

Test REANA Deployment at BNL

Chris Hollowell <hollowec@bnl.gov>

k8s-HEP Meetup
Dec 2, 2020

 **BROOKHAVEN** | Scientific Data and
NATIONAL LABORATORY | Computing Center

 U.S. DEPARTMENT OF
ENERGY

What is REANA?

- A platform for reproducible scientific data analysis
 - <https://www.reanahub.io/>
 - Users run workflows via 'reana-client', and can view job status, outputs/plots via web interface
- Open source software being developed at CERN
- Implements reproducibility via containerized workflows
 - Deployment via k8s
 - Version 0.7 added chart for helm-based deployment
 - User containers run in k8s
 - Can also interface with HTCondor, but currently CERN-specific
- Software is in active development - in “developer preview” release state



Test REANA Deployment at BNL

- Interested in testing REANA, as it may become an important component of future HEP analysis facilities
- Deployed a test instance at scale using helm on our 6-node staff k8s (v1.18) cluster
 - Followed directions available here:
<http://docs.reana.io/development/deploying-at-scale/>
 - Added reana helm repo and created a new namespace for reana

```
$ helm repo add reanahub https://reanahub.github.io/reana
$ helm repo update
$ kubectl create namespace reana
```
 - Created a back-end NFS storage provisioner, setting “name: reana-shared-volume-storage-class” for the storage class in the values file:

```
$ helm install reana-dev-storage stable/nfs-server-provisioner -f ./nfs.yaml --namespace reana
```
 - Deployed REANA itself, using created NFS storage backend

```
$ helm install reana reanahub/reana --namespace reana --wait --set shared_storage.backend=nfs
```
 - Some additional steps to initialize DB and add admin user

Test REANA Deployment at BNL (Cont.)

- Using default Traefik ingress controller
 - Can disable automatic traefik installation via traefik.enabled helm value
 - Appears to support other ingress controllers via ingress.annotations.kubernetes.io/ingress.class setting, but I did not test
- Web interface not open to the world, only internally at our facility
 - Need to access via SSH SOCKS proxy, SSH tunnel, or X11 forwarded/NX browser
- Currently using separate local accounts, not our LDAP/K5
 - In discussions with REANA developers about tying access into our IDP
 - Local test users currently signup via the web interface
 - Requires admin token approval before they can actually run workflows
 - Would prefer to disable web signup interface
 - Admins would just handle the entire account creation process
 - Requested the option to disable this feature in a github ticket

Test REANA Deployment at BNL (Cont.)

- REANA pods

```
$ kubectl -n reana get pods
```

NAME	READY	STATUS	RESTARTS	AGE
reana-cache-547d579864-xpgdd	1/1	Running	0	8d
reana-db-f5494cb59-6tmn2	1/1	Running	0	8d
reana-dev-storage-nfs-server-provisioner-0	1/1	Running	0	8d
reana-message-broker-748495898d-ccvgx	1/1	Running	0	8d
reana-server-77498d757f-5lms9	2/2	Running	0	8d
reana-traefik-777f695bdf-bgwdk	1/1	Running	0	8d
reana-ui-5669d4764-drbk8	1/1	Running	0	8d
reana-workflow-controller-764b9489bf-6tx7n	2/2	Running	0	8d

- REANA User Administration

```
$ kubectl -n reana exec -i -t reana-server-77498d757f-5lms9 -- /bin/bash
```

```
Defaulting container name to rest-api.
```

```
Use 'kubectl describe pod/reana-server-77498d757f-5lms9 -n reana' to see all of the containers in this pod.
```

```
root@reana-server-77498d757f-5lms9:/code# flask reana-admin user-list --admin-access-token XYZ
```

ID	EMAIL	ACCESS_TOKEN	
00000000-0000-0000-0000-000000000000	hollowec@bnl.gov	ABC	active
65ba0caa-1dac-4a34-a2eb-5dcb588a4fdc	caramarc@bnl.gov	XYZ	active

```
...
```

Test REANA Deployment at BNL (Cont.)

