



# An Introduction to Kubernetes and Rancher





# Adrian Goins

Director of Community and Evangelism

adrian-goins

adriandotgoins

adrian

oskapt

monachus

adrian.goins.tv

Tweet about the training with #RancherTraining  
and #RancherK8s



# AGENDA

1

Intro and Housekeeping

2

Installing K3s

3

Intro to Kubernetes and  
Kustomize

4

Installing Rancher

5

Getting Started with Rancher

6

Final Q&A



Your questions in the GoToWebinar questions panel. → This session is being recorded.

# Follow Along With Me

The repository for today's class is at

<https://github.com/rancher/k8s-intro-training>



ll be sent after the class. ▶ Post your questions in the GoToWebinar question

# Resources - Docs

<https://rancher.com/docs/>

The screenshot shows a web browser displaying the Rancher v2.0 documentation. The URL in the address bar is <https://rancher.com/docs/rancher/v2.x/manual-quick-start/deployment/manual-quick-start-manual-setup/>. The page title is "Manual Quick Start". The left sidebar has a dropdown menu set to "RANCHER 2.X" and contains several sections: Overview, Quick Start Guides (which is expanded to show Amazon AWS Quick Start, DigitalOcean Quick Start, Vagrant Quick Start, and Manual Quick Start), Deploying Workloads, Installation, Upgrades and Rollbacks, and Backups and Disaster. The "Manual Quick Start" section is currently selected. The main content area starts with a heading "Howdy Partner! This tutorial walks you through:" followed by a bulleted list: "Installation of Rancher v2.0", "Creation of your first cluster", and "Deployment of an application, Nginx". Below this is a "Quick Start Outline" section with a sub-bullet list: "1. Provision a Linux Host", "2. Install Rancher", "3. Log In", and "4. Create the Cluster". A small video player in the bottom right corner shows a man with glasses speaking.

# Resources - Community Support

<http://slack.rancher.io>

Rancher Users - [Jump to...](#)

#general

13,449 | 9:0 | Rancher Users | Home · point » for now 2.0 discussions Rancher staff

Today

Devopsguy 8:21 AM  
In production ready cluster using 3 nodes with etcd and controlplane roles.  
Where the nginx ingress should run if the master nodes doesn't have worker roles?  
rebrek 8:32 AM  
does some people here sometimes get answers? or are there only people asking for help?  
jan\_n 8:32 AM  
@rebrek: sure, sometimes there are answers here. Then there's paid support if you really want answers. 😊  
jan\_n 8:32 AM  
lol i guess there is @jan\_n i am not used to slack, (I only use irc most of the time), but for a newbie asking for a probably dumb question, it's not even a paid support that I would need but a paid course i guess 😊  
jan\_n 8:34 AM  
@rebrek: so did you see I answered your question?  
jan\_n 8:35 AM  
@jan\_n umm no? did you? you mean my question about kubectl?  
rebrek 9:08 AM  
I reply Today at 9:08 AM  
oh ok, sorry didn't notice the "1 reply" below my original message, thanks a lot @jan\_n. Amazing! It works now, thank you!  
Norman Schlatter 10:44 AM  
hello, how can I start a pod in debug mode?  
bd8392 11:17 AM  
replied to a thread: HI, I've seen this tutorial <https://youtu.be/sM5vjt-hvIA> which was very good! The teacher said, that it is possible to set up ...  
<https://rancher.com/docs/rancher/v2.x/en/catalog/multi-cluster-apps/> would this work for my purpose?  
rebrek 11:32 AM  
Is there a way to replace the load balancer traefik (which is the default provided by k3s if I understand correctly) with another one (Ingress) please?

Message #general

<http://forums.rancher.com>

RANCHER

all categories Categories Latest New (13) Unread Top

Category Topics Latest

Announcements 8 / month Rancher Secret Support 1 Rancher 24m

General 21 / month Update load balancer tags via API 2 Rancher 2h

Beta Lounge 85 Rancher deployment - many environments 5 Rancher 2h

Rancher 64 / month GlusterFS / BeeGFS on Rancher? 2 Rancher 6h

RancherOS 13 / month Custom catalog not showing templates 0 Rancher 7h

Longhorn 1 / month Rancher Ingress contr 1 Rancher 8h

Kubernetes 1.8? 1 Rancher 1h

# What's required to unleash the power of Kubernetes

DevOps  
Automation

Secure Application Deployment

Shared Tools & Services

Centralized  
Management

Security, Policy and User Management



Consistent Cluster Operations

Certified  
Kubernetes  
Distributions

Run Certified Kubernetes from Dev to Edge



Datacenter



Cloud



Dev



Branch



Edge

the Community at [slack.rancher.io](https://slack.rancher.io)

Follow Along at [github.com/rancher/k](https://github.com/rancher/k)

# Rapid growth of Kubernetes is exposing the need for centralized Management



Secure Application Deployment

Load Balancing

Routing

Metrics

Autoscaling

Canary

Git Deployments

Shared Tooling & Services



Prometheus

Jenkins

Grafana



HELM



Istio



envoy



fluentd

Security & Authentication



Active Directory

GitHub

SAML

Ping

okta

Policy management

Configuration enforcement

RBAC policies

CIS benchmark monitoring

Pod & network security policies

Simplified Cluster Operations & Infrastructure Management

Kubernetes version management

Visibility & diagnostics

Monitoring & alerting

Centralized audit

Node pool management

Cluster provisioning



RKE



Datacenter



Amazon EKS



Azure AKS



Google GKE



Cloud



K3S



Dev



Branch



Edge

# Let's Talk About K3s

- Super-lightweight CNCF-certified K8s distribution
- Different from non-production solutions like minikube/microk8s/kind
- Small binary / Uses 512MB of RAM for all of Kubernetes
- Easily installed with k3sup from Alex Ellis



```
fish: /Users/monachus/Documents/workspace/monachus/rancher/training/k8s:1
```

```
~/D/w/m/r/t/k8s|master✓
► set -x KUBECONFIG $(pwd)/kubeconfig
~/D/w/m/r/t/k8s|master✓
► kubectl get nodes
NAME      STATUS    ROLES      AGE      VERSION
training-a   Ready     master    60s     v1.17.7+k3s1
~/D/w/m/r/t/k8s|master✓
► _
```



# Kubernetes 101

- ✓ Pods
- ✓ ReplicaSets
- ✓ Deployments
- ✓ ConfigMaps
- ✓ Services
- ✓ Ingresses

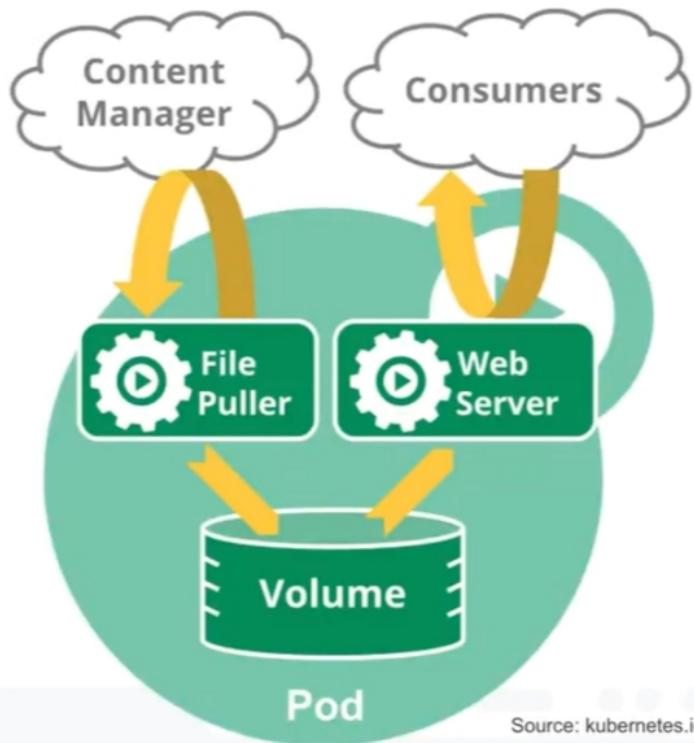


# Pods

- Smallest unit that can be deployed in Kubernetes
- Consist of one or more containers that are always scheduled together
- Each pod is given a unique IP address
- Containers in a pod can speak to each other via localhost



