Deploy a Kubernetes Cluster with NVIDIA DeepOps Automated Deployment

NetApp Solutions

Kevin Hoke
May 24, 2021

Table of Contents

Deploy a Kubernetes Cluster with NVIDIA DeepOps Automated Deployment 1

Deploy a Kubernetes Cluster with NVIDIA DeepOps Automated Deployment

To deploy and configure the Kubernetes Cluster with NVIDIA DeepOps, complete the following steps:

1. Make sure that the same user account is present on all the Kubernetes master and worker nodes.
2. Clone the DeepOps repository.

   ```
   git clone https://github.com/NVIDIA/deepops.git
   ```

3. Check out a recent release tag.

   ```
   cd deepops
   git checkout tags/20.08
   ```

   If this step is skipped, the latest development code is used, not an official release.

4. Prepare the Deployment Jump by installing the necessary prerequisites.

   ```
   ./scripts/setup.sh
   ```

5. Create and edit the Ansible inventory by opening a VI editor to `deepops/config/inventory`.
   a. List all the master and worker nodes under [all].
   b. List all the master nodes under [kube-master]
   c. List all the master nodes under [etcd]
   d. List all the worker nodes under [kube-node]

```
######
# ALL NODES
# NOTE: Use existing hostnames here, DeepOps will conf
######
[all]
hci-ai-k8-master-01     ansible_host=172.21.232.114
hci-ai-k8-master-02     ansible_host=172.21.232.115
hci-ai-k8-master-03     ansible_host=172.21.232.116
hci-ai-k8-worker-01     ansible_host=172.21.232.109
hci-ai-k8-worker-02     ansible_host=172.21.232.110

######
# KUBERNETES
######
[kube-master]
hci-ai-k8-master-01
hci-ai-k8-master-02
hci-ai-k8-master-03

# Odd number of nodes required
[etcd]
hci-ai-k8-master-01
hci-ai-k8-master-02
hci-ai-k8-master-03

# Also add mgmt/master nodes here if they will run non
[kube-node]
hci-ai-k8-worker-01
hci-ai-k8-worker-02

[k8s-cluster:children]
kube-master
kube-node
```

6. Enable GPUOperator by opening a VI editor to `deepops/config/group_vars/k8s-cluster.yml`.

```
# Provide option to use GPU Operator instead of setting up NVIDIA driver and
# Docker configuration.
deepops_gpu_operator_enabled: true
```

7. Set the value of `deepops_gpu_operator_enabled` to true.
8. Verify the permissions and network configuration.

```
ansible all -m raw -a "hostname" -k -K
```

- If SSH to the remote hosts requires a password, use -k.
- If sudo on the remote hosts requires a password, use -K.

9. If the previous step passed without any issues, proceed with the setup of Kubernetes.

```
ansible-playbook --limit k8s-cluster playbooks/k8s-cluster.yml -k -K
```

10. To verify the status of the Kubernetes nodes and the pods, run the following commands:

```
kubectl get nodes
```

```
rarvind@deployment-jump:~/deepops$ kubectl get nodes
NAME                 STATUS   ROLES    AGE     VERSION
hci-ai-k8-master-01   Ready    master   2d19h   v1.17.6
hci-ai-k8-master-02   Ready    master   2d19h   v1.17.6
hci-ai-k8-master-03   Ready    master   2d19h   v1.17.6
hci-ai-k8-worker-01   Ready    <none>   2d19h   v1.17.6
hci-ai-k8-worker-02   Ready    <none>   2d19h   v1.17.6
```

```
kubectl get pods -A
```

