

# Kubernetes/OKD at BNL

*Chris Hollowell* <[hollowec@bnl.gov](mailto:hollowec@bnl.gov)>

*Tejas Rao* <[raot@bnl.gov](mailto:raot@bnl.gov)>

Scientific Data and Computing Center (SDCC)

USATLAS Computing Facilities Face-to-Face Meeting - December 1, 2022



@BrookhavenLab

# Kubernetes at BNL

- Have had a vanilla kubernetes cluster for SDCC staff use for a number of years
  - Primarily utilized for testing
  - Have a [REANA](#) testbed deployed on this cluster
    - Framework for reusable analysis
  - Can be utilized by SDCC administrator staff to deploy staff-controlled services which require k8s
  - 1 control plane node, and 5 workers
    - No HA setup as primarily utilized for testing
- Increasingly, users have been interested in deploying their own internal services at our facility
  - k8s is a natural mechanism to provide this functionality
  - However, opening vanilla k8s API access to untrusted users (multi-tenant) so they can deploy services posed a number of security issues in the default k8s configuration:
    - Allows containers to run as root
    - Users can mount arbitrary system paths into containers
    - Of particular concern in an environment like ours with shared network filesystems with UID-based auth (NFS, Lustre, etc.)



# OKD at BNL

- **Setting up secure multi-tenancy in a vanilla k8s cluster is difficult**
  - Possible to work around issues through the setup of admissions controllers, RBAC, etc.
    - Not trivial and easily opens the door for administrator error
  - One of the reasons the large commercial k8s providers give tenants completely isolated/individual clusters on VMs
- **Another issue is there is a fast pace of development in vanilla k8s**
  - There are frequently major changes between “minor” releases
  - Important functionality sometimes stays “beta” for a long time, or is dropped
- **Therefore, decided to adopt [OKD](#) for our k8s needs**
  - The community release of Red Hat’s Openshift k8s platform
  - Secure out the box - suitable for multi-tenant use
    - Users are never root in containers by default
    - OKD/Openshift adopted at a number of other US national labs including FNAL and ORNL
  - Release model more consistent with an enterprise product than vanilla k8s
  - Simple integration with LDAP/OIDC identity providers
  - Provides users with a convenient easy to use management web interface (“Web Console”)
  - ***We have ported REANA to our OKD cluster, so are working on decommissioning our vanilla k8s***



# OKD/Openshift Considerations

- Namespaces are referred to as “projects” in OKD/Openshift
- `oc`, rather than `kubectl` is the standard OKD/Openshift CLI tool
  - Essentially an extended `kubectl` binary - supports all `kubectl` commands/constructs
    - Client tarball actually ships with `kubectl` hardlink to `oc`
  - Adds additional functionality, such as managing OKD/Openshift projects, and logging into the system
- Helm3 can be used with Openshift/OKD
  - Extremely convenient for deploying services
    - Helm is the defacto package/deployment manager for k8s
    - Openshift Templates are far less commonly used
  - Some helm charts and containers will not work out of the box for regular/unprivileged users, and need to be modified for use with OKD/Openshift
    - Can't define `ClusterRole/ClusterRoleBinding` objects
    - Can't run as root in containers
      - Each project is assigned a unique ranges of UIDs that can be used
        - Allows multi-tenancy



# OKD Clusters at BNL

- 2 OKD clusters brought online in 2022
  - One for ATLAS and the other for sPHENIX
    - sPHENIX cluster primarily for conditions DB and Panda use
  - **ATLAS cluster**
    - Primarily for Analysis Facility (AF) services that require k8s
      - [ServiceX](#) (20220418-1418-stable)
      - REANA (0.9.0-alpha.5)
      - Note that our analysis facility JupyterHub deployment does not require k8s and uses a modified [batchspawner](#) plugin to utilize our existing large batch (HTCondor/SLURM) farms
        - REANA is also capable of leveraging batch resources



