Kubernetes Installation and Configuration Fundamentals

by [Anthony Nocentino](https://app.pluralsight.com/profile/author/anthony-nocentino)

* [Description](https://app.pluralsight.com/library/courses/kubernetes-installation-configuration-fundamentals/description)

[Course Overview](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac)

[Course Overview](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac)

[Hi everyone. My name is Anthony Nocentino,](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=null) [enterprise architect and founder of Centino Systems.](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=6.615) [Welcome to my course, Kubernetes Installation and Configuration Fundamentals.](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=9.217) [Are you a systems administrator that needs to build](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=15.231) [and administer a Kubernetes cluster?](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=17.156) [If you do, then this is the course for you.](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=18.941) [First,](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=21.263) [we'll start off by introducing you to the Kubernetes and the](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=21.68042857142857) [world of deploying container‑based applications.](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=25.39357142857142) [We'll look closely at Kubernetes architecture, including](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=28.077) [cluster components and networking fundamentals.](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=31.297) [Then we'll look at how to build your own Kubernetes cluster,](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=35.617) [and we'll look at installation considerations, such as](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=38.568) [deploying on premises or deploying in the cloud.](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=41.665) [Together,](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=45.552) [we'll deploy clusters in both scenarios in some demonstrations. And](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=45.83316666666666) [we'll wrap it up with learning how we can interact with our cluster at](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=50.98188888888889) [the command line using kubectl and the basics of application and](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=54.21357142857143) [service deployments in Kubernetes.](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=59.45657142857142) [At the end of this course, you'll know how to build a Kubernetes cluster,](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=62.624) [both in local virtual machines and as a managed service](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=65.418) [with some of the major cloud providers.](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=68.72914285714285) [You'll learn what it takes to get an application up and](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=70.38066666666668) [running in your Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=73.48074999999997) [Before beginning this course,](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=75.016) [you should be familiar with the Linux operating system and](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=76.00914285714286) [administering it at the command line.](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=79.52585714285715) [You should have a firm understanding of TCP/IP‑based networking and also](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=81.679) [understand the fundamental concepts of containers.](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=85.91383333333336) [I hope you'll join me on this journey to learn how to build a](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=88.995) [Kubernetes cluster and deploying container‑based applications](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=93.52437499999996) [in Kubernetes in the course, Kubernetes Installation and Configuration Fundamentals.](https://app.pluralsight.com/course-player?clipId=5eb3e712-5e73-409c-9dc1-a86e870265ac&startTime=96.60050000000001)

[Exploring the Kubernetes Architecture](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a)

[Introduction, Course, and Module Overview](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a)

[Hello, this is Anthony Nocentino with Centino Systems.](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=1.14) [Welcome to my course, Kubernetes Installation and Configuration Fundamentals.](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=4.03) [This module is the Introduction and Exploring Kubernetes Architecture module.](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=8.68) [In this course,](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=13.7) [we're going to get you started on your journey into building](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=14.65) [and configuring your own Kubernetes cluster,](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=17.46) [whether it's on‑prem or in the cloud.](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=20.46) [So let's start off with a course overview.](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=23.34) [We're going to kick it off with an introduction,](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=25.32) [and then we're going to look at the theory and the concepts behind](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=27.1) [Kubernetes that makes, well, Kubernetes, Kubernetes in the module,](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=30.5) [Exploring Kubernetes Architecture.](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=34.79) [Once we have that theory in our brains,](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=36.94) [we're going to move on into installing and configuring Kubernetes. We're](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=38.52) [going to look at both on‑prem and cloud scenarios,](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=42.61) [doing a deep dive into how that really all pieces together](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=45.73) [in building our first cluster together.](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=48.88) [And then once we have that cluster up and running,](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=50.78) [I don't want to leave you hanging.](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=52.92) [We're going to go ahead and learn how to work with our](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=54.49) [Kubernetes cluster and do some basic operations,](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=56.87) [like deploying a service and an application into our](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=59.83) [newly built Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=62.75) [All right, In this module,](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=65.54) [we're going to kick it off with the discussion to](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=66.91) [answer the question, what is Kubernetes?](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=69.19) [And once we know that,](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=71.35) [then we're going to move on to exploring the Kubernetes architecture.](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=72.3) [We're going to zoom in very closely on the cluster's components and also some basic networking fundamentals.](https://app.pluralsight.com/course-player?clipId=ca261788-675d-41c2-98a4-a55d40946c9a&startTime=75.62)

[What Is Kubernetes? Kubernetes Benefits and Operating Principles](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da)

[Let's start our discussion with answering the question, what is Kubernetes?](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=1.14) [At its core, Kubernetes is a container orchestrator.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=5.34) [What this means is it's Kubernetes job to start and stop](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=8.64) [container‑based applications based on the requirements of the](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=11.95) [systems administrator or developer.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=14.83) [Now, one of the key facets of Kubernetes is workload placement.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=17.04) [If I need to deploy a container‑based application into a cluster,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=20.94) [how do I deploy it? On which servers does it physically live on?](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=24.74) [Does it need to be co‑resident with other services or](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=28.34) [containers inside of the cluster?](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=30.83) [If that's the case,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=32.81) [we can define that in our code that describes our](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=33.89) [application, and Kubernetes can manage that for us.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=37.15) [Now, speaking of Kubernetes managing things for us,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=40.54) [Kubernetes also provides an infrastructure abstraction. As a developer,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=43.08) [if I need to deploy an application into production,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=47.69) [I really don't want to have to care about which server it](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=50.49) [lives on or have to go configure a load balancer to send](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=53.24) [traffic to my application.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=55.97) [That's all going to be handled for me under the hood by Kubernetes.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=57.61) [One of the other core ideas behind Kubernetes is this concept](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=62.04) [called desired state. We can define what our applications or](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=65.17) [our services look like in code,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=68.81) [and it's the job of Kubernetes to make sure that our](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=70.73) [system meets that defined desired state.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=73.64) [And so perhaps our application or our system is composed of some collection](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=76.41) [of web application containers and database containers,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=80.68) [maybe some middleware or even a caching tier.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=83.79) [We can write the code that describes what our system looks](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=86.34) [like, hand it off to Kubernetes for deployment,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=89.43) [and it's Kubernetes job to make that happen, to make sure that](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=91.88) [our system is in that defined desired state.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=95.39) [Let's look at some of the key benefits of using Kubernetes,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=100.04) [and first off is speed of deployment.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=102.79) [Kubernetes gives us the ability to deploy container‑based applications very,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=105.64) [very quickly. And what this enables us actually to do is to get code](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=110.01) [from a developer's workstation into production fast,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=114.04) [and that gives us the ability to absorb change quickly.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=117.09) [The speed of deployment really allows you to iterate](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=120.44) [quickly and get new versions of code out,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=123.08) [enabling new key capabilities for your organization's applications.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=125.5) [Next up is Kubernetes' ability to recover very quickly. When we define our](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=130.21) [system and code and we define that desired state, so perhaps a collection](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=135.21) [of web app containers, and something causes our system to no longer be in](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=138.74) [that desired state, so perhaps a container crashed or a server failed,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=142.84) [well, it's Kubernetes' job to ensure that our application comes back to](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=146.53) [that defined desired state by deploying new containers supporting our](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=151.14) [applications,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=155.39) [making sure that we have a collection of web app containers up](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=156.28) [or really whatever resource it is that defines our application](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=160.14) [or our system's desired state.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=163.5) [And then finally,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=165.15) [Kubernetes allows us to hide infrastructure complexity in the cluster.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=166.04) [And so things like storage,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=169.96) [networking configuration, and workload placement are core functions of](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=171.52) [Kubernetes, and developers don't have to worry about these things when](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=175.42) [deploying container‑based applications into Kubernetes.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=179.16) [So , and now that we know what Kubernetes is and the benefits of Kubernetes,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=184.54) [let's look at some of the basic operating principles behind Kubernetes. And](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=188.29) [first up is desired state or declarative configuration.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=191.96) [This is where we define our application, or really,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=195.84) [our deployment's state and code.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=198.58) [We define what we want to deploy, and Kubernetes makes that happen for us.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=200.84) [It'll go and pull the specified container images and starts them up](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=204.65) [and possibly even allocates load balancers and public IPs if that's](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=208.25) [what's defined in our deployment's code. We write the code](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=211.59) [describing the deployment. Kubernetes does the work to bring it](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=215.18) [online in the desired state.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=218.09) [Next, controllers or control loops have the responsibility of](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=220.31) [constantly monitoring the running state of the system to make sure](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=223.79) [that the system is in that desired state.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=227.2) [And if it's not,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=229.84) [a controller will try to bring the system into the desired state.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=230.66) [And so, for example,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=234.04) [if we've defined that we want three web app containers online,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=235.35) [it's Kubernetes job, more specifically,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=238.9) [a controller's job, to ensure that three web app containers are](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=241.23) [online. A controller will start the three replicas up, and later,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=244.5) [if one of those fails or goes offline,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=247.92) [a controller will create a new web app container,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=250.14) [replacing the one that failed.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=252.63) [Now there are many different types of controllers](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=254.54) [available in Kubernetes for various scenarios,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=256.59) [and we'll cover them throughout this series of courses.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=259.22) [But the key concept here is controllers are what make changes](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=261.68) [to the system to ensure the desired state.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=264.66) [Another core principle is the Kubernetes API.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=267.54) [The Kubernetes API provides a collection of objects that we can use to build](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=270.74) [and define the systems that we want to deploy in code.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=275.31) [The objects defined in our code define the desired state of our](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=279.14) [applications or the systems that we want to deploy in Kubernetes.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=282.72) [The Kubernetes' API is implemented and available via the API Server.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=285.78) [The API Server is the central communication hub for information in a](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=292.14) [Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=296.28) [This is where we, as administrators and developers,](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=297.84) [interact with Kubernetes to deploy and manage workloads.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=300.98) [And that's also where the components of a Kubernetes cluster interact with each](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=304.44) [other to understand the current state of the system and to make changes to that state, if needed to ensure the desired state.](https://app.pluralsight.com/course-player?clipId=a3180fc3-2702-4f75-b3e5-730f9ae447da&startTime=308.39)

[Introducing the Kubernetes API - Objects and API Server](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b)

[So let's look more closely at the Kubernetes API.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=1.04) [Behind the scenes, you'll find they're a collection of API objects.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=4.33) [These objects are primitives that represent the state of the system,](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=7.92) [so things like Pods,](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=12.19) [which represent a deployed container, or nodes, which](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=13.18) [represent servers that do work in your system.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=16.11) [These API objects allow us to configure the state of the system,](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=19.03) [and we could do that, both declaratively and imperatively,](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=23.64) [where declaratively, we learned, means we can describe the implementation](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=27.05) [that we want to have, or we can describe that deployment and how we want](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=31.78) [our system to look. Imperatively means we sat down at the command line and](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=34.81) [executed a sequence of commands to get the system to be in the state that](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=38.26) [we want it to be in.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=41.96) [So as we continue along learning about the API server,](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=43.44) [the API server is a RESTful API that runs over HTTP or HTTPS using](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=46.54) [JSON. And this is going to be the sole way that we as administrators](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=51.25) [interact with our cluster, but it's also the only way that Kubernetes](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=55.96) [interacts with the cluster as well.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=60.69) [There are other facets of the system that will be exchanging information, and](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=62.47) [all of that has to go through the API server. Now when we're communicating to](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=65.63) [the API server and we're telling it to create things or to change objects into](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=70.36) [different configuration states, that information is serialized and then](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=75.14) [persisted into the cluster data store.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=78.72) [Now let's look at some of those core primitives, the Kubernetes API objects.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=82.64) [These are going to be the building blocks for our deployments](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=86.5) [inside of Kubernetes. And first up's Pods.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=89.48) [What Pods are, are a single or a collection of](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=92.02) [containers that we deploy as a single unit.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=95.22) [This essentially is our container‑based application.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=97.96) [Then we have controllers.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=100.84) [These are the things that keep our system in the desired state. So things](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=101.95) [like ReplicaSets and deployments are going to become core tools in our](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=105.63) [toolbox as we build deployments in Kubernetes.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=109.86) [Now,](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=112.99) [services provide a persistent access point to the applications that we deploy](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=113.41) [in Pods because, as things change under the hood and our Pods get redeployed](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=117.82) [perhaps by our controllers as they come up and down,](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=122.63) [those things will be constantly changing.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=125.24) [Well,](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=126.78) [it's the service's responsibility to provide a persistent access point to](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=127.06) [the applications provided by our Pods. and finally,](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=131.45) [there's storage.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=134.58) [Of course, we'll need to be able to store data somewhere,](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=135.17) [and Kubernetes gives us some storage objects so we can have persistent](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=137.27) [storage in our applications. Now this certainly is not an exhaustive list](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=141.77) [of all the API objects available in Kubernetes,](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=146.44) [but these are the key players, the core things that we'll build our deployments from.](https://app.pluralsight.com/course-player?clipId=948dd9a6-3c6a-440d-9fe6-b19671f28d3b&startTime=148.99)

[Understanding API Objects - Pods](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881)

[So let's look a little closer at each of these core](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=1.04) [API objects that we just introduced,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=3.67) [and we're going to start the conversation with Pods.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=5.39) [What a Pod is is a construct that represents one or more](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=7.3) [containers inside of a Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=11.45) [What it really is is those container‑based applications or](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=13.92) [services that you need to deploy into your Kubernetes cluster](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=16.85) [or your production environment.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=20.05) [From a Kubernetes standpoint, it's the most basic unit of work,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=21.54) [in fact, it's the unit of scheduling.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=24.8) [When we define a Pod,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=26.75) [we also can define some of the resources that it requires inside](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=28.31) [of the manifest that describes our deployment.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=31.94) [It's up to Kubernetes to ensure that those resources are available,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=35.04) [and schedules that Pods onto those resources in our cluster,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=37.99) [and brings our application up and running.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=40.98) [Another key attribute of Pods is that they're ephemeral.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=43.74) [What this means is that no Pod is ever redeployed.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=46.16) [If I deploy an application as a Pod based off of a container today,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=49.4) [and that Pod dies,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=53.1) [if I go ahead and redeploy that Pod again based off](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=54.28) [of that same container image,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=57.13) [no state is maintained between those two deployments.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=58.58) [Once that Pod goes away, it never ever comes back.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=61.4) [That new Pod is going to be deployed and provide those services of that](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=63.92) [container‑based application in that Pod deployment.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=67.68) [Another key attribute of Pods is the atomicity of a Pod,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=70.32) [they're either there or they're not.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=72.84) [Now, in a single‑container Pod, this kind of makes sense,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=73.37) [like a Pod's up and running, if the container dies,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=77.6) [well, yeah, that Pod's no longer available.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=79.66) [In a multi‑container Pod,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=81.61) [if I have more than one container deployed inside of a Pod,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=83.77) [if one of those containers dies, the entire Pod becomes unavailable,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=86.94) [and that Pod is no longer providing the services that it should.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=90.57) [Continuing the conversation about Pods,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=94.72) [what I want to introduce now is that it's Kubernetes' job to keep](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=96.67) [your Pods up and running using controllers,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=100.09) [but more specifically,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=102.41) [it's about keeping your application or your deployments in the desired state.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=103.8) [And so to do that, to make sure that things are in the desired state,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=107.81) [Kubernetes tracks the state of a Pod.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=111.08) [Is our Pod and the containers inside of that Pod up and running?](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=113.29) [Now of course our Pod could be up and running,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=117.13) [but the applications inside of that Pod could be throwing errors,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=119.55) [so Kubernetes also tracks the health of a Pod.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=122.55) [Is the application inside the Pod up and running?](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=125.44) [And so Kubernetes can check the health of an application](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=128.39) [running inside of a Pod with probes.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=130.97) [In our deployment code,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=133.1) [we can define a probe to check the health of our applications.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=134.37) [And so, for example, if we've deployed a web application,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=137.69) [we could write a probe that checks the URL and tests to see if the](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=140.75) [application is responding in an appropriate way.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=144.14) [And so if that probe fails,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=147.24) [then Kubernetes could understand that and make the](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=148.79) [appropriate adjustments to that individual Pod that might](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=151.5) [not be responding in a healthy way,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=154.72) [and this could be deleting the Pod and replacing it with a new one.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=156.37) [We're going to look much more closely at Pod internals](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=160) [and probes in an upcoming course, but for now,](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=162.61) [what I want you to understand is what they are and the role that](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=164.98) [they play inside of deploying applications in Kubernetes. We're going to look at them very closely later on.](https://app.pluralsight.com/course-player?clipId=68d4e70d-f9b7-4da5-84dd-1bfced5e2881&startTime=167.53)

[Understanding API Objects - Controllers](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc)

[So now that we know what Pods are and that they can come](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=0.94) [and go based on status and health, how does Kubernetes manage my Pod state?](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=3.54) [What's the thing that makes sure that our Pod is up, running, and healthy?](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=8.36) [Well, that's where controllers come in.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=12.77) [A controller defines the desired state for your cluster](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=14.83) [and the applications deployed in it.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=18.24) [It's the job of a controller to ensure that things stay in the desired state.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=20.44) [Controllers are exposed to you as workload resource API objects](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=24.72) [and are most commonly used to create and manage Pod‑based](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=28.92) [applications deployed in Kubernetes.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=31.96) [These API objects will create and configure the controllers](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=34.64) [needed to create and manage your Pods for you.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=38.44) [And they ensure your applications stay in the desired state.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=40.7) [So,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=44.64) [for controllers to define and manage the desired state](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=45.02) [of applications deployed as Pods,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=48.03) [controllers monitor and respond to the state and health of Pods deployed,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=49.97) [ensuring the desired number of Pods are up,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=55.38) [running, and healthy in a cluster.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=58.09) [And if the state or the health changes, the controller will respond accordingly,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=60.16) [trying to make sure that the system stays in the desired state.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=64.14) [The first controller that we're going to look at today is the ReplicaSet.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=67.4) [You've heard me describe this scenario a couple of times so far](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=71.04) [in the course where I want to have a collection of web](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=73.69) [application containers up and running in Pods.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=76.47) [Well, that's what a replica set models.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=79.54) [It allows us to define the number of replicas for a particular Pod](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=81.88) [that we want to have up and running at all times.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=86.02) [And so if I want to have three Pods of this](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=88.64) [particular web app running at all times,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=91.43) [well,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=93.6) [it's the replica set's job to make sure that those three](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=93.97) [Pods are up and running at all times.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=96.97) [If one of those Pods becomes unavailable or unhealthy,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=99.54) [for whatever reason,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=102.73) [it's Kubernetes' job to delete that failed Pod and deploy a](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=103.76) [new Pod with the hopes of that new Pod returning the system](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=107.6) [back to the desired healthy state.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=110.98) [When it comes to deploying applications in Kubernetes,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=113.74) [you generally won't create replica sets directly,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=116.32) [you will create deployments.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=119.69) [On the creation of a deployment,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=121.64) [it creates a replica set based on what's defined in the deployment.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=123.39) [And the deployment manages the state of the replica set,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=128.48) [so things like which container image to run and](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=132.42) [also the number of Pods to create.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=134.7) [But more interestingly,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=136.75) [the deployment controller manages the transition between two](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=137.86) [replica sets, and a practical example of this is moving](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=141.73) [between two versions of an application.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=145.18) [When we define our initial deployment, it creates a replica set,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=147.64) [let's say running version 1.1 of a container image for our application.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=150.47) [We can use a deployment to control the transition to a new collection of](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=155.44) [Pods, perhaps running version 1.2 of our application.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=159.71) [A deployment controller controls the transition and even](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=163.74) [gives us the ability to roll back if needed.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=167.17) [Deployments are core to the success of deploying applications in Kubernetes,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=169.69) [and we'll look at them in more detail in this course and in](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=174.2) [much more detail in upcoming courses.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=176.91) [Now, there are many more types of controllers available in Kubernetes,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=179.34) [not just ones based on Pods.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=182.63) [In future courses, as we continue along in our Kubernetes journey,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=184.44) [we'll look at things like nodes, services,](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=188.19) [and other controllers that provide value to our Kubernetes applications and infrastructure.](https://app.pluralsight.com/course-player?clipId=3134d6bb-5bc5-44a0-a0ec-fb4db02960bc&startTime=190.33)

[Understanding API Objects - Services](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378)

[So we just learned that Kubernetes controllers can start and stop Pods based](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=0.94) [on the health of the Pod and changes in application state.](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=5.82) [Well,](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=9.24) [what we need to know now is, how does Kubernetes add some](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=9.6) [persistency to all of that ephemerality?](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=12.97) [Well,](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=14.86) [that's where services come in. Services add](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=15.31) [persistency to the ephemerality of Pods.](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=19.21) [What a service is is the network abstraction for access to the](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=21.81) [services that Pods actually provide. And so what Kubernetes does for](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=24.9) [us is it persistently allocates an IP and DNS name for the application](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=28.63) [services that are provided by the collection of Pods that we want to](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=33.11) [front‑end with a service.](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=36.42) [And so, as Pods come and go based on their desired state, under the hood,](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=38.24) [Kubernetes is going to dynamically update the service with](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=42.19) [new information for those particular Pods.](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=45.07) [And so our users or other applications will simply access that](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=47.74) [front‑end IP address or DNS name, and Kubernetes will maintain the](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=51.46) [plumbing, or the infrastructure underneath, as Pods come and go based](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=55.45) [on their lifecycle and desired state.](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=58.82) [And so, as users or applications access that persistent IP,](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=60.94) [what Kubernetes will do is update the routing information to make sure that](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=64.49) [traffic comes in on that persistent IP and is routed directly to the Pods that](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=68.23) [are and healthy supporting that particular services.](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=72.67) [We can also leverage services to scale our application by adding and](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=75.94) [removing Pods based on the demands of that application, and services also](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=79.23) [provide a load balancing to distribute application load across the Pods providing that application services.](https://app.pluralsight.com/course-player?clipId=11d84091-668b-48f7-8302-2673a9d6f378&startTime=83.67)

[Understanding API Objects - Storage](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5)

[So, the next question you might have in your head is what about my data?](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=1.04) [Where does Kubernetes store data for persistent storage in the cluster?](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=3.93) [And so let's talk about some storage constructs that](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=8.59) [are available inside of Kubernetes.](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=10.66) [Initially, Kubernetes had the concept of a volume,](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=12.99) [which was storage backed by physical media that was](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=15.14) [directly accessible to a pod.](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=17.49) [And so as we deployed pods,](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=20.04) [we defined that we wanted this volume of this type of storage, and those](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=21.87) [two things were tightly coupled together in a deployment.](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=26.16) [And so we lost some flexibility in how we could administer](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=28.89) [pods and how we could administer storage.](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=31.23) [And so, after that, Kubernetes came up with the concept of a persistent volume.](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=33.54) [What a persistent volume is is pod independent storage that's](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=38.3) [defined by the administrator at the cluster level.](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=41.18) [And so when a pod wants access to that storage,](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=44.68) [it defines what's called a persistent volume claim.](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=47.38) [So, in the pod definition, we basically say I want 10 GB of this type of storage,](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=49.92) [and the pod claims that storage from the underlying](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=55.36) [cluster. This technique effectively decouples the pod from the storage inside of the cluster.](https://app.pluralsight.com/course-player?clipId=802ed8de-57de-421b-bcd5-39a8211b23f5&startTime=58.42)

[Kubernetes Cluster Components Overview and Control Plane](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9)

