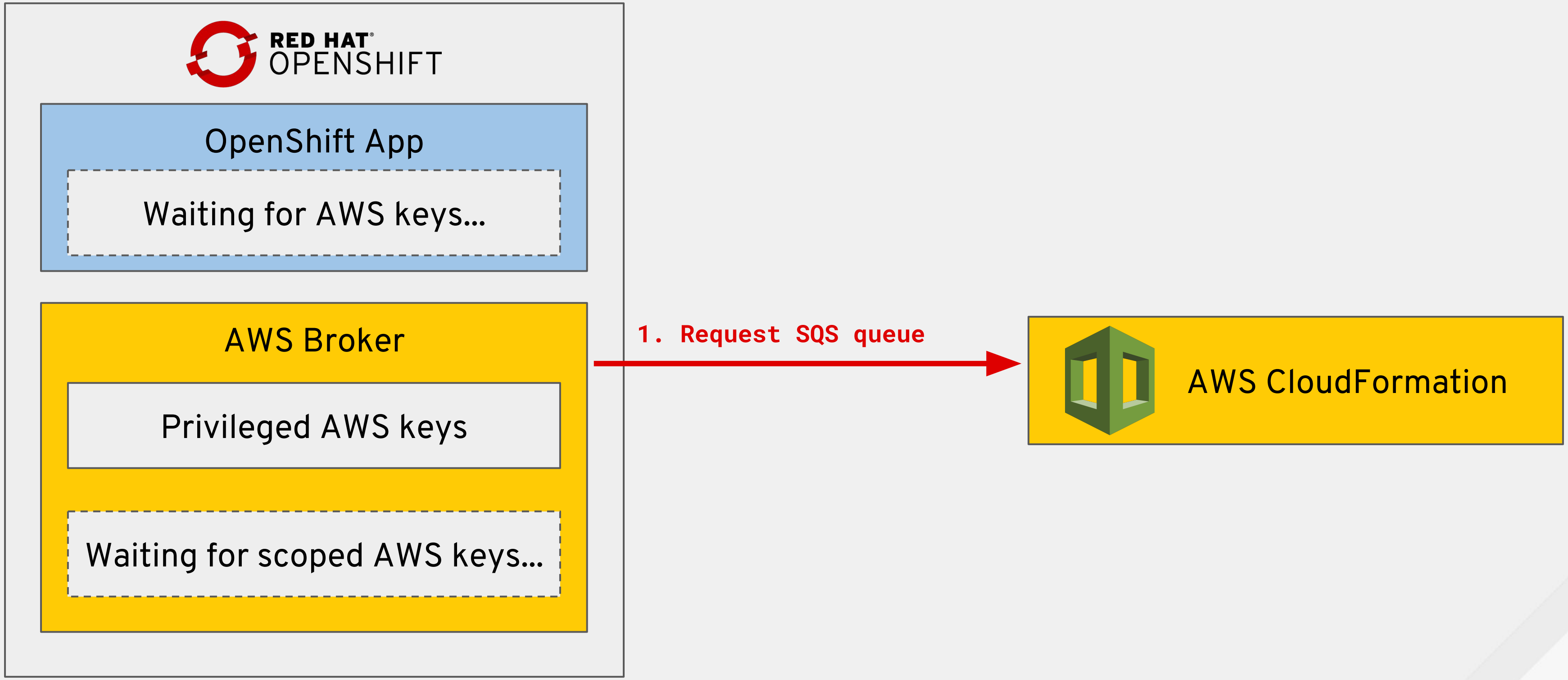
IAM - Creating Scoped AWS Keys



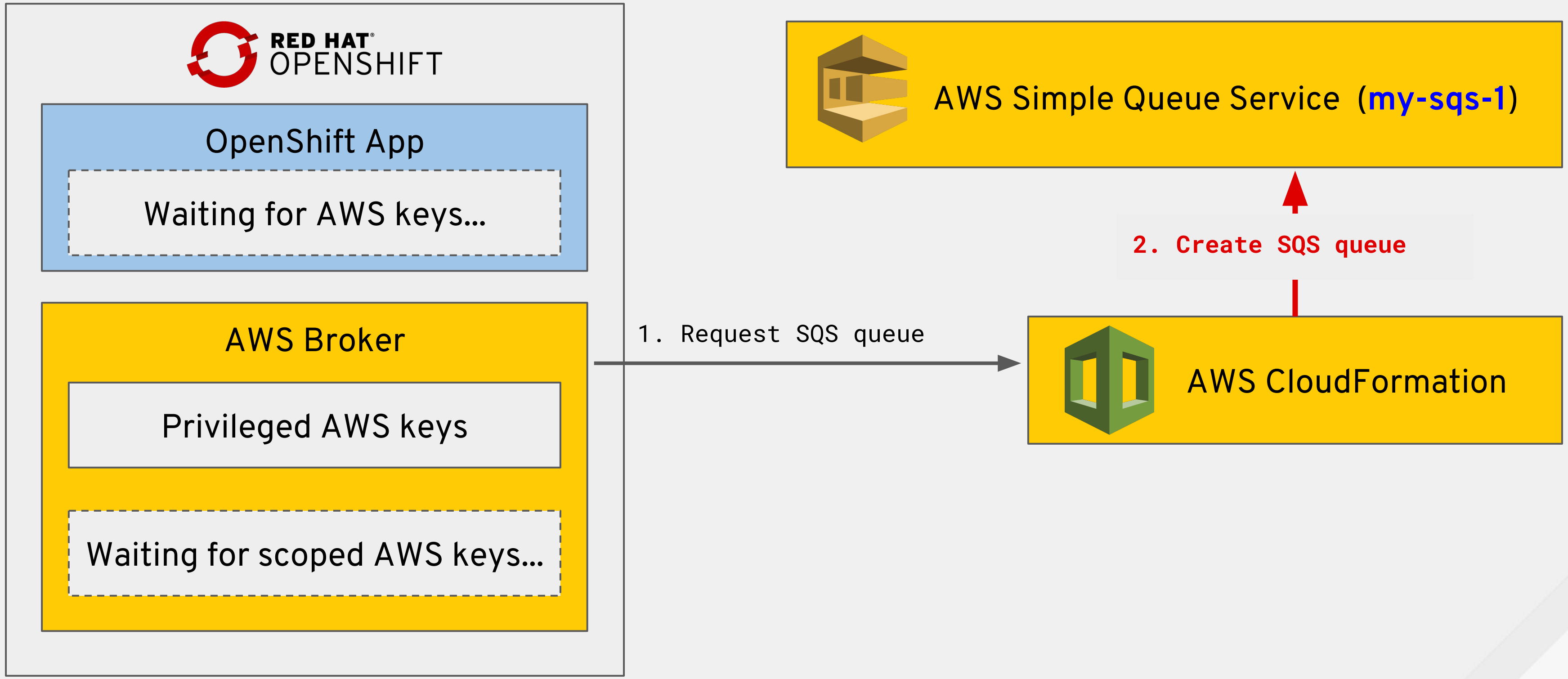
IAM - Creating Scoped AWS Keys



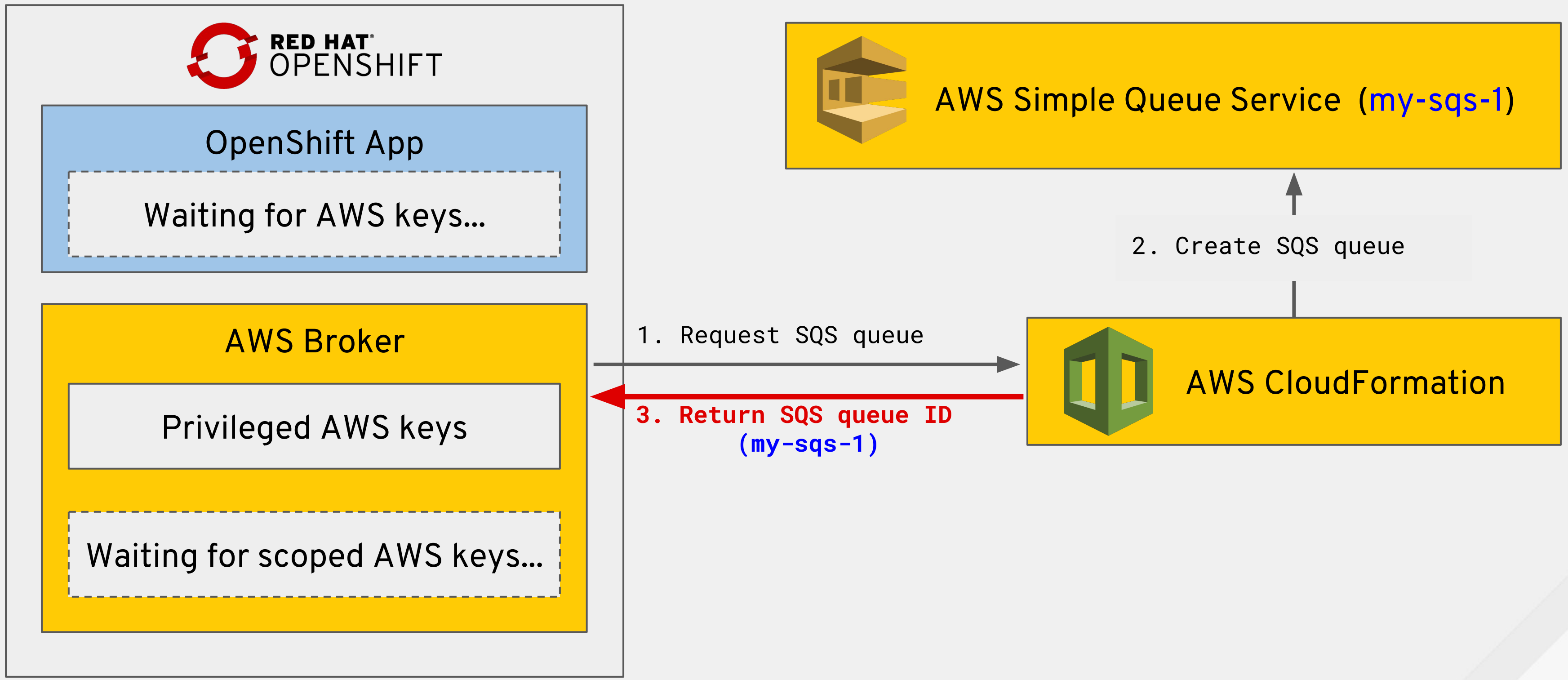
IAM - Creating Scoped AWS Keys



