Kubernetes in the IRIS-HEP SSL

Lincoln Bryant
Enrico Fermi Institute
University of Chicago

Chicagoland K8S Workshop
Jan 30, 2020
What is SSL?

- Innovation and training platform for IRIS-HEP
- "Path to Production" for new services, algorithms, etc
- Experiment with new technologies that may inform how we build future facilities for users
RIVER – our first SSL cluster

- Repurposed R&D cluster
  - ~3k cores (a bit vintage, but nice)
  - Each node:
    - 48 cores
    - 256 GB RAM
    - 2x800GB SSD
    - 10Gbps networking
  - Network:
    - 2x40g to campus 100g SciDMZ
Kubernetes Setup on SSL

- Kubernetes v1.15
  - single master
  - ~70 compute nodes
  - Calico for networking
- Kubeadm for cluster setup
- CentOS for the base OS, Puppet for all of the ancillary things surrounding K8S
Deploying on-prem can be challenging

- Cloud vendors are very happy to let you magically use as many resources as you like! (for only a small fee........)
  - LoadBalancers just work
  - Ingress just works
  - Storage just works
  - DNS just works
- K8S makes things a lot easier for developers, but can push much complexity onto admins on-prem!
How we’re tackling this today

- Load Balancer – MetalLB
- Ingress controller – Traefik
- Storage – Local Volume Provisioner
- DNS – External DNS
- Monitoring – Prometheus
Other observations

- What happens when a Node dies?
  - Sometimes pods are moved. Sometimes they hang out in ERROR state indefinitely. ([#55713](#55713) – open for 2 years!)

- Routing traffic in/out of K8S can be rather asymmetrical, especially when LoadBalancers are involved (by design, as best we can tell)

- Ingress really feels like it is only meant to work with HTTP(S) connections.

- Kubernetes has no Long Term Support releases—variants such as OpenShift may be a bit more conservative
Roadmap for SSL/RIVER

- Kubernetes update to the latest version
  - Need to also ensure all users have latest client binaries
- Rook is high on the priority list
  - Local Persistent Volumes have issues with provisioning granularity and clean up
- Integrating some kind of OIDC workflow
  - Would like to force users to re-auth with something like CI Logon or Globus semi-frequently
- Documenting the pattern of deployment for other sites who wish to be SSL-like
  - Many things to do outside of Vanilla deployment!
## Some early SSL users

<table>
<thead>
<tr>
<th>Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceX</td>
<td>Data transformation and delivery service for LHC analyses</td>
</tr>
<tr>
<td>Frontier Analytics</td>
<td>Analyze and improve data access patterns for ATLAS Conditions Data</td>
</tr>
<tr>
<td>perfSONAR Analytics</td>
<td>Network route visualization based on perfSONAR traces</td>
</tr>
<tr>
<td>Parsl / FuncX</td>
<td>Parallel programming in Python, serverless computing with supercomputers</td>
</tr>
<tr>
<td>Large-Scale Systems Group @ UChicago</td>
<td>Serverless computing with Kubernetes</td>
</tr>
<tr>
<td>Skyhook</td>
<td>Programmable storage for databases, scaling Postgres with Ceph object store</td>
</tr>
<tr>
<td>SLATE / OSG</td>
<td>Backfilling otherwise unused cycles on SSL with work from the Open Science Grid</td>
</tr>
</tbody>
</table>
Getting access

- Join the IRIS-HEP Slack and SSL Google Group
  - [https://groups.google.com/a/iris-hep.org/forum/#!forum/ssl-team](https://groups.google.com/a/iris-hep.org/forum/#!forum/ssl-team)
  - Send an email to lincolnb@uchicago.edu for Slack
- Send me an email to me asking for access, I’ll send you a kubernetes config and some information
- In the medium term future:
  - We will plan to have a sign-up portal and some sort of OAUTH workflow for new and returning users
Thanks!