[So now that we've introduced the basic principles of Kubernetes,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=1.04) [let's move on into exploring the Kubernetes cluster architecture and](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=4.35) [look at what a cluster is actually composed of.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=8.59) [And so let's look at cluster components.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=11.54) [The first cluster component is the control plane node.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=14.24) [The control plane node implements the major control functions of a cluster.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=17.41) [It coordinates cluster operations, monitoring,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=22.14) [Pod scheduling, and is the primary access point for cluster administration.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=25.14) [Next up is the Node, sometimes called the worker Node.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=29.64) [This is where our application pods actually run.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=33.24) [The Node has the responsibility of starting up pods and](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=36.44) [the containers supporting those pods.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=39.54) [Nodes implement networking,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=42.24) [ensuring the network reachability to the pods and](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=43.78) [services running on worker nodes.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=46.48) [Each node in the cluster contributes to the compute capacity of the cluster,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=49.04) [and clusters are generally composed of multiple nodes based on the](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=52.88) [scalability requirements of the workloads deployed.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=56.83) [And finally, nodes can be either virtual or physical machines.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=59.59) [The control plane node used to be called the master node,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=65.74) [and so in some documentation and resources on the web,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=68.54) [you will see the term master node used.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=71.48) [The modern name for this is control plane node.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=73.54) [I'm in the process of updating this series of courses,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=76.26) [and we'll update this to control plane node in each course,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=78.78) [updating the videos, code samples, and demos.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=82.04) [So if you get to a course that I haven't updated yet,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=84.51) [you will see this referred to as the master node.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=87.34) [This is conceptually the same as the control plane node,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=89.84) [and it will be updated as soon as I can get to that course.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=92.77) [Zooming in on the cluster's control plane node.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=96.14) [The control plane node,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=100.14) [which implements the major control functions of a cluster,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=101.38) [is comprised of several components, and the first is the API server.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=104.44) [The API server is the primary access point for](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=108.55) [cluster and administrative operations.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=111.44) [This is essentially the communication hub of our Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=113.49) [Now, the API server itself is stateless,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=117.54) [and so we need a place to be able to store the state of the system,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=120.23) [and that's where the cluster store etcd comes in.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=123.44) [This has the responsibility of persisting the state of our Kubernetes objects.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=126.74) [Next is the scheduler.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=131.74) [The scheduler tells Kubernetes which nodes to start pods on](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=132.83) [based on the pod's resource requirements and other attributes](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=136.57) [such as administrative policies.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=139.46) [And then there's the controller manager.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=141.29) [The controller manager has the job of implementing the lifecycle](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=143.82) [functions of the controllers that execute and monitor the state](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=146.89) [of Kubernetes objects such as pods.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=151) [Basically, it has the responsibility of keeping things in the desired state.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=153.64) [And we're also going to introduce kubectl here,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=157.62) [not really part of the control plane,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=160.28) [but it's how we're going to interact with the API](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=162) [server for administrative functions.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=164.28) [When we work with kubectl,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=166.44) [it's going to interact with the API server for us to be able to retrieve](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=167.88) [information and also to make those changing operations to help get our](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=171.47) [system into the state that we want it to be in,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=175.27) [such as deploying workloads.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=177.53) [It's our primary administrative command line tool for](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=179.54) [operating our Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=182.41) [And now we're going to deal with this right now.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=184.28) [I call it kubectl.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=186.22) [You can call it kube control, kube cuddle,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=187.66) [whatever you want to do, but for this course,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=190.05) [we're going to go with kubectl.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=191.57) [Now, within the control plane node,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=193.76) [let's look even closer at those individual control plane](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=196.66) [components that we just introduced.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=199.82) [We're going to look at the API server, the cluster store etcd,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=201.6) [the scheduler, and the controller manager.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=204.86) [Now, the API server is central to the control of your cluster.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=207.24) [It's core to all operations.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=211.19) [All configuration changes pass through the API server.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=213.54) [It's a very simple interface leveraging a RESTful API,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=216.31) [exposing RESTful operations like GET, PUSH,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=219.84) [POST, and so on.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=222.82) [Based on the operation that's going into the API server,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=224.14) [it's the responsibility of the API server to validate that operation and](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=226.78) [persist that object state into the Cluster data store,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=231.17) [which is etcd, and as we've introduced,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=234.59) [etcd is the cluster data store and is used to persist the](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=237.75) [state of those API objects as key value pairs.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=240.66) [Next up is the scheduler.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=245.14) [It's the job of the scheduler to watch the API server for unscheduled pods,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=246.68) [and then schedule those pods on nodes.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=251.85) [When scheduling pods,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=254.34) [the scheduler will evaluate the resources required by](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=255.49) [that Pod in terms of things like CPU, memory,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=258.43) [and storage requirements to ensure their availability when](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=261.64) [placing a Pod on a specific node in our cluster.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=264.95) [Further,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=267.88) [the scheduler has the responsibility of respecting any](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=268.47) [constraints that we defined administratively,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=271.3) [and so perhaps you want to keep two pods on the same node at all times.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=273.84) [That's called Pod affinity.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=277.29) [Or perhaps we want to do the opposite,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=278.86) [where we want to ensure that two pods are never on the same node.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=280.5) [That's called Pod anti‑infinity, and the Scheduler will handle that for us.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=283.12) [The Scheduler is capable of scheduling pods based on](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=287.44) [many different other attributes,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=290.24) [and we'll look at this more closely in an upcoming course.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=291.65) [And then finally, the controller manager.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=294.04) [It's the job of the controller manager to execute those controller loops.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=296.37) [Those controller loops implement the lifecycle functions of pods](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=300.57) [,and thus defined the desired state of the system.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=303.95) [It's the job of the controller manager to watch the current](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=306.46) [state of the system and update the API server to ensure that](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=309.51) [its heading towards the desired state.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=312.56) [The replica set API object is an example of a controller that we](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=314.64) [will be working with a bunch in this course.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=318.57) [This is the controller that is used to ensure that the correct number of](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=320.74) [pods or replicas are up and running in our deployments.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=324.7) [And so it's the responsibility of the replica set controller to](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=328.14) [communicate to the API server changes in its desired state.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=331.45) [And so this could be operations like creating new pods or deleting pods,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=335.3) [based on what the running state of the cluster is.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=338.94) [And as we've introduced,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=342.24) [there are many different other controllers available in Kubernetes,](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=343.26) [and we'll introduce more in this course and in upcoming courses as we continue our Kubernetes studies.](https://app.pluralsight.com/course-player?clipId=39344605-5400-4a92-8e6e-0667d8cbf7b9&startTime=346.19)

[Nodes](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9)

[Now that we've introduced the control plane node and its core components,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=1.04) [let's take some time to look at what a node does in our cluster.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=4.14) [A node is where our application Pods run.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=7.02) [The node starts up the Pods and ensures that the](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=9.71) [containers in those Pods are up and running.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=12.46) [Nodes also implement networking to ensure the reachability of](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=15.44) [the Pods running on the nodes in our cluster.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=18.67) [We could have many nodes in our cluster based on the](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=21.44) [scalability of requirements of the applications that we're](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=23.97) [deploying inside of Kubernetes,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=26.22) [and nodes can consist of either physical or virtual machines.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=27.74) [Now, within a node, there are some components that I want to introduce you to,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=31.35) [and first up is the kubelet.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=34.84) [The kubelet has the responsibility for starting up Pods on the node,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=36.63) [then there's the kube‑proxy,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=39.98) [which has the responsibility for Pod networking and implementing](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=41.4) [our services abstraction on the node itself,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=44.58) [and then there's the container runtime.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=46.97) [This is the actual runtime environment for our containers.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=49.18) [The container runtime has the responsibility of pulling the container](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=52.31) [image from our container registry and providing an execution environment](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=55.47) [for that container image and the Pod abstraction.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=59.84) [Now,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=62.74) [both the kubelet and the kube‑proxy communicate directly with the API server,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=62.91) [monitoring it for changes to the environment.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=67.71) [And so as Pods are scheduled to this individual node,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=69.67) [the kubelet, it will monitor the API server for that information.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=72.74) [And if it does have a scheduled Pod,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=75.87) [it'll start up the containers needed to support that Pod on the node.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=77.86) [Similarly, if there's a networking topology change that needs to be implemented,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=81.92) [such as adding a newly‑created Pod's IP to a service for load balancing,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=85.55) [it's the responsibility of the kube‑proxy to monitor the API](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=89.74) [server and make that modification on the node.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=93.08) [Let's look at those individual components of a node in more detail.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=97.64) [We're going to look at the kubelet, the kube‑proxy, and the container runtime.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=101.74) [Now,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=105.54) [I do want to call out that all of these components actually](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=105.81) [will run on all of the nodes in the cluster.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=108.5) [What this means is that even on the control plane node,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=111.14) [these three services, the kubelet, the kube‑proxy,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=113.95) [and the container runtime,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=116.56) [will run because there are going to be special purpose Pods deployed onto the](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=117.61) [control plane node providing the control plane services.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=122.31) [There's also going to be a kube‑proxy providing networking services,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=125.65) [and a container runtime as these execution environment for](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=128.67) [those Pods on the control plane node,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=132.27) [and so all of these services will exist on all nodes in our cluster.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=134.47) [Looking more closely at the kubelet first,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=138.46) [as we've discussed, it monitors to API server for changes.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=140.94) [And so, as Pods are scheduled onto individual nodes,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=144.3) [it's the kubelet that goes and asks the API server,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=147.94) [hey, do you have any work for me?](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=151.44) [And the API server answers that question, yes or no.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=153.26) [The kubelet is also responsible for Pod lifecycle,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=156.34) [meaning starting and stopping Pods and its containers in](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=159.22) [reaction to those modifications or those state changes that](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=163.01) [are being watched on the API server.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=166.1) [In addition to monitoring the API server for changes,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=168.44) [it's also the responsibility of the kubelet to](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=171.3) [report on both node and Pod state.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=173.35) [So if a node is reachable and reports its status as ready,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=176.34) [then it's available for scheduling of new Pods in our cluster.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=179.44) [The kubelets also have the responsibility of monitoring Pod state.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=182.28) [Is the Pod up and running?](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=186) [That's all reported back to the API server via the kubelet.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=188.04) [If there is a probe used for determining Pod health,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=190.96) [it's the responsibility of the kubelet to execute that probe.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=193.37) [Now,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=196.64) [the kube‑proxy has the responsibility of all networking components for nodes,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=196.85) [and it's most commonly implemented in iptables.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=201.04) [There are other modes for kube‑proxy, but for this course,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=204.04) [we're going to focus on iptables.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=206.55) [Kube‑proxy has the responsibility of implementing that services](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=208.44) [abstraction that we introduced a little bit ago,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=211.74) [and the kube‑proxy is also responsible for routing traffic to Pods.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=213.89) [And so as application requests come into the cluster,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=218.08) [kube‑proxy has the responsibility of ensuring that those requests land on](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=220.85) [the correct Pods executing on the nodes in our cluster. And the kube‑proxy](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=225.72) [also has the responsibility of load balancing,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=229.8) [making sure that traffic that's coming into multiple](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=232.15) [Pods is distributed in an even fashion.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=234.52) [The container runtime has the responsibility of downloading](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=237.44) [container images and starting and running containers.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=240.39) [The container runtime in Kubernetes is wrapped up in what's](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=243.41) [called the Container Runtime Interface, or CRI,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=245.79) [and this gives us the luxury of being able to swap out the container runtime,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=248.54) [choosing from several different supported container runtimes. Out of the box,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=251.8) [we're going to be using containerd.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=255.44) [This is the default container runtime In today's version of Kubernetes.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=257.03) [There are many other container runtimes available to you, and you can use](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=260.46) [those in Kubernetes as long as they're CRI, or Container Runtime](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=264.56) [Interface, compliant. Before Kubernetes version 1.20, Docker was the](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=268.03) [container runtime used. In version 1.20, Docker was deprecated and will be](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=273.19) [removed in 1.22, or later.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=278.34) [In practice, though,](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=280.74) [you can still use container images built with Docker in your Kubernetes cluster](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=281.82) [when using other container runtime as long as they're CRI compliant. In our cluster in this course, we'll be using containerd.](https://app.pluralsight.com/course-player?clipId=560c9b84-b139-40fa-8c77-c0d2a7632da9&startTime=286.77)

[Cluster Add-on Pods](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695)

[Cluster add‑on Pods are Pods that provide special](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=1.04) [services to the cluster itself, and the primary example of this is DNS.](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=3.92) [These special‑purpose Pods for DNS provide DNS services inside](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=9.54) [the cluster using the CoreDNS DNS server.](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=13.88) [The IPs for the Kubernetes service front ending these DNS server Pods and](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=17.72) [the search suffix of the domain is placed into the networking configuration](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=22.38) [of any Pods created by the cluster's API server,](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=26.23) [and so Pods and nodes and services will register their](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=29.48) [addresses with the DNS server when they're created.](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=32.81) [And since this is the DNS server used inside of the cluster for its services,](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=35.74) [it's commonly used for service discovery for](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=40.39) [applications deployed inside of a cluster.](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=42.84) [You'll find DNS Pods deployed in nearly every Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=45.4) [Other cluster add‑ons include ingress controllers,](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=49.94) [which are essentially advanced HTTP or Layer 7 load](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=52.37) [balancers and content routers.](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=55.89) [Also,](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=57.5) [another example of an add‑on Pod is the Kubernetes dashboard for](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=58.05) [web‑based administration of your Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=61.81) [Ingress and dashboard Pods are optional,](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=64.47) [and can be easily added on to any cluster if needed.](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=66.9) [Network overlays are also classified as cluster add‑on Pods. We're going to talk about those in more detail in the next module.](https://app.pluralsight.com/course-player?clipId=e73685d0-189c-4207-88fd-f9550289a695&startTime=70.44)

[Pod Operations](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07)

[So,](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=1.14) [now that we know the basic principles behind Kubernetes in](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=1.55) [terms of theory and we looked at the components that a](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=4.02) [Kubernetes cluster is made of,](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=6.81) [let's see Kubernetes in action and see how this all pieces together.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=8.44) [And we're going to start off with some pod operations.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=11.51) [Let's say we have a cluster and it's composed of a control](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=13.89) [plane node and a couple of worker nodes.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=16.93) [Using kubectl, we submit code to instruct Kubernetes to create a deployment.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=19.44) [And in that deployment, we've defined that we want three replicas of our pod.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=24.09) [That request is going to be submitted to the API server,](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=27.64) [and it's the responsibility of the API server to store](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=30.16) [that information persistently in etcd.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=32.86) [With that then,](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=35.94) [it's the responsibility of the controller manager to spin up those](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=36.72) [three requested replicas in that replica set.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=40.36) [And so that's going to create those three pods.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=43.36) [That request is then going to be submitted to the scheduler. The scheduler is](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=46.34) [then going to tell the API server that these pods need to be scheduled on](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=49.88) [nodes that it selects, and that scheduling information is persisted back to](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=53.75) [etcd. Which nodes did those pods get scheduled on? Well that's dependent upon](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=57.45) [the resources requested in the pods and the resources available in the nodes](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=61.68) [in the cluster..](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=66.19) [The kubelets on the nodes will ask the API server, hey,](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=67.54) [do you have any work for me, and then it's going to start spinning up](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=70.46) [those pods that were requested by the replica set.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=73.27) [Now everything is going along happy, happy,](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=75.95) [and the controller manager is monitoring the state of the replicas. And](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=78.11) [everyone's reporting that they're in a desired state.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=81.37) [But what happens if a node goes down?](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=83.94) [Well, that node is no longer reporting state,](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=85.86) [and the controller manager understands that we're now outside](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=88.7) [of the desired state in the terms of the number of replicas](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=91.79) [required for our replica set.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=94.85) [So the controller manager is going to submit for a new](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=97.44) [pod to be created and scheduled,](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=100.09) [and it's the responsibility of the scheduler to find a node to put that pod on.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=102.04) [And so it schedules that new pod to be deployed on a node](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=105.94) [to bring our replica set back into compliance with the](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=109.19) [desired state of three replicas.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=112.07) [Now,](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=114.24) [you might be wondering by looking at this picture, why didn't the](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=114.71) [scheduler schedule that pod onto the control plane node to maybe](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=117.19) [even out the workload a little bit between the control plane node](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=120.78) [and the worker nodes? Well, in the default configuration of Kubernetes,](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=124.17) [the scheduler taints the control plane node.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=128.28) [And what this means is is that the control plane node is only going to run](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=130.86) [pods that are system pod, and so in this case the API server,](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=134.3) [etcd, the scheduler, and the controller manager. And a user workload is going](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=138.35) [to be scheduled onto regular worker nodes in the cluster.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=142.5) [We can certainly adjust the configuration of the control plane node to](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=146.44) [untaint the control plane node so that user pods can be started up on it,](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=149.68) [but that's usually only good for test labs, not good for production systems.](https://app.pluralsight.com/course-player?clipId=a12438f5-fc25-4604-88b2-c0755f0d8f07&startTime=153.99)

[Service Operations](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c)

[With the basics of pod operations under our belt,](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=1.04) [let's see how we can extend that into services and provide a network](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=3.62) [abstraction or a single access point to our end users into the](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=6.78) [applications running inside of our cluster.](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=10.9) [Zooming out a little bit,](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=13.34) [we're not going to focus on things at the node level right now,](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=14.62) [we're going to look at things from the cluster level. And let's say](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=17.05) [we're running a deployment that creates a replica set which creates](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=20.01) [pods inside of our cluster.](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=23.29) [In this case,](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=25) [we're going to have four web application pods running inside of the cluster.](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=25.72) [We can expose access to these pods with a service. And in this case our](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=30.34) [service is going to be HTTP running on TCP port 80. With the service up and](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=34.01) [running, users in other applications can come along and access are web](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=38.59) [application pods via this fixed, persistent service endpoint,](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=42.52) [which is going to be an IP address or a DNS name.](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=46.63) [And as requests come in,](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=49.74) [they're going to be load balanced to all of the pods supporting that service.](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=51.02) [Now, the value of services,](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=54.66) [aside from simply having a persistent access endpoint into the application when](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=56.31) [combined with the functionality of a replica set, if a pod becomes unresponsive](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=60.67) [or is no longer available it's the responsibility of the replica set controller](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=65.52) [to remove that pod and deploy a new one.](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=70.63) [Now our users knew nothing of this because behind the scenes that](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=73.64) [service is going to stop load balancing requests to the failed,](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=77.07) [deleted pod and will continue load balancing requests across the](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=80.55) [remaining pods. Once that new pod is up and ready, requests will be](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=84.72) [load balanced to it as well. This functionality is core to building self‑healing applications in Kubernetes.](https://app.pluralsight.com/course-player?clipId=71f977d9-37f0-4a98-9039-1f1fc99e650c&startTime=88.72)

[Kubernetes Networking Fundamentals](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f)

[We just introduced services and accessing applications on Pods.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=1.04) [Now is an appropriate time to take a look at some](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=4.8) [Kubernetes networking fundamentals.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=7.5) [In Kubernetes, every Pod deployed will get assigned its own unique address.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=9.3) [And to facilitate for this, there are some networking requirements,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=14.12) [or rules,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=17.2) [that we have to live by when we build our networked topologies](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=18.07) [for supporting our Kubernetes infrastructures.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=20.47) [And the first rule is,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=22.78) [Pods on a node can communicate with all Pods on all nodes in a](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=24.42) [cluster without network address translation,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=28.76) [or NAT.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=31.25) [The next rule is, agents running on a node,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=31.78) [so things like system daemons and the kubelet,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=34.31) [have the ability to communicate with all running Pods on that node.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=36.89) [And so, essentially,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=40.66) [what we need to be able to to do is to design our network](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=42.07) [infrastructures such that all Pods on all nodes in a cluster have](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=44.69) [reachability to each other with the real IP addresses that are on](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=49.72) [the nodes and the Pods themselves.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=53.3) [And we'll look at an implementation of this in an upcoming](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=54.88) [module when we build a cluster together.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=57.61) [So let's look more closely at the networking fundamentals,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=61.54) [or how really Pods and nodes can communicate to each](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=63.85) [other in some various scenarios.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=67.08) [And so let's say we have a cluster up and running,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=68.76) [and we deploy a multi‑container Pod inside of our cluster.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=70.35) [Well, those two containers inside that Pod,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=74.19) [if they need to communicate to each other,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=76.63) [they're going to do that over localhost using namespaces.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=78.05) [Now, let's say we deploy some additional Pods.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=81.14) [Those Pods aren't self‑contained, so they can't communicate over localhost.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=83.14) [So they'll communicate to each other over a layer 2 software bridge on the](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=86.49) [node using the real IP addresses of the Pods themselves,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=90.49) [and so that's how they can communicate to each other.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=93.68) [Now, let's say we have to reach out onto a Pod onto another node.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=96.1) [Let's say the Pods in our first node there at the top need](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=99.8) [to communicate to that Pod on the bottom.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=102.32) [Well, that's going to happen on the real IP address of that Pod,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=104.34) [so we'll need to facilitate for layer 2 or layer 3 connectivity between the Pods](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=107.64) [on the top node there and the Pod on the bottom node there.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=112.09) [And so that's going to be really dependent upon our network infrastructure,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=114.97) [so we might have to work with our network engineering team to](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=118.56) [ensure that we have the ability to have layer 2 or layer 3](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=121.15) [reachability between the Pods on these nodes.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=124.03) [Another common scenario is,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=126.7) [if you don't control the underlying network infrastructure is](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=128.09) [to deploy what's called an overlay network,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=131.07) [and that overlay network gives the ability of all of these things to](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=132.78) [seem like they're on the same layer 3 network and communicating](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=136.24) [together on an individual Pod overlay network.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=139.79) [And the final case I want to look at with you today is external services.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=143.04) [Let's say we have some service in our cluster,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=145.9) [and we want to expose that to the world.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=147.94) [Well, we learned that that's going to be implemented on the kube‑proxy,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=149.91) [and we stand up that HTTP service, in this case,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=153.15) [and we want to expose that to the world.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=155.48) [And so that's how we can get external users into our applications on those](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=157.34) [services, and it's a responsibility for the service to then communicate to](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=161.83) [the individual Pods that are front‑ends.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=164.95) [So these are the four core networking scenarios that I wanted to introduce](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=167.34) [to you today, within a Pod, between Pods on the same node, between Pods on](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=171.22) [different nodes, and external services accessing things running on Pods](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=175.29) [inside of your Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=179.89) [Now, as you might expect, this is only scratching the surface,](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=181.5) [and we are going to have a dedicated course just to networking coming up in this Kubernetes series.](https://app.pluralsight.com/course-player?clipId=d37d5b82-a8ad-4120-896f-1652639fdf1f&startTime=184.68)

[Module Summary and What's Next!](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28)

[Alright, so here we are at the end of the module. We covered a lot.](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=1.04) [I hope I got you started on your journey of learning](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=4.18) [Kubernetes well. We learned what is Kubernetes and the](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=6.64) [basic principles behind its operations.](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=9.76) [We explored the Kubernetes architecture.](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=11.94) [We learned a lot about the individual cluster components and](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=14.27) [how they interoperate to provide those services that](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=17.18) [Kubernetes provides to us as a platform.](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=19.91) [And then we looked at some basic networking fundamentals so we can learn how](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=22.35) [information moves between resources inside of Kubernetes.](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=25.44) [Well, I think we're off to a good start, and I hope you do too.](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=30.24) [Now,](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=32.84) [please join me in the next module where we're going to install and configure our first Kubernetes cluster together.](https://app.pluralsight.com/course-player?clipId=29be37c4-4b7d-4615-b8cb-7fd70916da28&startTime=33.08)

[Installing and Configuring Kubernetes](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3)

[Module Overview](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3)

[Hello, this is Anthony Nocentino, enterprise architect with Centino Systems.](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=1.34) [Welcome to my course, Kubernetes Installation and Configuration Fundamentals.](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=4.95) [This module is Installing and Configuring Kubernetes.](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=9.51) [So far, we've looked at exploring the Kubernetes architecture,](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=13.54) [getting those basic principles underneath our belt.](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=16.48) [Now it's time to roll up our sleeves and learn how to install and configure](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=19.02) [our first Kubernetes cluster. First up, in this module,](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=22.61) [we're going to start off with some installation considerations.](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=26.34) [Basically, what do you need to know and do before you install Kubernetes?](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=29) [Then we're going to look at the installation overview.](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=34.54) [We're going to discuss the process of how we](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=36.7) [actually install a Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=38.67) [Then we'll look at where we can get Kubernetes from because,](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=40.46) [well, we need software to be able to install.](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=43.59) [Once we have all of that, we'll sit down and we will learn how to install](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=45.75) [our first Kubernetes cluster together with kubeadm.](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=49.25) [And then we'll wrap up the module with creating a Kubernetes](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=52.34) [cluster in the cloud, looking at some managed service offerings from several cloud providers.](https://app.pluralsight.com/course-player?clipId=83402031-fd52-4b6f-b737-042579911ec3&startTime=55.52)

[Installation Considerations](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597)

[Let's start this module off with discussing some installation considerations.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=1.24) [What this boils down to is where are you going to install Kubernetes?](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=5.64) [Where are you going to get the service from to deploy your applications?](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=9.24) [And first up is cloud.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=12.66) [And here there are really two major use cases when deploying Kubernetes in](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=14.76) [the cloud. You can deploy infrastructure‑as‑a service,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=18.74) [virtual machines and deploy a Kubernetes cluster on top of that.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=21.74) [With IaaS‑based deployments, the networking,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=25.82) [the hypervisor, and all of the infrastructure.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=28.58) [plumbing isn't something that you have to worry about,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=30.57) [but you do have to worry about the operating system,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=33.09) [patching it,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=35.41) [and installing and maintaining Kubernetes itself as](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=36.06) [software on those virtual machines.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=38.84) [And then there's PaaS offerings, or platform‑as‑a service offerings.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=41.44) [Kubernetes is available as a managed service,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=44.92) [and all the big cloud providers provide this.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=47.4) [And you can simply go ahead and consume Kubernetes as a service.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=49.71) [You don't have to worry about any of the underlying](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=53.94) [infrastructure or redundancy.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=56.1) [So one of the things that you do have to think about when](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=57.83) [you are going with a PaaS offering, or a managed service offering of Kubernetes,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=60.5) [is you lose a little flexibility in the versioning and other](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=64.31) [features that are available inside of Kubernetes,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=68.12) [so there's kind of a tradeoff there.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=70.52) [When you're building your own Kubernetes cluster](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=72.25) [using virtual machines in a cloud, well,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=74.7) [you get to control exactly which version of Kubernetes that you install.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=76.92) [But when you do that,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=81.34) [you will take on more responsibility in administering the cluster when](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=82.21) [compared to a manage service or a PaaS offering.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=86.08) [So another solution to getting Kubernetes is on premises.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=89.04) [You can install Kubernetes on bare metal physical machines,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=91.9) [or you can get Kubernetes on prem using virtual machines,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=94.91) [and that seems to be a pretty common deployment scenario.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=98.23) [And now that leaves us with one giant question.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=101.24) [Which one do you choose?](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=104.45) [Well, that's really dependent on a couple of things.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=106.14) [And what I like to think is that it's dependent a lot upon the skill set of](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=108.47) [the organization and the strategy of the organization.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=112.64) [If you and your teams are cloud ready and you're already in a cloud](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=115.84) [and you're moving forward as fast as you can,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=119.48) [well, yeah, it certainly makes sense to deploy in a cloud offering.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=121.33) [Now if you want to stay on prem, well, that's fine, too.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=125.54) [Again, it goes to the skills and the strategy of the organization.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=127.98) [Which one do you choose?](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=131.94) [Well,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=133.08) [take all of these things into consideration when you're working out](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=133.29) [your installation considerations for Kubernetes.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=136.28) [This course is about the installation and configuration of Kubernetes,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=138.67) [so we're going to focus a lot on installing Kubernetes in virtual machines.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=142.45) [But we are also going to look at some cloud scenarios](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=146.29) [together at the end of this module.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=148.66) [Now let's discuss some additional installation considerations that we want to](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=152.34) [take into account before we get started deploying Kubernetes.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=156.52) [In an earlier module,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=159.36) [we looked at Pod networking and how Pods need to communicate with each other.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=160.78) [So we do have to take that into account before we](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=164.23) [install our cluster networking.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=166.81) [Do we want to use an overlay network,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=168.48) [or do we want to work with a network engineering team to make sure](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=170.34) [that we have the correct layer 2 and layer 3 connectivity between](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=173.24) [the Pods and the nodes in our cluster?](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=177.2) [And we also want to ensure that there aren't any network IP](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=178.8) [range overlaps with what we're doing in our cluster and in the](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=181.72) [rest of our network infrastructure.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=184.8) [Then there's the question of scalability.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=186.25) [Do we have enough nodes in our cluster with the appropriate amount of](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=188.22) [resources on each node in terms of CPU and memory to support the](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=191.3) [workloads that we want to deploy into Kubernetes? We also need to take](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=194.96) [into account that we have enough nodes in our cluster to support the](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=198.51) [workload in the event of a node failure.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=201.49) [And then there's questions about high availability and disaster recovery.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=204.14) [If you've noticed so far, we've discussed a single control plane node cluster.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=207.35) [There's one control plane node doing all of the work in](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=211.61) [controlling everything in our cluster.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=214.37) [That's certainly not a highly available solution.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=216.47) [We'll learn in later courses in this series how we can take that](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=219.04) [control plane node and have replicas of the API server backed with](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=221.93) [multiple copies of etcd to provide redundancy for both the API](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=225.72) [server and the etcd data store.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=229.61) [Additionally, we need to take into account for disaster recovery scenarios,](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=232.04) [ensure that we have backup and recovery of that etcd data store in the event of a catastrophic failure.](https://app.pluralsight.com/course-player?clipId=5f4148df-26ef-4ce3-a052-f7c6a91da597&startTime=235.48)

