



Up & Running: Rancher

Course Overview & Rancher Architecture

Instructor



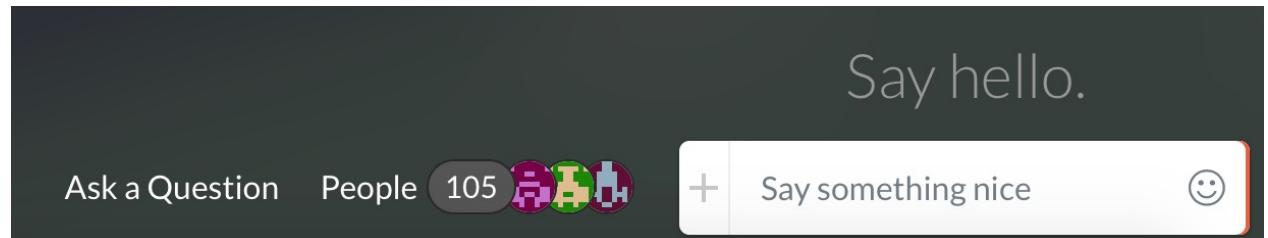
Luke Mwila

I'm a Senior Technical Evangelist at SUSE and is an AWS Container Hero. I specialize in cloud and DevOps engineering and cloud-native technologies.

I'm passionate about sharing knowledge through various mediums and engaging with the developer community at large.

Crowdcast

- Ask questions and upvote
- Look out for polls
- Say something nice in the chat
- Session is recorded



Agenda

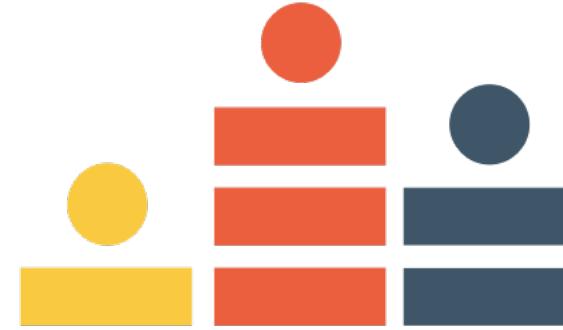
1. What To Expect
2. How To Sign Up
3. What is Rancher?
4. Why use Rancher?
5. Q & A
6. The Rancher Architecture
7. Q & A



What To Expect

Course Session Format

1. All classes will be live sessions held on Crowdcast every Wednesdays at 8am PT (for the next 8 weeks).
2. Each session will be an 1 hr 15 minutes long.
3. There is no certification at the end of the 8 weeks.



Rancher 2.6

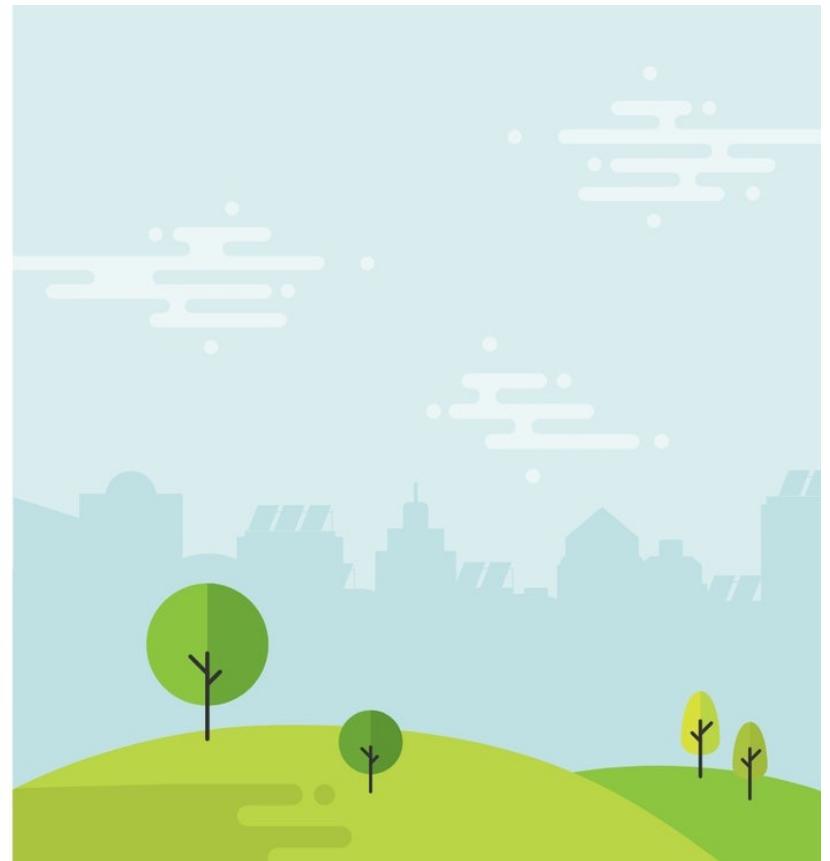
All the content in this course will be based on the latest version of Rancher.

