

Cluster Migration

Migrating a Kubernetes cluster from an older version (v1.19.3) running on CentOS 7.9 with Docker 20.10 to a new cluster running on Ubuntu VMs involves several steps. Here's a detailed migration plan to help you through the process:

1. Planning and Preparation

1. Assess the Current Environment:

- Document the existing cluster setup, including network configuration, node specifications, storage configurations, and any customizations or integrations.
- List all the workloads, services, and applications running in the cluster.
- Review the Kubernetes and Docker configurations, including any custom or third-party components.

2. Set Up the New Environment:

- Provision new Ubuntu VMs for the new Kubernetes cluster.
- Ensure these VMs have the required resources (CPU, memory, disk space) and network configurations.

3. Choose the New Kubernetes Version:

- Select the desired Kubernetes version for the new cluster. Ensure that it's compatible with the latest supported Containerd version and your applications.

2. Set Up the New Kubernetes Cluster

1. Install Kubernetes:

- Use a tool like `kubeadm`, `kops`, or a managed service to set up the new Kubernetes cluster on Ubuntu.
- Ensure that the new cluster's version is compatible with your applications.

2. Install Containerd (or Container Runtime):

- Install the recommended Containerd version or other compatible container runtimes on the new Ubuntu VMs.

3. Configure Networking:

- Set up networking plugins (e.g., Calico, Flannel) that are compatible with your new Kubernetes version.

4. Secure the Cluster:

- Apply security best practices, including configuring RBAC, network policies, and securing etcd.

3. Prepare for Migration

1. Document Current Workloads:

- Pull latest Kubernetes resource definitions (Deployments, Services, ConfigMaps, Secrets, etc.) and save them in YAML files.

2. Prepare NFS for the New Cluster:

- Ensure the NFS server is accessible from the new cluster nodes.
- Confirm that the NFS server is properly configured to handle the data storage requirements.

4. Migrate Workloads and Data

1. Restore Configuration and Data:

- Restore etcd backups to the new cluster (if necessary). This is often not required unless you're specifically migrating the etcd database.
- Apply the saved Kubernetes resource definitions to the new cluster using `kubectl apply -f` commands.

2. Migrate Persistent Data:

- If using persistent volumes, migrate the data according to the storage solution's recommendations (e.g., re-attaching or migrating snapshots).

3. Test the New Environment:

- Verify that all applications and services are running as expected in the new cluster.
- Perform tests to ensure functionality, performance, and connectivity.

5. Cut Over to the New Cluster

1. Switch Traffic:

- Update DNS records, load balancer configurations, or any other network configurations to point to the new cluster.

2. Monitor the New Cluster:

- Closely monitor the new cluster for any issues or performance problems.
- Ensure all systems are functioning as expected and that no critical issues are present.

6. Decommission the Old Cluster

1. Verify Migration:

- Ensure that everything has been successfully migrated and is functioning correctly.

2. Decommission Old Infrastructure:

- Safely shut down and decommission the old Kubernetes cluster and associated resources.

3. Cleanup:

- Remove any leftover configurations or resources related to the old cluster.

7. Post-Migration Tasks

1. Update Documentation:

- Update your documentation to reflect the new cluster setup and any changes made during the migration.

2. Review and Optimize:

- Review the performance and configuration of the new cluster and make any necessary adjustments.

3. Conduct a Post-Migration Review:

- Gather feedback and review the migration process to identify any areas for improvement.

By following these steps, you can effectively migrate your Kubernetes workloads from an older cluster on CentOS to a new cluster on Ubuntu, while ensuring minimal disruption to your services.