Update on the Common Application Platform

Michael Irwin - DCSS Oct 2021

Agenda

- What's this all about again?
- Current Status of the Platform
- Demo Time!
- Next Steps

What’s this all about again?

The Common Application Platform

Initiative

Establish an organizational unit that provides application technology infrastructure and shared services for all application development teams in the Division and provide as a service to all.

Outcome

- Teams will leverage a single common platform on which to run applications
- Shared application development and deployment services will be managed by a single team

The Original Goals, in Human Terms

- Let application teams focus on their applications
  - No need to think about cloud architecture, networking, or other details
  - No need to log into machines (and maintain them)
- Lower the barrier of entry to use application services
  - Provide training, templates, etc. on application services
- Support faster iteration and deployments
  - Teams only declare their desire state
  - Provide CI templates to meet common patterns and use cases
- Design with multi-tenancy from the start
  - Provide a solution that scales to the needs, providing consistency

Current Status of the Platform

Declarative Programming

- We have built the platform around declarative programming
- At the end of the day, Kubernetes is a declarative programming system with reconciliation loops
- Using controllers, we can support a variety of different types of objects

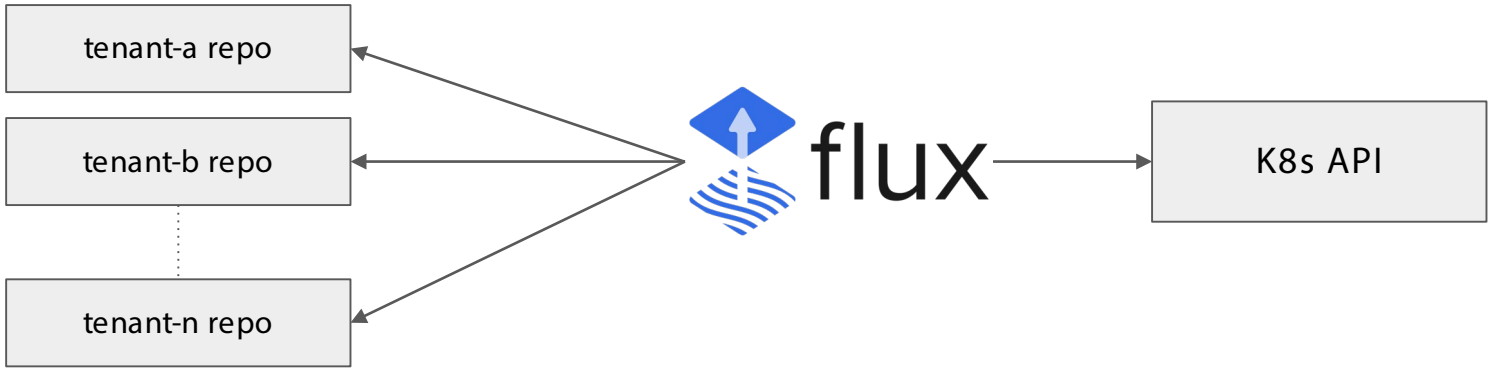
```
apiVersion: v1
kind: Pod
metadata:
  name: hello-world
spec:
  containers:
    - name: hello-world
      image: mikesir87/hello-world
```

Supporting Multi-Tenancy

- Namespaces allow for isolation of resources within the cluster
  - RBAC ensures you can only see the resources you're authorized to see
- Leveraging admission controllers, we can add additional policies
  - Validate domain name usage on Ingress and Certificates
  - Provide Pod Security Standard baseline enforcement