# Pods



Source: kubernetes.io



# Basic Pod Spec

```
apiVersion: v1
kind: Pod
metadata:
  name: myapp-pod
  labels:
    app: myapp
spec:
  containers:
  - name: myapp-container
    image: busybox
    command: ['sh', '-c', 'echo Hello Kubernetes! && sleep 1']
```



```
~/D/w/m/r/t/k8s|master✓ kubectl apply -f pod/pod.yaml
pod/myapp-pod created
~/D/w/m/r/t/k8s|master✓ kubectl get po
NAME      READY  STATUS        RESTARTS   AGE
myapp-pod  0/1   ContainerCreating   0          6s
~/D/w/m/r/t/k8s|master✓ kubectl get po -w
NAME      READY  STATUS        RESTARTS   AGE
myapp-pod  0/1   Completed    0          13s
myapp-pod  1/1   Running     1          14s
myapp-pod  0/1   Completed    1          15s
myapp-pod  0/1   CrashLoopBackOff  1          16s
myapp-pod  1/1   Running     2          35s
myapp-pod  0/1   Completed    2          36s
myapp-pod  0/1   CrashLoopBackOff  2          50s
myapp-pod  1/1   Running     3          67s
myapp-pod  0/1   Completed    3          68s
```



# Declarative and Repeatable Actions

- Use principles of infrastructure-as-code and configuration management to deploy clusters and workloads
  - Terraform
  - Ansible
  - Kustomize
- Keep templates under source control
- Take care not to commit secrets to the repository





# Kustomize

- Built into kubectl since 1.14 (-k)
- Allows easy templating and overrides
- Easily maintained in source control
- Allows separation of concerns between Dev and Ops

Docs at <https://kustomize.io>

For the rest of the Kubernetes portion, we'll be using Kustomize to generate and apply configs.

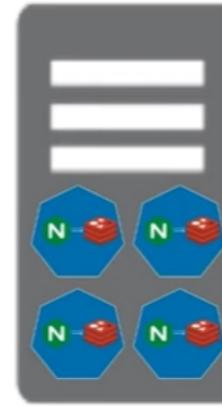
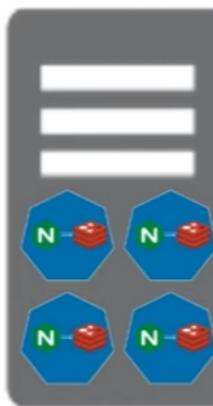
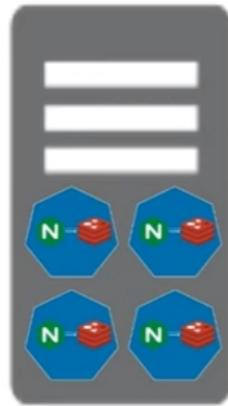


# Replica Set

- Defines the desired scale and state of a group of pods
- Introduces state management

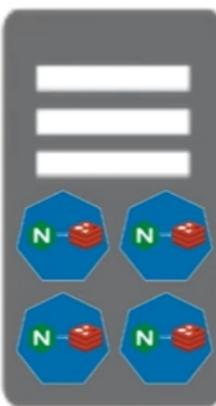
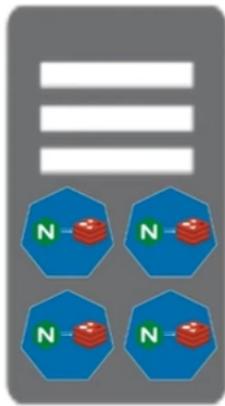


# Replica Set



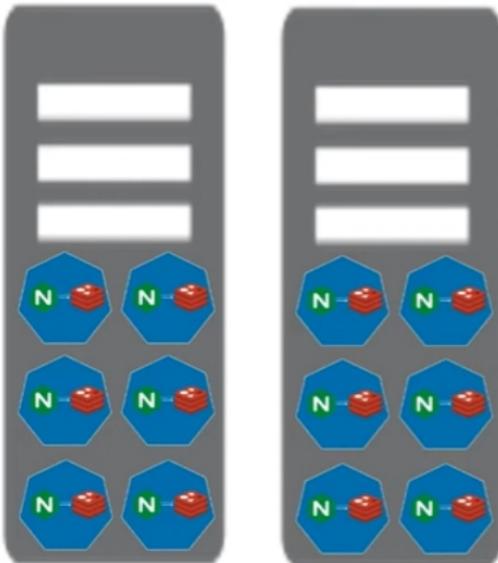


# Replica Set





# Replica Set



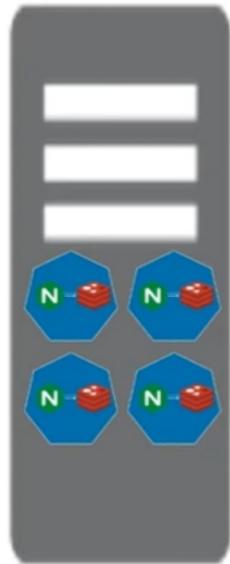


# Deployments

- Level of abstraction above ReplicaSets
- Deployments create and update ReplicaSets
- Allow you to easily scale and perform rolling upgrades

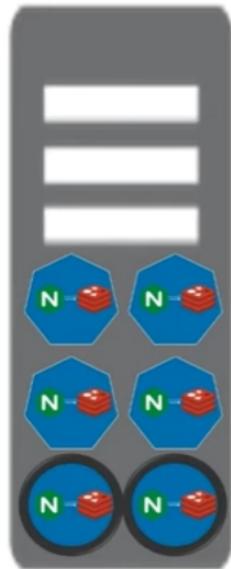


# Deployments



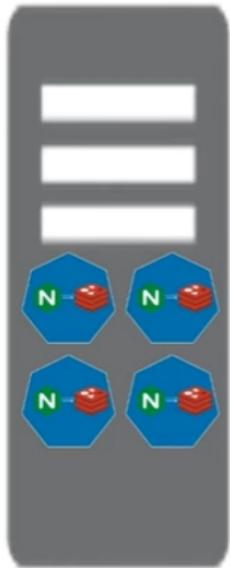


# Deployments





# Deployments



```
kubectl /Users/monachus/Documents/workspace/monachus/rancher/training/k8s
root@training-a:/etc/rancher/k3s# logout
Connection to training-a closed.
~/D/w/m/r/t/k8s|master✓
► kubectl create deploy nginx --image=nginx:1.16-alpine
deployment.apps/nginx created
~/D/w/m/r/t/k8s|master✓
► kubectl get deploy
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
nginx     0/1       1           0           6s
~/D/w/m/r/t/k8s|master✓
► kubectl get rs
NAME          DESIRED   CURRENT   READY   AGE
nginx-689d5b8c7  1         1         0       11s
~/D/w/m/r/t/k8s|master✓
► kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-689d5b8c7-rhgks  1/1     Running   0          62s
~/D/w/m/r/t/k8s|master✓
► kubectl delete deploy/nginx
```



```
fish ~/Users/monachus/Documents/workspace/rancher/training/k8s
~/D/w/m/r/t/k8s|master✓
► kubectl create deploy nginx --image=nginx:1.16-alpine
deployment.apps/nginx created
~/D/w/m/r/t/k8s|master✓
► kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-689d5b8c7-dsxsb   1/1     Running   0          5s
~/D/w/m/r/t/k8s|master✓
► kubectl get pods -o yaml ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► kubectl get pods/nginx-689d5b8c7-dsxsb -o yaml ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► _
```



```
~/D/w/m/r/t/k8s|master✓
► kubectl create deploy nginx --image=nginx:1.16-alpine
deployment.apps/nginx created
~/D/w/m/r/t/k8s|master✓
► kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-689d5b8c7-dsxsb   1/1     Running   0          5s
~/D/w/m/r/t/k8s|master✓
► kubectl get pods -o yaml ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► kubectl get pods/nginx-689d5b8c7-dsxsb -o yaml ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► kubectl describe po nginx-689d5b8c7-dsxsb _^&1 |less;
```