Howdy!
Welcome to Rancher

Username
admin

Password

Remember Username



Course Structure

Installing Rancher & Initial Setup Part 1 (Week 2)

1. Prerequisites for installing Rancher
2. Installation using Docker and Rancher Desktop

Installing Rancher & Initial Setup Part 2 (Week 3)

1. Installation using Helm
2. HA Rancher on RKE cluster
3. Backup and Restore
4. Security with RBAC and Auth Providers



Course Structure

Cluster Operations Part 1 (Week 4)

1. Importing Clusters
2. Creating Clusters
3. Modifying Clusters
4. The kubectl Shell
5. Working with the Rancher CLI/API



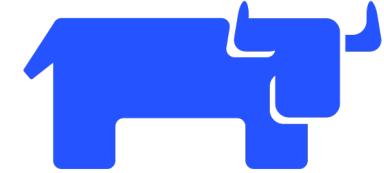
Course Structure

Cluster Operations Part 2 (Week 5)

1. Downstream Cluster Backup, Restore and Upgrade
2. Certificate Management
3. CIS Scanning

Cluster Operations Part 3 (Week 6)

1. Monitoring and Observability
2. Logging
3. Troubleshooting



Course Structure

Application Management (Week 7)

1. Project and Namespace Management
2. Deploying Workloads
3. Service Discovery and Load Balancers
4. Persistent Storage
5. Managing Helm Repositories
6. Continuous Delivery with Rancher Fleet



Course Structure

Where Do You Go from Here (Week 8)