- Importantly, user containers run as non-root user - UID 1000

```
$ kubectl -n reana get pod reana-run-job-bdb72671-7f1c-428d-9754-02b09b817ad8-h6pp2 -ojson
```

```
...
"nodeName": "kubnode04.sdcc.bnl.gov",
"priority": 0,
"restartPolicy":
"Never",
"schedulerName":
"default-scheduler",
"securityContext": {
  "runAsGroup": 0,
  "runAsUser": 1000
}, ...
```

```
kubnode04# ps auwxxf
```

```
...
1000 2456 0.0 0.0 36360 1664 ? S 09:19 0:00 \_ root -b -q
code/gendata.C(20000,"results/data.root")
1000 2457 0.3 0.1 322360 101820 ? S 09:19 0:00 \_ /usr/local/bin/root.exe -splash -b -q
code/gendata.C(20000,"results/data.root")
```

- REANA code is setting securityContext for job pod, from reana-job-controller in `kubernetes_job_manager.py`:

```
self.job["spec"]["template"]["spec"]["securityContext"] = client.V1PodSecurityContext(
    run_as_group=WORKFLOW_RUNTIME_USER_GID, run_as_user=self.kubernetes_uid)
```

Running an Example Workflow

- Installed reana-client package locally in my home directory on our SL7 interactive nodes, and successfully ran REANA's root6-fit demo workflow

Needed to upgrade setuptools locally first

```
$ pip install --user --upgrade setuptools
$ pip install --user reana-client
$ export REANA_SERVER_URL=https://kubmaster01.sdcc.bnl.gov:30443
```

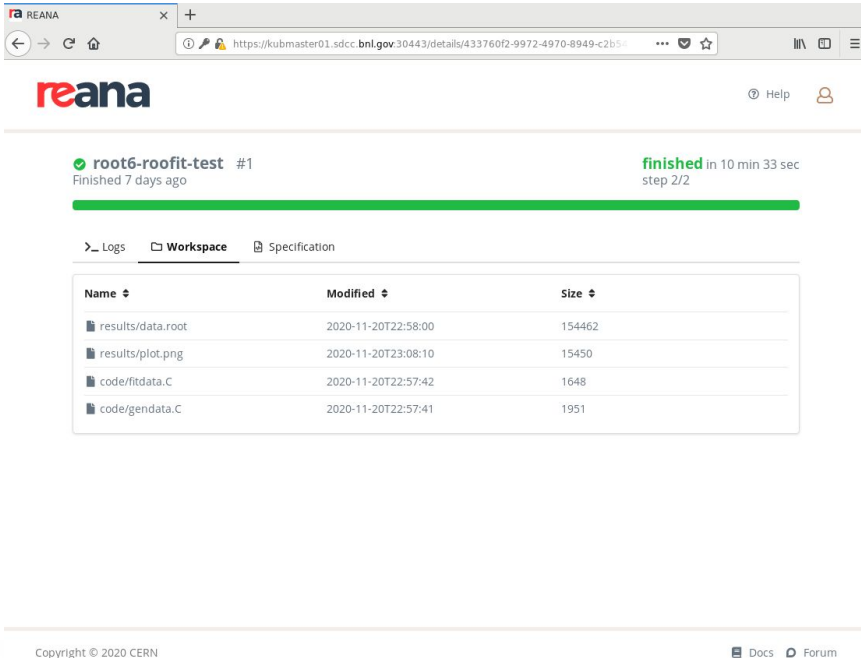
Set REANA_ACCESS_TOKEN to token listed in user's WebUI user profile

```
$ export REANA_ACCESS_TOKEN=XYZXYZXYZ
$ cd reana-demo-root6-fit
$ cat ./reana.yaml
```

```
# reana.yaml
version: 0.6.0
inputs:
  files:
    - code/gendata.C
    - code/fitdata.C
  parameters:
    events: 20000
    data: results/data.root
    plot: results/plot.png
workflow:
  type: serial
  specification:
    steps:
      - name: gendata
        environment: 'reanahub/reana-env-root6:6.18.04'
        commands:
          - mkdir -p results && root -b -q
            'code/gendata.C(${events},${data}) '
      - name: fitdata
        environment: 'reanahub/reana-env-root6:6.18.04'
        commands:
          - root -b -q 'code/fitdata.C("${data}","${plot}")'
    outputs:
      files:
        - results/plot.png
```