[Installation Methods](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1)

[Now, let's look at some installation methods for how we can get Kubernetes](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=1.24) [up and running, and first up is desktop installations.](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=5.1) [A desktop installation of Kubernetes is great for](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=8.16) [development environments or just playing around to learn](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=10.79) [how the Kubernetes platform works.](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=13.06) [Primarily,](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=15.08) [I use Docker Desktop for Mac when I just want to test various things and kick](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=15.88) [the tires inside of a Kubernetes platform rather than using a full‑blown](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=19.4) [installation or some managed service in the cloud.](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=23.76) [There is also a desktop version of Kubernetes for Docker for Windows as well.](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=26.37) [The installation method that's become the standard for installing and](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=30.38) [bootstrapping a Kubernetes cluster is a tool called kubeadm,](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=33.64) [or kubeadmin.](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=36.61) [This is a package that you can install, and with kubeadm you can bootstrap](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=38.34) [a cluster and get it up and running in a very fast way.](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=41.83) [This is the installation method that we will use in this course. And finally,](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=44.94) [as we introduced a second ago,](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=48.56) [there are cloud scenarios where you can deploy both IaaS and](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=49.89) [PaaS offerings In the various cloud providers.](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=52.98) [There are lots of different ways to get access to Kubernetes and get a](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=55.76) [cluster up and running. So go ahead and experiment one of these, but in](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=58.8) [this course we're going to focus on installing on virtual machines in a](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=62.15) [local on‑premises cluster using kubeadm. And we're also going to look at some clouds scenarios in this course as well.](https://app.pluralsight.com/course-player?clipId=ab5f2a68-44ce-4942-b2bf-2ace5886e3f1&startTime=65.46)

[Installation Requirements](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4)

[When you're installing Kubernetes on bare metal or, like we're](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=1.24) [going to do in this course, in virtual machines, we have some](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=3.82) [system requirements that we need to sort out before we get](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=5.87) [things up and running. And, well, the first thing is you need Linux.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=8.37) [We're going to focus on Ubuntu virtual machines in this course, but](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=11.73) [certainly Kubernetes is compatible with CentOS, RHEL, and many of](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=15.12) [the other major Linux distributions.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=18.82) [Windows worker nodes are available,](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=21.14) [but that topic is out of scope for this course. You'll need 2](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=22.93) [CPUs in your system and also 2 GB of RAM.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=26.02) [Now this is the bare minimum to bring up a cluster for a lab.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=28.88) [If you're sizing the cluster for a production workload,](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=32.1) [you will need an appropriate amount of resources to support your workload.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=34.44) [Next, you need to ensure that the swap is disabled on your system,](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=37.8) [and I'll show you how to do that in an upcoming demonstration.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=41.14) [In addition to the base system requirements that we just looked at,](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=43.84) [you'll also need a container runtime.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=47.32) [Your container runtime will need to be Container](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=49.29) [RuntimeInterface, or CRI, compatible.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=51.46) [Currently supported is containerd, Docker, and CRI‑O.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=53.77) [Now,](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=58.14) [as of Kubernetes version 1.20, Docker has been deprecated and](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=58.41) [will be removed in version 1.23 or later.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=62.33) [For this course,](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=65.41) [we're going to walk through an installation of containerd and build a](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=66.55) [cluster with that as our container runtime in our demos.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=69.76) [But I'm also going to provide example code for building a](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=73.14) [cluster with Docker in the course downloads.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=75.89) [From a networking standpoint,](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=78.46) [we've introduced so far the networking requirements for Kubernetes and](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=79.75) [pod networks, and so we already know that we need network connectivity](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=82.99) [between all of the nodes in the cluster.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=86.3) [Additionally, though, we also need to ensure that each system has a unique hostname and a unique MAC address.](https://app.pluralsight.com/course-player?clipId=982d8f0e-d017-4f76-8bd5-c243179677d4&startTime=88.54)

[Understanding Cluster Networking Ports](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4)

[Now I want to introduce you to the ports that are required to run](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=1.14) [Kubernetes in the event that you need to build firewalls or other](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=4.42) [security perimeters around these resources.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=7.64) [Now, we know that this is what our cluster looks like,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=9.94) [and we know that the control plane node provides a](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=12.57) [collection of services such as the API server,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=14.96) [etcd, and so on.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=17.31) [We also know that worker nodes need to access the API server.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=19.14) [Specifically,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=22.93) [the kubelet and the kube‑proxy on worker nodes will](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=23.67) [talk to the API server over TCP/IP.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=26.17) [And so now, let's go through each of these cluster components](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=29.54) [together and discuss the ports that are required and who uses](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=32.21) [those ports so that you can develop the firewall rules to help](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=35.81) [secure your Kubernetes platform.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=38.8) [The API server, by default,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=40.86) [runs on port 6443. Now that's configurable to any port number,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=42.44) [but that's the default port.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=46.64) [Who needs to talk to that?](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=47.84) [Well,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=49.64) [pretty much anything in the cluster and anything that](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=49.83) [needs to talk to it externally,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=52.88) [whether it's you administratively with kubectl at the command](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=54.3) [line or any pipeline tools that need to have access to the](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=56.99) [system. Etcd runs on 2379 and 2380.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=60.3) [Who needs to talk to etcd?](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=64.69) [The API server does because etcd is where the API server persists its data.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=66.59) [If you're running a redundant configuration of etcd,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=71.84) [the various replicas of etcd will need to communicate](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=74.64) [with each other over these ports.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=77.62) [And so if you start scaling that out for redundancy purposes,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=79.25) [these are the ports that are required for each of](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=82.2) [those additional etcd replicas.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=84.65) [Next is the scheduler, which runs on 10251, and it's used by, well, itself.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=86.84) [If you go look and see at what IP and port it's listening on,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=92.44) [it's listening on 10251, but only on localhost.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=95.6) [It's not exposed to the outside world. And the same](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=98.84) [goes for the controller manager,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=101.56) [which runs on port 10252, and it also is just listening on localhost.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=102.93) [And then finally, on the control plane node is the kubelet.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=108.64) [This runs on port 10250, and all of the control plane components will](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=111.84) [need access to it inside of our Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=115.33) [Now on the worker node side of the house, there's a](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=119.14) [kubelet running on each worker node,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=121.4) [and it runs on port 10250. And the control plane](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=123.45) [elements will need access to this kubelet.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=126.54) [And now, for something that we haven't introduced yet, the NodePort service.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=128.94) [It's a type of service that exposes our services and ports](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=133.14) [on each individual node in our cluster,](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=136.71) [and those port ranges are going to be allocated from 30000 to 32767.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=139.34) [Who needs access to these ports?](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=145.53) [Well, anything that would need access to the services published on those ports.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=147.74) [We're going to cover NodePort services in much detail in our networking course later in the series.](https://app.pluralsight.com/course-player?clipId=1b20338b-f7e9-4eb1-ad55-8327b233e2c4&startTime=151.61)

[Getting Kubernetes](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435)

[All right,](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=1.44) [so we've gone over the high‑level installation scenarios,](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=1.75) [we've gone through the system requirements,](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=4.62) [but what we haven't talked about is where do we actually](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=6.31) [get the Kubernetes software from. Well,](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=8.68) [Kubernetes is maintained on GitHub, so if you go to](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=10.98) [github.com/kubernetes/kubernetes, that's where you'll find the](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=13.12) [living, breathing project that is the Kubernetes software.](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=16.51) [So, honestly, if you're up for it,](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=20.24) [you can go ahead and contribute to this project as well.](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=22.39) [It's a very, very active and vibrant community.](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=24.86) [In addition to just the raw software that's available on this website,](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=28) [there is a ton,](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=31.1) [I mean a ton, of deep‑dive documentation available for you to](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=32.35) [learn about how things work at the deepest level.](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=36.68) [In fact, if you're up for it, you can go read the code itself.](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=39.3) [Now, for this course,](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=42.84) [and for most of the production systems that I support, I use Linux](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=43.91) [distribution repositories, both yum or apt depending on which Linux](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=47.21) [distribution I'm using. In this series of courses, we're going to use Ubuntu, so we're going to get Kubernetes from an apt repository.](https://app.pluralsight.com/course-player?clipId=b0f4845d-f2a5-4a1a-a790-a39ccf314435&startTime=51.28)

[Building Your Own Cluster](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10)

[So with the software in our hands,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=1.04) [it's now time to learn the steps that we need to take to](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=2.67) [build our first Kubernetes cluster together.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=4.83) [And, well,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=7.38) [step one is install and configure a container](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=8.17) [runtime and the Kubernetes packages.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=10.97) [Today, in this course,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=13.64) [we're going to learn how to install Kubernetes from packages](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=14.6) [and use containerd as our container runtime.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=16.88) [Then once the packages are installed, we need to create our cluster.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=19.9) [What this means is, using kubeadm,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=23.69) [we're going to bootstrap that first control‑plane node and get](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=25.46) [those critical cluster components up and running,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=28.97) [the API server, the controller manager, etcd,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=31.21) [and so on.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=33.89) [With the control plane up and running,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=34.76) [then we need to configure our Pod networking environment.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=36.75) [In this series of courses,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=38.96) [we're going to use an overlay network for Pod networking in our cluster.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=40.36) [Once we have our Pod network up and running,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=44.74) [we can then join additional nodes to our cluster for worker nodes](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=46.46) [that we can use for our application workloads.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=49.72) [Now,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=54.14) [let's look at some of the required software packages to get](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=54.45) [started working with our first Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=57.13) [Again, we're going to be using Ubuntu in this series of courses,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=59.79) [so this means we'll be using the app package manager for package installation.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=62.45) [First up, we'll need a container runtime.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=67.34) [We're going to use containerd in this series of courses.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=69.2) [But as we discussed a moment ago,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=72.14) [I'm going to give you code for both containerd and Docker installations.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=73.5) [In our upcoming demo,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=77.13) [we'll walk through the process of installing and](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=78.84) [configuring containerd together.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=81.24) [And in the course downloads,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=82.72) [I'll provide examples for both containerd and Docker since](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=84.09) [Docker is going to be around until version 1.23,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=87.01) [or later.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=89.85) [Next is the kubelet.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=90.94) [The kubelet is the thing that's going to drive the work](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=92.12) [on individual nodes in our cluster.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=94.45) [That comes from a package named kubelet.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=96.53) [There's also a package named kubeadm, or kube Adam,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=98.84) [or kube admin, whatever you want to call it,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=102.1) [but it's the tool responsible for bootstrapping our cluster](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=104.13) [and getting the cluster components up, running,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=107.5) [and configured.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=110.23) [It's also the tool that we're going to use to join](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=111.34) [additional nodes to our cluster.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=113.49) [And then finally, there's kubectl, or kube control,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=115.27) [or kube cuddle, whatever you want to call it again,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=118.38) [but it's the primary command‑line tool that we're going to use](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=120.5) [to administer the workloads in our cluster.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=123.47) [Now,](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=126.44) [I do want to call out that you want to install all four of](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=126.65) [these packages on all nodes in a cluster, regardless of if they're a control plane node or a worker node.](https://app.pluralsight.com/course-player?clipId=099a5e65-8d7c-4db0-89e0-079cebb29f10&startTime=129.81)

[Installing Kubernetes on VMs](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac)

[Now,](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=1.04) [let's go ahead and look at the sequence of commands that we need to use to get](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=1.39) [an installed Kubernetes on some Ubuntu virtual machines.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=5.09) [And I do want to point out here that we need to do this on all the](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=8.4) [nodes that we're going to have in our cluster,](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=11.3) [both control plane and worker nodes.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=13.11) [First up, we need to install our container runtime.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=15.74) [And on an Ubuntu system we'll do that with apt‑get install containerd.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=18.44) [When we get into the demos,](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=22.84) [we'll have some additional configuration steps that are needed to](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=23.91) [configure containerd to use the systemd cgroup driver.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=26.52) [The next thing that we're going to do here is we're going to add the GPG key](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=30.44) [for the apt repository where the Kubernetes packages live,](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=33.41) [and then we're going to execute this series of commands to add the](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=36.43) [Kubernetes apt repository to our local repositories list.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=40.01) [Now, don't worry about writing this all down or taking screenshots,](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=43.7) [these will be available to you in the course downloads.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=46.83) [And we're going to go through this process together](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=49.34) [in our upcoming demonstration.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=51.04) [Now, with that new repository installed,](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=52.64) [we need to tell apt to update its package list,](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=54.84) [and we can do that with apt‑get update.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=57.07) [And then we'll use apt‑get install to install the](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=59.44) [three remaining required packages, the kubelet,](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=61.94) [kubeadm, and kubectl.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=65) [Now,](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=67.15) [here's where we're going to deviate from the normal Linux package installation.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=67.48) [What we're going to do here is we're going to mark these packages with](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=71.54) [apt‑mark hold for the four packages that we need to install.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=74.71) [And the reason for this is we no longer want apt to](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=78.01) [maintain the upgrading of these packages.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=80.53) [We're going to service these packages outside of the normal security updates](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=83.21) [for the system so that we can control when we move between versions of](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=87.1) [Kubernetes independent of security patches being applied.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=91.27) [And this is needed because Kubernetes has a defined upgrade](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=94.73) [process when moving between versions. We're going to cover that process in a course later in this series.](https://app.pluralsight.com/course-player?clipId=523452d8-18d1-49e5-81b8-21c327ab12ac&startTime=98.14)

[Lab Environment Overview](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58)

[So we just looked at the code‑level detail to install](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=0.94) [Kubernetes on some Ubuntu virtual machines.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=3.87) [Now, let's zoom out a little bit and look at the lab](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=6.13) [environment that we're going to use in this course.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=8.65) [So, here you can see we have one control plane node and three worker nodes,](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=10.81) [and so that's the topology of the cluster that we're going to](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=14.68) [construct together in the upcoming demonstrations and use](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=17.34) [throughout this series of courses.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=20.76) [We're also going to have kubectl installed on the control plane node.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=22.07) [And so we'll drive all of our work from that particular](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=26.56) [node when we're working with our cluster.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=29.36) [The version of Ubuntu that we're using in the lab virtual machines](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=31.33) [is Ubuntu 18.04. I've tested all the labs with the current version](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=34.9) [of Kubernetes 1.21 on Ubuntu 18.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=39.43) [As versions of Kubernetes change,](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=42.54) [I will continue to update the code in the course downloads to ensure](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=44.21) [that they work with the latest version of Kubernetes.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=47.4) [So if you experience an issue,](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=49.84) [please reach out to me via email or in the course discussion board.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=51.55) [I'll check things out and update the code accordingly.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=54.74) [Now as for our lab VMs, my virtual machines are VMware Fusion.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=57.29) [You can use pretty much any hypervisor you want,](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=61.84) [as long as you have full network connectivity between](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=64.37) [all of the nodes in your cluster.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=66.59) [Minimum CP requirements stand, so I have two virtual CPUs per](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=68.84) [virtual machine in this lab environment.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=73.21) [And I also have 2 GB of RAM in each virtual machine.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=75.56) [I'm a little generous with the virtual machines from a disk space](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=79.06) [standpoint because honestly, it's kind of a harder thing to change](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=81.65) [after the virtual machine is configured, and so I'm allocating 100 GB](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=84.47) [for each virtual machine in this cluster,](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=88.38) [and we also want to make sure that Swap is disabled on](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=90.79) [each virtual machine that we're using.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=93.38) [Now I have all of the host names set for each of these individual](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=95.46) [nodes, and each of these node names is in a host file on each virtual](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=99.02) [machine in the cluster with the corresponding IP address, so that I](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=103.6) [don't have to rely on DNS for hosting resolution between any of the](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=106.79) [virtual machines, that's going to be hardcoded inside a host file on](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=110.63) [each node.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=114.31) [Now as for host names, here we go. For our control plane node, it's](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=115.15) [going to be c1‑cp1 for cluster 1‑control plane 1, and its IP address](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=118.6) [in my lab is going to be 172.16.94.10. You can use whatever IP in](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=124.2) [your lab that you want to use. Now if you're coming from a previous](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=129.59) [version of this course,](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=133.07) [this system used to be called c1‑master1, so please be](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=134) [aware and update any code accordingly. Any where you see](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=137.62) [c1‑master1 in any future course,](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=140.71) [go ahead and swap in c1‑cp1. I'll be updating the videos and the code](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=142.91) [in each course in this series as I get to them.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=148.37) [Next up is c1‑node1, so a pretty easy naming convention there,](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=151.74) [its IP address ending in 1.11, c1‑node2 ending in .12, c1‑node3](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=155.61) [ending in 13. And so that's our cluster, one control plane node and three worker nodes.](https://app.pluralsight.com/course-player?clipId=2f2ddc47-9399-4a98-8a32-9b084f105f58&startTime=158.89)

[Demo: Installing and Configuring containerd](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50)

[All right, It's taken us a long time to get here,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=1.04) [but it's time for our first demo together.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=2.99) [In this demo, what we're going to do is,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=5.1) [we're going to install some packages together,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=6.96) [containerd, kubelet, kubeadm, and kubectl.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=8.66) [And with all those packages installed,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=12.27) [we're going to look at some systemd units for some of these packages to](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=14) [understand how they operate with the underlying systemd system.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=16.93) [All right, so here we are in VS Code,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=23.14) [and let's get started with the process of installing the required](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=24.84) [packages to bootstrap our Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=28) [In VS Code here,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=30.73) [what I'm going to do is highlight the code at the](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=32.12) [top and I'll execute that code,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=34.04) [and it's going to show the output at the bottom in our environment.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=35.84) [So this way we're able to see both the code that I'm](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=39.23) [executing and the output at the same time.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=41.08) [I've already gone ahead and deployed the four virtual machines that we](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=43.93) [discussed in the presentation portion of the course.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=46.83) [So I have c1‑cp1, c1‑node1, 2, and 3 all up and running in my environment here.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=49.63) [I also right now have an SSH connection open to c1‑cp1,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=56.24) [as we can see at the bottom here.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=60.66) [And so let's go ahead and start the process of installing and configuring](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=62.36) [the proper software for our Kubernetes environment.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=66.66) [The first thing that we'll need to do,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=70.44) [and we're going to need to do this on every node in our cluster,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=71.52) [is to make sure that our swap is disabled.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=74.88) [And I've already actually done that for this virtual machine.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=78.11) [And so here you can see if I do a swapoff ‑a I get no output](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=81.24) [because I've already disabled the swap file.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=84.85) [So let me go ahead and show you how I did that.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=87.54) [I did that by editing the fstab.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=89.31) [Highlight this here, run that code at the bottom, and there we can see the](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=91.93) [contents of our fstab. On the second line here, we can see the at the swap](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=95.8) [is disabled because this line is commented out.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=101.48) [And so what you 'll want to do here is either delete that line,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=104.84) [comment it out,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=107.34) [save the file, and then reboot your system, and you're swap will be disabled.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=108.54) [And with that configuration complete,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=113.24) [let's move forward into the installation and configuration of containerd.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=115.06) [Now I do want to call out that the containerd installation right](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=119.84) [now is a little bit more complicated than I think it should be, and](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=122.69) [I expect that this process is going to become more streamlined as](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=126.16) [containerd becomes kind of the centerpiece of the container](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=130.49) [runtimes for Kubernetes.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=133.61) [And so what we're going to see here is, we have to perform some](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=135.84) [extra steps to get this containerd installation configured for](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=138.56) [us to use Kubernetes on top of it.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=142.39) [And what I expect is that this is going to become more simple, and so I](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=144.13) [promise to keep this code up to date with the latest method.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=147.14) [But this is the most current method for configuring and installing containerd.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=150.61) [And so let's walk through that process together.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=154.9) [The first thing that we'll need to do is to load some](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=158.34) [modules, and with lines 28 and 29 here,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=160.58) [I'm going to load both the overlay and the br\_netfilter module.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=163.22) [Now that's the runtime module.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=167.04) [That's right now in this configuration, but I also need to make](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=168.44) [sure that they're set on boot, and we can do that here on lines](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=171.64) [31 through 34 with this heredoc.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=175.07) [And that's going to write the appropriate information into](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=177.3) [the modules‑load.d directory in etcd.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=179.96) [So we'll run that code there.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=182.97) [It'll create that file in the appropriate directory to ensure that](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=184.26) [these modules get loaded when the system reboots. We also need to](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=187.87) [configure some sysctl parameters, and, again,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=192.81) [we want to make sure that those persist across reboots.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=195.48) [And so we'll grab this heredoc here from lines 38 to 42, run that code,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=198.24) [and it's going to right the appropriate sysctl parameters into](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=203.42) [/etc/sysctl.d in the file 99‑kubernetes‑cri.conf. With that file](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=207.27) [created, we can then apply that configuration with the code on line 46](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=214.05) [here, so sudo sysctl, space, ‑‑system.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=218.77) [Now those configurations that are in that file are](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=222.74) [applied to the running system.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=225.49) [With those configuration prerequisites complete, both the modules and the](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=228.14) [sysctl parameters set, it's now time to install containerd.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=232.07) [We'll make sure we haven't up‑to‑date package list, and we can do that](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=237.24) [with the code on line 50 here, sudo apt‑get update.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=240.12) [Once we have the updated package list, we can install containerd,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=243.94) [and the code to do that is on line 51, sudo apt‑get](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=247.07) [install ‑y containerd. Run that code,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=250.17) [and that's going to install containerd for us on this system.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=253.5) [Now, with containerd installed,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=256.94) [we need to apply some configuration specific to containerd,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=258.38) [and we'll do that with the code here on lines 55 and 56.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=262.39) [First up, we'll create a directory for a containerd configuration file to live](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=267.44) [in. On line 55 we have mkdir ‑p /etc/containerd to make that directory. Now on](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=271.91) [line 56 here, we can use the containerd command to generate a default](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=278.65) [configuration file. So there we see sudo containerd config default. That's](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=284.44) [going to generate a default configuration file, and we're going to pipe that](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=289.12) [output into sudo tee and write that output into /etc/containerd/config.toml,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=292.44) [and inside that configuration file will be the configuration attributes for](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=298.55) [containerd.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=304.34) [We're pretty good with the defaults except for one change.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=305.54) [We need to change the cgroup driver of containerd to systemd,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=308.64) [which is required for our kubelet, and we'll look at how](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=313.51) [to configure the kubelet in the next demonstration a](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=316.47) [little bit later in the course.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=320.17) [And what we're going to do here is grab the text here on lines 68 and 69.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=321.74) [That's our cgroup driver configuration, and copy that into our](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=326.11) [clipboard, and then open up the config.toml file in vim. And we're](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=329.73) [going to look for the string that we have on line 65 there that ends in](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=333.89) [containerd.runtimes.runc And below that,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=337.53) [we're going to paste in this text from 68 and 69 into that file.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=341.1) [So let's go ahead and find that information in our config file.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=345.17) [So jumping down to the bottom here,](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=350.04) [I'm going to look for containerd.runtimes.runc. Once I find that](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=351.92) [line, here we go, I'm going to add a new line and drop in that](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=356.54) [text that I copied from lines 68 and 69.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=362.88) [And so there we can see I added that configuration into the configuration file.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=365.94) [I'm going to write that out and save it. With that written out, let's go](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=370.54) [ahead and restart containerd, and we can do that with sudo systemctl](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=374.26) [restart containerd to make that configuration change for containerd, part of the running configuration for containerd.](https://app.pluralsight.com/course-player?clipId=a26cfcd6-7693-49b7-8afb-0340d4355b50&startTime=378.96)

[Demo: Installing and Configuring Kubernetes Packages](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6)

