INSTALLATION AND CONFIGURATION

Installation and Configuration

Installation Considerations

To begin the installation process, you should start experimenting with a single-node deployment. This single-node will run all the Kubernetes components (e.g., API server, controller, scheduler, kubelet, and kube-proxy). You can do this with Minikube for example.

Once you want to deploy on a cluster of servers (physical or virtual), you will have many choices to make, just like with any other distributed system:

- Which provider should I use? A public or private cloud? Physical or virtual?
- Which operating system should I use? Kubernetes runs on most operating systems (e.g. Debian, Ubuntu, CentOS, etc.), plus on container-optimized OSes (e.g. CoreOS, Atomic).
- Which networking solution should I use? Do I need an overlay?
- Where should I run my etcd cluster?
- Can I configure Highly Available (HA) head nodes?

To learn more about how to choose the best options, you can read the *Getting Started* documentation page.

With systemd becoming the dominant init system on Linux, your Kubernetes components will end up being run as *systemd unit files* in most cases. Or, they will be run via a kubelet running on the head node (i.e. **kubadm**).

Lab exercises in this course were written using Google Compute Engine (GCE) nodes. Each node has 2vCPUs and 7.5GB of memory, running Ubuntu 20.04. Smaller nodes should work, but you should expect slow response. Other operating system images are also possible, but there may be slight differences in some command outputs. Use of GCE requires setting up an account and will incur expenses if using nodes of the size suggested. You can view the *Getting Started* pages for more details.

Amazon Web Services (AWS) is another provider of cloud-based nodes, and requires an account; you will incur expenses for nodes of the suggested size. You can find videos and information of how to get started online.

Virtual machines such as KVM, VirtualBox, or VMware can also be used for lab systems. Putting the VMs on a private network can make troubleshooting easier.

Finally, using bare metal nodes with access to the Internet will also work for lab exercises.