Running an Example Workflow (Cont.)

```
$ pwd
/home/chris/reana-demo-root6-fit
$ reana-client run -w root6-roofit-test
[INFO] Creating a workflow...
root6-roofit-test.1
[INFO] Uploading files...
File /code/gendata.C was successfully
uploaded.
File /code/fitdata.C was successfully
uploaded.
[INFO] Starting workflow...
root6-roofit-test.1 has been queued
```

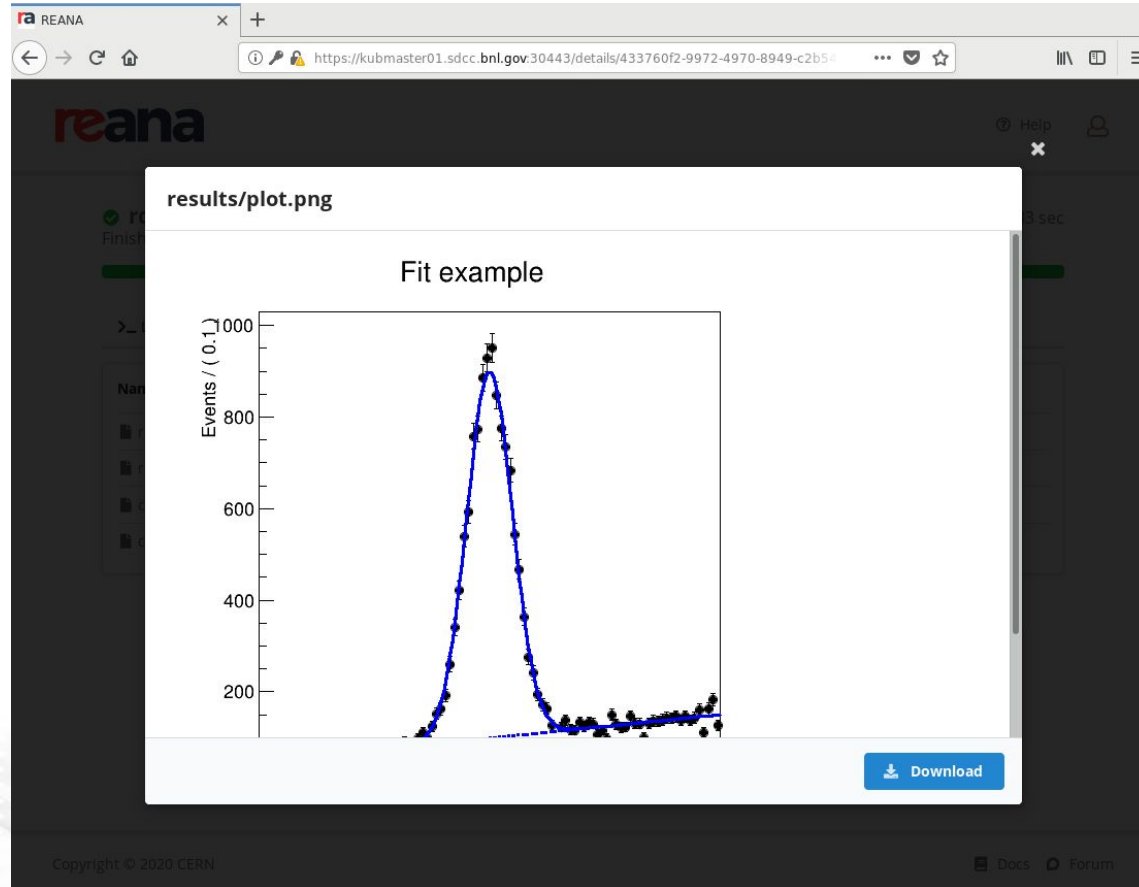


The screenshot shows the REANA web interface in a browser. The page title is "REANA" and the URL is "https://kubmaster01.sdcc.bnl.gov:30443/details/433760f2-9972-4970-8949-c2b5...". The main content area displays a workflow named "root6-roofit-test #1" which is "finished" in 10 min 33 sec, step 2/2. Below this, there are tabs for "Logs", "Workspace", and "Specification". The "Workspace" tab is active, showing a table of files:

Name	Modified	Size
results/data.root	2020-11-20T22:58:00	154462
results/plot.png	2020-11-20T23:08:10	15450
code/fitdata.C	2020-11-20T22:57:42	1648
code/gendata.C	2020-11-20T22:57:41	1951

At the bottom of the page, there is a copyright notice "Copyright © 2020 CERN" and links for "Docs" and "Forum".

