Kubernetes and Containers at UW-Madison

Stumbling Around in the Dark
Part 1 - CAE
“One must imagine Sisyphus happy.”
What do we do right now
What do we do right now

- Oracle database (The Eye of Larry) describes our world and who’s in it.

- FAI (..well, it’s mostly automated) takes us from PXE to installed Debian system with requested packages.

- CFEngine 2 (Broken the way we like it!) Configures services and does service reloads

- Custom scripts (Welcome to the land of technical debt!)
The wakeup call
Why change?

- Machine configuration is more tied to versions of the OS than we’d like. Containers may untangle that!

- Library tricks we use on lab machines, like LD_PRELOAD or CDE that we use for proprietary software look similar to Containers, and containers look more manageable. Hey! We could use this on the server!

- Some of what Kubernetes does looks like our custom scripts but cleaner and standardized. This could be less technical debt for us!
Why change?

- Containers/Kubernetes looks like a layer that we can use to make our rig more fungible…. Perhaps we move it to bare metal, hardware somewhere else on campus, or even the cloud in the long run.

- Perhaps, with continuous integration, we can actually increase our security by doing better at keeping packages up-to-date.
Containers - a new way of holding services
Containers - a new way of holding services
Containers - a new way of holding services

- Operating System
- Kernel
- Library
- Library
- Library
- Library
- E˴ɛɑ˙˔ȵɐʎɛ
- Parameters and Environment
- Filesystem Tree

CONTAINER

Virtual Network

External Network
Kubernetes to orchestrate

Kubernetes API Server

- Operating System Image
- Kublet
- Container
- Container

K8s "client"

etcd distributed database
The end (of CAE as is)

Maybe beyond Kubernetes, we need to think about serverless computing.

Also, may I coin the term RAC (Redundant Array of Clouds) here?
Part 2 - K8s Tricky bits
Kubernetes is...

- some form of greek for “helmsman”.
- from “the cloud”-- You’ll find that documentation for using bare metal is harder to find.
- a cult -- “Everybody is doing it.” There’s a lot of hype.
- available in many forms: some distributions, some whole operating systems, and some cloud API’s.
  
  https://www.cncf.io/certification/software-conformance/
- fast moving -- Expect new releases every quarter.
- using a bunch of go (golang) and YAML
- frequently shortened to K8s
Kubernetes confusions

- **How do I get network access?** In the cloud, people just use the load balancer. In bare metal, we can’t “just” do that. “Ingress Controller”

- **How do I integrate with persistent storage?** K8s wants to dynamically provision storage, but only some providers supply this. (Keep an eye on Rook for integrated storage.)

- **There’s 10,000 packages that integrate with Kubernetes, how do I know what to use?** Keep another eye on the Cloud Native Foundation projects: https://www.cncf.io/projects/

- **Is Kubernetes a panacea for security?** No, all containers need a bill of materials that can be analyzed and managed to keep the exposed surface area minimal.
Kubernetes confusions

- **How is logging done?**
  Containers tend to do logs to stdout/stderr. K8s stores logs for pods on the Node, you will need to use a tool like logstash or fluentd to copy them to a better location.

- **Is there one way to do backups?**
  Most of what is deployed usually comes from a code repository (git), but what about persistent volumes? Heptio.ARK seems to be one solution. I haven’t come to a conclusion here yet.
Not a real deployment

```
# kubectl apply -f deploy.yaml
```

Deployment Declaration
Version: 2
Metadata:
  Tags:
    This: that
Replicate: 2
Pod:
  Image: something:2.1
  Mount:
    Path: /somewhere
    Source: some-resource
I want:

2 A's

1 B

And give “A” An external network connection
Helm - one more layer of abstraction

- People think of Helm as kubernetes “packages”.
- Deploys templated YAML, using go templates.
- Unlike packages, one can install Helm “charts” more than once.
- Along with the new “Operator Framework”, the future is *interesting*.
- Problem: It’s easy to forget about the bill of materials.
Part 3 - Campus Cooperation
We are all trying different things

<table>
<thead>
<tr>
<th>Group</th>
<th>Orchestration method</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAE</td>
<td>Kubernetes 1.8 via typhoon</td>
</tr>
<tr>
<td>DoIT</td>
<td>Rancher 1.6 - cattle w everything turned off</td>
</tr>
<tr>
<td>SSEC</td>
<td>self hosted Kubernetes deployed on CENTOS</td>
</tr>
<tr>
<td>CHTC</td>
<td>minikube</td>
</tr>
<tr>
<td>Icecube</td>
<td>Kubernetes deployed on Atomic on Centos -&gt;</td>
</tr>
<tr>
<td>Library</td>
<td></td>
</tr>
<tr>
<td>Icecube 2 the revenge</td>
<td>RancherOS with K8s</td>
</tr>
<tr>
<td>CSL</td>
<td></td>
</tr>
<tr>
<td>DoIT2</td>
<td>shell scripts</td>
</tr>
<tr>
<td>DoIT3</td>
<td>AWS, elastic container service</td>
</tr>
</tbody>
</table>
Thank you, and join us! JOIN US!

- Mailing list: join-containers@lists.wisc.edu

- A meeting at the end of June will be announced on the mailing list.

- We want to do code-sharing on a new campus-provided GitLab site.
  - Can we have a new campus-provided GitLab site?

(https://xkcd.com/1988/)
Resources that have been useful to me

- Software Engineering Daily Podcast:  
  [https://softwareengineeringdaily.com/category/podcast/feed/](https://softwareengineeringdaily.com/category/podcast/feed/)
- PodCTL  
  [https://blog.openshift.com/tag/podctl/feed/](https://blog.openshift.com/tag/podctl/feed/)
- Google Kubernetes Podcast  
- Helm Project  
  [https://helm.sh/](https://helm.sh/)
- Operator Framework  
- Kelsey Hightower (great presenter from Google Cloud Engine)  
  "Kubernetes for Sysadmins" - from Puppet Conf 2016:  
  [https://youtu.be/HIXp0-M6SY](https://youtu.be/HIXp0-M6SY)
Kubernetes terms

- **Objects:**
  Pods, Replication Controllers, Deployments, Services, Config Map, Secret, Persistent Volume, Persistent Volume Claim, Volume Ingress Controller, Ingress

- **Other terms:**
  Namespace, RBAC, Node

- **This is just a cool feature:**
  source <(kubectl completion bash)
HA! Not YAML for a MVS 370 machine!

//KCnumberA JOB ,'Doby Johnson',MSGCLASS=H
//STEP1 EXEC PGM=ASSIST
//STEPLIB DD DSN=T70KAH1.ASSIST.LOADLIB,DISP=SHR
//SYSPRINT DD SYSOUT=* 
//SYSIN DD *
******************************************************************************

* Program: ASSIGNMENT
* Programmer: Doby Johnson (this is JCL from an NIU assignment)
******************************************************************************

/*

//FT05F001 DD *
   4500 -230  9
   0  +7  2

*/