[With containerd installed and configured,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=1.14) [now it's time to move forward and install the Kubernetes packages kubeadm,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=3.34) [kubelet, and kubectl,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=7.81) [and the first part of that process is to add Google's apt repository GPG](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=9.57) [key to our system so that we can trust that repository.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=14.66) [The next step then is to add the Kubernetes apt](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=18.54) [repository to our local repositories list,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=21.61) [and we can do that with the heredoc that's on lines 86 through 88.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=24.4) [Run that code together,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=28.89) [and that's going to configure that local apt](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=30) [repository on our local system here.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=32.98) [With that new repository added,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=35.43) [we're going to want to update the package metadata information](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=37.75) [for our system so that we can get the package information from](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=40.66) [that newly added repository, and we can do that with sudo apt‑get update.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=43.34) [Now let's take a peek at what's available to us to](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=48.74) [install from that new repository,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=52.16) [and I want to look at the kubelet packages to see the different](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=55.94) [versions of the kubelet that are available.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=58.83) [And I could do that with apt‑cache policy and then](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=60.62) [specifying the package in kubelet.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=63.25) [I want to pipe that into head and limit that to 20 lines of output.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=65.2) [In the output at the bottom here,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=68.86) [we can see the different versions of the kubelet that are](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=70.39) [available as packages in the repository.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=73.45) [So there we see 1.20.2‑00, 1.20.1‑00.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=75.56) [And so those are all the different versions of the kubelet](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=81.45) [that are available to us to install.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=84.52) [And what we're going to do here is, we're actually,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=86.94) [we're going to pin our installation to a specific version,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=88.64) [and we're going to install 1.20.1‑00,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=92.19) [and I have that set here as an environment variable on line 98.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=96.75) [And what we'll do then is, on line 99,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=101.14) [we'll specify the exact version of the packages that we want to install.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=103.79) [So there we see sudo apt‑get install ‑y,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=107.79) [and then the package name equals,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=110.87) [and then we're referencing that environment variable that we just declared,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=112.58) [version.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=115.73) [We're going to repeat that pattern for kubeadm and also kubectl,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=117.14) [specifying the version for each, making sure that they're the same,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=120.73) [all on 1.20.1‑00.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=123.75) [And the reason why I want to do that is because later on in this series,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=126.35) [of course, we're going to run an upgrade to a newer version of Kubernetes.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=130.31) [And so if you saw there was 1.20.2 is available,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=134.75) [so that way I have a version to upgrade to when we get to that course.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=137.61) [Let's go ahead and run this code together to get the kubelet,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=141.8) [kubeadm, and kubectl installed.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=144.56) [Now with our packages installed,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=146.82) [let's go ahead and mark all four packages with hold,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=148.39) [and we can do that with sudo apt‑mark hold,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=152.34) [kubelet, kubeadm, kubectl, and containerd.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=155.4) [And what that will do is prevent these packages from being](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=159.64) [updated when someone comes along and updates the system for](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=162.39) [security updates with apt.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=165.6) [Now on lines 104 and 105 here,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=168.33) [I have the code that'll allow you to install the latest](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=171.65) [and greatest version of Kubernetes.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=173.98) [I held us back one version intentionally and also showed](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=175.59) [you how to pick a specific version.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=178.27) [If you want to just install the latest version that's available,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=180.33) [you can use the code that's on line 104 and 105.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=182.73) [So now with the requisite packages installed,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=186.34) [let's look at the systemd units for a couple of the](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=188.3) [services that were installed, and first up is the kubelet.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=191.78) [I want to look at sudo systemctl status kubelet.service and look at the output](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=194.99) [that's generated for this particular system to unit. And we can see something](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=200.18) [interesting here, is that we see that the main process exited, code=exited,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=204.88) [status=255 failed with result exit=code.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=209.48) [And we also see that the systemd unit's status is](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=213.24) [activating, rather than active,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=216.21) [which would mean that the service is up and running. What's](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=218.02) [happening right now is the kubelet's actually crash looping](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=220.6) [because there's no cluster configuration yet. We haven't told the](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=223.14) [kubelet to do anything specific.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=226.84) [And so later on when we get into the next portion of the course](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=229.24) [we're going to learn how to bootstrap a cluster. And what that's](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=231.53) [going to do is write some information into a specific location](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=233.78) [the kubelet's looking for.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=237.22) [Well, that information's not available there yet,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=238.57) [and so the kubelet is going to be in what's called a crash loop,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=241.09) [looking in that specific location for the configuration information that's](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=243.47) [generated by the bootstrapping process for our cluster.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=248.39) [But we haven't done that yet.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=251.78) [And so hang on to that thought right now, and a little bit later](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=253.08) [in the next demo when we bootstrap the cluster,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=255.66) [we're going to revisit this and see when the kubelet is up and running](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=257.53) [and healthy because our cluster has been bootstrapped.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=261.06) [The other service that I want to look at is also containerd,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=264.24) [just to make sure that that's in good shape before we move](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=268.01) [forward with our demonstrations.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=270.23) [So sudo systemctl status containerd.service, run that code](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=272.08) [there. We can see it's active and running,](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=276.85) [so it's up and loaded and in a proper state.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=279.59) [So let's break out of this output here. And one final step is](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=283.14) [just to make sure that both of these services are set to start](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=287.44) [up when the system starts up.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=291.67) [We can do that with sudo systemctl enable kubelet.service.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=293.01) [So let's going ahead and run that code. And we'll do the same, sudo](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=297.58) [systemctl enable containerd.service to make sure that both of those](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=300.6) [services are set to start up when the system boots.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=305.15) [So what we've done right now is, we've installed the required packages](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=308.84) [to start building, or bootstrapping, our cluster, what you're going to learn about now in the next portion of the course.](https://app.pluralsight.com/course-player?clipId=1dbc3f75-0ea0-43eb-9f63-55df080260c6&startTime=312.35)

[Bootstrapping a Cluster with kubeadm](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3)

[With the software packages installed,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=1.14) [it's now time to start the process of bootstrapping or](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=3.05) [creating a cluster with kubeadm.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=5.83) [Kubeadm is a tool that we can use to manage the process of creating our cluster.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=7.92) [It's going to walk through various phases, and build and configure our](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=12.28) [cluster for us, and right now we're going to walk through each one of](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=15.82) [those steps together so that you know what's really going on during the](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=18.77) [creation of a Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=21.8) [Now, to bootstrap a cluster at the command line,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=24.14) [you'll type kubeadm init,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=26.18) [and it's going to begin the process of creating your cluster.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=28.44) [The first phase is the pre‑flight checks.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=31.1) [What kubeadm is going to do here for you is execute a series of checks](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=33.76) [that will help ensure a successful cluster creation.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=37.68) [So things like ensuring that you have the right permissions](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=40.74) [on the system that you're working with.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=42.83) [It's also going to verify that you have the appropriate system](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=44.53) [resources on this box in terms of CPU and memory,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=47.51) [and another important pre‑flight check that it executes is it](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=50.35) [checks to see if there's a compatible container runtime on this](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=54.21) [system, and if it's up and running.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=57.16) [If any of these checks fails,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=58.75) [kubeadm will report the error and stop the cluster creation process.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=60.34) [The next thing that kubeadm does for you is it creates](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=64.14) [a certificate authority for you.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=66.56) [Kubernetes uses certificates for both authentication and encryption.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=68.23) [We're going to look at this in more detail in an upcoming slide. In the](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=72.37) [next phase, what kubeadm does is it generates kubeconfig files for you for](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=75.04) [the various cluster components of Kubernetes, so that they can locate and](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=78.74) [authenticate against the API server.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=82.78) [We're going to look at these much more closely in an upcoming slide as well.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=85.54) [Then the next phase is generating static pod manifests.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=88.16) [Static pod manifests are going to be generated for each of the control](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=92.02) [plane pods, and what's going to happen here is they're going to be written](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=95.26) [to the file system and the kubelet is going to monitor that location. If it](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=98.31) [finds a pod manifest there, well the kubelet's going to start up the pods](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=103.08) [defined in that manifest, and we're going to look at this also in much more](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=106.64) [detail in a few moments. And then, after those static pod manifests are](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=109.92) [generated,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=113.62) [kubeadm is going to wait for the control plane pods](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=114.17) [to start up. So the API server,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=116.83) [etcd, and so on, will be started up as pods. Then kubeadm](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=118.36) [goes on to taint the control plane node.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=122.17) [We discussed this a little bit ago when we looked at](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=124.97) [pod operations earlier in the course.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=127.1) [What this means is when Kubernetes schedules pods,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=128.98) [it will never schedule user pods onto the control plane node, it will only](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=131.81) [ever schedule system pods to run on that control plane node.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=136.28) [Then, kubeadm generates a bootstrap token used for joining](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=139.44) [additional nodes to the cluster, and then finally,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=143.32) [in its final phase,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=145.8) [it will start up any add‑on pods, such as DNS and kube‑proxy pods.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=146.9) [The process that we just described here is the default process](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=151.74) [if you use kubeadm init with no parameters.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=154.7) [This bootstrapping process is very customizable,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=158) [and you can build more complex or customized clusters using kubeadm command](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=160.68) [line parameters, or even a YAML manifest configuration file to describe](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=165.12) [your cluster's configuration. In this module,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=169.53) [we're going to learn how to create a cluster with a configuration file,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=172.32) [since our cluster is using containerd rather than Docker as its](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=175.46) [container runtime. In that cluster configuration file, we will specify](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=178.88) [that we're using containerd as our container runtime, and we'll also](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=182.85) [define the C group driver for the kubelet,](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=186.39) [which is systemd.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=188.66) [The reason why we're using systemd as our C group driver is because Ubuntu is a systemd based system.](https://app.pluralsight.com/course-player?clipId=3941bfcd-4e7c-4d1d-8ec3-c5919d26fbf3&startTime=189.93)

[Understanding the Certificate Authority's Role in Your Cluster](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7)

[It's time now to zoom in on those facets of kubeadm](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=1.34) [that I called out a second ago,](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=4.23) [and we're going to start that conversation with the certificate authority.](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=5.73) [Kubeadm init, by default,](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=9.12) [will create a self‑signed certificate authority for you.](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=10.79) [And if desired or required in your organization,](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=14.04) [you can tell kubeadm init to integrate with an external PKI,](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=16.34) [or public key infrastructure, if that's needed in your organization.](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=19.61) [The certificate authority that's created by kubeadm init is](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=23.54) [used to secure cluster communications.](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=27.14) [It's going to generate server certificates that are used by](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=29.64) [the API server to encrypt the HTTP stream that is the API](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=32.13) [server's communication protocol.](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=36.16) [So that's going to be HTTP over TLS, in other words, HTTPS.](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=37.77) [In addition to securing cluster communications,](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=41.95) [the certificate authority is used to generate certificates for the](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=44.58) [authentication of users and cluster components like the scheduler,](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=47.39) [controller manager, and kubelet to authenticate the request of the API server.](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=51.39) [And so we'll have strong authentication methods for](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=56.84) [users that have to operate the cluster, any of the cluster components,](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=59.46) [and also the nodes in our cluster since the kubelets will use](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=62.72) [certificates to authenticate and verify their identity.](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=65.97) [The certificate authority files live in /etc/kubernetes/pki and will](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=69.54) [be distributed to each node in your cluster when you join additional](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=74.4) [nodes to your cluster using kubeadm init.](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=78.19) [This distribution process ensures that the nodes trust the self‑signed CA.](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=81.14) [For more information on how to customize your kubeadm init process,](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=85.55) [check out this link here.](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=89.15) [This article includes the many customizable parameters for kubeadm init,](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=90.57) [including how to specify an external PKI configuration, amongst many more.](https://app.pluralsight.com/course-player?clipId=47a15ecf-fa34-46e9-852a-fe0e69f734a7&startTime=95.49)

[kubeadm Created kubeconfig Files and Static Pod Manifests](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710)

[Next up is kubeadm‑created kubeconfig files.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=1.14) [What is a kubeconfig file?](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=5.14) [A kubeconfig file is a configuration file that defines how](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=6.38) [to connect to your cluster API server.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=9.85) [Inside a kubeconfig file, you're going to find the certificates](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=12.84) [used for client authentication. And you'll also find the](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=15.51) [cluster API server's network location.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=18.5) [Usually it's going to be its IP address or DNS name.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=21.07) [Often included in a kubeconfig file is the CA certificate of the CA that](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=24.08) [was used to sign the certificate of the API server.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=29) [This way,](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=32.27) [the client can trust the certificate presented by](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=32.55) [the API server upon connection.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=35.13) [Kubeadm creates a collection of kubeconfig files used by various](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=37.17) [cluster components, and those live in /etc/kubernetes.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=41.91) [Let's take a look at each one of those now.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=45.1) [The first kubeconfig file that's generated is admin.conf,](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=47.44) [which is the Kubernetes cluster administrator account,](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=50.76) [essentially a super user inside of our Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=53.5) [And we're going to use this to authenticate to our cluster today in our](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=57.1) [upcoming demo. Kubeadm also generates kubelet.conf. Kubelet.conf is used](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=60.32) [to help the kubelet locate the API server and present the correct client](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=65.7) [certificate for authentication.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=69.8) [Similar kubeconfig files exist for the controller manager and the](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=71.23) [scheduler, again, to tell these components where the API server is](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=74.99) [on the network and also which client certificates to use for](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=78.2) [authentication to the API server.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=81.44) [These kubeconfig files are generated by kubeadm during the cluster](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=83.54) [bootstrapping process and are used by these components to locate and](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=87.68) [authenticate the API server. Later on in this series of courses, we'll look](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=91.18) [at how to create kubeconfig files for individual users that will locate and](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=95.09) [authenticate to the cluster API server, and we'll also learn how to set](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=99.33) [appropriate permissions for users.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=103.34) [The next stop on the list in our closer look at what kubeadm](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=106.94) [init does for us is static pod manifests.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=109.71) [A manifest describes a configuration, and in this case it's going to describe](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=112.64) [the configuration of a pod. And so kubeadm init is going to generate static](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=116.61) [pod manifests for our cluster control plane components that it needs to](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=120.46) [bootstrap a cluster and get it up and running.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=124.46) [Those static pod manifests are going to live in](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=126.84) [/etc/kubernetes/manifests. And it's the job of kubeadm init to](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=129.01) [generate a static pod manifest for each of the core cluster control](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=134.04) [plane components, and that includes etcd, the API server,](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=137.97) [the controller manager, and the scheduler. So, kubeadm init generates the](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=141.7) [static pod manifests for each of the control plane pods,](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=146.44) [and they live in that special directory.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=149.01) [It's the job of the kubelet to watch this directory on the file](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=151.34) [system. And for any static pod manifests that it finds in this](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=154.97) [directory, it'll start up the pods described in those manifests.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=157.7) [And in this process here, it's the control plane pods.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=161.42) [This is how a cluster starts up after a reboot as well.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=164.51) [The kubelet is started by the systemd daemon on the node on boot. And then on](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=168.55) [the cluster we'll find the static pod manifests in this directory, and it will](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=173.2) [start up any of the pods defined in the static pod manifests. And in this case,](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=177.5) [it's going to be the control plane pods.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=181.86) [Now, you might be wondering,](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=184.04) [why do I need this special configuration to start up these pods?](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=185.15) [Well, this is what enables the startup of these core cluster components, well, without the cluster being up and running.](https://app.pluralsight.com/course-player?clipId=113b6ea4-9ebe-43d1-a5fc-2dadce715710&startTime=188.2)

[Pod Networking Fundamentals](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb)

[Now,](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=1.24) [that wraps up the conversation about kubeadm init and a](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=1.59) [closer look at each one of those phases.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=4.83) [Now,](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=7.04) [let's shift the focus over to the last thing that we need to talk about](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=7.26) [before we actually stand up our Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=10.03) [We're going to talk about Pod networking again.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=12.17) [Let's say we're designing a cluster.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=14.84) [We have the requirements of the Kubernetes networking model](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=16.24) [that we introduced earlier in this course.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=19.48) [We need to ensure that we have a single un NATed IP address per Pod and](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=21.4) [that all Pods have reachability between each other Now,](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=26.12) [we could, of course, use direct routing,](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=28.64) [and in this case,](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=30.45) [we would have to configure the infrastructure underneath our](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=31.24) [Pods and nodes to support full IP reachability between the Pods](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=33.73) [and nodes without NAT using real IPs, and sometimes that's just not feasible.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=37.8) [Perhaps you're in a cloud scenario or in a scenario where you don't](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=42.65) [have control over the underlying infrastructure,](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=46.31) [and that's where overlay networking comes in,](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=49.1) [sometimes called software‑defined networking.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=51.2) [With overlay networking,](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=53.04) [what happens is we get the appearance of a single layer 3](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=54.36) [network that the Pods can all communicate on. And some of the](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=57.14) [techniques that are used to make this happen are tunneling and](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=60.26) [encapsulation of the IP packets,](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=63.12) [and that's the responsibility of the overlay network to](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=65.45) [facilitate for those communications of those packets](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=68.08) [between the Pods in an unchanged way.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=70.44) [Now, some of the overlay networks that are available to us](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=72.87) [include Flannel, Calico, and Weave Net.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=75.62) [These are overly networks,](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=78.34) [each of which have different capabilities and features that may be](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=79.35) [interesting to you as you deploy your overline network. In our demos](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=82.12) [today, we'll be using Calico for our Pod network.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=86.16) [When deploying a Pod network,](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=89.14) [it will provide the IP address management for the Pods deployed.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=90.75) [Now, it's imperative that the network range must not overlap with](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=94.14) [other networks for your network infrastructure, so be sure to work](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=97.76) [with your network engineering teams to find an appropriate address](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=101.55) [range to use for your Pod network.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=104.41) [For a full discussion of overlay networking,](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=106.6) [I encourage you to check out this link here.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=108.83) [You'll find many more overlay networks available and other](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=110.73) [networking solutions that you can use for your Pods and nodes so](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=113.87) [that they can communicate to each other.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=117.61) [This discussion here is really just scratching the surface for what you need](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=119.01) [to know about networking to get a cluster up and running.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=122.68) [We have a full course dedicated the Kubernetes networking coming up later in this series.](https://app.pluralsight.com/course-player?clipId=d22cc231-ce3b-4077-8c19-92146ec0dafb&startTime=125.44)

[Creating a Cluster Control Plane Node and Adding a Node](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda)

[With all of the preliminaries out of the way,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=1.04) [it's time to use kubeadm to bootstrap the control](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=2.82) [plane node in our Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=6.22) [The first thing that we need to do is download a YAML manifest that](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=8.94) [describes our pod network that we want to deploy,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=12.38) [and that's calico.yaml.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=15.41) [This is the actual deployment manifest for our pod network.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=17.14) [Inside of that file is where you can specify the pod network IP address range.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=21.04) [We'll look at how to configure that in our upcoming demo.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=26.04) [If this URL changes for the Calico pod network,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=28.84) [I will keep the course downloads updated with the new URL.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=32.25) [Next, you'll need to create a cluster configuration file,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=35.84) [and there's an easy way to do that with kubeadm.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=39.64) [We can use kubeadm config print init‑defaults, and then write that out to file.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=41.83) [So here,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=47.61) [we're using tee to write the output to console and also into a](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=48.3) [file manifest named ClusterConfiguration.yaml.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=52.15) [Inside of that file is the configuration defaults for a Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=56.24) [In our upcoming demo,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=61.64) [we're going to use this command to generate that cluster configuration file.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=62.63) [And then we're going to make some required changes to that](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=66.67) [file in our lab environment together.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=69.32) [Once we have that cluster configuration file,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=71.57) [the next thing that we need to do is to execute sudo kubeadm init,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=74.04) [kicking off the kubeadm init process that we just walked through together.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=77.78) [We need to then pass in our cluster configuration file with the](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=81.94) [‑‑config parameter and specify the file location of the](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=85.07) [configuration file that we just created,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=89.82) [ClusterConfiguration.yaml.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=91.6) [We then need to specify the CRI socket of our container runtime.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=93.94) [And we're going to be using containerd in this course,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=98.31) [so here you can see the path /run/containerd/containerd.sock,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=100.51) [which is the file location of our CRI socket.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=105.07) [If you don't specify a CRI socket in the current version of kubeadm](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=109.04) [it will default to Docker and will error out.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=112.9) [I expect this to change and be updated to auto detect the](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=115.88) [container runtime in future releases of kubeadm.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=119.53) [When you get into the demos,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=122.03) [if you decide to use Docker as your container runtime for your lab,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=123.67) [you'll find the kubeadm init process is a bit different.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=127.74) [In fact, it's a lot simpler.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=130.84) [The reason is a lot of the configurations are auto detected and](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=132.64) [don't require cluster configuration files for specifying a CRI](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=136.21) [socket at the command line with a parameter.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=139.83) [But like I said a second ago,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=142.44) [I expect the containerd implementation to get a lot simpler,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=143.66) [and I'll update the course accordingly.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=146.93) [Now, once kubeadm init finishes,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=148.94) [all the control plane pods will be up and running.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=151.47) [Kubeadm init will also print out the commands and the parameters](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=153.94) [needed to join additional nodes to your cluster.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=156.98) [And it will also print out how to configure a](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=159.34) [kubeconfig file for an administrative user.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=161.9) [We'll review the join command in a moment,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=165.24) [but now let's look at how to work with that kubeconfig](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=167.55) [file for an administrative user.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=170.39) [The following steps will enable your currently logged in user to](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=172.74) [perform administrative functions on your cluster.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=175.79) [And first what we'll do is we'll create a directory for our](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=178.24) [kubeconfig file in our current HOME directory.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=181.3) [And then what we'll do is we'll copy the admin.conf kubeconfig file](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=184.37) [from /etc/kubernetes into our HOME directory.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=189.1) [And then what we'll do is adjust the permissions on that file so](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=193.49) [that we can access this with our current user.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=196.36) [This kubeconfig file allows the current user to perform](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=199.64) [administrative functions against the cluster.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=202.44) [And inside of that kubeconfig file, you'll find the required](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=204.64) [certificates to authenticate to the cluster and also the network](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=207.59) [location of our cluster's API server.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=211.02) [The final step that we need to take after creating our control plane](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=213.79) [node with kubeadm init is to deploy a pod network.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=217.05) [And we can do that with kubectl apply ‑f calico.yaml.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=220.45) [This will read the deployment manifest file that we just download,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=225.44) [send it into the API server, and then create the pod network in our cluster.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=229.11) [Once the pod network is finished,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=234.04) [then the DNS addon pods will be able to start up,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=235.55) [and we'll have a fully functioning control plane node.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=238.17) [The next thing that we need to do now is to add some](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=240.94) [additional nodes to our cluster for user workload.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=243.13) [Let's check out how to do that.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=245.89) [Nodes, or worker nodes, are where your user workloads are run in your cluster.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=249.14) [To join a node to a cluster, you first need to install all](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=253.74) [the required software packages onto your soon‑to‑be node,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=256.86) [so your container runtime, in our case that's containerd,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=260.95) [kubeadm, kubelet, and kubectl all need to be installed.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=263.7) [We then use kubeadm with the join parameter to start the](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=267.84) [process of joining a node to a cluster.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=270.67) [Kubeadm join takes some additional parameters,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=272.91) [such as the network location of the API server,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=275.86) [a bootstrap token used to join a node to a cluster, and a CA cert hash used](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=278.02) [to trust the certificate presented by the API server.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=283.03) [We'll look at some example code for that in a second.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=286.34) [When we execute kubeadm join on the node that we want to join to the cluster,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=288.54) [what's going to happen is that node is going to download some](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=292.63) [cluster information and configuration metadata.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=295.61) [It's then going to generate and submit a certificate signing request,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=298.34) [or CSR,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=301.47) [into the API server to generate a certificate that will be used by the](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=302.59) [kubelet on the node we're joining to the cluster.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=306.29) [And this is used by the kubelet to authenticate to the API server.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=308.69) [The CA is going to automatically sign that certificate signing request,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=313.14) [and then kubeadm join is going to download that certificate and store](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=317.04) [that on the file system of the soon‑to‑be node.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=321.21) [That certificate's going to live in /var/lib/kubelet/pki on the node.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=324.3) [Kubeadm join is then going to generate a kubeconfig file](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=329.13) [named kublet.conf. And that kubeconfig file is going to live](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=333.2) [in /etc/kubernetes on the node.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=336.58) [In that kubeconfig file is going to be a reference to that new client](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=339.74) [certificate and also the network location of the API server that we want to](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=343.07) [authenticate against for the cluster that this node is joining.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=347.16) [This process is called TLS bootstrapping.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=350.55) [The code for kubeadm join looks like this,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=355.14) [kubeadm join and then the IP address and port of the API server,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=357.84) [which in our lab is 172.16.94.10 on port 6443.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=362.09) [If you're using a different network address, please update that here.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=367.65) [The next parameter is the bootstrap token.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=371.64) [This effectively is a time ticket or password for joining a node to a cluster.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=374.15) [And then the next parameter after that is the CA cert hash.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=379.34) [The CA cert hash here is used to establish a certificate train of trust](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=382.74) [between the node joined in a cluster and the certificate presented by the API](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=386.31) [server for HTTP requests during the join process.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=390.9) [Recall,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=394.36) [a moment ago I told you that the output of kubeadm init when we](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=395.07) [create our cluster will print this command and the required](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=399.25) [parameters to the screen so that you can use that to join](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=402.9) [additional nodes to your cluster.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=405.7) [In our upcoming demos,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=407.54) [I'll show you just that, and I will also show you](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=408.71) [how to print the bootstrap token,](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=411.03) [the CA cert hash, and also the entire join command on demand so](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=412.79) [that you can get that information when you need it outside of that initial cluster creation process.](https://app.pluralsight.com/course-player?clipId=ab0e7471-d727-44d0-b8d2-6632f2e84eda&startTime=416.91)

[Demo: Creating a Cluster Control Plane Node](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb)