# OKD Cluster Details

- **ATLAS cluster running OKD 4.10, and is provisioned with:**
  - 7 Dell R640 Servers
    - 3 HA control plane nodes, 4 worker nodes
      - Running Fedora CoreOS (FCOS) 35 deployed via OKD Installer-Provisioned Infrastructure (IPI)
        - CRI-O used as container runtime
      - Specs:
        - 2x Xeon Silver 4210 CPU @ 2.20 GHz
        - 128 GB RAM
        - 4x 25 Gbps NICs
        - 3x 480 GB SSDs
  - NetApp A250 Storage Appliance
    - 14 x 1.92 TB NVME drives (~26 TB raw)
    - ONTAP NetApp OS allows dynamic PV provisioning via [Trident](#)



*ATLAS OKD Cluster Hardware*

# OKD Cluster Details (Cont.)

- Node layout

```
# oc get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
control0.usatlas.bnl.gov	Ready	master	90d	v1.23.3+759c22b
control1.usatlas.bnl.gov	Ready	master	90d	v1.23.3+759c22b
control2.usatlas.bnl.gov	Ready	master	90d	v1.23.3+759c22b
node0.usatlas.bnl.gov	Ready	worker	14d	v1.23.3+759c22b
node1.usatlas.bnl.gov	Ready	worker	14d	v1.23.3+759c22b
node2.usatlas.bnl.gov	Ready	worker	15d	v1.23.3+759c22b
node3.usatlas.bnl.gov	Ready	worker	22d	v1.23.3+759c22b

```
# oc get clusterversion
```

NAME	VERSION	AVAILABLE	PROGRESSING	SINCE	STATUS
version	4.10.0-0.okd-2022-03-07-131213	True	False	86d	4.10.0-0.okd-2022-03-07-131213

- HAProxy running on VM for control plane API service redundancy

- Plan to add additional nodes and storage in the coming years, based on utilization

- Likely more compute-farm-like nodes will be added for processing/workload oriented use cases like ServiceX/REANA

- May consider moving to fully/paid supported Red Hat Openshift product if clusters see wide adoption

- Will depend on pricing

# OKD Cluster Details (Cont.)

---

- Authentication tied to our [Keycloak](#) OIDC IDP

- Users login to web console to obtain a token which can be used with the oc CLI:

```
oc login --token=XYZ --server=https://api.usatlas.bnl.gov:6443
```

- API server accessible internally at SDCC, so `oc` typically used from our various interactive nodes



- OKD web console only available internally, at least for now

- Users/admins access from workstations onsite, or via VPN or SSH SOCKS proxy

- Users can setup Openshift/OKD routes to expose services internally

- Facility-level firewalls prevent them from being accessed outside SDCC
- Web services that needs to be opened to the world must go through a reverse proxy
  - Can be manually setup by SDCC administrator staff
  - Only available after approval/scanning, etc.



# OKD Web Console

The screenshot shows the OKD Web Console interface in a Mozilla Firefox browser. The browser address bar shows the URL: `https://console-openshift-console.apps.usatlas.bnl.gov/k8s/cluster/projects/servicex`. The console header includes the OKD logo, a user profile for Christopher Hollowell, and navigation icons. A left-hand sidebar contains a menu with categories: Administrator, Home, Projects (selected), Search, API Explorer, Events, Operators, Workloads, Pods, Deployments, DeploymentConfigs, StatefulSets, Secrets, ConfigMaps, and CronJobs.

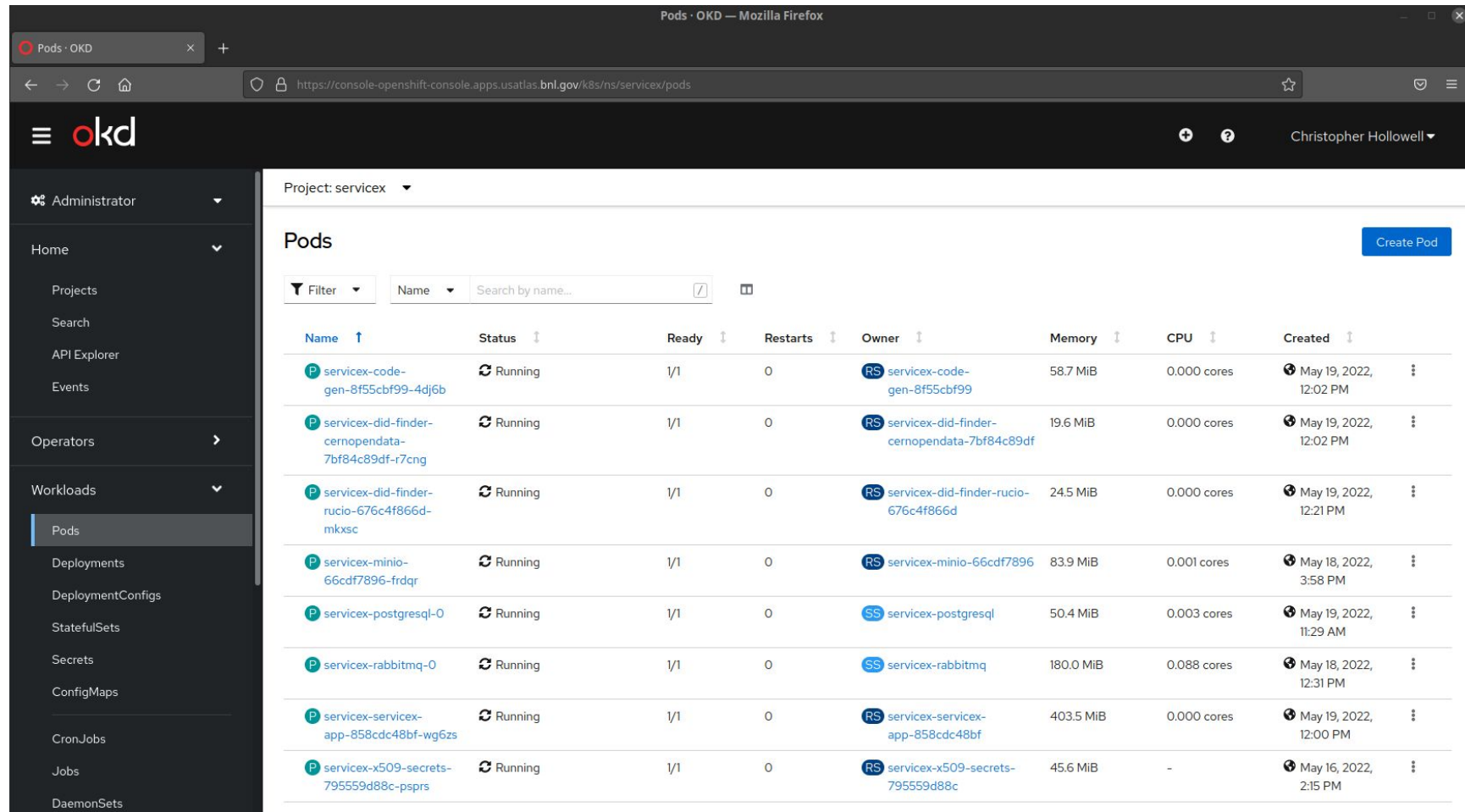
The main content area displays the 'Project details' for 'servicex', which is in an 'Active' state. The 'Overview' tab is selected, showing the following information:

- Details:** Name: servicex; Requester: hollowec; Labels: `kubernetes.io/metadata.name=servi...`; Description: No description.
- Status:** Active (indicated by a green checkmark).
- Utilization:** A table showing resource usage over a 1-hour period (3:00 PM to 3:30 PM).

Resource	Usage	3:00 PM	3:30 PM
CPU	92.7m	200m	[Line chart showing CPU usage]
Memory	866.2 MiB	4 GiB	[Line chart showing memory usage]
Filesystem	47.6 MiB	50 MiB	[Line chart showing filesystem usage]
Network transfer	80.81 Bps in, 81.57 Bps out	100 Bps, 50 Bps	[Line chart showing network transfer]
- Activity:** Ongoing. There are no ongoing activities. Recent events: There are no recent events.
- Inventory:** 6 Deployments, 0 DeploymentConfigs, 2 StatefulSets, 8 Pods, 0 PersistentVolumeClaims, 7 Services, 2 Routes.

OKD Web Console - Project Screen

# OKD Web Console (Cont.)

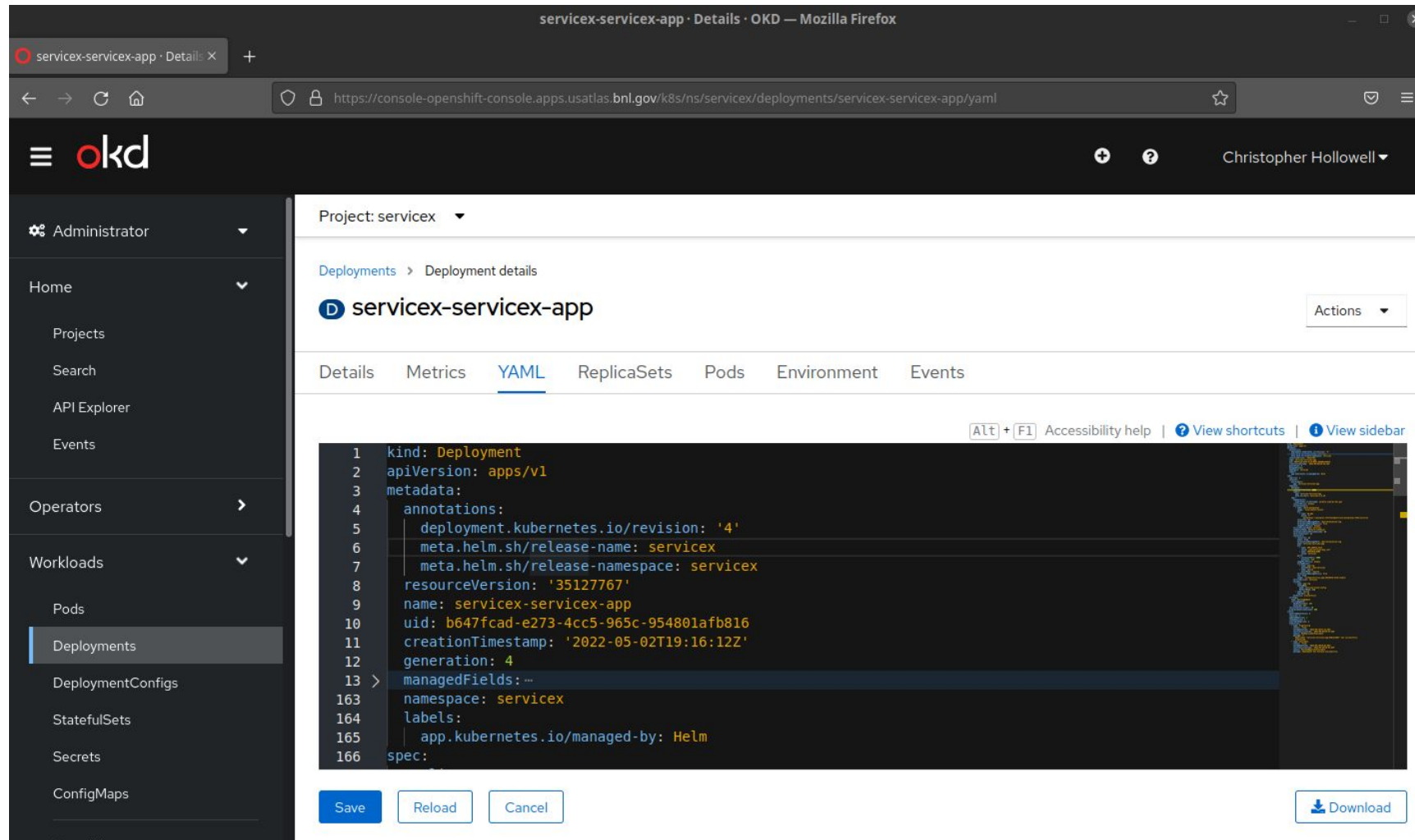


The screenshot displays the OKD Web Console interface in a Mozilla Firefox browser. The page title is "Pods · OKD". The URL is "https://console.openshift-console.apps.usatlas.bnl.gov/k8s/ns/servicex/pods". The user is logged in as "Christopher Hollowell". The left sidebar shows the navigation menu with "Pods" selected under the "Workloads" section. The main content area shows the "Pods" page for the "servicex" project, featuring a table of pod details.

Name ↑	Status ↓	Ready ↓	Restarts ↓	Owner ↓	Memory ↓	CPU ↓	Created ↓