Name: nginx-689d5b8c7-dsxsb  
Namespace: default  
Priority: 0  
Node: training-a/10.68.0.143  
Start Time: Thu, 06 Aug 2020 13:36:29 -0400  
Labels: app=nginx  
pod-template-hash=689d5b8c7  
Annotations: <none>  
Status: Running  
IP: 10.42.0.10  
IPs:  
  IP: 10.42.0.10  
Controlled By: ReplicaSet/nginx-689d5b8c7  
Containers:  
  nginx:  
    Container ID: containerd://c5b5be8e2f2844e37a483ff0ed70d97a3243fab0ffbaf281c2f48ee39434ec017  
    Image: nginx:1.16-alpine  
    Image ID: docker.io/library/nginx@sha256:0dfc8450deb8c7f06fbaac27e453ac3262df7d3a93639c4e2  
    Port: <none>  
    Host Port: <none>  
    State: Running  
      Started: Thu, 06 Aug 2020 13:36:30 -0400  
    Ready: True  
    Restart Count: 0  
    Environment: <none>  
    Mounts:  
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-pbj9f (ro)



```
fish /Users/monachus/Documents/workspace/monachus/rancher/training/k8s
~/D/w/m/r/t/k8s|master✓
► kubectl create deploy nginx --image=nginx:1.16-alpine
deployment.apps/nginx created
~/D/w/m/r/t/k8s|master✓
► kubectl get pods
NAME                      READY   STATUS    RESTARTS   AGE
nginx-689d5b8c7-dsxsb   1/1     Running   0          5s
~/D/w/m/r/t/k8s|master✓
► kubectl get pods -o yaml ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► kubectl get pods/nginx-689d5b8c7-dsxsb -o yaml ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► kubectl describe po nginx-689d5b8c7-dsxsb ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► _
```



```
~/D/w/m/r/t/k8s|master✓
► kubectl create deploy nginx --image=nginx:1.16-alpine
deployment.apps/nginx created
~/D/w/m/r/t/k8s|master✓
► kubectl get pods
NAME READY STATUS RESTARTS AGE
nginx-689d5b8c7-dsxsb 1/1 Running 0 5s
~/D/w/m/r/t/k8s|master✓
► kubectl get pods -o yaml ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► kubectl get pods/nginx-689d5b8c7-dsxsb -o yaml ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► kubectl describe po nginx-689d5b8c7-dsxsb ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► kubectl scale deploy/nginx --replicas=3
deployment.apps/nginx scaled
~/D/w/m/r/t/k8s|master✓
► kubectl get pods
NAME READY STATUS RESTARTS AGE
nginx-689d5b8c7-dsxsb 1/1 Running 0 3m6s
nginx-689d5b8c7-bdnxj 1/1 Running 0 4s
nginx-689d5b8c7-pkw2s 1/1 Running 0 4s
~/D/w/m/r/t/k8s|master✓
► _
```



```
fish ~/Users/monachus/Documents/workspace/monachus/rancher/training/k8s
~/D/w/m/r/t/k8s|master✓
► kubectl create deploy nginx --image=nginx:1.16-alpine
deployment.apps/nginx created
~/D/w/m/r/t/k8s|master✓
► kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-689d5b8c7-dsxsb  1/1     Running   0          5s
~/D/w/m/r/t/k8s|master✓
► kubectl get pods -o yaml ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► kubectl get pods/nginx-689d5b8c7-dsxsb -o yaml ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► kubectl describe po nginx-689d5b8c7-dsxsb ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► kubectl scale deploy/nginx --replicas=3
deployment.apps/nginx scaled
~/D/w/m/r/t/k8s|master✓
► kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-689d5b8c7-dsxsb  1/1     Running   0          3m6s
nginx-689d5b8c7-bdnxj   1/1     Running   0          4s
nginx-689d5b8c7-pkw2s   1/1     Running   0          4s
~/D/w/m/r/t/k8s|master✓
► kubectl set image deploy/nginx nginx=nginx:1.17-alpine_
```



```
fish /Users/monachus/Documents/workspace/monachus/rancher/training/k8s
~/D/w/m/r/t/k8s|master✓
► kubectl rollout status deploy/nginx
Waiting for deployment "nginx" rollout to finish: 1 out of 3 new replicas have been updated...
^C
~/D/w/m/r/t/k8s|master✓
► kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-689d5b8c7-dsxsb   1/1     Running   0          4m1s
nginx-689d5b8c7-bdnxj   1/1     Running   0          59s
nginx-689d5b8c7-pkw2s   1/1     Running   0          59s
nginx-597f95cdd7-wgkj7   0/1     ErrImagePull   0          43s
~/D/w/m/r/t/k8s|master✓
► kubectl describe po nginx-597f95cdd7-wgkj7 ^&1 |less;
~/D/w/m/r/t/k8s|master✓
► kubectl rollout undo deploy/nginx
deployment.apps/nginx rolled back
~/D/w/m/r/t/k8s|master✓
► kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
nginx-689d5b8c7-dsxsb   1/1     Running   0          5m26s
nginx-689d5b8c7-bdnxj   1/1     Running   0          2m24s
nginx-689d5b8c7-pkw2s   1/1     Running   0          2m24s
~/D/w/m/r/t/k8s|master✓
► -
```



```
~/D/w/m/r/t/k8s|master✓
► kubectl edit deploy/nginx
deployment.apps/nginx edited
~/D/w/m/r/t/k8s|master✓
► kubectl rollout status deploy/nginx
Waiting for deployment "nginx" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 1 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 2 out of 3 new replicas have been updated...
Waiting for deployment "nginx" rollout to finish: 1 old replicas are pending termination...
Waiting for deployment "nginx" rollout to finish: 1 old replicas are pending termination...
deployment "nginx" successfully rolled out
~/D/w/m/r/t/k8s|master✓
► -
```



```
~/D/w/m/r/t/k/d/base|master✓
► cat kustomization.yaml
resources:
- deployment.yaml
- service.yaml
configMapGenerator:
- name: index
  files:
  - configs/index.html
~/D/w/m/r/t/k/d/base|master✓
► cat configs/index.html
<html>
<head>
<title>Hello from nowhere!</title>
</head>
<body>
<h1>Hello Class!</h1>
<p>This is the default environment.</p>
</body>
</html>
~/D/w/m/r/t/k/d/base|master✓
► _
```



```
fish /Users/monachus/Documents/workspace/monachus/rancher/training/k8s/deployment/base
~/D/w/m/r/t/k/d/base|master✓
➤ kubectl apply -k .
configmap/index-gm4g8b4899 created
service/nginx created
deployment.apps/nginx created
~/D/w/m/r/t/k/d/base|master✓
➤ cat service.yaml
apiVersion: v1
kind: Service
metadata:
  labels:
    app: nginx
    name: nginx
spec:
  ports:
  - port: 80
    protocol: TCP
    targetPort: 80
  selector:
    app: nginx
  sessionAffinity: None
  type: NodePort
~/D/w/m/r/t/k/d/base|master✓
➤ _
```



```
fish ~/Users/monachus/Documents/workspace/monachus/rancher/training/k8s/deployment/overlay/staging
~/D/w/m/r/t/k/d/o/staging|master✓
► cat kustomization.yaml
namePrefix: staging-
commonLabels:
  environment: staging
bases:
- ../../base
patches:
- image.yaml
- replica_count.yaml
configMapGenerator:
- name: index
  behavior: replace
  files:
  - configs/index.html
~/D/w/m/r/t/k/d/o/staging|master✓
► _
```



```
~/D/w/m/r/t/k/d/o/staging|master✓
► cat kustomization.yaml
namePrefix: staging-
commonLabels:
  environment: staging
bases:
- ../../base
patches:
- image.yaml
- replica_count.yaml
configMapGenerator:
- name: index
  behavior: replace
  files:
    - configs/index.html
~/D/w/m/r/t/k/d/o/staging|master✓
► cat configs/index.html
<html>
<head>
<title>Hello from staging!</title>
</head>
<body>
<h1>Hello Class!</h1>
<p>This is the staging environment.</p>
</body>
</html>
~/D/w/m/r/t/k/d/o/staging|master✓
```



```
~/D/w/m/r/t/k/d/o/staging|master✓
► cat image.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
spec:
  template:
    spec:
      containers:
        - name: nginx
          image: nginx:1.17-alpine
~/D/w/m/r/t/k/d/o/staging|master✓
► _
```



```
~/D/w/m/r/t/k/d/o/staging|master✓
► cat image.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
spec:
  template:
    spec:
      containers:
        - name: nginx
          image: nginx:1.17-alpine
~/D/w/m/r/t/k/d/o/staging|master✓
► cat replica_count.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
spec:
  replicas: 1
~/D/w/m/r/t/k/d/o/staging|master✓
► -
```



```
fish /Users/monachus/Documents/workspace/monachus/ranches/training/k8s/deployment/overlay/staging
~/D/w/m/r/t/k/d/o/staging|master✓
► kubectl apply -k .
configmap/staging-index-7c47666mfg created
service/staging-nginx created
deployment.apps/staging-nginx created
~/D/w/m/r/t/k/d/o/staging|master✓
► kubectl get pods
NAME READY STATUS RESTARTS AGE
staging-nginx-74dd844-qjk8n 1/1 Running 0 4s
~/D/w/m/r/t/k/d/o/staging|master✓
► kubectl get deploy
NAME READY UP-TO-DATE AVAILABLE AGE
staging-nginx 1/1 1 1 10s
~/D/w/m/r/t/k/d/o/staging|master✓
► kubectl get configmap
NAME DATA AGE
staging-index-7c47666mfg 1 18s
~/D/w/m/r/t/k/d/o/staging|master✓
► cd ..../..
```



```
~/D/w/m/r/t/k/d/o/production|master✓
► cat kustomization.yaml
namePrefix: prod-
commonLabels:
  environment: production
bases:
- ../../base
patches:
- image.yaml
- replica_count.yaml
configMapGenerator:
- name: index
  behavior: replace
  files:
  - configs/index.html
~/D/w/m/r/t/k/d/o/production|master✓
► _
```



```
~/D/w/m/r/t/k/d/o/production|master✓
```

```
► cat kustomization.yaml
namePrefix: prod-
commonLabels:
  environment: production
bases:
- ../../base
patches:
- image.yaml
- replica_count.yaml
configMapGenerator:
- name: index
  behavior: replace
  files:
    - configs/index.html
```

```
~/D/w/m/r/t/k/d/o/production|master✓
```

```
► cat configs/index.html
```

```
<html>
<head>
<title>Hello from Production!</title>
</head>
<body>
<h1>Hello Class!</h1>
<p>This is the production environment.</p>
</body>
</html>
```

```
~/D/w/m/r/t/k/d/o/production|master✓
```



```
fish ~/Users/monachus/Documents/workspace/monachus/rancher/training/k8s/deployment/overlay/production
~/D/w/m/r/t/k/d/o/production|master✓
► cat replica_count.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
spec:
  replicas: 3
~/D/w/m/r/t/k/d/o/production|master✓
► cat image.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx
spec:
  template:
    spec:
      containers:
        - name: nginx
          image: nginx:1.16-alpine
~/D/w/m/r/t/k/d/o/production|master✓
► -
```



```
~/D/w/m/r/t/k/d/o/production|master✓
```

```
► kubectl get deploy
```