[Alright,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=1.14) [let's get into a demo and look at how we can create our](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=1.5) [first Kubernetes cluster together.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=3.79) [Once that's up and running, we're going to deploy a Pod network,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=5.94) [and then we're going to look at those systemd units again to see what changed](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=8.62) [from the last time when we installed the kubelet and how it's going to react](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=11.85) [differently now that we have a cluster up and running because of the static](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=15.38) [Pod manifests generated by kubeadm init.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=18.39) [Then we're going to look at those static Pod manifests](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=22.24) [for each of our control‑plane Pods,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=24.17) [and then we're going to join a couple of nodes to our new cluster.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=25.74) [Here we are logged into c1‑cp1,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=31.94) [and let's go ahead and begin the process of creating our](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=33.89) [first Kubernetes cluster together.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=37.21) [Now, this process that we're going to go through is for containerd.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=39.64) [If you're interested and still need the Docker demonstrations,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=42.82) [those are available to you in the course downloads here in the docker directory.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=45.78) [This demo that we're going to walk through together, we'll be using containerd.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=50.14) [Now,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=53.73) [the first part that we need to go through for creating our cluster is to](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=53.95) [download the deployment manifest for the Calico Pod network,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=57.54) [and I have the code do that here on line 11,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=61.4) [and so we'll use wget to retrieve the calico.yaml file from the Calico website.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=63.72) [Inside that manifest is a setting that describes the Pod Network IP range,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=69.94) [and that setting is CALICO\_IPV4POOL\_CIDR.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=74.3) [So let's grab this string here, and put that in our clipboard,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=78.11) [and then open up the file that we just downloaded,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=80.9) [calico.yaml, and find that setting inside the manifest.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=83.06) [Now, what we see here is the value for that field is 192.168.0.0/16.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=87.44) [All Pods are going to allocated IPs from that network range,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=94.44) [and so we want to make sure that that network range doesn't](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=97.85) [overlap with other networks in our infrastructure.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=100.25) [If it does, you can set that value here.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=103) [We're going to leave that as default,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=104.93) [and so let's go ahead and break out of this file and get back to our console.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=106.54) [Moving forward in the demo,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=112.34) [the next part is creating a kubeconfig init configuration file.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=113.47) [That configuration file is going to define the settings of the](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=119.54) [cluster that kubeadm is going to build for us.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=122.47) [We can generate a default file with the code here on line 24,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=125.84) [kubeadm config print init‑defaults.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=129.71) [We're going to take the output of that and write that into](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=133.24) [a file named ClusterConfiguration.yaml.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=135.32) [Let's run that code together and review the output of a](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=138.34) [default ClusterConfiguration file.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=141.61) [Scrolling up a little bit, in this output here,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=144.2) [we can see,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=146.55) [I do want to show you that there is one error that's going to be thrown.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=147.2) [In this current version of kubeadm that we're using for this demo,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=151.2) [it's going to look for Docker, and well,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=154.54) [Docker is not installed.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=156.32) [And so that's an active bug that will be fixed in future kubeadm installation,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=157.24) [so we can safely ignore that warning here.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=161.87) [Looking inside of the output of this ClusterConfiguration document,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=164.82) [we see localAPIEndpoint: advertisedAddress: 1.2.3.4.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=168.79) [That's the IP address of the API server,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=173.48) [and so we're going to want to update that to our IP](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=176.32) [address of our control‑plane node,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=179.9) [which in our cluster is going to be 172.16.94.10 or](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=181.24) [whatever IP address you're using in your lab.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=186.07) [The next thing I want to call out is in nodeRegistration: criSocket,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=189.34) [there we see /var/run/dockershim.sock.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=192.7) [We're going to want to update that from the default,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=194.95) [which is Docker, to the container runtime that we're using,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=198.8) [which is containerd.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=201.52) [One other element that we're going to add to this document that's](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=203.04) [not in the configuration yet is defining the cgroupDriver for the](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=205.71) [kubelet and setting that the systemd,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=210.28) [and so we're going to add some configuration there in a moment.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=211.74) [Scrolling down a little bit further in this file,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=214.8) [we can see kubernetesVersion is V1.20.0,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=217.26) [and we're going to want to update that to V1.20.1,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=220.55) [which is the version of the Kubernetes packages that](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=226.48) [we installed in the previous demo.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=228.8) [And so let's go ahead and begin the process of](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=231.08) [updating those four elements together.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=233.45) [Now,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=236.64) [I've written some code here that will swap out those values](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=236.9) [with sed \_\_\_\_\_ that we get through this process in a little](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=239.67) [more streamlined of a fashion.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=242.69) [And so on line 34 here,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=244.3) [I have sed that's going to swap out the advertisedAddress from 1.2.3.4](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=245.98) [to the actual address of our control‑plane node,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=250.46) [which is 172.16.94.10.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=253.71) [Let's go ahead and run that code to make that change.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=256.74) [Moving forward to the next update that we have to make,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=259.15) [we're going to change the criSocket to point to containerd,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=262.1) [which is the container runtime on our node here.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=266.27) [And so there on line 38, we have criSocket: \/var\/run\/dockershim\.sock/,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=269.24) [and we're going to change that to \/run\/containerd\/containerd\.sock.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=273.4) [Run that code there to make that update and move forward to the next](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=280.04) [change that we need to make to our ClusterConfiguration,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=283.25) [which is to set the cgroupDriver to systemd for the kubelet.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=286.04) [Now,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=290.34) [this configuration doesn't exist in the default](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=290.52) [document that's generated by kubeadm,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=293.52) [and so we're going to add that to our ClusterConfiguration file](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=296.05) [with this heredoc here starting on line 42.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=298.97) [And in that code there, you can see on line 46,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=302.39) [cgroupDriver is set the systemd for the kubelet.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=304.82) [Run that code there to append that to the file that we're working with.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=308.01) [Now,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=312.44) [let's hop into that file to review those changes and make that](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=312.74) [one final change for our Kubernetes version.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=316.11) [Down in the output here,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=318.9) [we can see that the localAPIEndpoint: advertisedAddresse is now 172.16.94.10,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=320.24) [which is the IP address of our Control Plane‑Node.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=326.96) [We also see that the criSocket is now /run/containerd/containerd.sock,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=330.74) [and so that update was successful to the configuration document.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=334.34) [Going down a little bit further,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=338.85) [let's go ahead and change the kubernetesVersion from v1.20.0 to v1.20.1.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=340.89) [Now, remember,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=350.97) [this is going to be whatever version of Kubernetes that you](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=351.68) [installed in the previous demonstration.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=354.17) [You want to make sure that those match.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=356.24) [So if you move ahead in a version, please be sure to update that here.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=357.39) [Scrolling down a little bit further, there at the bottom,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=361.3) [we can see our KubeletConfiguration defining the cgroupDriver as systemd.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=364.03) [Let's save this file out and move forward into the next part of our](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=368.71) [demo where we bootstrap our cluster together.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=373.07) [And so on line 58 here, I have sudo kubeadm init,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=376.41) [and then the parameter, ‑‑config=ClusterConfiguration.yaml.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=380.84) [That's the configuration document that we just built up together.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=385.49) [We're also going to specify the criSocket and point that at containerd.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=389.14) [Let's highlight all of that code, and run that together,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=393.43) [and begin the process of bootstrapping our cluster.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=397.09) [Now,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=400.04) [I'm going to speed this process up a little bit so we don't](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=400.25) [have to wait for the whole deployment process,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=402.88) [and then we're going to review the output together.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=405.02) [Alright, with that finished, we can our Command Prompt is returned.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=407.24) [Let's scroll to the top and review the output of that](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=410) [cluster initialization process together.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=412.93) [So in the output here, we can see we're using Kubernetes version 1.20.1.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=415.64) [The next phase is preflight checks where that goes through](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=420.24) [this preflight checks that we talked about during the](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=423.12) [presentation portion of the course.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=425.6) [In the next phase, it creates the ca for us,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=428.64) [or that certificate authority,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=431.99) [and then goes through a series of commands to generate](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=433.23) [certificates for each one of the control‑plane Pods.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=436.45) [Then it goes into the kubeconfig phase,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=439.84) [generating and writing out the kubeconfig files for admin.conf,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=441.91) [kubelet.conf, controller‑manager.conf, and scheduler.conf.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=445.86) [Moving forward,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=448.84) [it then creates the static Pod manifests for each of the control‑plane Pods.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=450.19) [There, we see kube‑apiserver, kube‑controller‑manager, and kube‑scheduler.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=454.53) [It writes those all into /etc/kubernetes/manifests.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=458.53) [After the Control‑Plane Pods static Pod manifests are generated,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=463.66) [it also creates a static Pod manifest for etcd.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=466.75) [And then at that point in time,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=470.24) [once all the Control‑Plane Pods static Pod manifests are in place,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=471.5) [it goes into a wait state, waiting for all those static Pods to start up.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=474.95) [And so there, we see wait for control‑plane.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=478.69) [And then a few seconds later,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=480.52) [we see all control‑plane components are healthy after 24 seconds here,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=482.23) [in my case.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=486.63) [Moving forward into the output here at the bottom,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=487.78) [we can then see it creates the addon Pod,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=492.24) [so there we see CoreDNS is created and kube‑proxy.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=494.99) [Now, the most crucial output in all of this here is seeing the output that says,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=499.84) [Your Kubernetes control‑plane has initialized successfully.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=503.72) [And so when you see that, you know your control‑plane node is up and running.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=506.83) [Now,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=511.64) [we see some code here that describes how we can](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=511.85) [connect to our cluster as a user,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=514.31) [and that's the code that we reviewed in the presentation portion that will](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=516.42) [copy the kubeconfig file for admin.conf from /etc/kubernetes into our HOME](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=519.54) [directory and then sets the permissions on that.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=525.94) [And then moving forward, at the bottom there,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=528.17) [we can see the the join command that we'll use in an upcoming](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=530.3) [administration on how to join a node to the cluster.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=533.21) [And we see kubeadm join and then the IP address of our API server,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=536.33) [which now is populated with the correct address,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=541.01) [172.16.94.10.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=543.23) [And so, with our control‑plane node up and running,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=546.94) [let's move forward and execute those commands that we](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=549.54) [need to access our control‑plane node.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=552.32) [And on line 68 here, we can see mkdir ‑p.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=554.97) [In our HOME directory, we're going to create a hidden directory, .kube.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=558.98) [Run that code and then copy /etc/kubernetes/admin.conf into that directory that](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=563.5) [we just create it and write that into a file named config.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=571.21) [That's going to copy that admin.conf kubeconfig file into](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=574.45) [our user's HOME directory so that we can use that to](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=578.59) [connect to our Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=581.08) [And then next,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=583.64) [we're going to change the permissions on that so that](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=584.22) [our regular user can access that file.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=585.95) [With our cluster up and running, we can now deploy our Pod network.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=588.32) [We could do that with kubectl apply ‑f calico.yaml.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=592.94) [Run that code there.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=596.74) [What that's going to go ahead and do is create a collection of resources](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=598.9) [that define the Pod network in our cluster for us.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=602.4) [And so,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=605.71) [now what's happening is a bunch of Pods and resources are being created](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=605.96) [to implement that Pod network within our cluster.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=610.76) [And now, let's take a peek at what's happening inside of our cluster,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=614.54) [and we can do that with kubectl get pods,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=617.66) [and we're going to get pods across all namespaces.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=620.27) [Well, really haven't discussed what a namespace is yet,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=623.11) [but it's a way for Kubernetes to organize resources.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=625.83) [And we're going to focus on some system Pods right now,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=628.23) [which are things like to control‑plane Pods and the](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=630.96) [Calico Pods that we just deployed.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=633.65) [In the output at the bottom here,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=635.41) [we can see a collection of Pods that are in different statuses.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=637.15) [Let's review some of the output together.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=640.32) [And so we see our control‑plane Pods, etcd,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=642.64) [apiserver, controller‑manager, and scheduler.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=645.52) [Those are all up and running because we just completed](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=648.2) [that cluster initialization process.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=650.95) [We also see kube‑proxy is up and running on this node.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=653.14) [That's going to implement service networking on this individual node.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=655.79) [We then see our coredns Pods are in the status of ContainerCreating,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=660.24) [so what that means is those Pods are out pulling the](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=664.26) [container images to start those Pods apps.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=667) [And similarly, we see two calico Pods that are in the status of PodInitializer,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=669.44) [or creating the Pod, and then also ContainerCreating,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=674.1) [most likely pulling the container image down again.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=676.76) [If I use the up‑arrow and execute that command again,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=680.84) [we can see that our Pods are still in that same state.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=683.69) [And so,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=686.44) [rather than continuing to iterate and hit the up‑arrow](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=687.25) [and press Enter over and over again,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=690.42) [I can add the ‑‑watch parameter to that command,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=692.84) [and what that's going to do is output to the screen the](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=697.24) [current state of all the Pods in the system.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=700.1) [And as the status of those Pods change, it'll be updated and written to console,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=702.94) [and so way,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=706.58) [we can kind of track the deployment of the remaining Pods that](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=707.17) [aren't quite up and running yet in our cluster,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=711.11) [specifically, our Pod network Pods or calico Pods,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=713.04) [and also, those coredns Pods.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=716.1) [A few moments later, we'll find that all of the pods are now up and running,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=718.34) [including our calico Pod network Pods and also our DNS Pods.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=721.32) [We can check the status of our control‑plane node with kubectl get nodes,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=725.44) [so let's go ahead and run that code there,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=729.62) [and we'll see that our control‑plane node,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=731.54) [c1‑cp1, has a status of Ready.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=734.32) [We can see that it's roles are currently control‑plane and master,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=736.69) [and it's up and running on version v1.20.1.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=739.62) [That's the version of Kubernetes that we deployed together.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=744.34) [We have a functioning cluster up and running.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=747.24) [Now, we still have to join some nodes to the cluster,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=750.04) [but before we do that, let's check out some systemd units again.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=752.25) [In the demonstration where we installed the kubelet,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=756.94) [we saw that the kubelet service was crash looping because there were](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=760.24) [no static Pod manifests for the kubelet to start.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=763.88) [We hadn't initialized our cluster yet.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=766.17) [Well, now that we have initialized our cluster,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=768.07) [let's check out the status of the kubelet systemd service one last time,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=769.87) [and we can do that with sudo systemctl status kubelet.service.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=774.55) [And in the output at the bottom there,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=778.74) [we can see that the status of the kubelet is now active (running) because](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=780.89) [it has static Pods to start up when the service starts.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=784.94) [Let's take a peek at those static Pod manifests that](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=789.74) [live in /etc/kubernetes/manifests.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=792.25) [Run that code there, and we can see we have a static Pod manifest for etcd,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=795.74) [the apiserver, the controller‑manager, and the scheduler.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=799.56) [If we look inside each one of these static Pod manifests,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=802.65) [these are going to describe what that Pod looks like that it needs to start up.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=806.55) [And so in this case here, we have the static Pod manifest for etcd.yaml.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=810.43) [Looking inside here at this code,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=815.64) [we can see the various configuration elements for etcd.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=817.09) [Break out of this output and do the same for the](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=820.77) [static Pod manifest for the apiserver, and we can see its configuration as well.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=824.86) [And so it later on in this series of courses,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=829.43) [we'll dive deeper into the static Pod manifest](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=831.36) [configurations for the control‑plane Pods.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=834.57) [One final thing that we're going to look at in this](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=838.24) [demonstration is where the kubeconfig files live for the](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=839.94) [control‑plane Pods on the control‑plane node,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=843.14) [and those are in /etc/kubernetes.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=845.81) [And so there, we see admin.conf, controller‑manager.conf,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=848.45) [and scheduler.conf,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=851.12) [and we also see the kubeconfig file for the kubelet on the control‑plane node,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=852.86) [kubelet.conf.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=857.64) [And so, with that demo here, we have a fully‑functioning,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=860.14) [up‑and‑running cluster with a Pod network deployed.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=862.58) [Now,](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=865.96) [join me in the next demo where we'll join a worker node to the cluster for some user workload.](https://app.pluralsight.com/course-player?clipId=18a74fb8-708b-4c8f-964b-41758e7245cb&startTime=866.27)

[Demo: Adding a Node to Your Cluster](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4)

[Now it's time to join a worker node to our cluster.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=1.34) [You'll find a lot of the steps in this process are very](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=4.54) [similar to setting up a control plane node,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=7) [in terms of getting and installing software.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=9.18) [The key difference is, once everything is installed and configured,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=11.44) [we'll use kubeadm join to join the node to our existing cluster.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=14.52) [And so here I have a session open to c1‑node1,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=18.84) [and let's get started joining this node to our cluster](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=21.26) [by ensuring that swap is disabled.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=24.52) [And so on line 7 here, I have the code to do just that.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=27.11) [Swapoff ‑a, if we get no output, we know the swap is off.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=29.76) [And if we need to disable that swap, we'll hop into our fstab,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=34.04) [and ensure that our swap entry is either commented out or removed from the file,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=37.51) [and there we can see it's commented out.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=42.88) [Let's hop back out into our script here,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=45.14) [and move forward in configuring this particular node.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=48.54) [We're going to start off with installing our container runtime,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=52.79) [which is containerd,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=55.47) [which is going to require us to load the overlay in](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=56.51) [br\_netfilter modules at runtime here,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=59.69) [and also execute this heredoc to make sure that those](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=61.39) [modules load when the system boots.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=64.73) [Moving forward,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=67.12) [let's go ahead and create the required sysctl parameters for our](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=68.17) [container runtime with the heredoc on lines 31 through 35.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=71.2) [Run that code there to save that into the sysctl.d directory,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=75.35) [so that's available on reboot, and then on line 39 here,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=79.43) [we'll execute sysctl ‑‑system to load those settings](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=83.44) [now in our current running system.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=86.92) [With those prerequisites done, we can now install containerd.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=90) [So I'll issue an apt‑get update for that,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=94.91) [and then execute sudo apt‑get install containerd,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=97.84) [to install the containerd container runtime.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=100.4) [With containerd installed,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=102.9) [we'll go ahead and move forward with configuring containerd.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=106.32) [So just like we did on the control plane node,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=109.23) [we'll need to make a directory for the containerd](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=111.16) [configuration in /etc/containerd.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=114.5) [And then we'll use containerd to generate a default](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=117.34) [configuration file with sudo containerd/config default,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=119.65) [and then pipe that output to tee,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=123.64) [and then create the configuration file in /etc/containerd/config.toml.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=125.25) [Now we need to make a modification to that configuration file to ensure](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=132.04) [our container runtime is using the SystemdCgroup driver.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=135.65) [I'm going to copy the code from lines 61 and 62,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=139.28) [and then hop into the containerd configuration file,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=142.35) [config.toml,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=145.7) [and we're going to look for that line that ends in containerd.runtimes.runc,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=146.76) [and paste in this code in that location.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=152.08) [So let's go ahead and find that in our configuration file here.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=154.77) [We'll add a new line, and paste our configuration in,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=157.4) [and get that in a proper format, and write that configuration file out.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=160.67) [And with that config,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=164.82) [we will then need to restart containerd with sudo systemctl](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=166.03) [restart containerd to affect that change,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=170.76) [to make sure that we're using the SystemDCgroup driver within containerd.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=172.98) [And so now we can install our Kubernetes packages,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=178.24) [and we start with adding the gpg key for Google's app repository,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=180.41) [and then adding the app repository to our local repositories lists.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=184.79) [Once that's added, we'll then update our package list with apt‑get update,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=190.54) [and get a quick peek with apt‑cache policy kubelet to see all the](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=194.72) [different versions of the kubelet that are available.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=198.13) [And we're going to go with 1.20.1,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=200.47) [which is the version that we used on our control plane node.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=201.97) [And with that, we can go ahead and install our Kubernetes packages.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=205.31) [So I'm going to set the version environment variable,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=207.95) [just like we did previously, and then we're going to install the kubelet,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=210.54) [kubeadm, and kubectl, and we're going to pin that to the version 1.20.1‑00,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=213.65) [which is the version of the package that we want to install.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=219.96) [With the package installation finished,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=222.91) [let's use apt‑mark hold to put a hold on the kubelet kubeadm,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=225.17) [kubectl, and containerd.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=229.31) [So I'll run that code on line 92 here to do that.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=230.59) [If you want to install the latest version of Kubernetes,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=232.91) [I have the code here on lines 96 and 97 to do just that,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=235.56) [but make sure that you match the version that you're](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=239.66) [using on your control plane node.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=241.63) [Moving forward,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=243.28) [let's see the status of our kubelet with sudo systemctl status kubelet.service,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=245.82) [and we'll find that the kubelet is crash looping right now,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=251.98) [because we have not yet joined this node to our cluster.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=255.09) [We'll break out of the output at the bottom,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=259.02) [and we'll also check the service status for containerd with](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=260.63) [sudo systemctl status containerd.service.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=264.72) [We can see that the containerd service is active and running,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=267.83) [and so that's in good shape there to move forward.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=271.86) [We'll want to make sure that both the kubelet and containerd are](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=274.17) [configured to start up when the system reboots,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=278.37) [and we can do that with sudo systemctl enable kubelet.service,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=280.78) [and we'll do the same for containerd.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=285.26) [And so now it's time to join the node to the cluster.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=287.74) [We have all of the software installed.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=290.41) [We need to join the node to the cluster now,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=292.11) [and let's hop back out onto c1‑cp1, to begin that process.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=294.1) [To join a node to a cluster, you need both a Bootstrap token,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=298.78) [and also the CA cert hash.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=302.37) [You could have copied this information from the output of kubeadm init,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=304.67) [and used that information, but let's say that you didn't.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=308.16) [I'm going to show you where you can find that information in your cluster.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=310.46) [First up is the Bootstrap token.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=314.14) [The Bootstrap token is a timed ticket and has a 24‑hour lifecycle.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=315.98) [So after 24 hours, you have to generate a new one.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=319.78) [If we do a kubeadm token list,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=323.14) [it will print any active Bootstrap tokens to screen,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=325.24) [and you can see here, the only Bootstrap token,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=327.96) [and that it has one hour left, as indicated here in the TTL column.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=330.31) [If you have no Bootstrap tokens,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=334.34) [you can generate a new one with kubeadm token create.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=335.98) [And here's our new token in the output here.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=339.71) [Again, we see that warning about Docker not being installed.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=342.16) [This is safe to ignore, and it will be cleared up in future versions of kubeadm.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=345.24) [The other key piece of information that you need to join a](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=348.88) [node to a cluster is the CA cert hash, and so on line 24 here,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=353.03) [I have the code that goes and extracts the CA cert hash from](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=357.26) [actual CA certificate in /etc/kubernetes/pki.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=361.64) [If I run this code here, it'll print that information to the console.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=366.23) [So there we it begins with 073.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=369.37) [Now we could take all of this information and piece it together,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=371.79) [and make the join command with kubeadm joins,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=375.19) [specifying the Bootstrap token, and then specifying the CA cert hash,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=378.38) [or we could use the command on line 129 here,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=382.28) [to generate that join command for us.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=385.47) [And so let's do that with kubeadm token create ‑‑print‑join‑command.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=387.54) [We run that code, and there at the bottom there,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=392.84) [we'll get a well‑formed join command with the](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=395.36) [proper parameters and their values, including the location of the API server,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=397.47) [the Bootstrap token, and the CA cert hash.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=401.52) [So let's grab this text and throw that into our clipboard.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=404.54) [And so, with that join command in our clipboard,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=408.34) [let's hop back onto our individual worker node.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=411.77) [So I'm going to SSH back into c1‑node1.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=415.22) [Now that I'm on c1‑node1, you can see here,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=419.04) [I have the text for the join command.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=421.41) [I'm going to take that command that I just copied into my clipboard,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=423.59) [and paste that over this existing text,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=426.23) [because this is a previous token and a previous CA cert hash.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=428.55) [And so let's get that onto our screen here,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=431.87) [and we'll go ahead and format that so it's a little more readable.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=434.14) [And now, let's walk through this join command together.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=438.74) [We have sudo kubadm join, and then the IP address of our API server,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=441.64) [172.16.94.10, on port 6443.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=446.04) [We then have the Bootstrap token specified with ‑‑token,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=450.25) [and then the Bootstrap token that was generated when we](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=453.41) [used the print‑join‑command command.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=456.13) [We then also have the CA cert hash defined there,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=458.14) [so we see that beginning with 0735.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=461.45) [And so with that, we can then highlight this code and run it,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=463.35) [and that will then join c1‑node1 to the cluster on c1‑cp1.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=467.34) [In the output at the bottom there,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=472.13) [we could see it's executing the preflight checks,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=474.02) [and then it's waiting for the kubelet to perform the TLS Bootstrap process,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=476.09) [which we discussed during the presentation portion of the course.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=479.83) [And so now, that's all we need to do on the individual worker node.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=484.64) [Let's hop back out onto our control plane node,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=488.95) [c1‑cp1, and check the status of things.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=491.51) [If I do a kubectl get nodes now, we can see c1‑cp1 is up, running, and ready.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=494.51) [We see that c1‑node1 is not ready,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=501.44) [because what's happening right now is it's deploying the Calico pod network](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=504.19) [onto that new node that we just added to the cluster,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=507.84) [and it's also deploying the kube‑proxy pod onto c1‑node1.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=510.37) [If we do a kubectl get pods ‑‑all‑namespaces ‑‑watch,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=515.24) [we can see that it's in the midst of deploying those pods onto the](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=520.39) [new node that we just joined to the cluster.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=525.84) [Let's go ahead and break out of there, and get our console back,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=527.77) [and now we should see that c1‑node1 is ready,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=532.23) [because now the calico pod networking pods are deployed onto the node,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=537.2) [as well as the kube‑proxy.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=541.01) [And so there we can see that c1‑node1, its status is up and ready,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=542.55) [and it's running version v1.20.1.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=546.64) [Now we just installed c1‑node1 together.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=548.51) [I encourage you, the viewer,](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=554.04) [to go ahead and install c1‑node2 and c1‑node3 to](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=555.67) [complete the build of your cluster.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=559.09) [Just repeat the process of the script here in updating the](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=561) [node name as you go through the installation process to the node that you're working on.](https://app.pluralsight.com/course-player?clipId=94c391b5-f7af-4030-a6ef-4247837ca9a4&startTime=565.28)

[Managed Cloud Deployment Scenarios: AKS, EKS, and GKE](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963)

[So far we've discussed manually installing Kubernetes in virtual machines.](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=1.04) [That technique can be applied to both on‑prem virtual](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=5.12) [machines or IaaS VMs in the cloud.](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=7.81) [But what if you want to use them in service?](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=10.26) [And so I want to take a second here to walk you through some of the managed](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=12.64) [solutions that are available from the major cloud providers,](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=15.55) [and first up is Elastic Kubernetes Service,](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=18.04) [EKS.](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=20.65) [This is a managed service offering from Amazon Web Services, or AWS.](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=21.33) [Here is a link to a getting started guide so that you can go ahead and get](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=26.14) [started using Kubernetes in the cloud as a service right away.](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=29.64) [Google Kubernetes Engine, or GKE, is available from Google.](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=33.84) [And so you can go ahead and consume that service if you want to as well.](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=37.06) [And here is a link to a how‑to document to get you started there.](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=40.06) [And last but not least, of course, is Azure Kubernetes Service from Microsoft,](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=43.4) [AKS.](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=47.67) [Here is a link to a walkthrough document there at the](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=48.36) [bottom on building your own cluster in AKS.](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=50.77) [The process for deploying a cluster in a managed solution in any of](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=54.24) [these cloud providers is effectively the same.](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=57.42) [Using the command line tooling or web portals,](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=59.47) [you'll go ahead and authenticate against your cloud provider,](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=61.68) [you'll deploy a cluster,](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=64.03) [you'll then download a kubeconfig file to your local machine so that you](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=65.46) [can use kubectl with a client certificate to authenticate to the API server that's in the cloud for your cluster.](https://app.pluralsight.com/course-player?clipId=ca7d2b09-9af3-40bf-8565-a34c0dda8963&startTime=68.59)

[Demo: Creating a Cluster in the Cloud with Azure Kubernetes Service](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf)