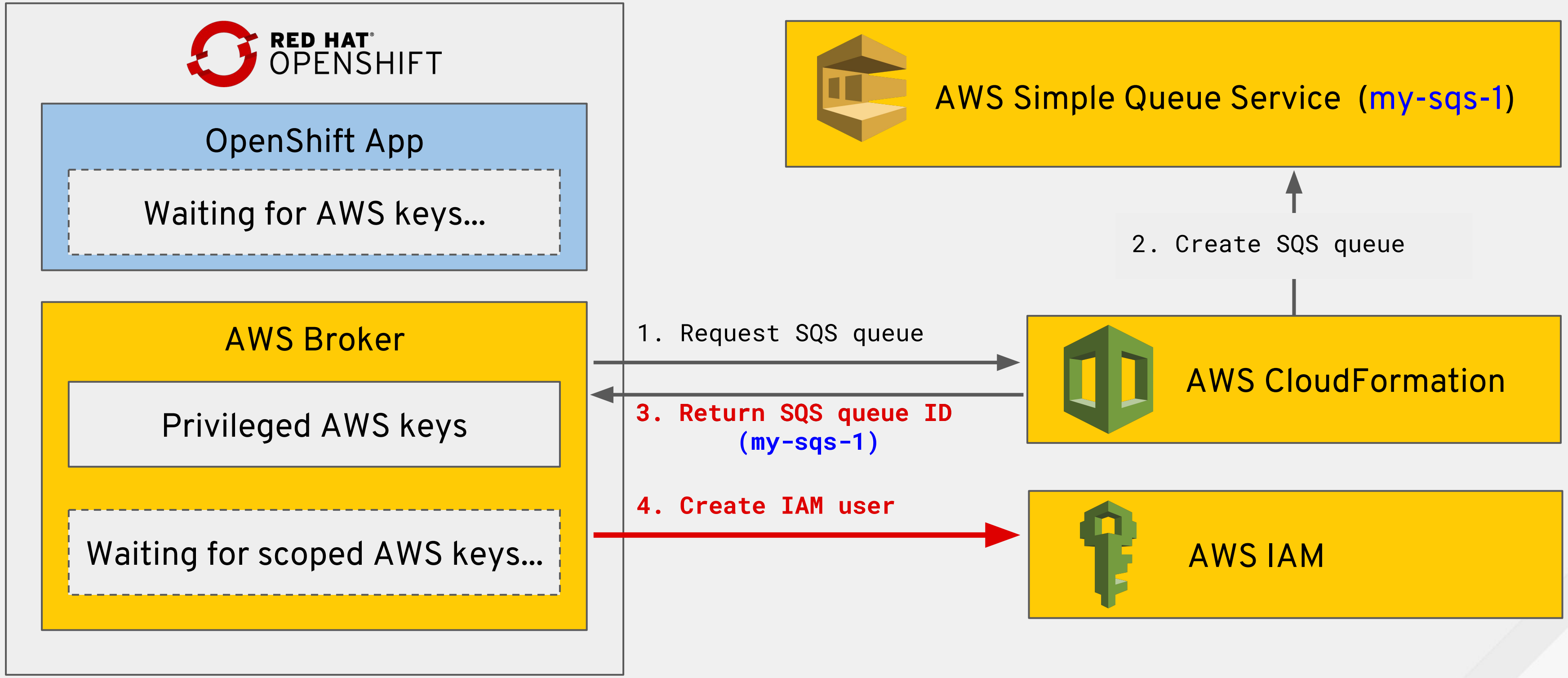
IAM - Creating Scoped AWS Keys



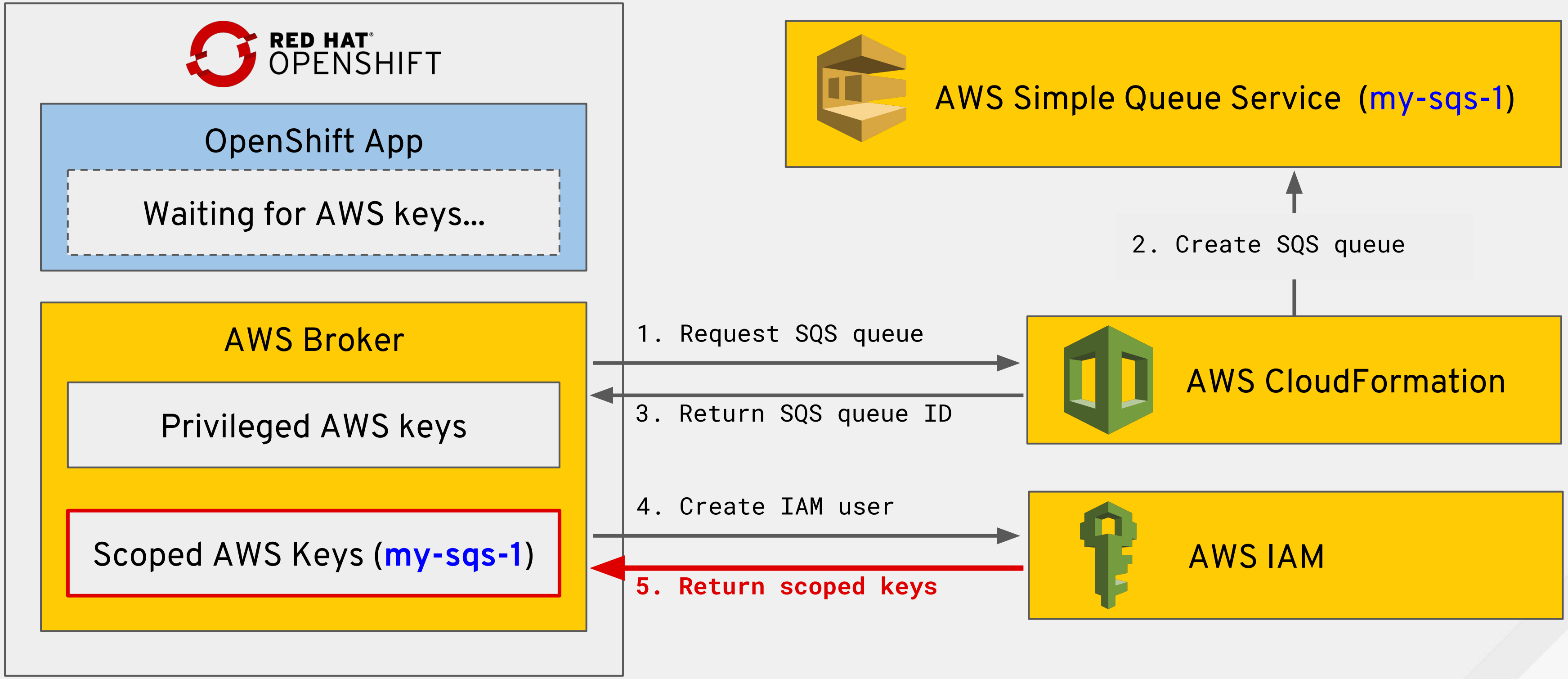
IAM - Creating Scoped AWS Keys



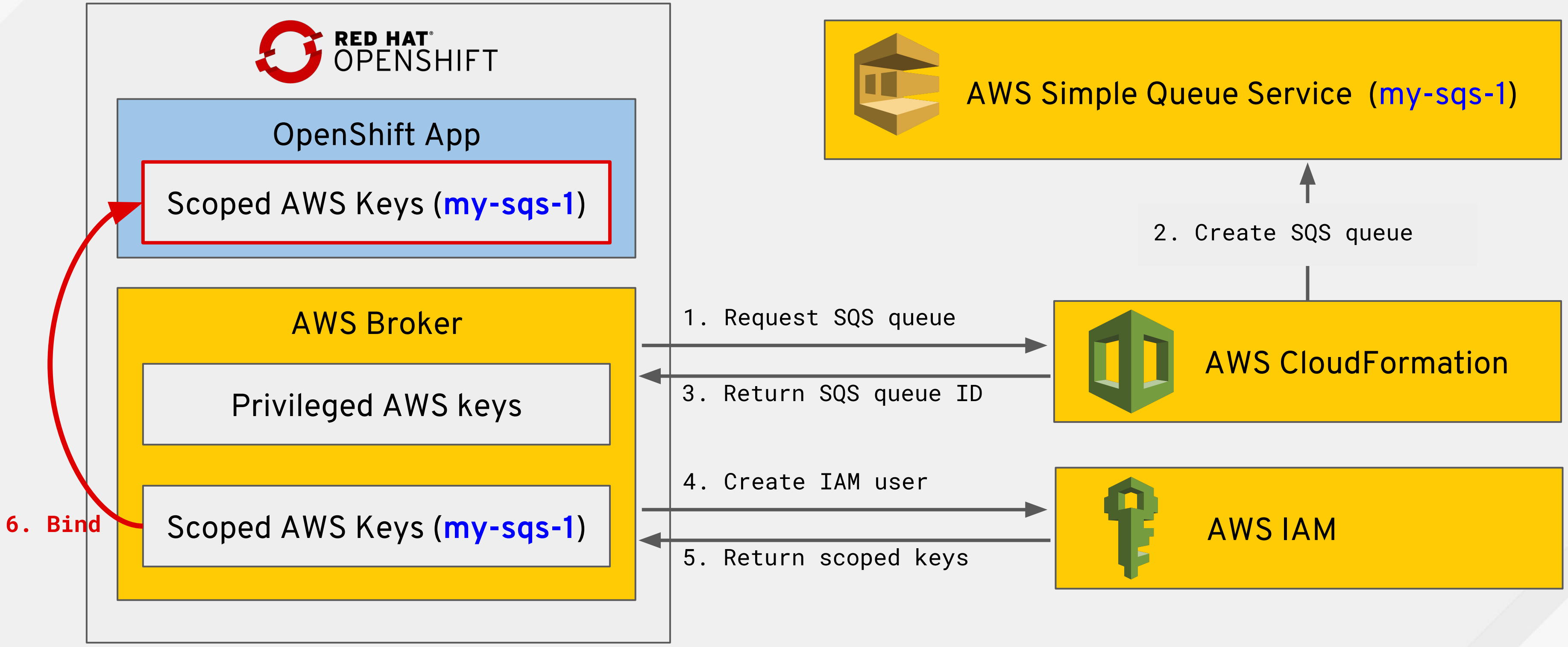