Running an Example Workflow (Cont.)



Openshift at BNL

- Besides our staff-only k8s cluster, BNL also has an OKD/openshift (OKD 4.6) cluster for user service deployment use
 - Why? Because k8s RBAC/security is complex, making multi-tenant k8s difficult to securely setup
 - In k8s cloud providers (GKE, EKS, etc.) customers provision their own personal clusters - not one big k8s cluster that every customer shares
 - k8s security seems to be somewhat of an afterthought, with loose policies in place by default
 - Critical PodSecurityPolicy functionality not enabled by default on the kube-apiserver commandline
 - Pod Security Policies is not an officially released feature in the current v1.19 version - still considered beta
 - As a result, not uncommon to see typical k8s with Docker clusters setup where every user is a full admin, and pods are all running as the root user
 - Less of a problem if only trusted staff/services are utilizing it

Openshift at BNL (Cont.)

- Given our heavy use of POSIX uid-auth-based network filesystems, we cannot allow users or external organizations to be root on our networks
 - Being root on the network (ability to open privileged [<1024] ports), or on a host that mounts this storage, means root on the storage, and one can delete/modify files at will
 - One reason Singularity has been adopted in our community for the portability-of-compute use case instead of Docker, and Docker use is generally not permitted on DoE HPC systems
 - With Singularity regular users are never root in containers (without a user namespace mapping)
- Can be solved by network isolation/partitioning, and pod overlay networks help
 - But by default in k8s, users can escape these and access the host's network
 - CNIs like Calico and Weave support NAT
 - Users can instantiate pods with `"hostNetwork: true"` in the pod YAML

Openshift at BNL (Cont.)

- In contrast, Openshift/OKD comes with a secure/restrictive policy configured by default
 - Suitable for multi-tenant use, and on networks where uid-auth-based network filesystems are being used
 - Pods run by regular/non-admin users are never root
 - Not possible for unprivileged users to access the host's network
 - Setting `"hostNetwork: true"` in the pod YAML is forbidden
 - One issue is that some helm charts expect cluster-level admin privileges
- Some other Openshift advantages
 - Openshift is a commercially supported enterprise product
 - Provides users with a convenient dashboard/web interface
- All reasons why Openshift/OKD is being adopted at DoE national labs like ORNL, FNAL and BNL

REANA on Openshift

- Unfortunately, was not able to get at REANA deployment to install with helm on Openshift as a regular unprivileged user:

```
$ helm install --devel reana reanahub/reana --wait --namespace reana --set traefik.enabled=false
Error: rendered manifests contain a resource that already exists. Unable to continue with
install: could not get information about the resource:
clusterrolebindings.rbac.authorization.k8s.io "reana-manage-deployments" is forbidden: User
"chris" cannot get resource "clusterrolebindings" in API group "rbac.authorization.k8s.io" at the
cluster scope
```

- Chart expects admin-level permission to set ClusterRole
 - Similar to issue reported by UC, who submitted a pull request to address: <https://github.com/reanahub/reana/pull/291>
 - In discussions with developers about how to resolve

Conclusions

- REANA is platform for reproducible scientific analysis
 - Users run workflows in containers for reproducibility
 - Deployment/orchestration via k8s
- We've deployed a usable test v0.7 REANA cluster at SDCC/BNL
 - Running on our staff k8s cluster
 - Utilized provided helm chart to deploy
 - Ideally would run on our Openshift/OKD cluster instead
 - Issue with helm chart and unprivileged deployment
 - Discussion with developers in REANA gitter
- In contact with the developers about some additional desired features
 - Disabling of local user-signup in the web interface
 - Integration with our IDP/K5
 - The ability to submit jobs to our HTCondor and Slurm clusters