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
staging-nginx	1/1	1	1	71s
prod-nginx	3/3	3	3	5s

```
~/D/w/m/r/t/k/d/o/production|master✓
```

```
► kubectl get pods
```

NAME	READY	STATUS	RESTARTS	AGE
staging-nginx-74dd844-qjk8n	1/1	Running	0	77s
prod-nginx-6df449679d-r44wj	1/1	Running	0	11s
prod-nginx-6df449679d-6p6qw	1/1	Running	0	11s
prod-nginx-6df449679d-kd87r	1/1	Running	0	11s

```
~/D/w/m/r/t/k/d/o/production|master✓
```

```
► -
```



# ConfigMaps

- Used to override container-specific data like
  - Configuration files
  - Environment variables
  - Entire directories of data
- Automatically updated within container when changed
  - nginx ingress controller
- Best practice is to version ConfigMaps and perform rolling update
  - Kustomize will do this for us

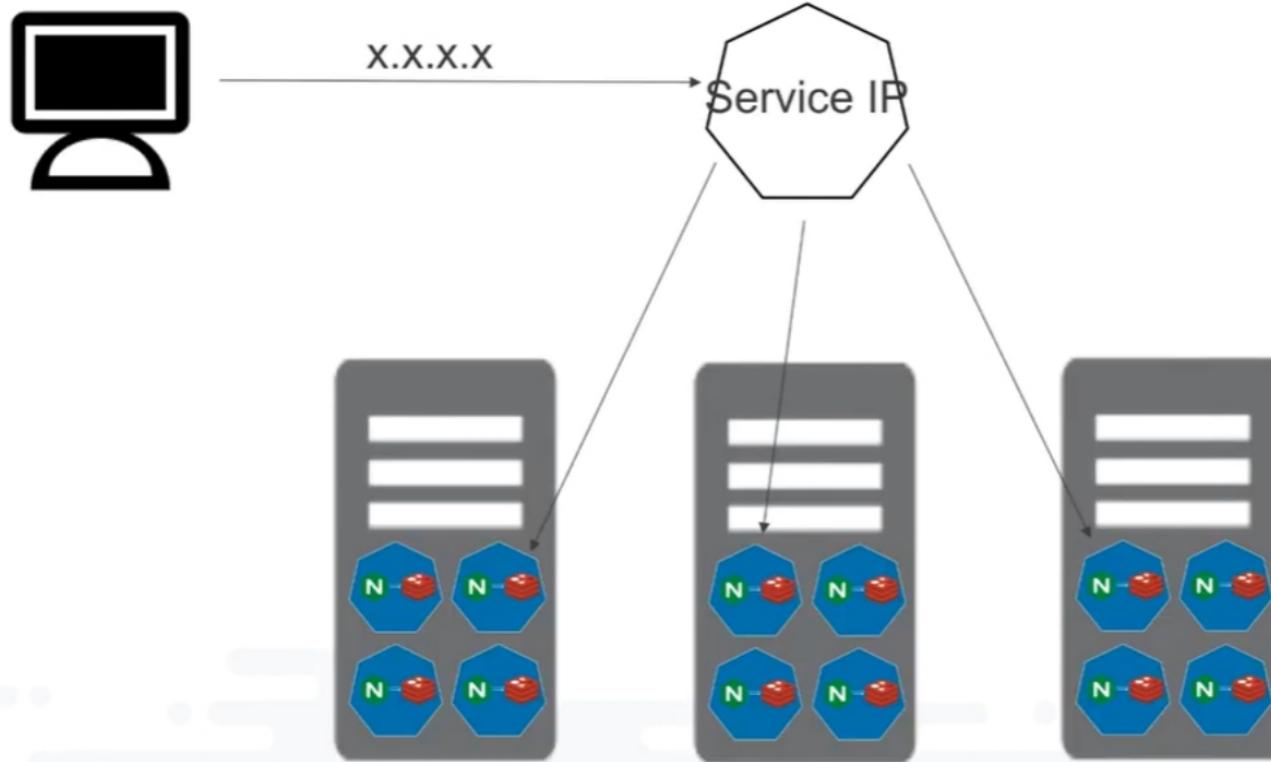


# Services

- Define a DNS entry that can be used to refer to a group of pods
- Provide a consistent endpoint for the group of pods
- Different types: ClusterIP, NodePort, LoadBalancer



# Services



```
~/D/w/m/r/t/k/d/o/production|master✓
```

```
► kubectl get service
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
kubernetes	ClusterIP	10.43.0.1	<none>	443/TCP	43m
staging-nginx	NodePort	10.43.63.43	<none>	80:32513/TCP	9m40s
prod-nginx	NodePort	10.43.65.200	<none>	80:30258/TCP	8m34s