IAM - Creating Scoped AWS Keys



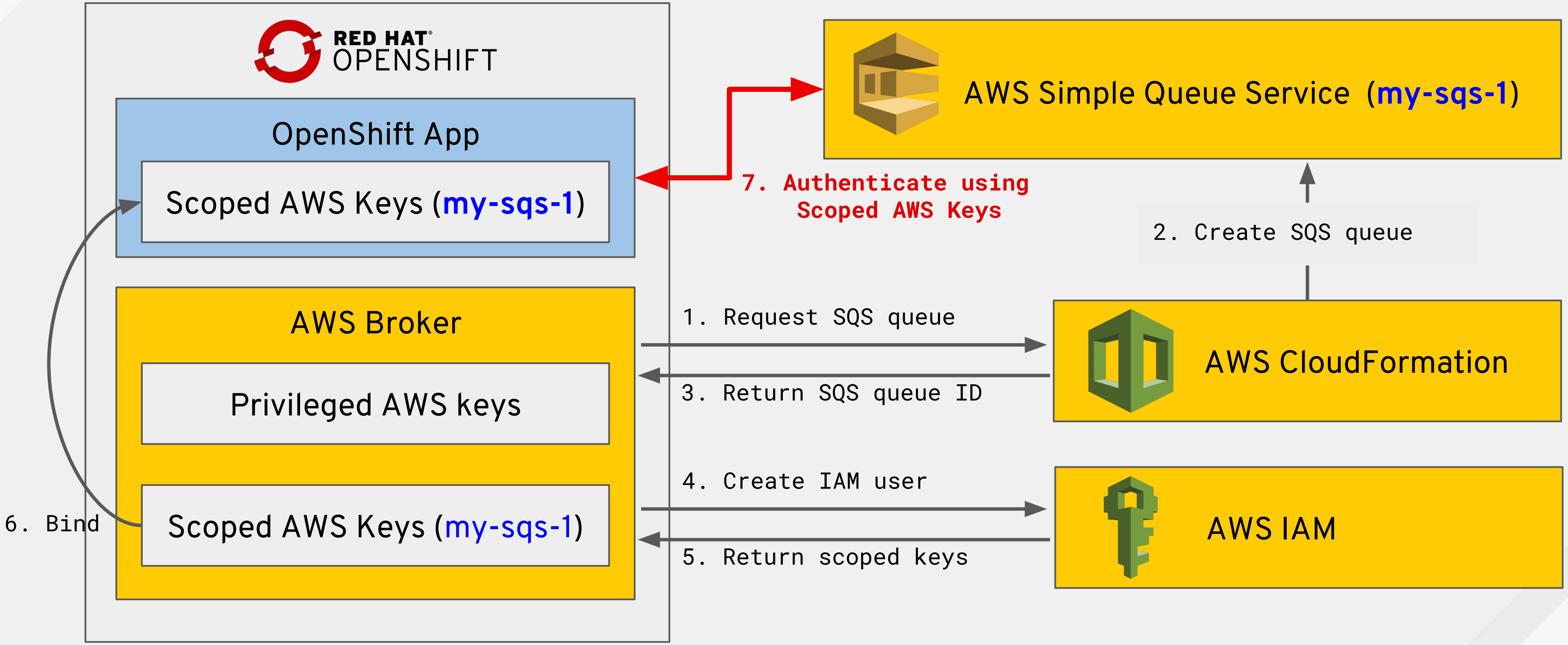
IAM - Creating Scoped AWS Keys



IAM - Creating Scoped AWS Keys



IAM - Creating Scoped AWS Keys



Installing the AWS Broker on OpenShift: *3 methods*

AWS

SERVICE BROKER

Installing on an *existing OpenShift cluster*

- AWS provides an **OpenShift template** for installing the AWS Broker
  - Helper script creates required TLS certs and installs AWS Broker