Using GitOps

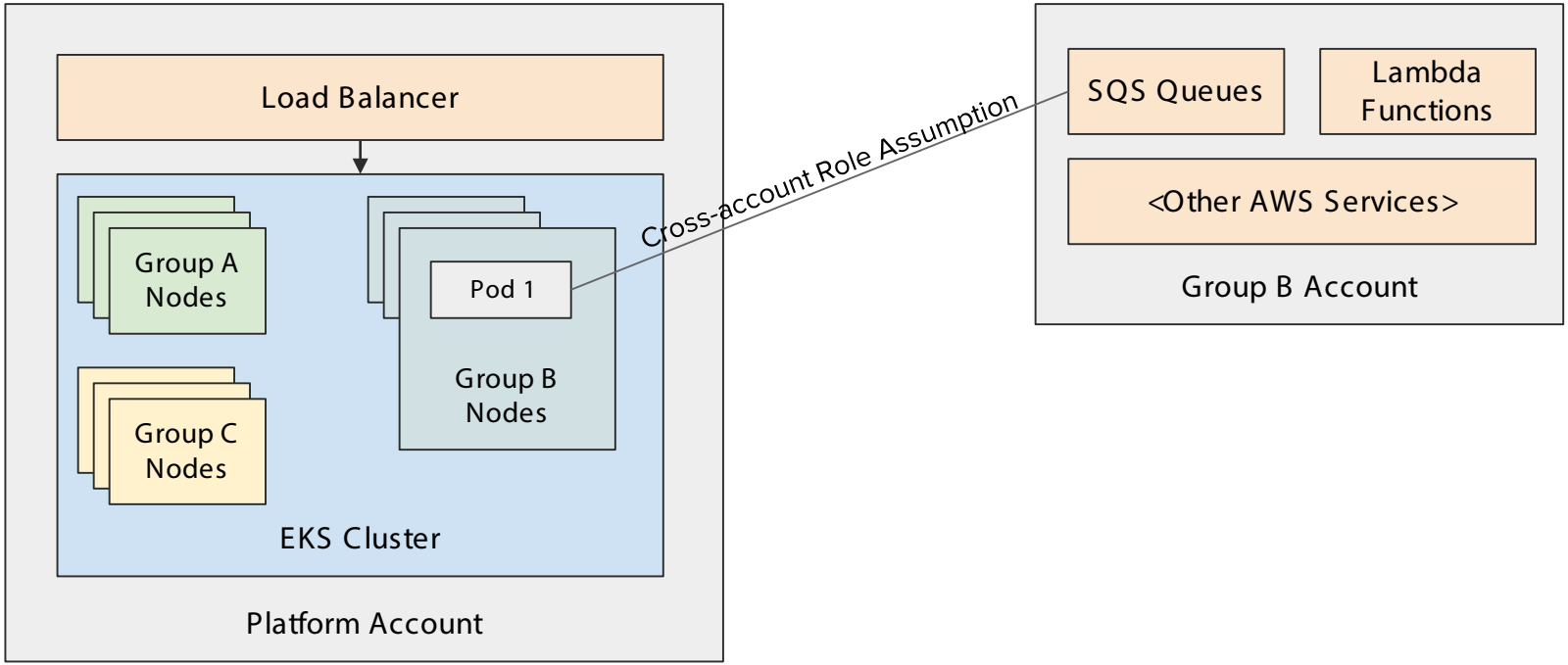
- We are using GitOps to deploy changes
  - Agent runs in the cluster and pulls in state from repos it watches
  - More secure by keeping all of the credentials in the cluster
  - Repos are always the source of truth



Designing for Multi-tenancy

- Every tenant has its own namespace
  - Tenant is pretty loose…can be an app or an env within an app
- Using various node pools for security and cost accounting
  - Does provide an additional security boundary as well
- GitOps (Flux v2) makes deployment super nice and easy
  - Each tenant has a manifest repo where they define their desired state
  - SSH key and webhook configuration to pull manifests is automated

What it *kinda* looks like...



Using the Platform, at a High Level

1. Request platform access
2. A tenant is created and configured
3. Define manifests in the tenant manifest repo
4. The manifests are applied
5. The apps startup and are accessible
6. Debug your app using the dashboard, kubectl, or CLS
7. Loop back to step 3

What's running on it?

- Currently have 24 tenant namespaces
  - **ITSO/ITSL** - 9 applications in production
  - **NI&S** - Customer Portal and all of their feature branch deployments
  - **SIS** - have piloted the Certs Manager
  - **Platform Team** - the dashboard, docs, etc.
- Several GitLab runners and all of their jobs

Demo Time!

Next Steps

Our Current Limitations

- Team-size is small…very small…
  - Limits our ability to do 24/7 support
- No support (yet) for apps that need to talk to IPv6-only addresses
  - While we can support IPv6 inbound (dual-stacking the load balancer), EKS doesn't support IPv6 yet
- Adding more user documentation

Pilot-Fest!

- Challenge to see how many *pilot* apps we can onboard in November
  - Recognize that support is currently limited to business hours
- Nov 3 from 2:00-4:00 - onboarding training and workshop!
  - Will cover basics and get you up and running with an application
- We will have office hours throughout the month of November and help available on Slack
- If you're interested, contact Michael Irwin or Justin Strickland by October 29

Thanks again!