

Load Testing a KumoScale Deployment

This section describes an example of how to set up a load test using the open source software tool FIOLoadGen.

Process for Using FIOLoadgen for Load Testing

FIOLoadgen is an open-source tool for generating load on FIO (the EOSIO-based Interchain Standard) enabled blockchains. It is used to stress-test and benchmark FIO enabled blockchains by simulating a large number of transactions and measuring the performance of the blockchain under different conditions. FIOLoadgen can be used in conjunction with Kubernetes to run load tests on a FIO blockchain that is deployed on a Kubernetes cluster.

Here is a general overview of the process:

Step 1. Deploy FIO Deploy on Kubernetes: You can use a tool like FIO Deploy to create and deploy a number of Kubernetes resources, such as pods and services, to run the FIO blockchain.

Step 2. Run FIOLoadgen on Kubernetes: You can create a Kubernetes deployment that runs FIOLoadgen and points to the FIO blockchain that you deployed in the previous step. You can use a Kubernetes ConfigMap to store the configuration file for FIOLoadgen, and you can use a Kubernetes Job to run the load test.

Step 3. Analyze the results: After the load test is complete, you can analyze the results of the load test, such as the number of transactions per second and the average transaction latency, to identify performance bottlenecks.

For more detailed information and usage you can visit the following link: <https://joshua-robinson.medium.com/storage-benchmarking-with-fio-in-kubernetes-14cf29dc5375>

Example of Load Testing a KumoScale Deployment using FIOLoadgen

Note: This is just one example of how a load test may be set up and the type of information that used for testing. There are other tools available; choose the one best for your environment.

For this example we are using KumoScale in Managed Mode with a Kubernetes cluster configured with storage classes:

- kumoscale-simple, a non-replicated storage class
- kumoscale-replicated, a replicated storage class
- kumoscale-replicated3, a replicated storage class

Step 1. Deploy FIO Deploy on Kubernetes

1. Confirm that you have

- allocated sufficient space to support the volumes that will be created.
- installed and configured FIODeploy to access your Kubernetes environment

2. Run FIODeploy

```
$/fiodeploy -s
```

You will get information on the Kubernetes configuration like:

```
Checking oc/kubectl CLI is available
✓ kubectl command available

Checking port 8080 is free
✓ port 8080 is available

Checking you are logged in to Kubernetes with kubeadmin
? unable to check login state with kubectl
```

You will be asked for the namespace to use. In this example we select the default (fio):

```
FIOLoadgen will use a create a new namespace to support the test environment
What namespace should be used [fio]?
✓ namespace 'fio' is available
```

FIOLoadgen will display the available storage classes and ask which to include in testing. We include all three:

```
Checking available storageclasses
- kumoscale-replicated-class
- kumoscale-replicated-class3
```

```
- kumoscale-simple-class
You may select multiple storageclasses. Press <ENTER> to end your selection.
Storageclass name: kumoscale-replicated-class
Storageclass name: kumoscale-replicated-class3
Storageclass name: kumoscale-simple-class
Storageclass name:
✓ workers will be deployed to 3 storageclasses
```

You will be prompted for the number of workers (nodes) to create for each storage class. We enter **20**:

```
How many fio workers (1-20) per storageclass [3]? 20
✓ 20 worker pods will be deployed to each required storageclass
```

You will be asked how you want to manage tests. We enter **remote**:

```
FIOLoadgen
To manage the tests, FIOLoadgen can use either a local daemon on your machine (local), or
deploy the management daemon to the target environment (remote)
How do you want to manage the tests (local/remote) [local]? remote
✓ fioservice will run in 'remote' mode
```

A summary appears with information on the configuration selected:

```
Deployment Summary

Namespace      : fio
Storageclass   :
- kumoscale-replicated-class
- kumoscale-replicated-class3
- kumoscale-simple-class
FIO Workers    : 20
FIO service    : remote
```

You are asked whether you are ready to create the test configuration. We enter **y(es)** and receive information on the environment created.

```
Ready to deploy (y/n) ? y

Starting deployment

Creating namespace (fio)
✓ namespace created OK

Deploying the FIO worker statefulset(s)
✓ statefulset.apps/fioworker-kumoscale-replicated-class created
✓ statefulset.apps/fioworker-kumoscale-replicated-class3 created
✓ statefulset.apps/fioworker-kumoscale-simple-class created

Waiting for worker pods to reach ready state
✓ All worker pods ready

Submitting the deployment for the fioservice daemon
✓ deployment created
```

FIOLoadgen displays the service for testing and URL for initiating load tests.

```
Adding port-forward rule (port 8080)
✓ pod/fioservice-c4fc46677-v6ts7 is ready
✓ port-forward created successfully (pid=602)

Access the UI at http://localhost:8080. From there you may submit jobs and view
```

Step 2. Run FIOLoadgen on Kubernetes:

Access and run the load test by going to the FIOLoadgen UI at the above URL <http://localhost:8080>. Use the UI to set up jobs and run tests as shown in the following screenshots. An example of how to set up and view the results is shown in the following screenshots:

- 1. Select the job. For example, randw7030.job, where transactions are split 70% read and 30% write.

136 rows, 2 selected

Next: [Snapshot Management](#)