- Install in just a few commands.

```
$ wget https://s3.amazonaws.com/[..]/deploy-awsservicebroker.template.yaml
$ wget https://s3.amazonaws.com/[..]/deploy_aws_broker.sh

$ bash deploy_aws_broker.sh
```

- Full instructions at https://bit.ly/2I18UMw

Installing in a *development environment*

- **CatASB** deploys **OpenShift + Service Catalog + Brokers**
  - Developer tool built from Ansible Playbooks
  - Uses `oc cluster up` to start OpenShift

- Deploy an OpenShift cluster + the AWS Broker in a few commands

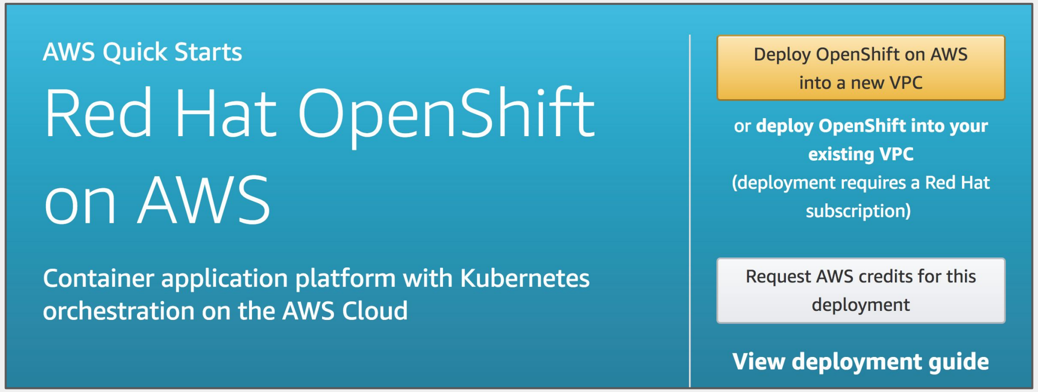
```
$ git clone https://github.com/fusor/catasb.git
$ echo "deploy_awsservicebroker: True" >> catasb/config/my_vars.yml

$ cd catasb/local/linux
$ ./run_local_setup.sh
```

- https://github.com/fusor/catasb

Installing with an *AWS QuickStart*

- AWS offers an **OpenShift on AWS** quickstart
  - Creates an OpenShift cluster on AWS
  - Has a parameter to enable the AWS Broker on the created cluster
- Free AWS credits available for evaluation purposes


AWS Quick Starts

Red Hat OpenShift on AWS

Container application platform with Kubernetes orchestration on the AWS Cloud

Deploy OpenShift on AWS into a new VPC
or **deploy OpenShift into your existing VPC**
(deployment requires a Red Hat subscription)
Request AWS credits for this deployment
View deployment guide

- https://aws.amazon.com/quickstart/architecture/openshift/

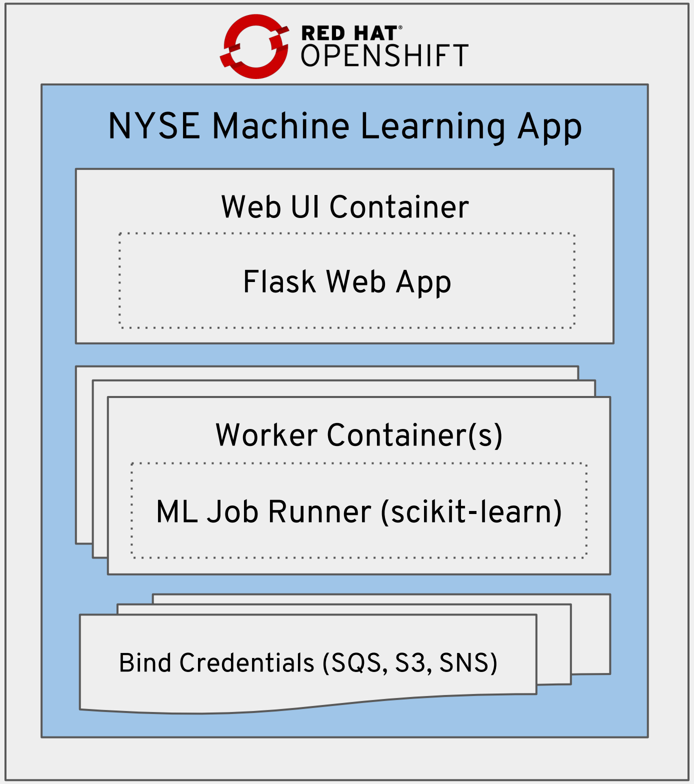
AWS Service Broker - *Roadmap*

AWS Broker - Roadmap

- Asynchronous Bind Support
- Advanced Parameter Validations
- More AWS Services

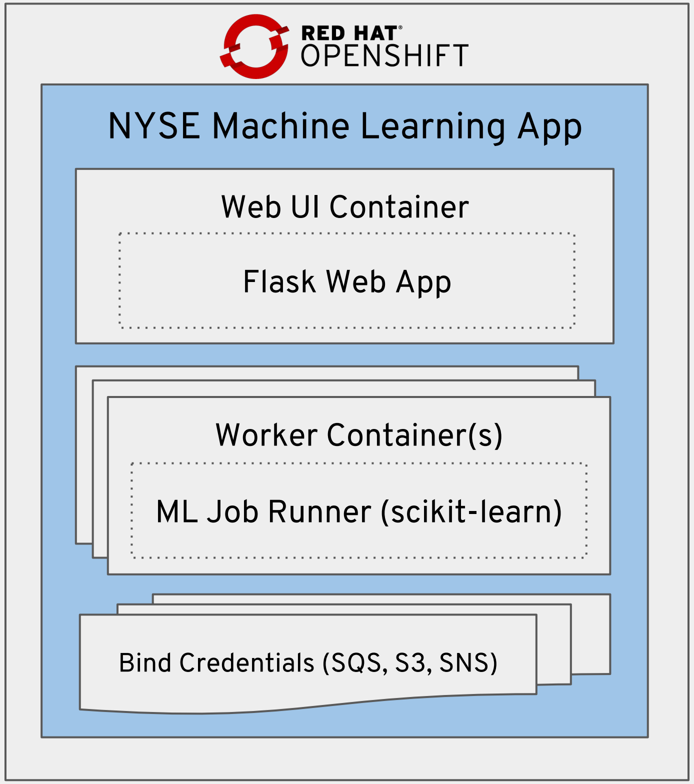
- We want your feedback!