[All right, let's get into a demo.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=1.84) [We're going to look at creating a managed service cluster in](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=3.11) [the cloud in Azure Kubernetes Service.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=5.75) [All right, so here we are logged into c1‑cp1,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=9.84) [and let's get started with the process of creating an](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=12.03) [Azure Kubernetes Service cluster.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=15.08) [The first thing that we need to do is to ensure that we](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=16.91) [have the Azure CLI client installed.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=19.18) [And so let's go ahead and walk through the steps of doing that together.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=21.22) [The first part of that process will be adding the](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=24.8) [repository to our local repositories lists.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=27.09) [And then next we will then add the GPG key for](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=30.24) [Microsoft's repository to our local keychain.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=32.73) [Once we have the repo installed and it's trusted,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=36.35) [we'll then update our package metadata with sudo apt‑get update and then](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=39.25) [install Azure CLI with sudo app‑get install azure‑cli.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=43.59) [With Azure CLI installed, it's time to log into our Azure subscription.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=48.62) [If you don't have an Azure subscription, I have a link here.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=55.04) [We can sign up and get free access to a free Azure subscription.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=58) [And so to log in, we'll use the command az login and run that code.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=62.04) [That's going to then initialize what's called a device login.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=66.64) [I'm going to log in to my Azure account off screen,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=70.75) [and in a moment I'll get the command line back, and I'll know](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=73.15) [that I'm authenticated to my Azure subscription.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=75.44) [Now that I'm successfully logged in,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=79.64) [I'm going to make sure that I'm pointing at the right subscription in Azure.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=81.17) [So I'll use az account set ‑‑subscription.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=84.42) [I'm going to point it at my Demonstration Account subscription.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=87.54) [And so now with the tooling installed and logged into](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=90.22) [our current Azure subscription,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=92.96) [I can then begin the process of deploying an AKS cluster.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=94.53) [And that starts off with creating a resource group,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=97.91) [which is a way to organize resources in Azure.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=100.84) [And we can do that with the command az group create.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=103.63) [I'm going to give it a name, Kubernetes‑Cloud,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=106.13) [and a location which is an Azure region,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=108.22) [and that's going to be centralus for our demo here.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=110.74) [So I'll run that code to create my Azure resource group.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=112.6) [The provisioningState comes back as Succeeded,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=116.04) [so we know that that was successful.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=118.03) [Moving forward,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=120.94) [let's take a look at the various versions of](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=121.62) [Kubernetes that are available to me in AKS.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=124.21) [And I can find that out with the command az aks get‑versions,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=126.87) [and then specify a location as centralus,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=130.79) [and modify that output into a table format.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=133.57) [And like we discussed in the presentation portion of the course,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=135.73) [this kind of goes into your cloud selection process](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=138.74) [because the versions of Kubernetes that are in the cloud](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=141.45) [are defined by your cloud provider.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=144.42) [And so here in the output,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=146.08) [we can see the various versions of Kubernetes that](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=147.38) [are available for us to consume.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=150.11) [And so, after that, it's then time to create our Azure Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=152.54) [Now you can do that with the command az aks create,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=157.61) [specifying our resource group that we just created,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=160.71) [Kubernetes‑Cloud.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=162.95) [We're required to have the generate‑ssh‑keys switch.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=164.14) [And then I'm going to give that cluster a name,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=167.74) [which is CSCluster, and define the node‑count as 3.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=169.07) [So,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=172.67) [we'll highlight that code and run that code together to start the](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=172.96) [deployment process of our AKS cluster in the cloud.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=176.79) [Now, I'm going to speed up the video here until the deployment is finished.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=180.44) [All right, so our deployment's finished.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=185.14) [We can see that the provisioningState is Succeeded.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=186.77) [Let's move forward in the demo.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=189.6) [If we didn't have kubectl installed on the system that we're working on,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=191.64) [we can use the command az aks install‑cli to download](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=195.14) [kubectl and install that on the local system.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=199.22) [But we're on c1‑cp1 which already has kubectl installed.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=201.55) [And so the next part of the process is going to be getting the](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=205.94) [credentials from AKS onto our local system so that we can locate](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=208.94) [and authenticate to our cluster.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=214.24) [And we can do that with az aks get‑credentials and](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=216.44) [then specifying the resource group, which is Kubernetes‑Cloud,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=219.39) [and the cluster that we just created together,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=222.38) [which is CSCluster.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=224.84) [Now we run that code there.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=226.32) [That's going to download the kubeconfig file for the cluster](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=227.82) [and then merge that into our local kubeconfig file in our](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=230.58) [current user's home directory.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=233.89) [And what that will do is give us two different cluster](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=236.24) [configuration contexts in our local system.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=238.68) [And so, to look at those we can use kubectl config get‑contexts.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=241.8) [And at the bottom here now we have two cluster contexts available to us.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=246.52) [And so we can use a cluster context to tell kubectl](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=250.38) [which cluster to send commands to.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=253.63) [And so here you can see the current cluster context,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=255.3) [as indicated by the asterisk, is our AKS cluster.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=258.23) [And that second cluster in there is our local kubeadm‑based cluster.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=261.54) [And so any commands that we execute right now will be sent to our AKS cluster.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=265.21) [That AKS cluster context was added when I downloaded the credential from AKS.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=270.24) [If we need to switch a cluster context,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=275.24) [I can use the command kubectl config use‑context,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=277.54) [and then specify the context name or the cluster name,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=281.22) [which is CSCluster for our Azure cloud.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=283.93) [But that's already been set when we merged that configuration in.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=287.26) [Now, just to make sure that we're pointing at the right cluster,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=291.14) [we can use a command like kubectl get nodes because we know](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=293.84) [that cluster topology is going to be slightly different than](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=296.69) [what we're pointing at locally.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=298.66) [And so, if I do kubectl get nodes,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=300.44) [we can see I have three worker nodes in this cluster.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=302.2) [What we don't see here in the output is the control plane node.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=306.04) [That's abstracted away for us in Azure Kubernetes Service.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=308.8) [What we do see here is just the worker nodes that](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=311.56) [are supporting our user workload.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=314.27) [And if we do a kubectl get pods ‑‑all‑namespaces,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=316.23) [we also won't see any of the control plane pods,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=320.47) [like the API server, etcd, the controller manager,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=323.42) [and so on.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=326.53) [We do see things like coredns, kube‑proxy,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=327.24) [and some additional pods that are used by AKS to report metrics.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=329.85) [When we're all done with our AKS cluster and we want to point back to our](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=334.94) [local kubeadm‑based cluster that we built together,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=338.68) [we can switch that cluster context by specifying kubectl config use‑context,](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=341.29) [and then that context name, which is kubernetes‑admin@kubernetes.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=346.67) [So we'll switch back to our local cluster, and to](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=351.22) [confirm that we'll do a kubectl get nodes.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=353.9) [And there we see our local cluster's nodes.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=356.94) [We see the control plane node and then the three worker](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=359.07) [nodes that are a member of that cluster.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=362.04) [I also have commented out here the command to delete this AKS cluster](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=364.44) [if you need to get rid of it from your subscription.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=368.12) [And you can do that with az aks delete, specifying the resource group name and then the cluster name.](https://app.pluralsight.com/course-player?clipId=f920b081-c13f-4fda-964b-87e27dd50dcf&startTime=370.4)

[Module Summary and What's Next!](https://app.pluralsight.com/course-player?clipId=eb938d60-63ec-4672-99ce-2296d2f5634a)

[All right.](https://app.pluralsight.com/course-player?clipId=eb938d60-63ec-4672-99ce-2296d2f5634a&startTime=1.34) [So here we are at the end of the module, and we certainly covered a lot.](https://app.pluralsight.com/course-player?clipId=eb938d60-63ec-4672-99ce-2296d2f5634a&startTime=1.74) [We've looked at some installation considerations and what you](https://app.pluralsight.com/course-player?clipId=eb938d60-63ec-4672-99ce-2296d2f5634a&startTime=4.55) [need to know before you install Kubernetes.](https://app.pluralsight.com/course-player?clipId=eb938d60-63ec-4672-99ce-2296d2f5634a&startTime=7.41) [Then we looked at an installation overview and talked about where to get](https://app.pluralsight.com/course-player?clipId=eb938d60-63ec-4672-99ce-2296d2f5634a&startTime=9.84) [Kubernetes and then installed our first cluster together with kubeadm.](https://app.pluralsight.com/course-player?clipId=eb938d60-63ec-4672-99ce-2296d2f5634a&startTime=13.04) [We then went on and discussed managed cloud solutions and did a](https://app.pluralsight.com/course-player?clipId=eb938d60-63ec-4672-99ce-2296d2f5634a&startTime=17.34) [deployment in Azure Kubernetes Service.](https://app.pluralsight.com/course-player?clipId=eb938d60-63ec-4672-99ce-2296d2f5634a&startTime=20.41) [Well I hope you enjoyed that module.](https://app.pluralsight.com/course-player?clipId=eb938d60-63ec-4672-99ce-2296d2f5634a&startTime=24.94) [Why don't you join me in the next module where we'll start working with our Kubernetes cluster?](https://app.pluralsight.com/course-player?clipId=eb938d60-63ec-4672-99ce-2296d2f5634a&startTime=26.51)

[Working with Your Kubernetes Cluster](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d)

[Module Overview](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d)

[Hello.](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=0.94) [This is Anthony Nocentino with Centino Systems.](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=1.37) [Welcome to my course, Kubernetes Installation and Configuration Fundamentals.](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=3.67) [This module is Working with your Kubernetes Cluster.](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=8.41) [So let's check out where we've been so far in the course.](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=12.34) [We started off with the introduction.](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=15.02) [We looked at what Kubernetes really is in the module, Exploring](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=16.69) [Kubernetes Architecture, we built our first cluster together in](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=20.49) [Installing and Configuring Kubernetes.](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=24.13) [Now it's time to sit down and roll up our sleeves and start](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=26.09) [working with our Kubernetes cluster. We're going to break down](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=29.22) [this module into two big chunks.](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=32.91) [We're going to start off with using kubectl to interact with our cluster,](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=34.68) [and then we're going to look at some basic application deployments into our cluster.](https://app.pluralsight.com/course-player?clipId=c6672018-a5c6-42f4-aa68-0e76ab4acc9d&startTime=38.51)

[Introducing and Using kubectl](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2)

[Kubectl, kube control, or kube cuddle, take your choice,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=1.04) [it's the primary CLI tool for controlling workloads in your Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=4.22) [Now,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=9.74) [what we're going to use is kubectl to perform operations against our cluster.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=10.05) [Basically, we're going to create, read, update,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=14.64) [or delete pretty much any kind of resource in Kubernetes.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=17.55) [Now, remember, in Kubernetes, everything goes through the API server,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=21.04) [and so kubectl is your primary way to interact with the API server.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=25.13) [And so any time you need to make a new thing or query something](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=29.42) [that exists or make a modification to something,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=32.75) [this is the primary CLI tool for doing that.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=35.7) [You're going to perform operations on resources, the Kubernetes API objects.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=38.54) [Using kubectl is how you're going to manipulate objects like Pods,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=43.79) [deployments, services, and others.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=47.9) [And so we're going to perform operations against resources.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=51.24) [And then finally,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=53.92) [the other facet that we're going to look at today in this module is output.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=54.86) [If there's output from the commands that we're executing,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=59.25) [then we can define the format that we're going to get that output in.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=62.01) [And so perhaps we want to have more detailed output with additional](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=65.44) [attributes exposed or we want to print output as a particular type like](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=68.29) [JSON or YAML. We can do this all with kubectl. And we're going to start](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=72.17) [our deep dive into how kubectl works with the operations that you can](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=78.59) [perform, or really, what do you want to do?](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=82.15) [And so let's jump right in and talk about the core operations that you'll](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=84.33) [likely use at the command line with kubectl every day.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=87.69) [First up is apply or create,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=91.94) [Apply and create are the primary operations for sending deployments and](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=94.74) [the creation of resources to the API server. Run allows you to start a](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=98.97) [single or bare Pod when it's not managed by a controller and then also](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=103.74) [specifying the container image at the command line, so basically,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=108.07) [bootstrapping the most basic Pod configuration. One of my favorite](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=111.44) [commands is explain.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=115.54) [This gives you the documentation for API resources,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=117.07) [and what this does is it shows you the documentation for a particular](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=120.47) [Kubernetes API object or resource listing the description and the](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=123.99) [fields needed to construct that API object.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=128.41) [This is a very valuable command when you're working at the command line.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=131.19) [Kubectl combined with delete will delete a specified resource.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=134.27) [And then there's kubectl get. What kubectl get will do is](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=138.61) [display basic information about the specified resource](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=142.63) [type. And so far in this course, we've looked at nodes and Pods in our cluster,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=146.25) [and we're going to use this plenty more as we get into more advanced topics.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=150.85) [Next,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=154.13) [let's look at describe. Kubectl describe is used to display very](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=154.75) [detailed information about a particular resource,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=158.85) [and it is extremely valuable in troubleshooting](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=161.39) [scenarios with Kubernetes resources,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=163.67) [and we'll look at it in great detail in an upcoming demo.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=165.72) [Next up is kubectl exec. Exec allows you to run a](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=169.64) [command inside a container on a Pod, and so this is very similar to docker exec.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=173.57) [And then finally, kubectl logs.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=178.74) [Kubectl logs allows you to view the logs that are written to](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=181.74) [stdout from a container running inside of a Pod.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=185.7) [And so this is very valuable for troubleshooting issues with your](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=188.67) [applications that are running inside of containers inside of Pods.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=191.61) [Now this is a short list of what I think are the most](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=194.83) [critical kubectl operations to get you started.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=197.98) [There is a huge list available to you here at the documentation](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=201.04) [page for kubectl on the Kubernetes website,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=204.44) [and I strongly encourage you to check that out there.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=207.29) [When working at the command line,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=212.04) [we're going to combine kubectl with an operation like the ones](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=213.19) [that we just introduced with a resource.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=216.6) [Basically, what do you want to perform that operation against?](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=218.64) [And we've introduced things like nodes and Pods and services. And honestly,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=222.24) [there are many,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=226.62) [many more objects available inside of Kubernetes that we can work](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=227.16) [with, and so that's how we go ahead and specify what type of resource](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=229.86) [we want to perform the operation against.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=233.31) [Now,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=235.24) [here you can see in parentheses an alias to represent that](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=235.73) [particular type of resource, so nodes, no, Pods,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=239.26) [po, services, svc, and that's a good way so that we can get real](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=241.88) [quick at the command line executing these commands.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=246.29) [Now there's a huge list of resources that are available, and you](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=249) [can certainly check that out at that link there,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=252.08) [but I'm going to show you some techniques at the command line where you](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=254.34) [can discover the resource names and the resource aliases because I want](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=257.43) [you to get very proficient at the command line without having to refer](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=261.32) [back to the documentation to get things done, and we'll do that in an](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=264.02) [upcoming demonstration together.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=267.76) [The final thing that I want to look at with kubectl today is modifying output.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=270.24) [We can specify kubectl's output format by adding](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=274.38) [additional flags to our commands.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=277.67) [The first format that I want to introduce you to](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=280.14) [today is wide. Using the wide option,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=282.35) [we can specify that we want kubectl to output additional information about our](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=284.82) [Kubernetes objects that have been deployed in our cluster.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=289.43) [We can also output our Kubernetes objects as YAML or JSON.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=292.15) [YAML and JSON formats are at the core of how Kubernetes describes things](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=296.48) [declaratively, giving us the ability to describe our configurations in code.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=301.08) [We can use kubectl to output YAML or JSON, and this is a very valuable way to](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=305.2) [get configuration data out of our cluster and describe the resources that have](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=310.72) [been deployed in our cluster.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=314.98) [We can persist this to file and exchange it with other systems,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=316.29) [down‑level environments, or developers, if we need to.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=320.11) [One final option that I want to introduce you to today here is a dry‑run.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=323.44) [When combined with the yaml output modifier, you can use this to generate](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=327.94) [YAML for resources that you want to create in your cluster.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=332.52) [But dry‑run doesn't create the object in the API server; it](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=335.67) [just out puts the YAML for the object.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=339.46) [And so this is a great tool for quickly generating YAML](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=341.77) [for resources that you want to create, so things like deployments,](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=344.85) [services, and more, and we'll see this in action later in the module.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=348.34) [If you want to dive deeper into how the output options work, check out this link here.](https://app.pluralsight.com/course-player?clipId=e557e615-669d-4a3e-8222-1c2723427fc2&startTime=352.84)

[A Closer Look at kubectl](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3)

[So let's bring all that together and learn how we can](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=1.04) [use kubectl at the command line.](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=3.84) [When we're sitting at the command line,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=5.94) [we use the command kubectl, we'll specify a command, in other words,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=7.04) [an operation,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=10.26) [right, that's the thing that we want to do. We'll specify a type or a resource,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=11.08) [right, what do we want to do it to, and then,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=14.97) [if we need to, we can specify an individual named object. And of course,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=16.96) [there's the optional flags that we can append to any command,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=20.82) [and so let's look at one command together.](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=23.5) [If we do something like kubectl get, right,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=25.71) [that's the operation, pods, that's the resource, and what if we](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=27.75) [needed to get the information about a particular pod?](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=31.23) [Well,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=33.24) [we would just append that on. So, kubectl get pods would list all of the](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=33.5) [pods in the default namespace. Kubectl get pods pod1 will give me the](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=37.33) [information about just that particular pod,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=41.79) [and then I can append on some optional flags.](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=44.28) [What if I wanted to get that pod's information output as YAML? I](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=46.36) [would append ‑‑output=yaml to get that info.](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=50.28) [One other command I want to show you guys is kubectl create, right,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=54.24) [again, the operation, what I want to create, the deployment, that's](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=57.56) [the resource. If I need to give it a name,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=60.78) [of course I would, nginx, and then I specify an image name, and that's how we](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=62.84) [could do a basic deployment at the command line using kubectl.](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=67.16) [Now,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=71.44) [I'm going to throw two links at you here, one for the reference documentation](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=71.65) [and one for a cheat sheet for using kubectl, and I strongly encourage you to](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=74.63) [check these out. They're fantastic resources,](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=78.53) [and you might find a small piece of information in there that](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=80.96) [can help you get through a scenario more efficiently or](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=83.93) [easily than you're doing it today. And so, go ahead and check out those two links, very good resources.](https://app.pluralsight.com/course-player?clipId=1ad9a4f1-422b-42a8-9f11-cadca1f2b9a3&startTime=86.21)

[Demo: Using kubectl: Nodes, Pods, API Resources and bash Auto-Completion](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1)

[So let's get into a demo where we're going to look at using kubectl.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=0.74) [We're going to use kubectl and work with some nodes,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=4.17) [pods, and other API resources, and I'm going to throw in a treat here for you.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=6.81) [I'm going to show you how to configure bash auto‑completion so](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=11.32) [you don't have to remember all the syntax and shortcuts for](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=14.47) [kubectl or even resource names.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=16.96) [Alright,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=20.14) [so here we are logged into c1‑cp1. Let's get started with](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=20.4) [working with our Kubernetes cluster using kubectl.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=23.93) [The first command that I want to show you here is kubectl](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=27.84) [cluster‑info, and this is useful for listing and inspecting which](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=30.51) [cluster you're pointing at in your current context.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=34.49) [So I'm going to highlight that code there and run kubectl cluster‑info. And](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=37.62) [at the bottom here, we can see it says Kubernetes control plane is running at](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=41.84) [https://172.16.94.10 on port 6443. That is the local API server running on](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=45.39) [c1‑cp1 in our kubeadm‑based cluster.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=54.44) [Now one of the most common operations that you'll use with](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=58.74) [kubectl is get, and so let's do that together with kubectl get](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=61.22) [nodes. And what that will do is then print out some critical](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=65.83) [information about the resource.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=69.51) [And in this case, that's going to be a node.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=71.33) [And at the bottom there,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=73.64) [let's walk through this output together. We have a row of information for](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=74.61) [each node in our cluster, and let's walk through each one.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=78.98) [We have c1‑cp1. We see its status is ready, so it's able to](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=82.22) [take on workload. In this case, on our control plane node,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=86.05) [that's going to be control plane pods.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=89.05) [We see its role is currently control plane and master.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=91.24) [It's been up for about 24 hours, and the version 1.20.1.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=93.95) [We see additional rows. For c1‑node1, 2, and 3,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=98.34) [and those all have a status of ready, meaning that they can take user workload.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=101.75) [Now, we can add the output modifier, ‑o wide, to get](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=106.34) [additional information about a resource.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=111.05) [And so, in this case here, we're going to say kubectl get nodes ‑o wide,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=113.45) [and what that will do is give me additional information about the resource, in](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=118.26) [this case, our nodes. And so, in addition to NAME, STATUS,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=122.39) [ROLES, AGE, and VERSION, we also have additional fields.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=125.62) [So we have the INTERNAL‑IP address of the node, so there we see 172.16.94.10](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=129.61) [for c1‑cp1. In some cloud scenarios, we'll see the external IP populated.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=135.1) [Since we're doing this on‑prem, that's set to none.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=140.11) [We have our OS‑IMAGE, so that's going to be Ubuntu 18.04.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=142.84) [The kernel that we're running and wrapping off the end of the screen there. We](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=146.54) [can also see the information about our container runtime.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=149.69) [And in our lab here,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=153.13) [that's containerd version 1.3.3, and we have a row of](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=154.38) [information for each node in our cluster.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=158.59) [So let's look at the pods that are currently running in our cluster.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=162.44) [And if I do kubectl get pods,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=166.17) [we get the answer of No resources found in default namespace.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=168.24) [Now remember, a namespace is a way to organize resources together.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=173.21) [And when we run kubectl get pods, that's going to](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=177.47) [point at the default namespace. And, well,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=180.51) [there's no workload up and running in the cluster yet,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=182.28) [so we have no resources found in the default namespace.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=184.91) [But there are some pods that are up and running in what's](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=188.24) [called the kube‑system namespace, and that's where the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=190.81) [system pods were run in our cluster.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=193.17) [And so I can say, kubectl get pods ‑‑namespace and then specify kube‑system.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=195.77) [And then I can see all of the system pods that are running,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=202.12) [including the control plane pods, our pod networking pods, and our DNS pods.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=206.16) [And so in the output at the bottom here, we see the name of](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=210.46) [the pod, and then next, we see ready,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=213.23) [which tells us if the containers defined in the pod are up and](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=215.52) [running. Then after that, we have status,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=218.39) [which tells us the current state of the pod.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=220.85) [Earlier, in a previous demo, when we deployed our pod network,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=223.24) [we saw the statuses container creating and pod initializing based on the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=226.2) [deployment state of the pod at that point in time and then a transition to](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=230.14) [running once everything was up running and ready.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=233.48) [Next we see restarts,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=236.24) [which is the number of times a container restarted inside of a pod,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=237.47) [and that that pod was defined about 25 hours ago. We could](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=241.58) [also combine kubectl get pods with ‑o wide, and so we can get](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=245.93) [additional information about a pod.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=251.49) [And so let's do that for our pods in our system namespace, so](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=253.84) [kubectl get pods ‑‑namespace kube‑system ‑o wide.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=256.83) [Run that code together, and we'll get additional information about a pod.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=262.05) [And so we see NAME, READY, STATUS, RESTARTS, and AGE, which is the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=266.53) [regular information that we just walked through.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=270.32) [Well, we have additional information now.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=272.74) [We have IP, NODE, NOMINATED NODE, and READINESS GATES.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=273.99) [Let's look closely at IP and NODE. So on the IPs,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=278.04) [we see that some of the pods are on the pod network 192.168, and some](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=281.74) [pods are on our virtual machines network 172.16.94.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=286.63) [Depending on the role that those pods play in our cluster,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=291.63) [that's what network they'll be attached to.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=295.19) [So, for example,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=296.74) [our DNS pods will be servicing DNS requests inside the cluster on](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=297.72) [the pod networks, so those are deployed on the pod network](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=302.01) [192.168.00, which we defined in our earlier module when we](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=304.6) [created our pod network together.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=310.15) [Some other pods are on the actual network that our](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=312.84) [infrastructure is on, so 172.16.94.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=316) [And so those are exposing services outside of the cluster.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=319.38) [So, for example, the API server is available outside of the cluster.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=322.53) [It's going to be listening on 172.16.94.10,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=326.26) [which is the real address of the control plane node.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=330.06) [Additionally, we see four kube‑proxy pods up and running.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=333.04) [There's a kube‑proxy pod running on each node in a cluster.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=337.14) [Now recall, kube‑proxy has the responsibility of implementing](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=340.29) [service networking on each individual node.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=343.41) [And so there will be a kube‑proxy pod on each individual node.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=345.94) [So there we see a kube‑proxy pod running on c1‑cp1, c1‑node1, c1‑node2,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=349.51) [and c1‑node3 on the real network IP address 172.16.94.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=354.55) [They're exposed to the real network so that they can receive those requests](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=360.71) [coming in from outside of the cluster and route that information to the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=364.05) [correct services and pods running inside the cluster.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=367.05) [Now, we can also combine kubectl with get all. And what get all will do](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=369.94) [is list all current resources that are running in a cluster, and I can](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=374.44) [also combine that with ‑‑all‑namespaces.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=378.76) [And what that will do is give me every resource that's up and](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=381.42) [running in my cluster across all namespaces.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=383.87) [And so this is a valuable command.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=386.08) [They give you a quick view of what's going on in your entire cluster's space.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=387.94) [And so the first part of the output here is pods, and we](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=392.94) [just walked through that together.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=395.13) [And so let's skip forward into the remainder of the output, and](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=396.42) [we'll see some other API resources defined in our cluster. We](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=400.34) [can see some services, daemon sets, deployments and replica sets are defined.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=403.62) [We'll be diving into each of these in much more detail in some upcoming courses.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=408.14) [But for now,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=411.96) [the key concept that I want to cover here is using kubectl to display](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=412.82) [all of the resources that are defined in our cluster.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=416.72) [Now, moving forward, let's ask the Kubernetes API server in our cluster](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=419.64) [about all the types of API objects that it knows about, and I can do](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=424.09) [that with the command kubectl api‑resources.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=427.96) [And I'm going to pipe that output into more because there is a large](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=431.24) [collection of API objects available for us to work with. Now in the output](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=433.95) [here, we see things like the name of the API object.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=438.01) [We see short names or aliases.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=441.64) [And so this is the way that we can address a particular](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=443.33) [object at the command line if we need to.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=445.89) [So, for example,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=448.13) [if you want to address nodes rather than typing the entire word](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=448.98) [nodes, we can type the alias or the short name no.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=452.03) [After the short name, we see the API version,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=455.84) [which is a way of grouping and versioning resources in the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=457.85) [API, so we see all of those are on v1. We also can see if](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=461.7) [an object is namespace or not.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=465.6) [So there we see true or false,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=467.5) [depending on if that particular object can be in a namespace or not. And then](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=468.76) [the object KIND at the end in that last column there, we see the different](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=473.16) [object names. And so, if we need to work with an alias,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=477.42) [let's get a quick,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=481.67) [simple demonstration of that, kubectl get no, and I'll still](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=482.53) [get that same output as if I typed out nodes.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=486.1) [And so, at the command line, you'll get used to using different various aliases,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=488.56) [depending on what your favorites are. Now within that huge list of API](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=492.91) [resources, one of the quick ways to be able to filter down to find what](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=498.01) [you want is to type that output in the grep.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=501.84) [And so here, we're going to look for the string pods from](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=504.04) [the output of kubectl api‑resources.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=507.39) [And again,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=510.01) [that's a quick way for me to kind of pare down that list to discover](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=510.66) [an API resource that I might want to work with.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=514.48) [Now if I need to know more about a particular API resource,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=517.54) [that's where the command explain comes in, and this becomes one of](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=520.84) [my more useful commands at the command line when I'm building](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=523.21) [workloads and constructing workloads.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=526.92) [And so let's look at kubectl explain,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=528.69) [and we're going to look at the object pod. Now we can put](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=531.18) [any object type in here from that listing of API objects](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=533.9) [that we just went through,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=537.76) [but we're going to walk through the pod example together.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=538.5) [So kubectl explain pod, and I'll pipe that into more. What](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=540.87) [explain gives you is the documentation about that API object and](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=545.03) [so that we can learn little bit more about what it takes to](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=549.37) [construct this type of object.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=552.95) [And so here we see the KIND, we see the VERSION, so KIND is](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=554.94) [Pod, VERSION is v1. This is a v1 pod.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=558.1) [We see a description that tells us what it is that we're working with. So a](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=561.74) [pod is a collection of containers that can run on a host.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=564.7) [Then it goes down to the actual fields that are required to construct a pod.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=567.84) [And so if we needed to deploy a pod,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=572.05) [this is what will be required, the apiVersion, the kind, a metadata, and](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=573.82) [a spec. If I wanted to dive a little bit deeper to learn about what in](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=577.99) [the spec is required to describe a pod, well,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=584.32) [let's look at that. I can do kubectl explain pod.spec and dive deeper](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=587.29) [into the description of a pod and learn about what I need to specify](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=592.21) [when I need to describe a pod in code.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=596.73) [And so this, again, is useful for discovering those various attributes.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=599.49) [And so if I go down in the output here,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=603.99) [you can see that one of the required fields is a container,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=605.88) [which makes sense because a pod runs containers,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=609) [and that's a very useful way to discover how to build that.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=612.04) [If I want to even go deeper, I can say kubectl explain pod.spec.containers](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=615.46) [and run that and dive deeper into that object. And here would be the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=620.94) [fields used to describe a particular container.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=624.64) [If I go down in this output here at the bottom,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=627.38) [we can see things like an image, which makes sense.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=629.25) [I'm going to run an image inside of a container to define what](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=631.7) [container image I want to be started up inside of a pod.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=635.64) [So useful information there, again,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=638) [diving into the documentation to learn about these things at the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=640.57) [command line quickly without having to go look them up either on](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=644.55) [the web or using some other resources.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=647.98) [Now this command, kubectl explain pod ‑‑recursive, is a valuable command](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=650.74) [because what it will do is it will output all of the fields that are](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=655.59) [part of an API object and march down recursively through all of the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=659.68) [fields available in the API object, in this case, a pod. And so what](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=663.24) [this will do is it will give you the output of the fields for the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=667.25) [particular API object, but without the description.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=670.62) [So if, perhaps, you forgot a particular field name,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=673.97) [this is a valuable way to go and just retrieve that information](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=676.32) [quickly. So let's break out of that output here and look at one of](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=678.93) [my other favorite commands that I use very,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=682.15) [very frequently at the command line when I'm working with resources](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=684.3) [defined in my cluster, and that's kubectl describe. And so on line 60](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=687.49) [here, I have kubectl describe. I'm going to describe what a node and](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=691.14) [then a particular node, c1‑cp1.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=695.05) [Now, I'm describing a node here,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=697.02) [but we could describe any other resource. It could](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=698.51) [be a pod, it could be a service.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=700.38) [We're going to focus on nodes at this point in time.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=701.99) [Kubectl describe nodes c1‑cp1.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=705.04) [Let's go and pipe that into more and walk through this output together.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=707.38) [So, describe gives you some very detailed information about an API object,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=711.34) [and this is extremely valuable when it comes to](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=715.59) [troubleshooting things that are running in your cluster.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=718.05) [And so in the output at the bottom here, we see the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=720.94) [name of the resource, in this case, it's a node,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=723.06) [which is c1‑cp1. The Roles, control‑plane, master, and we have some labels](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=725.08) [and annotations. Labels and annotations are a way for Kubernetes to track and](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=729.83) [monitor objects that are running in the cluster,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=734.69) [and we'll look at those in much more detail in an upcoming course.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=737.14) [In addition to that information,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=741.14) [we have the creation timestamp when this API object was created.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=743.26) [There we see the taint associated with the control plane node.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=746.52) [That's going to have the taint of NoSchedule.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=749.9) [This is the taint that prevents user pods from running on this node](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=752.03) [and allows only system pods to run on this node.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=756.45) [Moving forward in the output,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=760.44) [we see conditions, which describe the current state of the node](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=761.76) [in terms of things like network availability,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=765.12) [MemoryPressure,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=767.56) [DiskPressure, and PIDPressure. Moving forward in the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=768.41) [output, we also see things like addresses.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=772.28) [There's internal IP and then a hostname, the capacity of the node, so the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=774.47) [amount of CPU that it's contributing to the cluster,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=779.14) [its storage, memory, and so on.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=781.8) [We have some additional system information with regards to](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=784.94) [things like the kernel version, OS image,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=787.67) [the operating system that it's running, CPU architecture, and much more,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=790) [so very valuable deep‑dive information about that.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=793.22) [We have the current pods that are up and running on the node,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=796.84) [and so there we can see the collection of system pods that](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=799.5) [are running on the control plane node.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=802.29) [And so that was kubectl describe. For our control plane node c1‑cp1, we](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=805.04) [can do the same for c1‑node1 and dive deeper into this particular node's](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=809.97) [configuration at the command line here.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=815.35) [So there we can see things like its conditions and](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=817.86) [its status, if it's up and ready,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=820.29) [and all of the resources that are available on it, and so on.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=822.14) [So at the command line,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=826.54) [what we have the ability to do is interact with the API server. And one of](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=827.9) [the things that we want to be able to do when we're working with Kubernetes](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=833) [is to do these things quickly and discover these things at the command line,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=835.64) [so that's one of the reasons why we walk through how to retrieve API objects](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=839.18) [and their documentation because I want to be able to do that quickly at the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=843.04) [command line.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=846.31) [Similarly,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=846.77) [if I'm working with kubectl, I can just ask kubectl for help,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=847.46) [and I can do that with kubectl ‑h. And then in the output, it](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=851.55) [has a well‑formed organized method of exposing the various](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=855.11) [operations that you can perform.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=858.99) [So there we see basic commands like create, expose, and run, and](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=860.62) [then going down into more advanced commands for things like](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=865.12) [deployments and also cluster management. So very valuable](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=867.93) [information is available in the help.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=871.37) [In addition to the basic output right off of kubectl ‑h, I can](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=874.34) [combine that with an operation, so kubectl get ‑h, and then I](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=878.37) [can get more detailed information about a particular operation](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=883.06) [that I want to execute.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=886.77) [In this case, it's get. And so looking at this,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=887.54) [in addition to the normal help that you would see,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=890.72) [we also get a collection of very valuable examples.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=893.91) [These examples here are very useful for helping you remember](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=896.68) [more advanced command lines and texts.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=899.18) [The final operation that I want to call out here in the help is kubectl create.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=902.04) [This is a command used to create resources in the cluster imperatively,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=906.15) [and it's something that we'll be doing very frequently. And as you're](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=909.83) [working through future demos, be sure to remember that this is here for you](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=912.81) [as a resource. And now, last in the demo, but certainly not least, here's](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=916.14) [that treat that I want to share with you.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=921.01) [I want to show you that enable bash auto completion for kubectl.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=922.31) [And so, let's walk through that process together.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=926.45) [On line 71, I have apt‑get install ‑y bash‑completion, and that's](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=928.94) [going to install bash completion on our system.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=933.74) [And on my particular system here, we can see that it was already installed.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=936.58) [I'm then going to, on line 72, echo in some configuration information into](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=941.04) [my local bashrc. I'm going to reread that with the source command, and then](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=947.15) [I'm going to show you at the command line at the bottom here what bash auto](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=952) [completion provides. Now, normal bash completion would do something like](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=955.34) [this where it would complete the command based on what you've typed at the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=958.68) [command line.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=962.52) [So I type kube, I hit double tab, I get the auto completion of](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=963.02) [the three commands that match that string.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=966.56) [In that case,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=968.63) [kubectl is the one that we want. Now, to extend that, we get auto](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=969.26) [completion in kubectl for operations. So here you can see if I type g](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=973.79) [and hit double tab now, it auto completes to get. In addition to](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=978.19) [operations, it also does that for resources.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=982.13) [So if I type po and then double tab, you can see it'll auto complete the pod.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=984.79) [If I double tab again,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=989.66) [you'll see it auto completes to all of the different resources that are](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=990.77) [available in my API server that match that string, in this case, pods. And](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=994.62) [so we'll go ahead type pods there. Also, I can then add something like ‑‑ to](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=998.87) [see any of the different modifiers that are available to me at the command](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=1004.16) [line, so if I double tab now,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=1007.45) [we can see all the various different modifiers that are available to me at](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=1008.85) [this point in the command line string that I'm building.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=1013.06) [Let's go ahead and type something like all,](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=1015.76) [and then I'll do a double tab on that, and we can see it's going to limit the](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=1017.84) [list of modifiers down to the modifiers that match the string ‑‑all. And so I'll](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=1020.67) [go ahead and auto complete that with ‑namespaces.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=1025.38) [Execute that code there.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=1029.34) [We can see how we can use that to quickly work at the command line](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=1030.72) [to execute commands and discover commands and discover resources that are available in our cluster.](https://app.pluralsight.com/course-player?clipId=310d3eec-367c-4aa3-8343-9b7b669fcbb1&startTime=1034.09)

