
AWS
re:Invent

CON207

Define AWS service resources with AWS Controllers for Kubernetes

Jay Pipes

Principal Open Source Engineer

AWS

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

A familiar story



© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Alice is a web developer

- Develops Node.js application
- Builds application into immutable Docker image
- Chooses an embedded database for simple storage



And of course . . .







Alice deploys her app to Kubernetes

```
kubectl apply -f deployment.yml
kubectl apply -f service.yml
kubectl apply -f ingress-nginx.yml
```



Everything is going great, but then . . .

- 10 users try using the site at once
- The embedded database can’t keep up
- Alice realizes she needs a concurrent database



Alice deploys PostgreSQL to Kubernetes

```
kubectl apply -f pg-secret.yml
kubectl apply -f pg-volume-claim.yml
kubectl apply -f pg-deployment.yml
kubectl apply -f pg-service.yml
```





But now Alice is in the DBA game

- Not what Alice had in mind
- She wants to focus on her app
- What can she do?



AWS to the rescue?

- Alice hears about Amazon Relational Database Service (Amazon RDS)
- No more Alice the DBA
- But there’s a problem . . .



Where is Alice’s cozy Kubernetes experience?

Create database

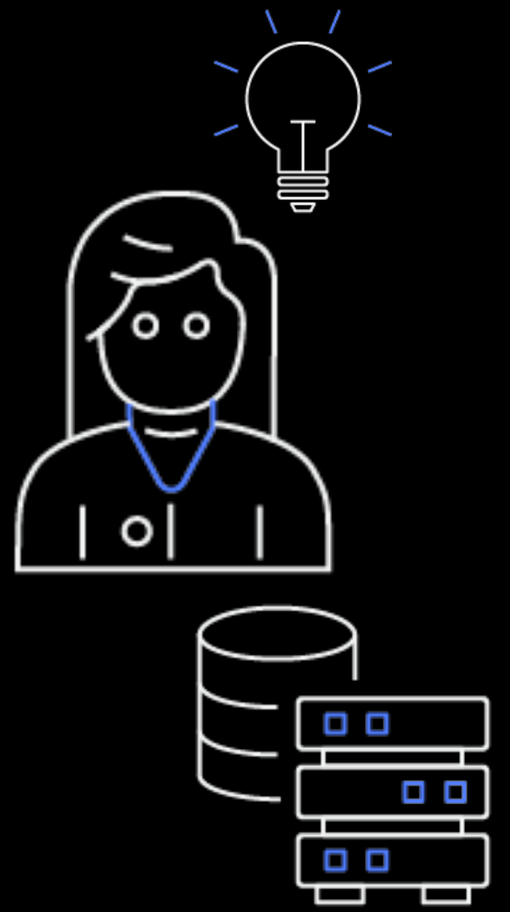
Choose a database creation method Info

- (•) **Standard Create**
  You set all of the configuration options, including ones for availability, security, backups, and maintenance.
- ( ) **Easy Create**
  Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type Info

- ( ) Amazon Aurora
- ( ) MySQL
- ( ) MariaDB
- ( ) PostgreSQL
- ( ) Oracle
- (•) Microsoft SQL Server



She doesn’t have to use the console

- AWS Command Line Interface (AWS CLI)
- AWS CloudFormation
- Terraform



But none of those are Kubernetes

And Alice really loves Kubernetes







kubectl apply -f db.yml

```
apiVersion: rds.services.k8s.aws/v1alpha1
kind: DBInstance
metadata:
  name: mydb
spec:
  dbInstanceClass: db.m1.large
  dbInstanceIdentifier: mydb
  engine: PostgreSQL
...
```





AWS Controllers for Kubernetes (ACK)

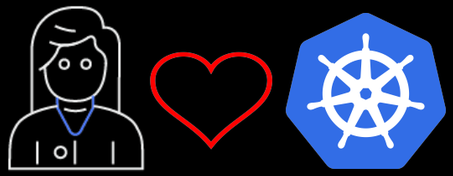


AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

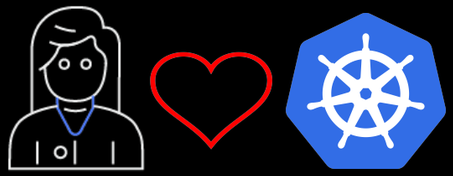
ACK: A Kubernetes experience for AWS APIs

- A collection of Kubernetes custom controllers
- AWS service API resources just another Kubernetes manifest
- ACK service controller manages life cycle of AWS resources



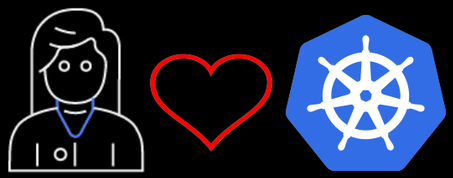
ACK design

- Code generation for everything, including controller implementation
- Consult with AWS service teams
- Not EKS specific; runs on any Kubernetes
- No AWS CloudFormation behind the scenes



Cross-account resource management

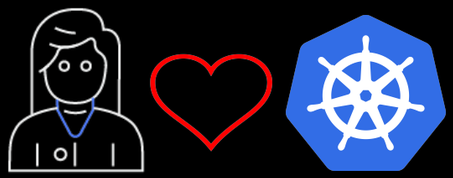
- Avoid installing ACK controllers in many clusters
- Cluster admin associates' AWS account IDs to namespace
- Application developer creates CR in namespace
- ACK controller looks up role ARN
- Controller calls `STS::AssumeRole` to pivot client



Authorization and access control

- Two RBAC systems in play (K8s and IAM)
- No overlap between K8s and IAM RBAC
- Use IAM roles for service accounts

Learn more at https://aws.github.io/aws-controllers-k8s/user-docs/authorization/



What about secret stuff?

```
apiVersion: rds.services.k8s.aws/v1alpha1
kind: DBInstance
metadata:
  name: mydb
spec:
  dbInstanceClass: db.m1.large
  dbInstanceIdentifier: mydb
  engine: PostgreSQL
  masterUserPassword: UhmPlainText!?
...
```





Ah, that’s better

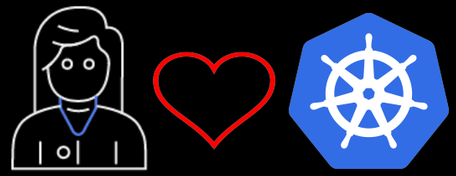
```
apiVersion: rds.services.k8s.aws/v1alpha1
kind: DBInstance
metadata:
  name: mydb
spec:
  dbInstanceClass: db.m1.large
  dbInstanceIdentifier: mydb
  engine: PostgreSQL
  masterUserPassword:
    name: dbsecrets
    key: masterUserPassword
...
```





On our roadmap

- Common AWS tag representation
- “Adopt a resource”
- Common rate limiting and throttling
- Service controller release roadmap: https://github.com/aws/aws-controllers-k8s/projects/1



Additional resources

Kubernetes Slack community #provider-aws channel

https://github.com/aws/aws-controllers-k8s

Thank you!

Jay Pipes

@jaypipes

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Please complete the session survey

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.