Let's build a hybrid cloud app!


RED HAT OPENSHIFT
NYSE Machine Learning App
Web UI Container
Flask Web App
Worker Container(s)
ML Job Runner (scikit-learn)
Bind Credentials (SQS, S3, SNS)

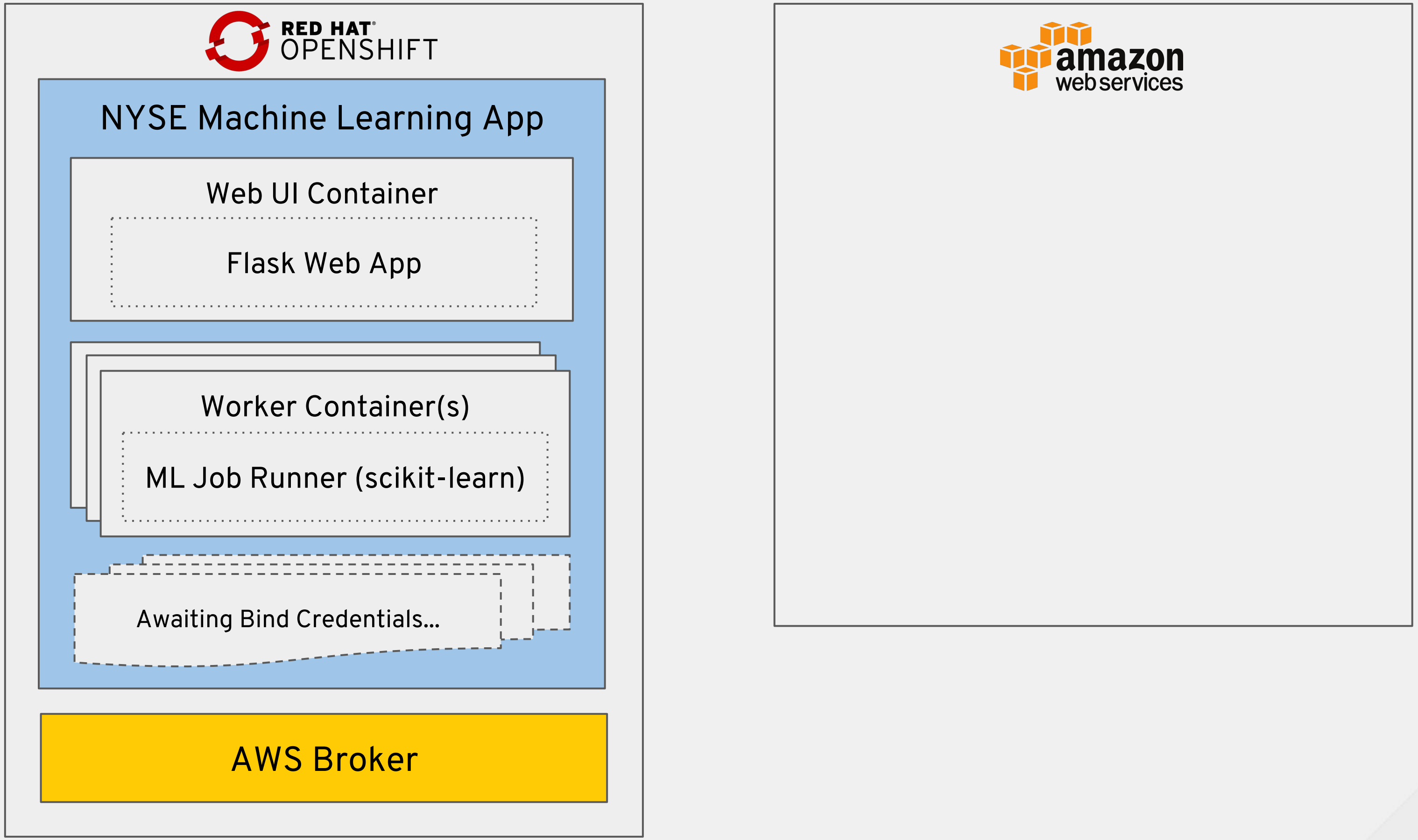
Purpose

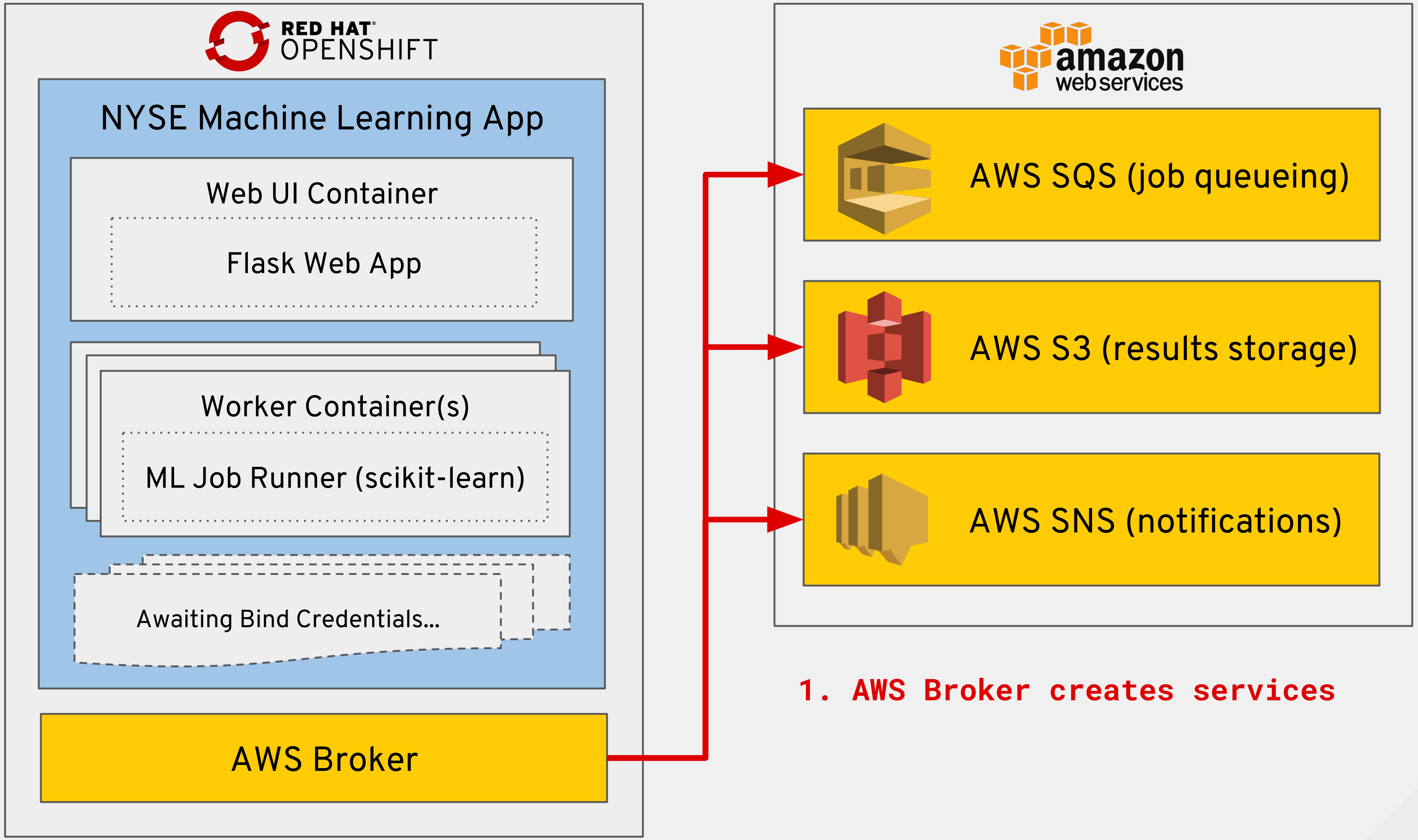
- Pull share prices from New York Stock Exchange
- Use affinity propagation to cluster related stocks
- Render graphic grouping related stocks


RED HAT OPENSHIFT
NYSE Machine Learning App
Web UI Container
Flask Web App
Worker Container(s)
ML Job Runner (scikit-learn)
Bind Credentials (SQS, S3, SNS)