servicex-code-gen-8f55cbf99-4dj6b	Running	1/1	0	servicex-code-gen-8f55cbf99	58.7 MiB	0.000 cores	May 19, 2022, 12:02 PM
servicex-did-finder-cernopendata-7bf84c89df-r7cng	Running	1/1	0	servicex-did-finder-cernopendata-7bf84c89df	19.6 MiB	0.000 cores	May 19, 2022, 12:02 PM
servicex-did-finder-ruccio-676c4f866d-mkxsc	Running	1/1	0	servicex-did-finder-ruccio-676c4f866d	24.5 MiB	0.000 cores	May 19, 2022, 12:21 PM
servicex-minio-66cdf7896-frdqr	Running	1/1	0	servicex-minio-66cdf7896	83.9 MiB	0.001 cores	May 18, 2022, 3:58 PM
servicex-postgresql-0	Running	1/1	0	servicex-postgresql	50.4 MiB	0.003 cores	May 19, 2022, 11:29 AM
servicex-rabbitmq-0	Running	1/1	0	servicex-rabbitmq	180.0 MiB	0.088 cores	May 18, 2022, 12:31 PM
servicex-servicex-app-858cdc48bf-wg6zs	Running	1/1	0	servicex-servicex-app-858cdc48bf	403.5 MiB	0.000 cores	May 19, 2022, 12:00 PM
servicex-x509-secrets-795559d88c-psprs	Running	1/1	0	servicex-x509-secrets-795559d88c	45.6 MiB	-	May 16, 2022, 2:15 PM

*OKD Web Console - Project Pod Listing*

# OKD Web Console (Cont.)



Project: servicex

Deployments > Deployment details

**D** servicex-servicex-app Actions

Details Metrics YAML ReplicaSets Pods Environment Events

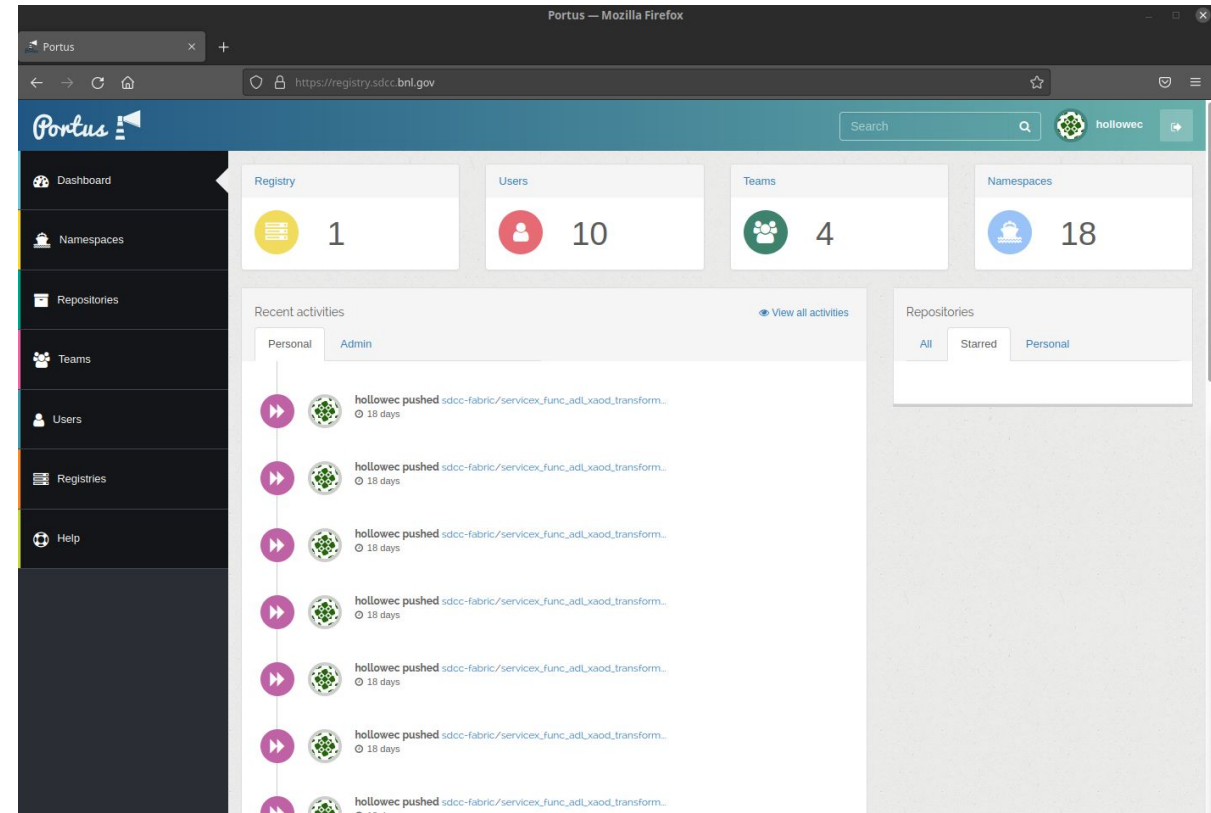
```
1 kind: Deployment
2 apiVersion: apps/v1
3 metadata:
4   annotations:
5     deployment.kubernetes.io/revision: '4'
6     meta.helm.sh/release-name: servicex
7     meta.helm.sh/release-namespace: servicex
8   resourceVersion: '35127767'