```
~/D/w/m/r/t/k/d/o/production|master✓
```

```
► cat ../../base/service.yaml
```

```
apiVersion: v1
```

```
kind: Service
```

```
metadata:
```

```
  labels:
```

```
    app: nginx
```

```
  name: nginx
```

```
spec:
```

```
  ports:
```

```
  - port: 80
```

```
    protocol: TCP
```

```
    targetPort: 80
```

```
  selector:
```

```
    app: nginx
```

```
  sessionAffinity: None
```

```
  type: NodePort
```

```
~/D/w/m/r/t/k/d/o/production|master✓
```

```
► -
```



```
fish ~/Users/monachus/Documents/workspace/monachus/rancher/training/k8s/deployment/overlay/production
▶ kubectl get service
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)        AGE
kubernetes     ClusterIP  10.43.0.1    <none>       443/TCP       46m
staging-nginx  NodePort   10.43.63.43  <none>       80:32513/TCP  12m
prod-nginx    NodePort   10.43.65.200 <none>       80:30258/TCP  11m
~/D/w/m/r/t/k/d/o/production|master✓
▶ curl training-a:32513
<html>
<head>
<title>Hello from staging!</title>
</head>
<body>
<h1>Hello Class!</h1>
<p>This is the staging environment.</p>
</body>
</html>
~/D/w/m/r/t/k/d/o/production|master✓
▶ curl training-a:30258
<html>
<head>
<title>Hello from Production!</title>
</head>
<body>
<h1>Hello Class!</h1>
<p>This is the production environment.</p>
</body>
</html>
~/D/w/m/r/t/k/d/o/production|master✓
▶
```

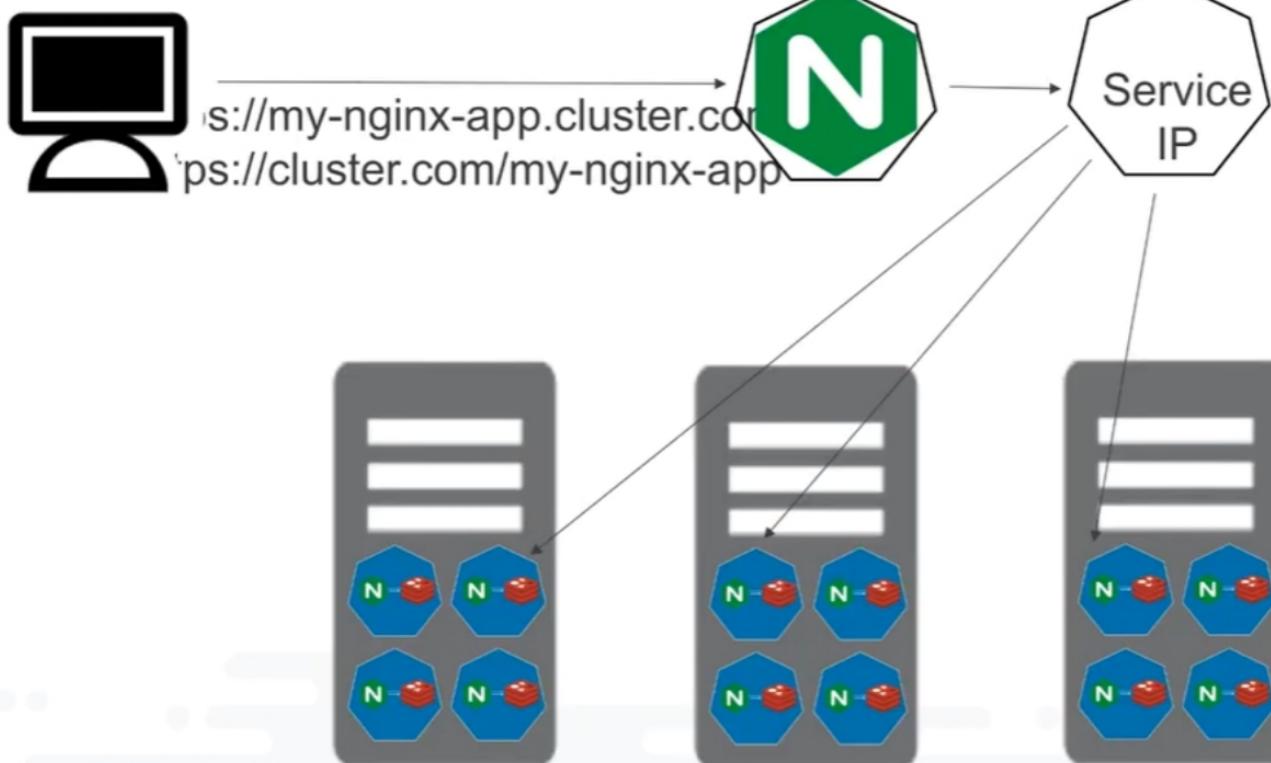


# Ingresses

- Define how traffic outside the cluster is routed to inside the cluster
- Used to expose Kubernetes services to the world
- Route traffic to internal services based on factors such as host and path
- Ingress is usually implemented by a load balancer (Nginx, HAProxy, etc...)



# Ingresses



```
~/D/w/m/r/t/k/d/o/i/single|master✓
➤
~/D/w/m/r/t/k/d/o/i/single|master✓
➤ cat kustomization.yaml
namePrefix: dev-
commonLabels:
  environment: dev
resources:
- ingress.yaml
bases:
- ../../base
patchesStrategicMerge:
- service.yaml
~/D/w/m/r/t/k/d/o/i/single|master✓
➤ _
```



```
~/D/w/m/r/t/k/d/o/i/single|master✓
```

```
► cat service.yaml
```

```
apiVersion: v1
```

```
kind: Service
```

```
metadata:
```

```
  labels:
```

```
    app: nginx
```

```
  name: nginx
```

```
spec:
```

```
  ports:
```

```
  - port: 80
```

```
    protocol: TCP
```

```
    targetPort: 80
```

```
  type: ClusterIP
```

```
~/D/w/m/r/t/k/d/o/i/single|master✓
```

```
► -
```



~/D/w/m/r/t/k/d/o/i/single|master✓

► kubectl get deploy

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
staging-nginx	1/1	1	1	16m
prod-nginx	3/3	3	3	15m
dev-nginx	1/1	1	1	5s

~/D/w/m/r/t/k/d/o/i/single|master✓

► kubectl get service

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
kubernetes	ClusterIP	10.43.0.1	<none>	443/TCP	50m
staging-nginx	NodePort	10.43.63.43	<none>	80:32513/TCP	16m
prod-nginx	NodePort	10.43.65.200	<none>	80:30258/TCP	15m
dev-nginx	ClusterIP	10.43.74.194	<none>	80/TCP	13s

~/D/w/m/r/t/k/d/o/i/single|master✓

► \_



```
fish ~/Users/monachus/Documents/workspace/monachus/rancher/training/k8s/deployment/overlay/ingress/single
▶ cat ingress.yaml
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: demo-ingress
  annotations:
    nginx.ingress.kubernetes.io/rewrite-target: /
spec:
  rules:
  - host: training-a.cl.monach.us
    http:
      paths:
      - path: /
        backend:
          serviceName: nginx
          servicePort: 80
~/D/w/m/r/t/k/d/o/i/single|master✓
▶ -
```



```
~/D/w/m/r/t/k/d/o/i/single|master✓
```

```
► kubectl get ingress
```

NAME	HOSTS	ADDRESS	PORTS	AGE
dev-demo-ingress	training-a.cl.monach.us	10.68.0.143	80	72s

```
~/D/w/m/r/t/k/d/o/i/single|master✓
```

```
► curl training-a.cl.monach.us
```

```
<html>
<head>
<title>Hello from nowhere!</title>
</head>
<body>
<h1>Hello Class!</h1>
<p>This is the default environment.</p>
</body>
</html>
```