It can take a few minutes for all the pods to run.

```
rarvind@deployment-jump:~/deepops$ kubectl get pods -A
NAMESPACE                NAME                                                              READY   STATUS
default                  gpu-operator-74c97448d9-ppdlc                                     1/1     Running
default                  nvidia-gpu-operator-node-feature-discovery-master-ffccb57dx9wtl   1/1     Running
default                  nvidia-gpu-operator-node-feature-discovery-worker-2lr9t           1/1     Running
default                  nvidia-gpu-operator-node-feature-discovery-worker-616x7           1/1     Running
default                  nvidia-gpu-operator-node-feature-discovery-worker-jf696           1/1     Running
default                  nvidia-gpu-operator-node-feature-discovery-worker-tmtwv           1/1     Running
default                  nvidia-gpu-operator-node-feature-discovery-worker-z4nlh           1/1     Running
gpu-operator-resources   nvidia-container-toolkit-daemonset-7jbl4                          1/1     Running
gpu-operator-resources   nvidia-container-toolkit-daemonset-x5ktb                          1/1     Running
gpu-operator-resources   nvidia-dcgm-exporter-5x94p                                        1/1     Running
gpu-operator-resources   nvidia-dcgm-exporter-7cbrl                                        1/1     Running
gpu-operator-resources   nvidia-device-plugin-daemonset-n8vrk                              1/1     Running
gpu-operator-resources   nvidia-device-plugin-daemonset-z7j6s                              1/1     Running
gpu-operator-resources   nvidia-device-plugin-validation                                   0/1     Completed
gpu-operator-resources   nvidia-driver-daemonset-7h752                                     1/1     Running
gpu-operator-resources   nvidia-driver-daemonset-v4rbj                                     1/1     Running
gpu-operator-resources   nvidia-driver-validation                                          0/1     Completed
kube-system              calico-kube-controllers-777478f4ff-jknxg                          1/1     Running
kube-system              calico-node-2j9mr                                                 1/1     Running
kube-system              calico-node-czk76                                                 1/1     Running
kube-system              calico-node-jpdxn                                                 1/1     Running
kube-system              calico-node-nwnvn                                                 1/1     Running
kube-system              calico-node-ssjrx                                                 1/1     Running
kube-system              coredns-76798d84dd-5pvgf                                          1/1     Running
kube-system              coredns-76798d84dd-w7l2j                                          1/1     Running
kube-system              dns-autoscaler-85f898cd5c-qqrbp                                   1/1     Running
kube-system              kube-apiserver-hci-ai-k8-master-01                                1/1     Running
kube-system              kube-apiserver-hci-ai-k8-master-02                                1/1     Running
kube-system              kube-apiserver-hci-ai-k8-master-03                                1/1     Running
kube-system              kube-controller-manager-hci-ai-k8-master-01                       1/1     Running
kube-system              kube-controller-manager-hci-ai-k8-master-02                       1/1     Running
kube-system              kube-controller-manager-hci-ai-k8-master-03                       1/1     Running
kube-system              kube-proxy-5znxk                                                  1/1     Running
kube-system              kube-proxy-fk6h6                                                  1/1     Running
kube-system              kube-proxy-hphfb                                                  1/1     Running
kube-system              kube-proxy-qzxhr                                                  1/1     Running
kube-system              kube-proxy-rkjds                                                  1/1     Running
kube-system              kube-scheduler-hci-ai-k8-master-01                                1/1     Running
kube-system              kube-scheduler-hci-ai-k8-master-02                                1/1     Running
kube-system              kube-scheduler-hci-ai-k8-master-03                                1/1     Running
kube-system              kubernetes-dashboard-5fcff756f-dmswt                              1/1     Running
kube-system              kubernetes-metrics-scraper-747b4fd5cd-4q4p2                       1/1     Running
kube-system              nginx-proxy-hci-ai-k8-worker-01                                   1/1     Running
kube-system              nginx-proxy-hci-ai-k8-worker-02                                   1/1     Running
kube-system              nodelocaldns-2dmjr                                                1/1     Running
kube-system              nodelocaldns-b7xrw                                                1/1     Running
kube-system              nodelocaldns-jrhs2                                                1/1     Running
kube-system              nodelocaldns-jztzs                                                1/1     Running
kube-system              nodelocaldns-wgx84                                                1/1     Running
```