9   name: servicex-servicex-app
10  uid: b647fcad-e273-4cc5-965c-954801afb816
11  creationTimestamp: '2022-05-02T19:16:12Z'
12  generation: 4
13  managedFields: ...
163 namespace: servicex
164 labels:
165   app.kubernetes.io/managed-by: Helm
166 spec:
```

Save Reload Cancel Download

OKD Web Console - Editing Deployment YAML

# Local Docker Registry

- We provide users with local private Docker registry where they can store containers for use with OKD
  - Portus user interface
    - Can manage user teams, visibility of containers, etc.
  - Tied into SDCC's IPA for authorization
  - Also only currently accessible internally at BNL
    - Users utilize with local workstations onsite, VPN or ssh SOCKS proxy
      - <https://registry.sdcc.bnl.gov>
  - Eliminates dependence on Dockerhub, or other external registries for critical services



*Private Registry Portus Web Interface*

# Conclusions

---

- Deployed a production OKD Cluster for ATLAS at BNL/SDCC
  - Services including ServiceX and REANA already running on the cluster
- Unlike vanilla k8s, OKD/Openshift provides a secure default configuration that is suitable for multi-tenant use
  - Users are never root in containers by default
- Deployed a local/private Docker container registry service for our users
  - Can be utilized with OKD
  - Eliminates dependence on external registries for important services
- Plan to add additional hardware to our OKD clusters as utilization increases