[Application and Pod Deployment in Kubernetes and Working with YAML Manifests](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97)

[So, now that we know how to interact with our cluster at the command line,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=1.04) [let's move the conversation to application deployment in Kubernetes.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=4.05) [Now, the first thing I want to discuss is imperative configuration.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=7.57) [When you're using imperative configuration,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=11.18) [you're generally going to be executing commands at](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=13.26) [the command line one at a time,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=15.8) [and you're going to be operating on one object at a time.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=18.41) [So, for example, if I want to create a deployment,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=21.71) [I can use kubectl create deployment nginx,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=24.89) [and specify the image as nginx.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=28.53) [I punch this in at the command line, I hit Enter,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=30.46) [the command is sent to the API server, and the object is created,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=33.04) [but I'm only operating on one object at a time on the command line.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=35.98) [Now, similarly, I can operate on other types of objects certainly.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=40.59) [I can say kubectl run nginx, and it will create a pod for me running nginx.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=43.35) [That's a fine way to manage a system,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=48.09) [but if your application stack starts to grow and your](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=50.15) [configurations and your deployments become more complex,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=53.51) [managing each individual object at the command line isn't really a](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=56.25) [sustainable way to manage or maintain your system.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=60.6) [Were going to want to do things declaratively,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=63.55) [and this is a core principle behind kubernetes,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=65.43) [where we define our desired state in code.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=68.83) [Earlier in the course, we introduced the concept of manifests,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=72.28) [first when we deployed our pod network,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=75.33) [and second when we bootstrapped our cluster with static pod manifests.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=77.29) [Those manifests described those configurations in code,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=80.84) [and Kubernetes was able to use them and bring the system up to the](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=84.74) [desired state as described in those manifest files.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=88.44) [And we can do the same for our own applications.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=92.5) [We can define our configurations in code using](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=95.4) [manifests written in YAML or JSON,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=97.54) [and feed those into the API server with commands like kubectl apply.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=99.88) [In this case here, you can see kubectl apply ‑f deployment.yaml,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=104.64) [and the contents of deployment.yaml will have the description of the](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=109.76) [thing that I want to deploy inside of Kubernetes.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=113.56) [Let's look at our first manifest together,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=117.94) [and we're going to look at a deployment manifest.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=120.07) [We could have manifests for many different types of](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=122.74) [API objects available in Kubernetes,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=125.08) [but we're going to start our conversation off with deployments.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=127.11) [The first thing that you're going to find in any manifest is the apiVersion,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=130.84) [and in this case here, since we're using a deployment,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=134.82) [it's going to be apps/v1.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=137.18) [As the Kubernetes API develops and changes to give users stability](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=139.14) [as to how API objects are defined and behave,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=143.76) [it's versioned.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=146.85) [This way,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=147.79) [we know a v1 deployment will always look and behave a](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=148.72) [particular way when we call that in our code.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=151.78) [If a newer version of an API is released,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=154.64) [and the API objects either change in definition or behavior,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=157.33) [that can be a breaking change, and so in our manifest we specify the API version,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=161.27) [and this gives us control over which version of an API object we're consuming,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=166.1) [and thus adding stability to the objects defined in our manifests.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=171.16) [The next thing that you're going to find in a manifest is kind,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=175.74) [or the kind of object that we want to define.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=178.44) [In this case, we're defining a deployment.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=180.99) [Remember in an earlier demo when we did kubectl api resources](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=183.57) [to list all of the API resources available,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=187.72) [those are the objects that we can use here.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=190.03) [We'll need some metadata to describe what we're working with here,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=192.74) [and in our case, we'll just give our deployment a simple name,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=195.78) [and we're going to call it hello‑world.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=198.55) [Next is the spec.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=200.2) [This defines the implementation details of the deployment object.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=201.82) [And the first thing that we're going to define is the number of replicas.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=205.73) [This is the number of pods that we want up and running for this deployment,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=209.44) [and we're going to start off with 1.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=212.52) [The selector is a way for a deployment to know which](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=214.34) [pods are a member of this deployment,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=217.34) [and we'll be diving into labels and selectors in much](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=219.48) [more detail in an upcoming course.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=222.26) [And then there's the template.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=224.54) [This is the section that's used to define the pods created by this deployment.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=225.88) [You'll also hear this called the pod template.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=230.54) [In here, we'll have another metadata section, and we'll define some labels.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=233.14) [These are matched with a selector in the deployment spec above,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=237.4) [and these are assigned to each pod created by this deployment,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=240.63) [and that's how the deployment is able to track which](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=243.93) [pods are a member of this deployment.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=246.57) [And then finally, there's another spec.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=248.94) [Here's where we will define the containers started](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=250.69) [by the pods in this deployment, and we define an image to do that.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=253.17) [Here we're going to be using a simple hello‑world](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=256.9) [application from the Google Container Registry,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=259.38) [and we'll also need to give this container a name,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=261.62) [and we're going to call it hello‑app.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=264.62) [Now, to deploy this deployment,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=266.34) [we can save this information into a file named deployment.yaml,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=268.49) [and then we'll use kubectl apply ‑f,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=271.87) [and then pass in that file name as a parameter.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=275.01) [That will then read the file and feed that manifest into the API server,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=277.61) [and then the API server will go and make that happen for us,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=281.94) [affecting the desired state of the application in the cluster,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=284.77) [and in this case,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=288.47) [it's starting up one replica of our hello‑world application pod.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=289.55) [Now you might be thinking, how am I going to remember all of this?](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=293.54) [Well, to build YAML manifests for deployments,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=296.26) [or really any objects that I want to work with,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=299.13) [well, remember in the last demo when we talked about kubectl explain,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=301.55) [you can use kubectl explain to quickly look up the fields for an object to help](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=305.76) [you fill out the implementation details for your manifests.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=310.91) [Additionally, we had previously discussed the dry‑run output modifier.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=314.54) [We can use that to generate basic manifests like this very quickly.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=318.35) [Let's look at dry‑run in more detail now.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=323.14) [Now,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=327.54) [we can certainly write the YAML for our deployment manifest by hand and ensure](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=327.94) [that we get all of the fields and all the spacing right,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=331.95) [but what I want to show you now is a way to generate the](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=334.14) [YAML needed to create a deployment, or really any API object,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=337.07) [on the fly at the command line quickly and correctly.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=340.99) [A few slides back,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=344.64) [I showed you how to create a deployment imperatively at the command line.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=345.6) [We can take that code and its parameters and combine](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=349.54) [it with ‑‑dry‑run=client ‑o yaml,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=352.45) [and what this will do is generate the YAML for the](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=355.8) [API object that you want to create,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=359.25) [but it won't send it to the API server for creation.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=361.08) [What dry‑run client ‑o yaml will do is write the object YAML to stdout.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=364.34) [We can combine that with file redirection and write the output into a file,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=370.37) [and so here you can see we're sending the output](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=374.44) [into a file named deployment.yaml.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=376.63) [With this YAML written into the deployment.yaml file,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=379.52) [we can use kubectl to send that deployment into our API server](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=383.14) [for creation or use this output as a starting template for a](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=387) [more complex deployment scenario.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=391.19) [I can't stress enough how helpful the dry‑run parameter can be to](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=393.84) [quickly generate correct YAML representation of objects,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=398.2) [nearly any object, not just deployments.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=402.01) [So I encourage you to use it frequently at the command line](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=404.43) [to help you get proficient at generating manifests quickly](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=407.32) [and correctly at the command line.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=410.32) [Now it's time to discuss what the API server is actually](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=414.44) [doing for us when creating a deployment,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=417.83) [and we're going to discuss the application deployment process in Kubernetes.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=419.78) [And at this point,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=423.64) [we're going to bring together a lot of the theory and concepts that](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=424.5) [we've been going through throughout this course,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=427.63) [and so really,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=429.48) [this is the time when we're going to bring all of those elements together.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=430.53) [And so let's say we have a cluster and we're sitting at the](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=433.74) [command line with kubectl and we want to deploy an application](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=436.57) [into Kubernetes and we say kubectl apply.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=440.41) [We then pass in some sort of manifest describing the](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=443.02) [objects that we want to create.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=446.11) [So let's say we want to create a deployment.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=447.35) [That deployment will create a replica set,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=449.39) [and that replicas set will create our pods based on](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=451.84) [the pod spec in the template,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=454.74) [and kubectl is going to send our request into the API server.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=456.48) [The API server is going to parse that information defined in the](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=461.23) [manifest and store those objects persistently in etcd.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=464.87) [The controller manager is watching for any new](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=468.99) [objects that it needs to know about.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=472.46) [Since we define a deployment,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=474.74) [it's going to start up a controller for that deployment,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=476.38) [and that will create a replica set.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=478.67) [That replica set is going to create the required number of](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=480.74) [pods to support the configuration and write that information](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=483.79) [about the pods back to etcd.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=487.15) [Now,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=489.51) [the scheduler is watching etcd to see if etcd has any pods](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=489.86) [that haven't been assigned to nodes yet,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=494.58) [and if it finds any unscheduled pods, it will schedule,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=496.54) [or in other words, assign a pod to run on a particular node in a cluster.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=500.34) [Each pod object is updated in etcd with the](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=505.14) [assigned node that it needs to run on.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=507.99) [Now, no pods have started yet.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=510.67) [What we have is the objects for the deployment,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=513.44) [the replica set,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=516.41) [and the pods with their scheduled node information all written into etcd.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=517.79) [So how do the pods actually start?](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=522.94) [Well, the kubelets on the nodes are watching the API server asking,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=525.14) [do you have any work?](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=529.34) [Do you have any work?](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=530.53) [And if it finds a pod scheduled for that node,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=531.62) [it's then going to send a message to the container runtime](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=535.22) [on that node to pull down the appropriate container images](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=538.46) [specified in that pod spec, and start the pod up on that node.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=542.4) [If that pod is a member of a service,](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=546.74) [then that service's information is updated in kube‑proxy on that node.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=549.32) [This entire process is how pods get deployed inside of Kubernetes.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=553.86) [In this example, we described a deployment.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=558.4) [Similar processes exist for deploying other types of objects in](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=561) [Kubernetes where the workflow might vary slightly.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=564.52) [I just wanted to highlight this example for you today to](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=567.24) [connect the dots of all the various things that we've discussed throughout this course.](https://app.pluralsight.com/course-player?clipId=633aa541-cd47-4007-a089-1b4f6a8e8a97&startTime=569.95)

[Demo: Imperative Deployments and Working with Resources in Your Cluster](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c)