11. Verify that the Kubernetes setup can access and use the GPUs.

```
./scripts/k8s_verify_gpu.sh
```

Expected sample output:

```
rarvind@deployment-jump:~/deepops$ ./scripts/k8s_verify_gpu.sh
job_name=cluster-gpu-tests
Node found with 3 GPUs
Node found with 3 GPUs
total_gpus=6
Creating/Deleting sandbox Namespace
updating test yml
downloading containers ...
```

```
job.batch/cluster-gpu-tests condition met
executing ...
Mon Aug 17 16:02:45 2020
+-----------------------------------------------------------------------------
------+
| NVIDIA-SMI 440.64.00    Driver Version: 440.64.00    CUDA Version:
10.2     |
|-------------------------------+----------------------
+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile
Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util
Compute M. |
|===============================+======================+================
======|
|   0  Tesla T4            On   | 00000000:18:00.0 Off |
0 |
| N/A   38C    P8    10W /  70W |      0MiB / 15109MiB |      0%
Default |
+-------------------------------+----------------------
+----------------------+
+-----------------------------------------------------------------------------
------+
| Processes:                                                       GPU
Memory |
|  GPU       PID   Type   Process name                             Usage
|
|=============================================================================
======|
|  No running processes found
|
+-----------------------------------------------------------------------------
------+
Mon Aug 17 16:02:45 2020
+-----------------------------------------------------------------------------
------+
| NVIDIA-SMI 440.64.00    Driver Version: 440.64.00    CUDA Version:
10.2     |
|-------------------------------+----------------------
+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile
Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util
Compute M. |
|===============================+======================+================
======|
```

```
|   0  Tesla T4            On   | 00000000:18:00.0 Off |
0 |
| N/A   38C    P8    10W /  70W |      0MiB / 15109MiB |      0%
Default |
+-------------------------------+----------------------
+----------------------+
+-----------------------------------------------------------------------------
------+
| Processes:                                                                  GPU
Memory |
|  GPU       PID   Type   Process name                                  Usage
|
|=============================================================================
======|
|  No running processes found
|
+-----------------------------------------------------------------------------
------+
Mon Aug 17 16:02:45 2020
+-----------------------------------------------------------------------------
------+
| NVIDIA-SMI 440.64.00    Driver Version: 440.64.00    CUDA Version:
10.2     |
|-------------------------------+----------------------
+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile
Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util
Compute M. |
|===============================+======================+================
======|
|   0  Tesla T4            On   | 00000000:18:00.0 Off |
0 |
| N/A   38C    P8    10W /  70W |      0MiB / 15109MiB |      0%
Default |
+-------------------------------+----------------------
+----------------------+
+-----------------------------------------------------------------------------
------+
| Processes:                                                                  GPU
Memory |
|  GPU       PID   Type   Process name                                  Usage
|
|=============================================================================
======|
|  No running processes found
```

```
|
+-----------------------------------------------------------------------------
------+
Mon Aug 17 16:02:45 2020
+-----------------------------------------------------------------------------
------+
| NVIDIA-SMI 440.64.00    Driver Version: 440.64.00    CUDA Version:
10.2     |
|-------------------------------+----------------------
+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile
Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util
Compute M. |
|===============================+======================+===============
======|
|   0  Tesla T4            On   | 00000000:18:00.0 Off |
0 |
| N/A   38C    P8    10W /  70W |      0MiB / 15109MiB |      0%
Default |
+-------------------------------+----------------------
+----------------------+
+-----------------------------------------------------------------------------
------+
| Processes:                                                                GPU
Memory |
|  GPU       PID   Type   Process name                             Usage
|
|=============================================================================
======|
|  No running processes found
|
+-----------------------------------------------------------------------------
------+
Mon Aug 17 16:02:45 2020
+-----------------------------------------------------------------------------
------+
| NVIDIA-SMI 440.64.00    Driver Version: 440.64.00    CUDA Version:
10.2     |
|-------------------------------+----------------------
+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile
Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util
Compute M. |
|===============================+======================+===============
```

```
======|
|   0  Tesla T4            On   | 00000000:18:00.0 Off |
0 |
| N/A   38C    P8    10W /  70W |      0MiB / 15109MiB |      0%
Default |
+-------------------------------+----------------------
+----------------------+
+-----------------------------------------------------------------------
------+
| Processes:                                                       GPU
Memory |
|  GPU       PID   Type   Process name                             Usage
|
|=======================================================================
======|
|  No running processes found
|
+-----------------------------------------------------------------------
------+
Mon Aug 17 16:02:45 2020
+-----------------------------------------------------------------------
------+
| NVIDIA-SMI 440.64.00    Driver Version: 440.64.00    CUDA Version:
10.2     |
|-------------------------------+----------------------
+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile
Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util
Compute M. |
|===============================+======================+================
======|
|   0  Tesla T4            On   | 00000000:18:00.0 Off |
0 |
| N/A   38C    P8    10W /  70W |      0MiB / 15109MiB |      0%
Default |
+-------------------------------+----------------------
+----------------------+
+-----------------------------------------------------------------------
------+
| Processes:                                                       GPU
Memory |
|  GPU       PID   Type   Process name                             Usage
|
|=======================================================================
======|
```

```
|  No running processes found
|
+-----------------------------------------------------------------------------
------+
Number of Nodes: 2
Number of GPUs: 6
6 / 6 GPU Jobs COMPLETED
job.batch "cluster-gpu-tests" deleted
namespace "cluster-gpu-verify" deleted
```

12. Install Helm on the Deployment Jump.

```
./scripts/install_helm.sh
```

13. Remove the taints on the master nodes.

```
kubectl taint nodes --all node-role.kubernetes.io/master-
```

This step is required to run the LoadBalancer pods.

14. Deploy LoadBalancer.
15. Edit the `config/helm/metallb.yml` file and provide a range of IP ddresses in the `Application Network` to be used as LoadBalancer.

```
---
# Default address range matches private network for the virtual cluster
# defined in virtual/.
# You should set this address range based on your site's infrastructure.
configInline:
  address-pools:
  - name: default
    protocol: layer2
    addresses:
    - 172.21.231.130-172.21.231.140#Application Network
controller:
  nodeSelector:
    node-role.kubernetes.io/master: ""
```

16. Run a script to deploy LoadBalancer.

```
./scripts/k8s_deploy_loadbalancer.sh
```

17. Deploy an Ingress Controller.

```
./scripts/k8s_deploy_ingress.sh
```

Next: Deploy and Configure ONTAP Select in the VMware Virtual Infrastructure (Automated Deployment)