```
~/D/w/m/r/t/k/d/o/i/single|master✓
```

```
► _
```



# Kubernetes 101

- ✓ Pods
- ✓ ReplicaSets
- ✓ Deployments
- ✓ ConfigMaps
- ✓ Services
- ✓ Ingresses



Rancher Docs: Documentation +

https://rancher.com/docs/ 133% Search

DOCS REQUEST A DEMO PRICING SUPPORT CONTACT

RANCHER WHY RANCHER? PRODUCTS CUSTOMERS RESOURCES COMPANY GET STARTED

Rancher 2.x

Rancher manages all of your Kubernetes clusters everywhere, unifies them under centralized RBAC, monitors them and lets you easily deploy and...

[Read the docs](#)

RKE

Rancher Kubernetes Engine

Rancher 1.6

If you haven't yet migrated to Rancher 2.x, you can still find documentation for 1.6 here. This is only for legacy users of the 1.6 product.

[Read the docs](#)

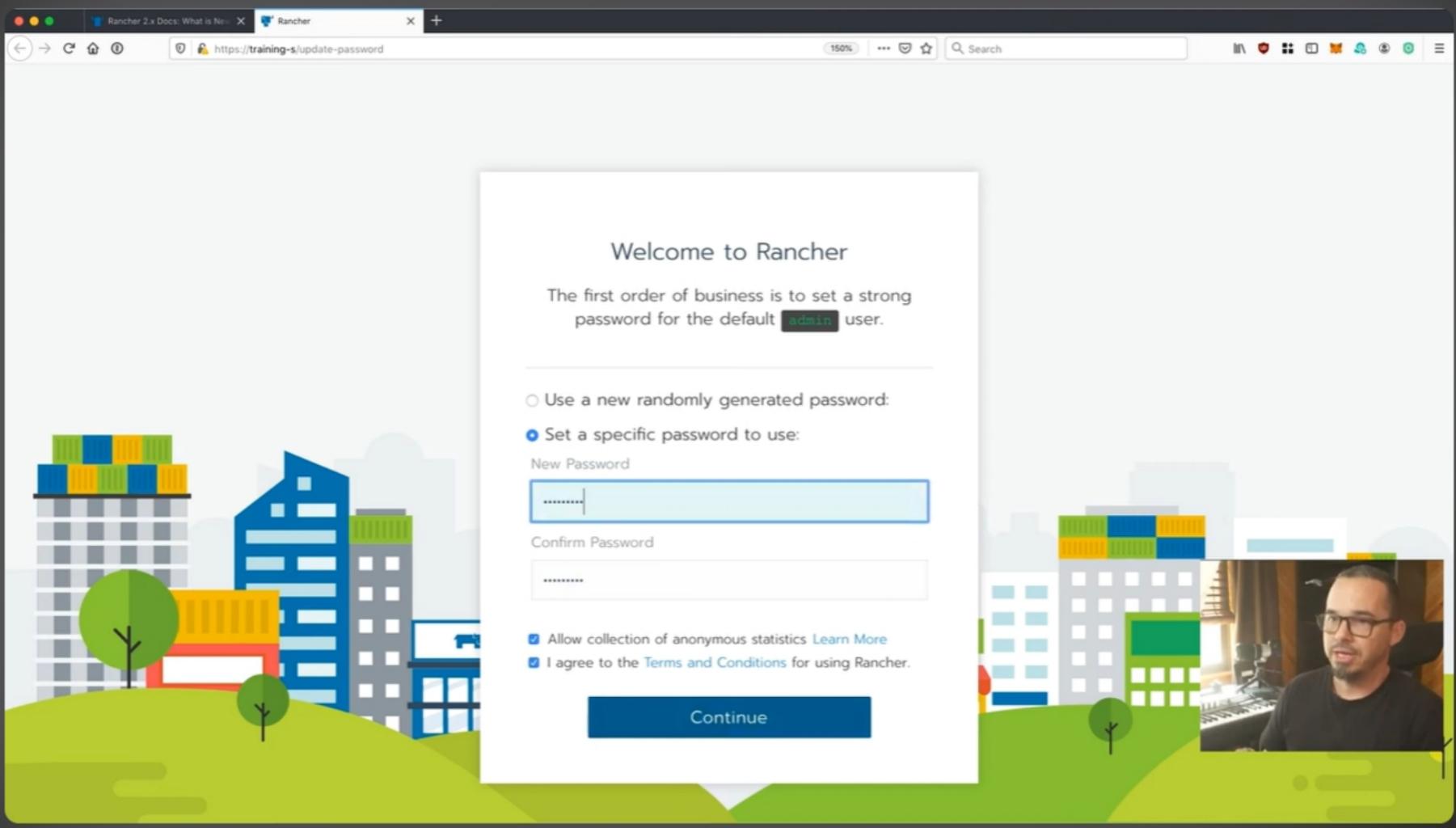
K3S

RancherOS

RancherOS is the lightest, easiest way to run Docker in production. Engineered from the ground up for security and speed, it runs all system services an...

[Read the do](#)





## Welcome to Rancher

The first order of business is to set a strong password for the default `admin` user.

- Use a new randomly generated password:
- Set a specific password to use:

New Password

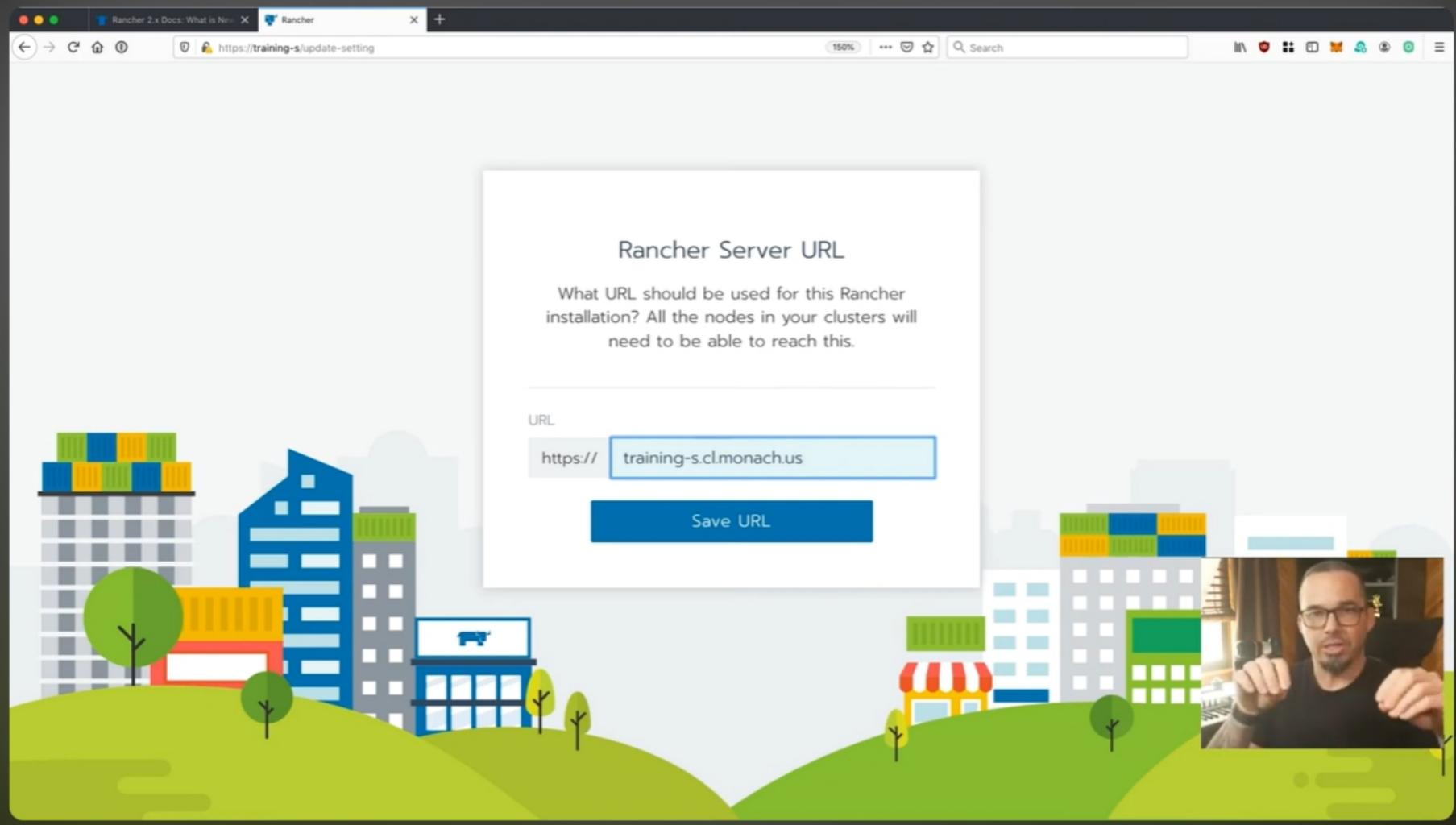
.....

Confirm Password

.....

- I agree to the [Terms and Conditions](#) for using Rancher.
- Allow collection of anonymous statistics [Learn More](#)

[Continue](#)



## Rancher Server URL

What URL should be used for this Rancher installation? All the nodes in your clusters will need to be able to reach this.

URL

https://

Save URL

Rancher 2.x Docs: What is New

Rancher

https://training-s/g/clusters/add/select

150% ... Search

Global Clusters Apps Settings Security Tools

## Add Cluster - Select Cluster Type



### From existing nodes (Custom)

Create a new Kubernetes cluster using RKE, out of existing bare-metal servers or virtual machines.



### Import an existing cluster

Import an existing Kubernetes cluster. For **K3S backed clusters**, Rancher can manage some aspects of the cluster configuration, such as version upgrades. For standard Kubernetes clusters, the provider will manage provisioning and configuration.

With RKE and new nodes in an infrastructure provider



Amazon EC2



Azure



DigitalOcean



Linode



vSphere

With a hosted Kubernetes provider



Amazon EKS



Azure AKS



Google GKE



fish :/Users/monachus/Documents/workspace/monachus/rancher/training/k8s/deployment/overlay/ingress/single