1. What didn't we cover and where can you learn it?
2. FAQ and Main issues Rancher users face



How To Sign Up

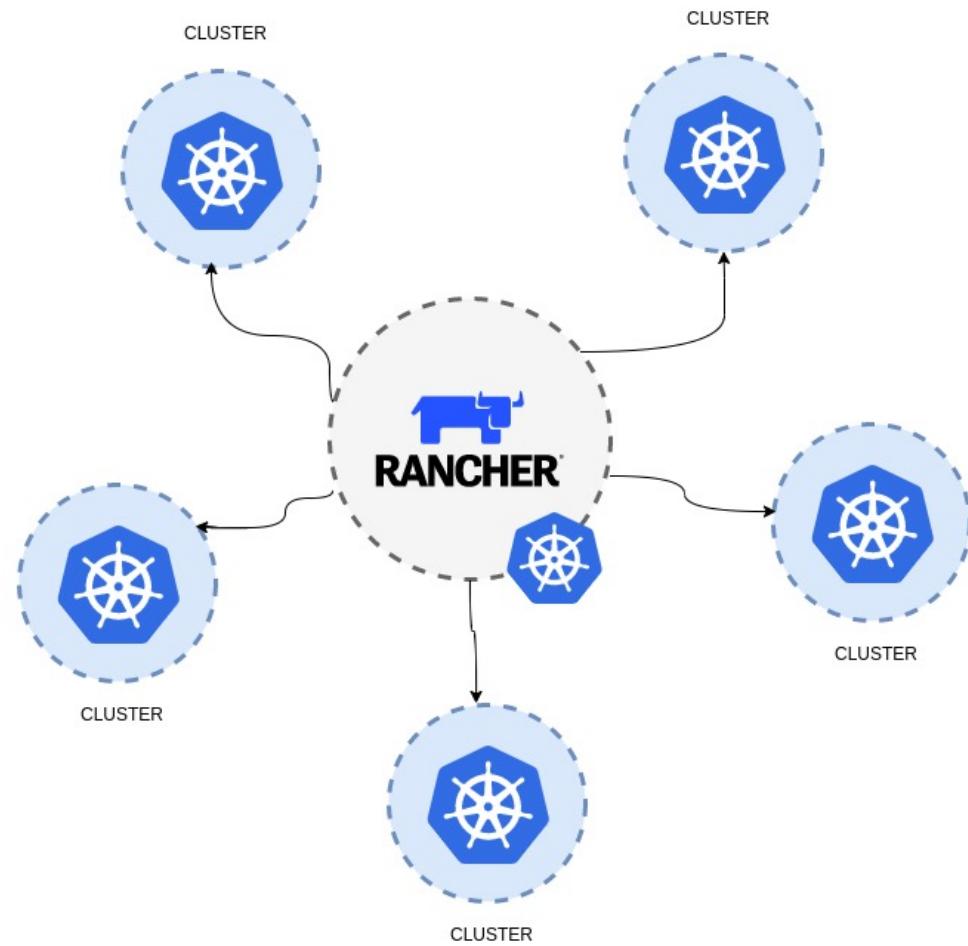
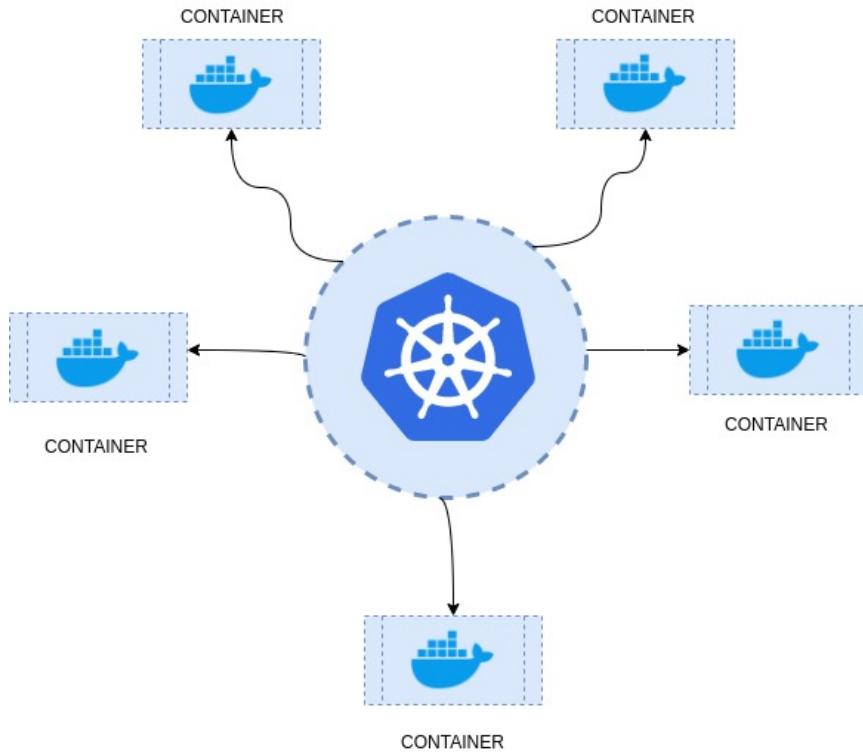


What is Rancher?

It is a Kubernetes cluster management platform. Rancher is designed to address the challenges presented by multi-cluster management.

*As Kubernetes orchestrates, manages and deploys containers,
Rancher in turn orchestrates, manages, and deploys Kubernetes
clusters.*