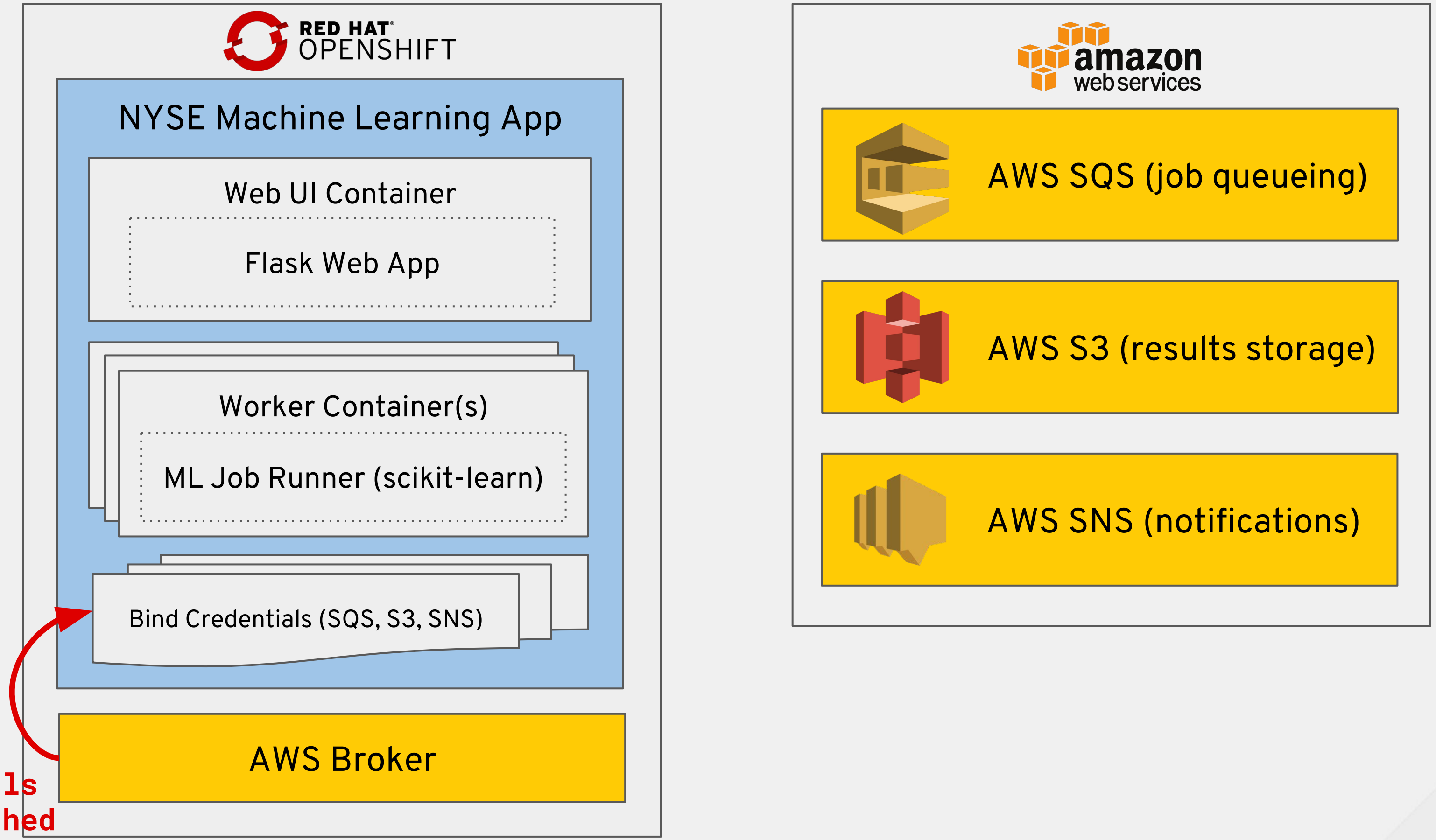
Architecture

- Web UI (flask)
  - Web UI accepts new jobs
  - Add jobs to **AWS SQS** queue
- Worker (scikit learn + python)
  - Pick up jobs from **AWS SQS** queue
  - Build graph showing stock clusters
  - Store results in **AWS S3**
  - Send results notification w/ **AWS SNS**

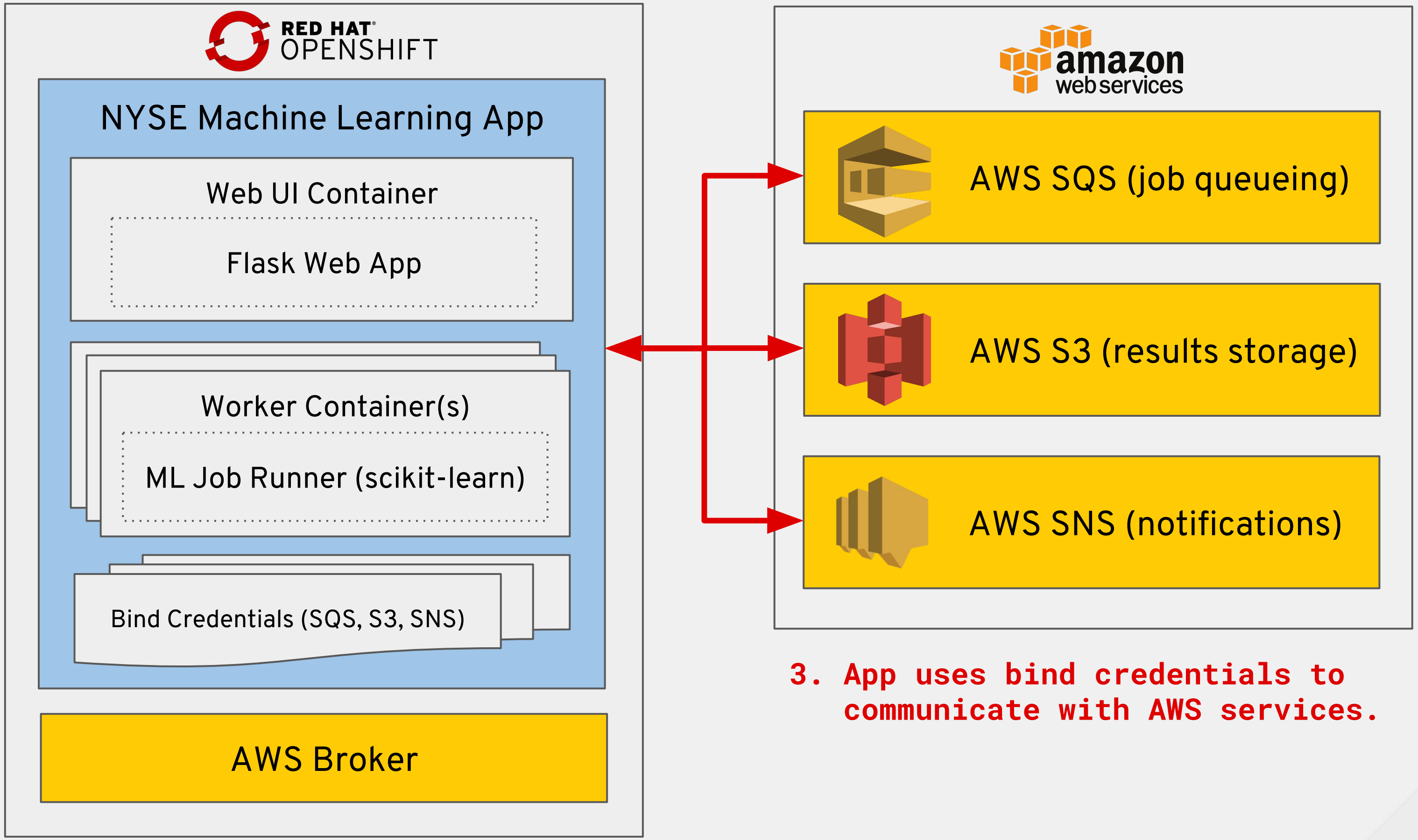

RED HAT OPENSHIFT
NYSE Machine Learning App
Web UI Container
Flask Web App
Worker Container(s)
ML Job Runner (scikit-learn)
Awaiting Bind Credentials...
AWS Broker
amazon web services



1. AWS Broker creates services



2. AWS bind credentials are attached



3. App uses bind credentials to communicate with AWS services.

Demo - Machine Learning on the NYSE with AWS

Where to go from here

Resources

- AWS Broker
  - Docs - https://bit.ly/2jIm0zO
  - Getting Started Guide - https://bit.ly/2I18UMw
  - AWS service APBs on GitHub - https://bit.ly/2JN802G
  - AWS QuickStart (free credits!) - https://amzn.to/2x6m1ph

- Automation Broker (base project)
  - Home: http://automationbroker.io/
  - YouTube Channel: https://bit.ly/2w704aD
  - Freenode IRC: #asbroker

Your feedback is *appreciated*!

RED HAT SUMMIT

THANK YOU

plus.google.com/+RedHat

facebook.com/redhatinc

linkedin.com/company/red-hat

twitter.com/RedHatNews

youtube.com/user/RedHatVideos