```
root@training-s:~# docker run -d --restart=unless-stopped -p 80:80 -p 443:443 -v /opt/rancher:/var/lib/rancher rancher/rancher:v2.4.5
3375452ceaa060530e4df132764b12d6bb8a86ba82138f8477a186468eec60a8
root@training-s:~# exit
logout
Connection to training-s closed.
~/D/w/m/r/t/k/d/o/i/single|master✓
► curl --insecure -sfL https://training-s.cl.monach.us/v3/import/c89tc5cbh7ffcw7skm8cj4bfcdwmcjn4pxvc8vx9ffdpgkn7tzkw.yaml | kubectl apply -f -
error: no objects passed to apply
~/D/w/m/r/t/k/d/o/i/single|master✓
► curl --insecure -sfL https://training-s.cl.monach.us/v3/import/c89tc5cbh7ffcw7skm8cj4bfcdwmcjn4pxvc8vx9ffdpgkn7tzkw.yaml | kubectl apply -f -
clusterrole.rbac.authorization.k8s.io/proxy-clusterrole-kubeapiserver created
clusterrolebinding.rbac.authorization.k8s.io/proxy-role-binding-kubernetes-master created
namespace/cattle-system created
serviceaccount/cattle created
clusterrolebinding.rbac.authorization.k8s.io/cattle-admin-binding created
secret/cattle-credentials-21a1e23 created
clusterrole.rbac.authorization.k8s.io/cattle-admin created
deployment.apps/cattle-cluster-agent created
daemonset.apps/cattle-node-agent created
~/D/w/m/r/t/k/d/o/i/single|master✓
► _
```



Rancher 2.x Docs: What is New    Rancher

https://training-s/g/clusters

150% ⌂ Search

Global Clusters Apps Settings Security Tools

Clusters Global Projects in demo Add Cluster

Clusters demo Default System

Delete Active

State Active

Search

Provider Nodes CPU RAM

K3s v117.7+k3s1 1 n/a n/a

Provider	Nodes	CPU	RAM
K3s v117.7+k3s1	1	n/a	n/a

Rancher 2.x Docs: What is New

Rancher

https://training-s/p/c-lttw7-p-jzfzj/workloads

150% ⚡ Search

demo Default Resources Apps Namespaces Members Tools Try Dashboard

Workloads Load Balancing Service Discovery Volumes Import YAML Deploy

Redeploy ↕ Pause Orchestration || Download YAML Delete

Search

State	Name	Image	Scale
Active	dev-nginx	nginx:1.16-alpine	1
Active	prod-nginx	nginx:1.16-alpine	3
Active	staging-nginx	nginx:1.17-alpine	1

Namespace: default



```
kubectl /Users/monachus/Documents/workspace/monachus/rancher/training/k8s/deployment/overlay/ingress/single
root@training-s:~# docker run -d --restart=unless-stopped -p 80:80 -p 443:443 -v /opt/rancher:/var/lib/rancher rancher/rancher:v2.4.5
3375452ceaa060530e4df132764b12d6bb8a86ba82138f8477a186468eec60a8
root@training-s:~# exit
logout
Connection to training-s closed.
~/D/w/m/r/t/k/d/o/i/single|master✓
► curl --insecure -sfL https://training-s.cl.monach.us/v3/import/c89tc5cbh7ffcw7skm8cj4bfcdwmcjn4pxvc8vx9ffdpgkn7tzkw.yaml | kubectl apply -f -
error: no objects passed to apply
~/D/w/m/r/t/k/d/o/i/single|master✓
► curl --insecure -sfL https://training-s.cl.monach.us/v3/import/c89tc5cbh7ffcw7skm8cj4bfcdwmcjn4pxvc8vx9ffdpgkn7tzkw.yaml | kubectl apply -f -
clusterrole.rbac.authorization.k8s.io/proxy-clusterrole-kubeapiserver created
clusterrolebinding.rbac.authorization.k8s.io/proxy-role-binding-kubernetes-master created
namespace/cattle-system created
serviceaccount/cattle created
clusterrolebinding.rbac.authorization.k8s.io/cattle-admin-binding created
secret/cattle-credentials-21a1e23 created
clusterrole.rbac.authorization.k8s.io/cattle-admin created
deployment.apps/cattle-cluster-agent created
daemonset.apps/cattle-node-agent created
~/D/w/m/r/t/k/d/o/i/single|master✓
► kubectl delete -k _
```



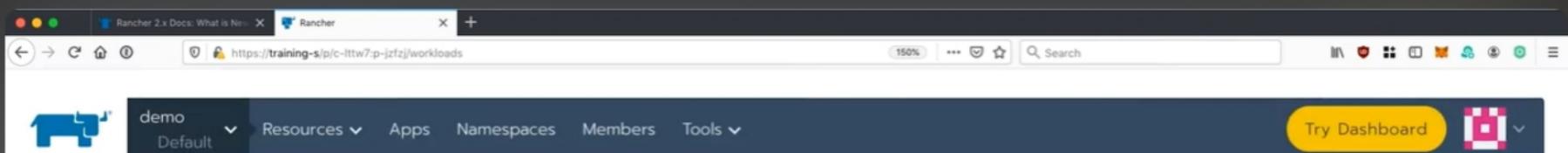
Rancher 2.x Docs: What is New

Rancher

demo Default Resources Apps Namespaces Members Tools Try Dashboard

https://training-s/p/c-ltw7-p-jzfjzj/workloads

150% Search



Workloads Load Balancing Service Discovery Volumes

Redeploy Pause Orchestration Download YAML Delete

Search

	State	Name	Image	Scale
prod-nginx	Active	prod-nginx	nginx:1.16-alpine	3
staging-nginx	Active	staging-nginx	nginx:1.17-alpine	1

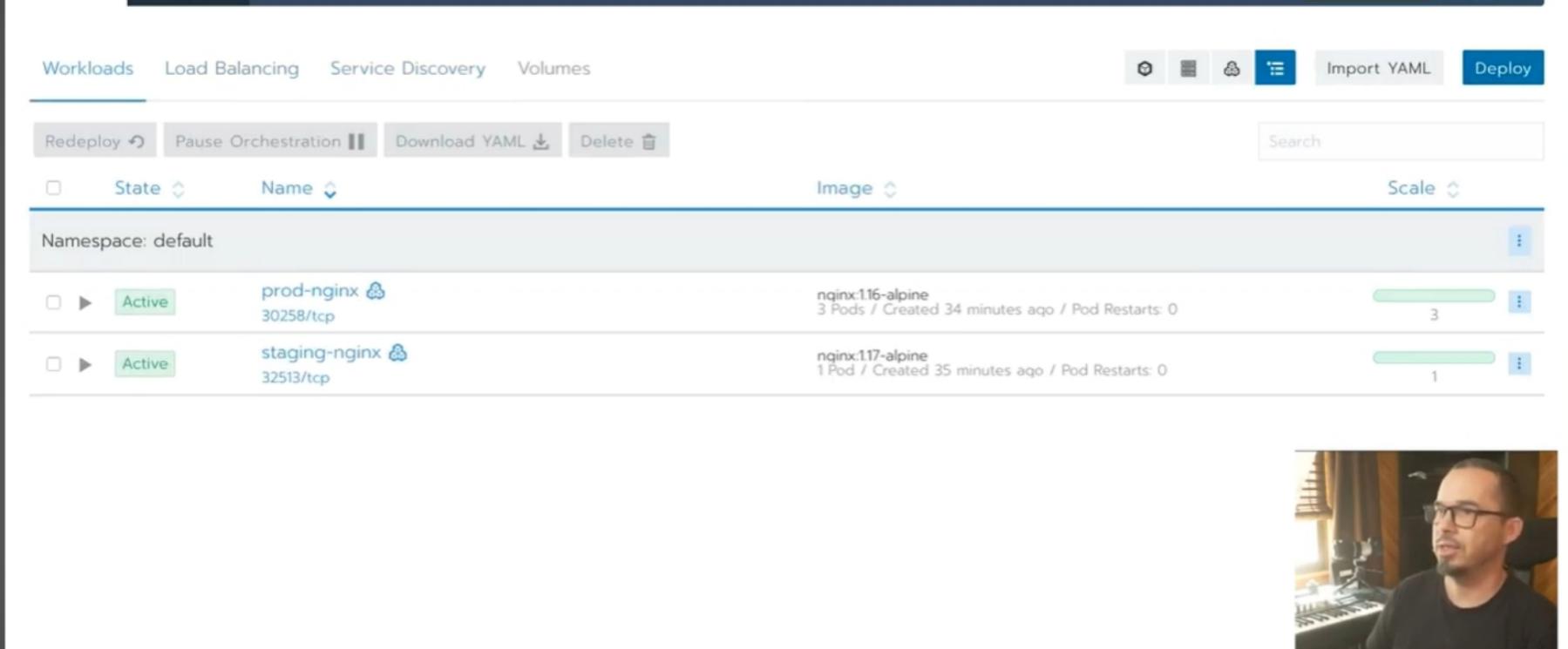


Namespace: default

v2.4.5 Help & Docs Forums Slack File an Issue

English Download CLI

https://training-s/p/c-ltw7-p-jzfjzj/workloads/run



v2.4.5 Help & Docs Forums Slack File an Issue

English Download CLI

https://training-s/p/c-ltw7-p-jzfjzj/workloads/run





demo

Dashboard: demo

Put this into `~/.kube/config`:

```
apiVersion: v1
kind: Config
clusters:
- name: "demo"
  cluster:
    server: "https://training-s/k8s/clusters/c-lttw7"
    certificate-authority-data: "LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUJpRENDQ\\
VMZ0F3SUJB01CQURBS0JnZ3Foa2pPUFFRREFqQTdNUnd3R2dZRFZRUUtFeE5rZVc1aGJXbGoKY\\
kdsemRHVNvaWE10YjNkbk1Sc3dHUVLEVlFRREV4SmrlVzVoYldsamJhbHpkR1Z1WlhJdFkyRXdIa\\
GNOTWpBdwPREEyTVRneE1qUTBXaNNOTXpBd09EQTBNVGd4TWpRMFdqQTdNUnd3R2dZRFZRUUtFe\\
E5rZVc1aGJXbGpiR2x6CmRHVNvaWE10YjNkbk1Sc3dHUVLEVlFRREV4SmrlVzVoYldsamJhbHpkR\\
1Z1WlhJdFkyRXdXVEFUmdjcWhrak8KUFFJQkJnZ3Foa2pPUFFNQkJ3TkNBQVM2cTdpYWgzNWRO\\
UpqbisreEzyZmI5Ty80Nk1EaUk5TU8vZndl3F1TgpjZ2h0ZUg4M2RsVVFMVECaW11bHF4aDVsR\\
FV0eWRZMWI0Q2x1dXBuR0FUN295TxjdJVEFPQmd0VkhROEJBZhFCkJBTUNBcVF3RhZRFZSMFRBU\\
UgvQkFVd0F3RUIvekFLQmdncWhrak9QUVFEQWd0SUFEQkZBaUE3VU5FTktDZloKKzli0TZBbkdyW\\
ndnVndJL3c0d3RkWE4xMX1PYm1YRnJTQUloQU1KbTBDbWEvQjdNQ1dxZWp4UVlnQ1JscjRmMgp5S\\
k1LNUZla1VDZWlGOud5Ci0tLS0tRUEIENFUlRJRkldQVRFLS0tLS0="

users:
- name: "demo"
  user:
    token: "kubeconfig-user-pmfpf:vq2mds84fckv99nttq4w5cj96jnnvpzxqx6pdw96pvzxrtz5x6th"

contexts:
- name: "demo"
```



Rancher 2.x Docs: What is New

Rancher

demo Default Resources Apps Namespaces Members Tools Try Dashboard

https://training-s/p/c-ltw7-p-jzfzj/workloads

150% ... Search

Workloads Load Balancing Service Discovery Volumes Import YAML Deploy

Redeploy	Pause Orchestration	Download YAML	Delete	Search
State	Name	Image	Scale	
Namespace: default				
<input type="checkbox"/>	<input type="checkbox"/> demo	monachus/rancher-demo	<div style="width: 30px; height: 10px; background-color: red;"></div> 3	<span>...</span>
	Updating	Deployment does not have minimum availability.		
<input type="checkbox"/>	<input type="checkbox"/> prod-nginx	nginx:1.16-alpine	<div style="width: 100%; height: 10px; background-color: green;"></div> 3	<span>...</span>
<input type="checkbox"/>	<input type="checkbox"/> staging-nginx	nginx:1.17-alpine	<div style="width: 10px; height: 10px; background-color: blue;"></div> 1	

# Rancher Academy is here!

The screenshot shows the landing page for the 'Certified Rancher Operator: Level 1' course. At the top, it says 'Certified Rancher Operator: Level 1' and 'Run Kubernetes Everywhere'. Below that, there's a summary table:

Start Date	Duration	Price
Self-Paced	5 Weeks	Free

Below the table is a 'Course Description' section with text about Kubernetes becoming the de facto standard for deploying container-backed applications. It also lists what you'll learn, including topics like RKE, Rancher Server, Kubernetes Clusters, and Kubernetes security.

On the right side, there's a detailed course breakdown:

Organization	RANCHER
Effort	3-5 Hours
Level	Introductory
Subject	Installing And Using Rancher For Deploying And Managing Kubernetes

At the bottom left is a large blue button with the Rancher Academy logo, which features a white bull icon and the text 'RANCHER ACADEMY'.

- Video introductions
- Theory work
- Demonstrations
- Hands-on labs
- Quizzes
- A graded final exam
- Certification



Gain in-depth knowledge and the confidence to best deploy, use, and manage Kubernetes with Rancher.

Go to: [academy.rancher.com](https://academy.rancher.com)

# 1000+ sign-ups on launch day!

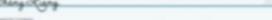


Shmulik Ahituv • 2nd  
Head of DevOps & IT at Zebra Medical Vision- I'm hiring!  
2th • Edited •

I'm usually not down with all these certification exams but this one is different.  
I'm a big fan of [Rancher Labs](#) products and vision.  
IMHO, Rancher did to Kubernetes what Docker did to containers technology.  
They've made [#kubernetes](#) accessible, relatively easy to setup a 'production grade level' cluster without requiring a PhD, made it vendor agnostic, Oh and the best part- all the rancher products ([#rancher](#), [#rke](#), [#k3s](#), [#longhorn](#) etc.) are FREE, open source and backed by [CNCF](#).

RANCHER ACADEMY  
CERTIFICATE OF COMPETENCY

This certificate is awarded to  
**Shmulik Ahituv**  
for demonstrating competency in the completion of  
**Certified Rancher Operator: Level One**

  
Shmulik Ahituv  
CEO, Rancher Labs  
June 1, 2020



71 • 5 Comments



# Thank You For Attending!



Slides and video will be sent after the class.



Post your questions