[Alright, so let's get into a demo, and in this demo,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=1.44) [we're going to deploy some resources using two different techniques.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=4.1) [We're going to look at things imperatively and declaratively.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=7.21) [Using those techniques, we'll learn how to deploy resources into our cluster,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=10.08) [focusing on deploying deployments, pods,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=14.19) [and services, and once we have things up and running,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=17.02) [we'll look at how we can make changes to existing resources in our cluster,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=19.79) [using both declarative and imperative techniques.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=23.09) [So here we are, logged into c1‑cp1,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=28.14) [and let's begin the process of deploying resources imperatively in our cluster.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=29.84) [What we're going to do first here is to create a deployment together,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=35.04) [and I have the code on line 8 to do just that.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=37.64) [And so on line 8, we see kubectl create.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=40.54) [What do we want to create?](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=42.95) [A deployment.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=44.08) [where we'd give that deployment a name, hello‑world.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=45.06) [And then we're going to specify the container image that we want](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=47.27) [to run in the deployment with the parameter,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=50.39) [‑‑image.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=52.95) [And so here for the image,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=54.34) [we're using a simple hello‑world app container image](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=55.37) [from Google Container Registry, hello‑app,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=58.3) [with a tag of 1.0.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=60.89) [And so when I highlight that code and run that there,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=62.69) [and at the bottom we can see deployment.apps/hello‑world created.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=65.42) [And so what this code is going to do is create a deployment,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=69.84) [which will create a ReplicaSet with one pod in it.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=72.62) [So it creates a one‑replica deployment.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=75.88) [Moving forward, let's go ahead and create a bare pod,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=78.34) [or a pod that's not managed by a controller.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=82.13) [And to create a bare pod, we use the command kubectl run,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=85.54) [and then we're going to give it a name, hello‑world‑pod,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=89.04) [and then we need to specify the container image that](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=91.03) [we want to run inside of that pod, and we do that again with the parameter,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=93.68) [‑‑image,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=97.54) [and we're going to use our simple hello‑world application again for this pod.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=98.18) [We'll run that code, and we can see here at the bottom,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=102.94) [pod/hello‑world‑pod created.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=105.4) [And so let's check out the status of our deployment,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=108.84) [and our bare pod, and I want to see if both of those pods are up and running,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=111.79) [and I can do that with kubectl get pods.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=115.68) [In the listing here, I have our pod that's associated with our deployment,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=118.58) [that's the first one in the list there, and then our bare pod,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=122.31) [which is hello‑world‑pod, which is the second one in the list there.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=125.29) [We can see that both of those pods' status is running.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=128.75) [Looking at the pod that's associated with our deployment,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=131.69) [we can see in the name of the pod, the name of the deployment,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=134.34) [which is hello‑world, and then we see the string,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=137.49) [5457b44555.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=139.69) [That's what's called the pod template hash,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=143.04) [and is unique amongst ReplicaSets within a deployment.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=144.83) [The last part of the pod name there is a unique](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=148.54) [identifier for a pod within a ReplicaSet,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=150.55) [so gnxsk is a unique name.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=153.72) [And so all of that together, with the deployment name,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=157.17) [hello‑world, the pod template hash, 5457b,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=159.92) [and so on, plus the pod unique identifier,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=162.07) [gnxsk, will give us the unique value for the pod name.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=165.07) [So let's look at these pods from a different angle.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=169.72) [Let's use kubectl get pods ‑o wide,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=172.4) [to display some additional information about the](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=175.73) [pods that we have up and running, and I want to zoom in on IP and NODE.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=178.59) [So we can see both of these pods have IPs that have](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=182.72) [been allocated from the pod network, 192.168.0.0/16.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=186.15) [We can also see the nodes that these individual pods have been scheduled to.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=191.94) [So the pod associated with our deployment,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=195.63) [the first one there, was scheduled to c1‑node3,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=197.32) [and then our bare pod was scheduled to c1‑node2.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=200.14) [Now I want to point out that Kubernetes is a container orchestrator.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=204.54) [It has the job of starting containers on nodes,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=208.65) [and we can see that we have two pods up and running on two different nodes,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=211.37) [and while those pods on those nodes started containers.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=215.47) [And so what I want to do now is I want to SSH into an individual node,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=219.46) [and show you how you can view the containers running on that node,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=225.14) [even the ones that have been started by Kubernetes.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=229.2) [And so we're going to go ahead and open an SSH connection into c1‑node3,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=231.94) [so I want to copy and paste this code from the top,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=235.06) [down to the bottom, and SSH into c1‑node3.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=237.67) [That's the node where our pod that's running from our deployment,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=241.74) [was scheduled to and is now running on.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=245.24) [The way that we can get a listing of the containers running on a node that are](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=247.73) [running with container D is to use the utility called crictl.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=252.72) [And so I have the code to do just that on line 30 here.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=256.01) [So we have sudo crictl, and then the parameter, ‑‑runtime‑endpoint.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=259.43) [And then we specify the cri socket for containerd,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=265.44) [and then the command that we want to run there at the end,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=268.89) [ps.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=271.27) [And so if I highlight that code and run it,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=272.64) [we can see that there are three containers running on this node in our cluster,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=274.1) [the first one being our hello‑app, which is part of our deployment.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=278.88) [There are two other containers running on this node,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=283.14) [one for our pod network, there we see calico‑node,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=285.36) [and we also see a container named kube‑proxy,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=288.18) [which is supporting our kube proxy pod, which is also running on this node.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=290.26) [If you are still running Docker, here is the command to do just the same.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=295.94) [If you aren't running containerd, you can do sudo docker ps,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=299.65) [to get a listing of the containers that are running on your node.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=302.55) [So let's go ahead and exit back out onto c1‑cp1,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=306.24) [and look at things from a couple other different angles.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=311.36) [I want to show you some troubleshooting techniques that](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=315.63) [might be useful for working with your pods,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=317.77) [first up, using kubectl logs.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=319.81) [And so on line 41 here, I have the command kubectl logs hello‑world‑pod.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=322.15) [We can use the command kubectl logs,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=326.27) [which is very valuable to retrieve the logs from a container](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=328.67) [running inside of a pod in our cluster.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=332.3) [And so any information written to standard out will be](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=334.12) [captured and available to you via kubectl logs.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=337.15) [And so this is valuable when you have an application that's in trouble,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=340.06) [or crashing, or a pod that won't start,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=343.29) [so a very useful troubleshooting technique here.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=345.24) [And so for our scenario, I have kubectl logs, and then a pod name.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=347.94) [In this case, it's going to be hello‑world‑pod.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=352.07) [So if I highlight that code, and run that,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=354.42) [we can see the log from our container inside of our pod.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=356.56) [In this case here, we have just one entry,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=359.52) [which includes a date and timestamp,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=361.1) [and then a string that says server listening on port 8080.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=362.81) [Moving forward.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=366.74) [I also want to show you that we can attach a shell to a running pod.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=367.5) [And to do that, we can use a technique called kubectl exec.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=373.14) [And I want to point out that you can use kubectl exec to](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=376.27) [start a process inside of a container, inside of a pod,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=379.21) [and you can use this to launch any process,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=382.81) [as long as that executable is available inside the container.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=385.24) [In this scenario, we're going to attach a shell,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=388.44) [so that I can show you how to have a shell to a](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=390.26) [container running inside of a pod.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=392.52) [And so let's walk through that technique together.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=394.14) [On line 47, I have kubectl exec ‑it.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=397.45) [The parameter ‑it will allow you to attach an interactive terminal.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=400.53) [We then specify the name of the pod that we want to attach it to.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=404.7) [In this case, it's hello‑world‑pod.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=407.26) [Then we have space minus minus, and then a space,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=409.44) [which is a delimiter, and then after that,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=411.54) [you specify the command that you want to run,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=413.54) [in this case, /bin/sh, which is a shell.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=415.52) [And so when I run that code,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=419.28) [it's going to give me an interactive shell to the](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=420.84) [container running inside of that pod, and at the bottom here,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=423.61) [you can see I have a root shell open.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=426.22) [We can now execute any command that exists inside of the container.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=428.22) [And so the first thing I want to show you is hostname.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=432.68) [When I execute hostname, it'll print out the hostname of the pod stdout.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=435.65) [And so here you can see the hostname, which by default will match the pod name,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=439.71) [in this case, it's going to be hello‑world‑pod.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=443.24) [If I do ip space addr,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=445.33) [I can look at the network configuration of this individual container,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=448.25) [running inside of this pod.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=452.36) [Then we can see in the output, the IP of this pod is 192.168.131.62.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=453.82) [When we're all done, we can use exit to exit out of the container,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=458.93) [and get back onto our localhost, which is c1‑cp1.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=464.77) [Now,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=468.18) [remember that first kubectl create deployment](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=469.79) [command that we executed in this demo?](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=472.56) [What that did is it created a deployment for us,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=474.56) [which then created a ReplicaSet, which then created a pod,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=477.1) [and I want to show you how that all pieces together here inside of our cluster.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=479.68) [And so we'll first off do kubectl get deployment hello‑world,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=484.36) [and that's going to list the information about our deployment,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=488.64) [hello‑world.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=491.85) [And so at the bottom here, we see the name,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=492.48) [hello‑world, we see that one of one pods is ready,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=494.19) [and that our deployment was created just under 9 minutes ago.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=496.59) [If I do a kubectl get replicaset,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=499.94) [we can see the ReplicaSet supporting the deployment.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=501.66) [So there we see hello‑world, and then the pod template hash, 5457b, and so on.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=505.17) [We also can see the desired state and current state,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=510.44) [and that one pod is up running, and ready,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=513.32) [in our ReplicaSet.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=515.79) [Now, if I do a kubectl get pods,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=518.04) [we can see the pods that are up and running in the cluster,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=519.67) [and now you can see where that name is built from,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=522.08) [the deployment name, hello‑world, then the pod template hash,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=524.47) [and then that pod's unique identifier, which is gnxsk.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=527.64) [So let's look a little more closely at the deployment,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=532.34) [at its ReplicaSet, and its pods,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=535.38) [and we'll use one of my favorite Kubernetes commands,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=537.87) [kubectl describe, to do that.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=540.48) [And so on line 64 here, I have kubectl describe deployment hello‑world.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=542.44) [I'm going to pipe that output into more,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=546.5) [and here at the bottom we can see the the output of](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=548.25) [kubectl describe for our deployment.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=550.15) [So we can see the name, hello‑world, and the namespace default,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=552.84) [and that it was created just a few minutes ago.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=555.88) [Moving forward in this output, here we see pod template,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=558.1) [which is the pod template for each of the pods that](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=561.51) [is created by this deployment.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=564.27) [So inside of there, we can see containers,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=566.14) [hello‑app, and then the container image that we're working with,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=568.2) [our hello‑app from gcr.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=570.61) [Going down to Events,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=572.34) [we can see in Events that this deployment has an](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=574.25) [event that says ScalingReplicaSet, because the deployment creates the ReplicaSet,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=578.24) [and then a ReplicaSet will go on and create the pods.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=583.24) [So in the Events output at the bottom,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=585.12) [we see ScalingReplicaSet from deployment‑controller,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=587.2) [Scaled up ReplicaSet hello‑world‑5457b44555 to 1.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=589.88) [So that's creating that ReplicaSet,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=596.05) [which will in turn create that one pod for us.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=598.14) [Let's go ahead and look more closely at the ReplicaSet,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=600.79) [and we can do that with kubectl describe replicaset hello‑world.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=602.95) [In this output here, we see the name, hello‑world,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=607.06) [and then the pod template hash.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=609.37) [In Replicas, we can see the current state is 1,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=610.98) [and desired is 1, and right below that, we see Pod Status one 1 is running.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=613.51) [So we know currently we are in the desired state for this ReplicaSet.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=617.27) [Moving forward, we see the pod template for this ReplicaSet.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=621.44) [This matches the current state that is in our deployment.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=624.09) [So there we see our containers, hello‑app,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=627.23) [running the container image from gcr, hello‑app with the tag of 1.0.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=629.35) [In the Events section for the ReplicaSet,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=633.93) [we can see an event that created a pod, and so let's look at that output.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=635.93) [We see SuccessfulCreate from replicaset‑controller,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=640.21) [Created pod: hello‑world, and then our pod template hash,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=642.93) [and then the pod identifier ending in gnxsk.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=646.47) [Now let's go a level deeper and get kubectl describe](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=651.44) [information about our running pod.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=654.36) [I want to show you a technique of how we can very quickly auto‑complete a](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=656.44) [pod name at the command line with Bash auto‑completion.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=660.4) [So I'm going to take the highlighted text here on line 75,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=663.15) [and thread it into my clipboard, and in the command line at the bottom,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=665.99) [I'm going to paste that,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=669.43) [and here we have kubectl describe pod hello‑world and a dash.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=670.63) [And so if I do a double tab now,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=674.26) [it'll try to auto‑complete to any pod that has the](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=675.96) [name that begins with hello‑world‑.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=678.81) [And so we have two, right?](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=681.84) [We have the one associated with our deployment,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=682.88) [and our bare pod,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=684.65) [and so I'll just type a 5 there to give it enough](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=685.5) [information to auto‑complete to the full pod name.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=687.95) [By hitting tab again, we can see it auto‑completes the full pod name for us.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=690.76) [And so we have kubectl describe pod, and then our pod name,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=694.7) [hello‑world‑5457b, and so on.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=698.8) [I'm going to pipe that output into more,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=701.18) [and here we can see the information associated with an individual pod.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=703.2) [So there we have the pod name.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=706.96) [We also can see the node that the pod is running on,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=708.84) [so c1‑node3, and its node IP address.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=711.05) [Going down a little bit further, we can see the pod IP address.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=714.72) [There we have 192.168.206.127.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=718.07) [We can see that this pod is controlled by the ReplicaSet,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=722.64) [hello‑world‑5457b, and so on,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=726.01) [and then we can see the runtime information about the container,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=728) [running inside of this pod.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=731.2) [Going down a little bit further, I want to jump to the Events section,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=733.2) [and in the Events section,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=737.54) [we have some really good information about the lifecycle of this pod,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=738.31) [and so let's walk through each one of these records.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=741.97) [First up, we see Scheduled from the default‑scheduler,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=745.04) [and the message is, Successfully assigned default,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=747.92) [which is the namespace/, and then that's our pod name,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=750.8) [hello‑world‑5457b, ending in gnxsk to c1‑node3.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=754.99) [So that's when the scheduling decision was made,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=758.93) [and that's part of this pod's events.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=761.89) [On the next line, we see Container image,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=764.74) [and there's our container image, hello‑app,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=767.32) [with the tag of 1.0.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=769.52) [We can see that it was already present on this machine,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=770.62) [and so we didn't have to re‑pull that down from the](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=773.35) [container registry; it already existed.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=776.08) [So then we can jump right to creating the container image,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=777.99) [and that's the third event in our output here,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=780.77) [Created container hello‑app.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=782.82) [The last record we have in the Events is, Started container hello‑app.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=784.94) [And so that process of scheduling, pulling,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=788.88) [creating, and starting, is all part of this pod's lifecycle.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=792.71) [If you create a deployment and your pods don't come up,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=796.68) [use kubectl describe on either the deployment,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=799.81) [the ReplicaSet, or one of the pods in your deployment,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=802.8) [and check out the events for those resources,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=805.39) [looking for any events or errors that can help you understand what went wrong.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=808.17) [This is my go‑to place to help me troubleshoot deployment](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=812.64) [failures and pods failing to start up.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=815.6) [Now, if you want to dive deeper into deployments,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=818.38) [check out my course, Managing Kubernetes Controllers and Deployments,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=820.75) [coming up later in this path.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=824.18) [I have a link there for that.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=825.48) [In that course, we'll talk about rollouts,](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=827.04) [controlling rollouts, rollbacks, and updating our application, so deep‑dive stuff covered in that course.](https://app.pluralsight.com/course-player?clipId=5e4f2bca-4ec2-4546-a24c-855063d35b3c&startTime=829.12)

[Demo: Exposing and Accessing Services in Your Cluster](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8)

[All right,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=0.84) [so let's move forward into the next part of our demo where we're going](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=1.18) [to learn how to expose our deployment as a service and so we can access](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=3.97) [our application inside of our cluster.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=8.11) [Now, to create a service for our deployment,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=10.84) [we can do that with the code on line 89 here,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=12.97) [kubectl expose deployment, and then the deployment name,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=15.64) [hello‑world \.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=18.74) [The ‑‑port parameter is the port that the service is going to listen on.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=19.7) [In this case, it's going to be on port 80.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=24.51) [We then have a parameter target port,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=26.84) [which is the port that our Pod is listening on.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=29.13) [So we saw earlier that our application is listening on port 8080,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=31.58) [so I'm going to highlight this code and run that to create](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=35.27) [the service for this particular deployment.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=39.02) [And at the bottom, we can see that service hello‑world is exposed.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=41.84) [And so let's look at the information associated with that service,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=45.24) [and we can do that with kubectl get service,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=48.32) [specifying the service name hello‑world.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=50.46) [At the bottom here in the output, we can see the name of the service,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=53.44) [the type is ClusterIP, and then the IP address associated with the service,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=56.11) [or this is where we're going to send user traffic to is 10.100.236.189.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=60.02) [There's no external IP associated with this.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=66.64) [In a cloud scenario, you might see a public IP address there.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=69.01) [The port that this service is listening on is port 80 on TCP.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=72.64) [Let's look at this from another angle using kubectl describe,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=77.44) [with kubectl describe service hello‑world,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=81.26) [and let's walk through this output together.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=83.46) [We see a lot of the same information that we saw just a second ago,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=85.84) [so we have the name, the type, and the IP address of the service.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=88.29) [We see the port that we're listening on and also the target port.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=91.36) [What I want to call out here is the Endpoints.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=94.74) [Endpoints are the IP and port pairs for each of the](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=97.34) [Pods that are a member of the service.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=100.76) [And currently there's only one Pod supporting this service,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=102.64) [and so here we see the Pod IP and port 192.168.206.127 and port 8080.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=105.51) [And so when traffic comes in on the service, it'll get routed to this endpoint.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=112.75) [And so let's go ahead and do just that. Let's access](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=117.74) [the service inside of the cluster.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=120.79) [And so what I'm going to do is I'm going to grab the IP](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=124.24) [address of the service and throw it into my clipboard and](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=126.14) [use curl to access our application,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=128.48) [which is a simple Hello World web application,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=131.85) [and curl is a command line web browser.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=134.29) [If I type curl and then the protocol http://,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=137.34) [and I'll paste in the service IP,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=140.58) [and then type in :80 for the port that we're listening on. When I press Enter,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=144.64) [this is going to send traffic to the service IP, which will then get routed](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=150.34) [into the Pod that's supporting this particular service.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=153.94) [So let's go ahead and press Enter and look at our output.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=157.28) [Our output, simple Hello, World application running on version 1.0.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=159.84) [The application also prints out the hostname.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=163.94) [So here we can see the actual Pod name is the hostname.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=166.21) [So we see hello‑world, then our Pod template hash,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=169.87) [and then the Pod ID ending in gnxsk.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=172.36) [And so what happened there is we hit the service IP. The service then routes](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=176.84) [that traffic into the individual Pod supporting the service.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=180.08) [If we scaled our application out,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=183.84) [there would be additional endpoints registered into the service,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=186.34) [and it would be up to kube‑proxy to load balance that traffic](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=188.8) [amongst all of the Pods supporting the service.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=192.18) [Now in the next part of the demo,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=196.84) [what I want to show you here is how to use kubectl to generate YAML or JSON](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=198.34) [for resources that you have deployed in your cluster.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=203.17) [In this case, we're going to look at a deployment.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=205.32) [And so on line 116 here I have kubectl get deployment hello‑world,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=208.24) [and then the output modifier ‑o yaml, and I want to pipe that output into more.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=212.9) [And so this will give us a YAML representation of the deployment object.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=217.14) [This YAML representation also includes runtime](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=222.19) [information about the actual object itself.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=225.44) [And so if we look inside of here, we see the YAML associated with the deployment,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=228.25) [apiVersion apps/v1, kind is Deployment,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=231.89) [but we also have a bunch of runtime information.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=235.33) [This can be useful for monitoring in configuration management scenarios,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=238.64) [but not so great as a source for manifest for declarative deployments.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=242) [We'd have to remove all of this runtime information.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=246.2) [I'm going to show you a technique in a few moments where we use dry run](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=248.56) [to help us generate these manifests very quickly and correctly at the](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=251.85) [command line without runtime information.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=256.59) [If we also wanted to see this from a different angle,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=259.44) [I can do the same, but for JSON.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=261.48) [So kubectl get deployment hello‑world, then with the output modifier ‑o json,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=263.56) [pipe that into more.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=268.75) [And here we can see the JSON representation of this object,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=270.05) [including its runtime information.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=273.48) [And so let's go ahead and clear out of that.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=276.04) [And before we move forward into our next demo,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=277.84) [I want to delete the resources that we created imperatively and recreate](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=280.37) [all of those declaratively in our upcoming demo. So let's walk through the](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=284.19) [process of deleting some resources together.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=287.36) [Before we delete anything, let's look at what we have.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=290.24) [And so with kubectl get all, we can see I have my deployment,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=292.61) [which created my replicaset, which created the Pod supporting that deployment.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=296.14) [We also see that our bare Pod is in there,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=300.58) [and I also have two services up and running,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=302.56) [one that we just created, our hello‑world service,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=305.42) [and a service for the API server that's available inside of the cluster.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=308.09) [So let's delete our service,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=312.44) [and we can do that with kubectl delete service hello‑world.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=313.98) [I'm going to go ahead and delete our deployment with](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=317.34) [kubectl delete deployment hello‑world.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=319.21) [What's going to happen when I do kubectl delete deployment hello‑world,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=322.04) [that'll delete the deployment, which will then delete the replicaset,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=325.49) [which will then delete all the Pods associated with that replicaset.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=328.57) [So there's a cascading delete that happens there.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=331.82) [We then need to delete our bare Pod since it's not](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=334.44) [associated with any controller, and we can do that with kubectl delete pod,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=336.85) [specifying a Pod name, hello‑world‑pod.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=341.78) [That's going to block for a moment until that Pod is actually deleted,](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=344.44) [and then we'll get our console back.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=347.75) [With that Pod deleted, if I do a kubectl get all now, all that we have left is that single cluster service for Kubernetes available.](https://app.pluralsight.com/course-player?clipId=8fdfeb30-2fbb-4b29-808c-bf3775f101f8&startTime=349.64)

[Demo: Declarative Deployments and Accessing and Modifying Existing Resources in Your Cluster](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c)

[Moving forward,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=1.14) [let's look at how we can deploy resources declaratively in our cluster.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=1.9) [We just walked through how to create things imperatively at the command line,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=6.48) [but we want to get to where we're deploying things](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=9.69) [declaratively in code in our cluster, and so let's start that process together.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=11.95) [On line 136 here,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=16.82) [I have an example of where we can use a dry‑run=client to help](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=18.76) [create a YAML manifest quickly and correctly,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=22.72) [and so let's walk through the code to do just that.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=25.39) [We have kubectl create deployment hello‑world,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=28.24) [and then specifying the image that we want to run.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=30.66) [So really, that's no different than we saw in the previous demo.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=32.6) [We then add on dry‑run=client and ‑o yaml,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=36.04) [and what that will do is create the YAML manifest for that deployment](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=39.73) [named hello‑world running that container image.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=45.05) [And that'll give us that output, the standard out.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=47.25) [I'm going to pipe that into more.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=49.22) [And so let's go and run this code and look at what we get.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=50.31) [In the code at the bottom, we have the YAML that describes what we want to do.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=53.04) [We want to create a deployment with that container image.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=56.98) [So let's break out of this output and then run that code one more time,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=61.44) [but instead of writing it to console, we'll write it to file.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=64.28) [And so here, the only difference from the previous command is on line 144,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=68.04) [where we're redirecting that output into a file named deployment.yaml file.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=72.38) [And so from here we could take this deployment.yaml and](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=77.14) [build on a more complex deployment, if we need to,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=79.8) [adding and changing the configuration inside of the](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=82.82) [deployment.yaml file if we needed to.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=85.39) [And so let's take a peek at that just to make sure that we have what we want,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=88.13) [and there we can see our code inside of there.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=90.91) [For our demonstration here,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=92.97) [we're going to take that file as is and deploy that in our cluster,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=94.69) [and so let's look at the code to do just that.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=99.13) [On line 152 we have kubectl apply ‑f deployment.yaml.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=101.33) [And so what this will do is read that deployment.yaml file and](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=106.39) [send it into the API server for creation.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=109.37) [So let's run that code and see what we get.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=111.97) [A few moments later at the bottom we can see](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=114.52) [deployment.apps/hello‑world is created.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=116.76) [So we created our deployment declaratively in code.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=119.26) [Now, we can do the same thing for our service.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=123.54) [In a previous demo, we use the command kubectl expose deployment,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=126.11) [and then specified the port as 80 and the target port as 8080.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=130.02) [We can also add on dry‑run=client ‑o yaml,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=133.54) [and do the same thing, converting that imperative code into declarative code.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=137.67) [And here we see at the bottom, we have our YAML representation of that.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=142.17) [We'll take that code and run that into file, just like we did a few moments ago.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=146.54) [But this time we're going to take this output for our service creation](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=150.73) [and redirect that into a file named service.yaml.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=153.88) [We'll take a peek inside of there and look at the code inside of service.yaml,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=157.44) [and here we can see the YAML manifest for the service that we want to create.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=161.32) [I can't stress enough how valuable this technique is to quickly and](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=167.44) [correctly create manifests that you can either deploy right away or](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=171.06) [make simple modifications to that text before you send it into the](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=174.78) [API server for creation.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=178.02) [So let's go ahead and create that service with kubectl apply ‑f service.yaml.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=180.84) [At the bottom here we can see service/hello‑world created.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=185.36) [Let's check out the status of our deployment and our service,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=190.04) [and we can do that with kubectl get all, run that code there,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=192.98) [and here we can see all the resources deployed into our default namespace.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=197.13) [In our default namespace we see our deployment hello‑world. That](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=201.74) [created the replicaset here at the bottom. That replicaset created](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=204.76) [the Pod that we have up and running.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=208.17) [We see our hello‑world service and also our Kubernetes](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=210.64) [API service that are in the cluster.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=213.15) [So next now,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=216.74) [let's look at how we can make a change to an existing resource in our cluster.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=217.66) [Since we have the code that describes our deployment,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=221.71) [we can make a modification to that file and then just resubmit that into the](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=225.2) [API server and then make that change to our deployment.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=228.99) [So let's do that together and edit deployment.yaml and make a](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=232.29) [modification to what we have running in our cluster.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=236.25) [So in here is the code that describes our deployment.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=239.84) [If we go down into the spec, I want to jump over in replicas and go from 1 to 20.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=242.74) [And so what I want to do is to change the number of replicas](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=249.44) [supporting our deployment from 1 to 20, and we're changing that in the code.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=251.99) [But that doesn't change the application yet.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=256.14) [I have to feed that code into the API server to effect that change.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=257.69) [And we do that with kubectl apply ‑f, and we're going to send in deployment.yaml.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=261.74) [At the bottom we can see deployment.apps/hello‑world configured.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=266.4) [We've effected that change to the desired state in our cluster.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=270.81) [And now Kubernetes will go and spin up those 19 additional Pods to](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=274.25) [bring us up to 20 replicas supporting our deployment.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=278.22) [And so if I do a kubectl get deployment now for hello‑world,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=281.64) [we can see in the output I have 20 of 20 Pods that are up,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=285.08) [running, and ready.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=288.31) [And if I do a kubectl get pods and pipe that into more,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=289.19) [we can see the the whole collection of Pods that are up and](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=292.09) [running and supporting our deployment now.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=295.02) [Now what happened behind the scenes when I scaled that deployment out,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=297.84) [each one of those Pods that was a member of that deployment](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=301.94) [also got registered as an endpoint in the service and will](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=304.25) [automatically start receiving workload.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=308.05) [And so if I do a kubectl get service now,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=310.44) [we'll go ahead and grab the IP address of this new service,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=313.64) [since I had to delete the previous service that we created](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=316.08) [imperatively and created a new one declaratively,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=318.37) [the IP address changed.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=320.94) [And so want to grab that IP address and throw that in my clipboard.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=322.07) [We're going to use curl again, http://,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=325.14) [and paste in the new IP address associated with our hello‑world service.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=328.36) [If I press Enter now,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=334.04) [I'm going to access the service on the service IP and then get load](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=335.4) [balanced to 1 of the 20 Pods supporting this application.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=338.16) [So there I got load balanced to a Pod ending in 7j8tc.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=342.22) [Let's try this a few more times, and we can see each time that I access the pod,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=346.44) [I get load balanced to a different Pod.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=350.46) [So the next one we see is qvc6s, 5w5cg, we hit that one twice,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=352.63) [and then we got load balanced to another Pod.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=358.88) [And so kube‑proxy is distributing that workload amongst 20](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=360.36) [Pods that are a member of this service.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=364.62) [Now let's say I didn't have that deployment manifest and I needed to scale my](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=368.04) [application or even make any other change to my application,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=372.04) [and I don't have that deployment manifest.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=375.45) [Well,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=377.88) [I can use the command kubectl edit to edit a resource that's](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=378.15) [available in the cluster that's already up and running.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=382.96) [And so let's look at the code to do this.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=385.84) [Kubectl edit, what do I want to edit?](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=387.91) [I want to edit the deployment named hello‑world.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=389.86) [And so when I run this code,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=393.14) [what it's going to do is retrieve the object via the API server and](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=394.4) [present that back to me in a local text editor.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=398.98) [And so now I don't need to make a modification to the code and then send it in.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=401.91) [What I'm going to do here is make the modification to this object](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=406.63) [in my text editor and then when I save this out,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=409.52) [it'll send this code back into the API server and effect that change.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=412.09) [And so here I'm going to change replicas from 20 to 30.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=416.68) [And when I save this out,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=421.44) [this is going to immediately be changed in our cluster effecting that change.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=423.04) [So there we see deployment.apps/hello‑world was edited.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=427.01) [And so now the desired state of our cluster changed from 20 Pods to 30 Pods,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=430.11) [and Kubernetes will go and scale up and create 10 more Pods in our](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=434.66) [cluster at the point in time in which that object was saved and sent back](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=438.35) [into the cluster when I exited my text editor.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=441.99) [If I do a kubectl get deployment now for hello‑world,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=444.54) [we can see 30 of 30 Pods are up and running.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=448.6) [I want to show you one other way to be able to scale an](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=452.04) [application at the command line, and we can do that with the](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=454.59) [command kubectl scale deployment. Which one?](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=457.69) [hello‑world and specifying the ‑‑replicas parameter. Here we're going](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=460.52) [to set our number of replicas to 40, and at the bottom there we can](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=464.76) [see deployment.apps/hello‑world is scaled.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=468.08) [And if I check the status of the deployment with kubectl get](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=470.99) [deployment hello‑world we can see 40 of 40 Pods are ready.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=473.72) [Now I do want to call out that I continue to scale up the number of Pods here.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=479.04) [We certainly could have reduced the number of Pods in any of these](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=482.55) [demonstrations to change the desired state of our application.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=485.29) [I'll leave that as an exercise to you, the viewer, to experiment with that.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=488.39) [And so that's a wrap for this demo.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=492.54) [Let's go ahead and clean up our resources that we deployed, and I'm going to](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=494.09) [use kubectl delete deployment hello‑world to delete that and then kubectl](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=498.17) [delete service hello‑world to delete our service.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=503.29) [And then I'll do a kubectl get all, and here we can see we're kind](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=506.34) [of in an intermediate state, where those Pods are all getting](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=509.88) [terminated actively for us behind the scenes.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=512.98) [And so we still have a couple that are up and running, and a few moments later,](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=516.03) [all of these Pods will be shut down and deleted from the cluster, and all](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=519.01) [that we'll be left with Is that Kubernetes service running in our cluster.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=521.94) [Let's refresh kubectl get all one last time to see, and all that we have left is our Kubernetes cluster API service.](https://app.pluralsight.com/course-player?clipId=2cee0169-e801-4c16-a894-c9962b6dad6c&startTime=526.34)

[Module Summary and Thank You!](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3)

[Here we are at the end of our module, and we introduced how we can](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=1.14) [interact with our cluster focusing on using kubectl to retrieve](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=4.15) [different types of information out of our cluster.](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=8.04) [And then we looked closely at how to deploy some applications and what](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=10.84) [is really happening behind the scenes. We discussed both imperative and](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=14.73) [declarative deployment methods, and hopefully I made the case for using](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=18.84) [declarative deployment methods.](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=22.43) [So here we are at the end of our course, and I really hope you](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=25.84) [enjoyed listening to this and that we laid the appropriate](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=28.66) [foundation for your Kubernetes studies.](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=31.18) [We covered a lot of ground together so far.](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=33.05) [We discussed the Kubernetes architecture, we did a deep dive into](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=35.17) [installing a Kubernetes cluster, both on‑prem and in the cloud, and](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=38.4) [we looked at how to interact with our cluster using kubectl and did](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=42.76) [some application deployments.](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=46.15) [It's truly been a pleasure recording this course,](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=47.74) [and I thank you so much for listening, and most importantly,](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=50.32) [learning with me.](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=53.38) [I hope you enjoyed the course. Join me again soon, here at Pluralsight.](https://app.pluralsight.com/course-player?clipId=1478fcf3-8af4-435f-9de2-7ee0947b46a3&startTime=54.31)