MLBench Helm Documentation

MLBench development team

Dec 07, 2020

MLBENCH

CHAPTER

ONE

MLBENCH: DISTRIBUTED MACHINE LEARNING BENCHMARK HELM CHART

The Helm Chart is used to deploy MLBench to a Kubernetes cluster. The source can be found in the Helm repository .

1.1 Chart Details

This Chart deploys the following:

- 1 x MLBench Dashboard/Master Node with Port 80 exposed (Dashboard and REST API)
- 2 x MLBench Worker Nodes, connecting to the REST API of the Dashboard, with Port 22 (SSH) exposed inside the cluster

1.2 Prerequisites

- Helm
- Helm needs to be set up with service-account with `cluster-admin` rights:

1.3 Installing the Chart

To install the chart with the release name `my-release` and values file `values.yaml`:

```
$ git clone https://github.com/mlbench/mlbench-helm.git
$ cd mlbench-helm
$ helm install -f values.yaml --name my-release ./
```

1.4 Configuration

The following tables list configurable parameters of the MLBench chart and their default values. Entries without default values are mandatory.

Specify each parameter using the `--set key=value[,key=value]` argument to `helm install`.

Alternatively, a YAML file that specifies the values for the parameters can be provided while installing the chart. For example,

```
$ helm install --name my-release -f values.yaml stable/dask
```

Tip: You can use the default `values.yaml`

1.4.1 Dashboard/Master Node

Parameter	Description	Default
`master.enabled`	Whether to deploy the master node or not	`true`
`master.name`	The name of the node	`master`
`master.image.repository`	The Docker Registry to use	`mlbench/mlbench_master`
`master.image.tag`	The tag of the image to use	`latest`
`master.image.pullPolicy`	The K8s imagePullPolicy	`Always`
`master.service.type`	The K8s service type	`NodePort`
`master.service.port`	The port to expose in K8s	`80`

1.4.2 Worker Nodes

Parameter	Description	Default
`worker.sshKey.id_rsa`	The SSH Private Key	(not shown)
`worker.sshKey.id_rsa`	The SSH Public Key	(not shown)

1.4.3 Hardware Limits

Important: These values are mandatory.

Parameter	Description	Default
`limits.workers`	The maximum number of workers that can be comissioned	
`limits.cpu`	The maximum number of cpu cores that can be comissioned per worker	
`limits.gpu`	The maximum number of GPUs that can be comissioned per worker	

1.4.4 Google Cloud Storage

If deploying to the Google Cloud, use these to set the shared storage for workers.

Parameter	Description	Default
`gcePersistentDisk.enabled`	Whether to use Google Cloud Storage	`false`
`gcePersistentDisk.pdName`	The name of the persistent Disk to use	

1.4.5 Weave

Settings concerning WeaveNet, a Networking Solution between K8s pods. Necessary in some cases where the SourceIP of a Pod defaults to the IP of the Node it's on, which can cause troubles with MPI execution.

Parameter	Description	Default
`weave.enabled`	Whether to use WeaveNet	`false`

1.4.6 NVIDIA Device Plugin

Needed to support NVIDIA GPUs in workers (unless already provided by your K8s provider.

Parameter	Description	Default
`nvidiaDevicePlugin.enabled`	Whether to use the NVIDIA Device Plugin	`false`

CHAPTER

TWO

INDICES AND TABLES

- genindex
- modindex
- search