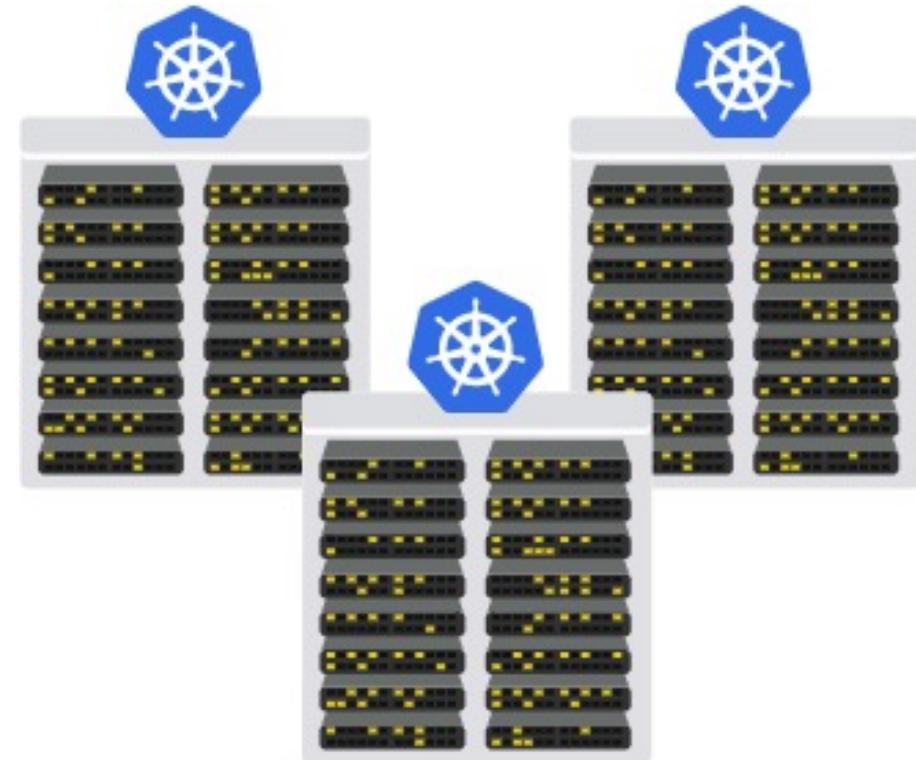
What is Rancher?



Why use Rancher?

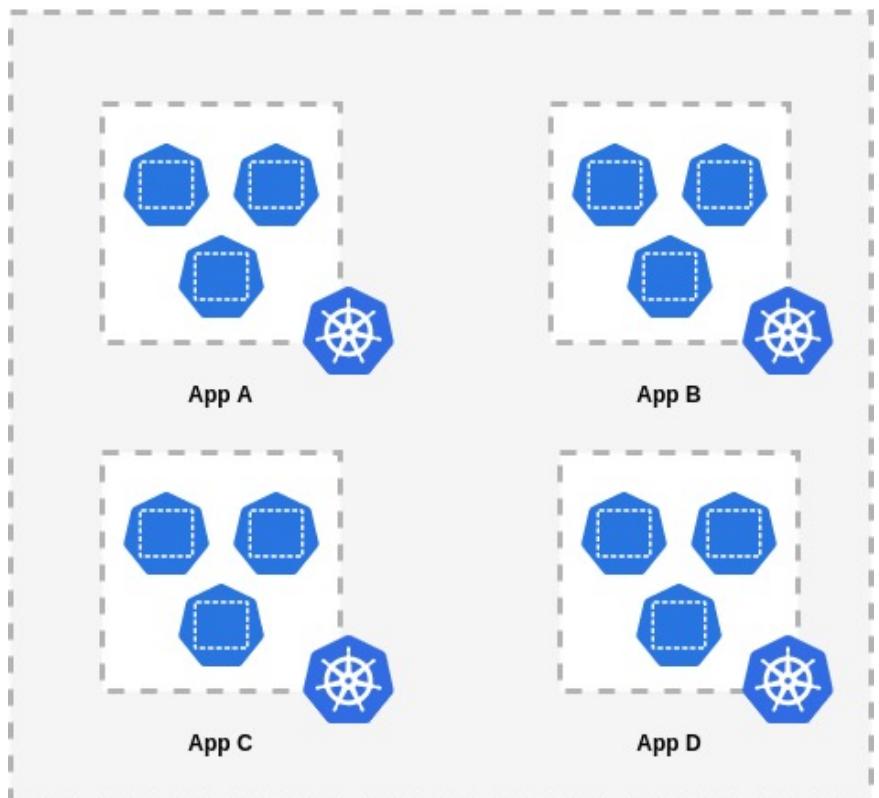
Challenges with Multi-Cluster Management

Cluster optimization is hard. This becomes increasingly complex the more clusters you have to deal with.

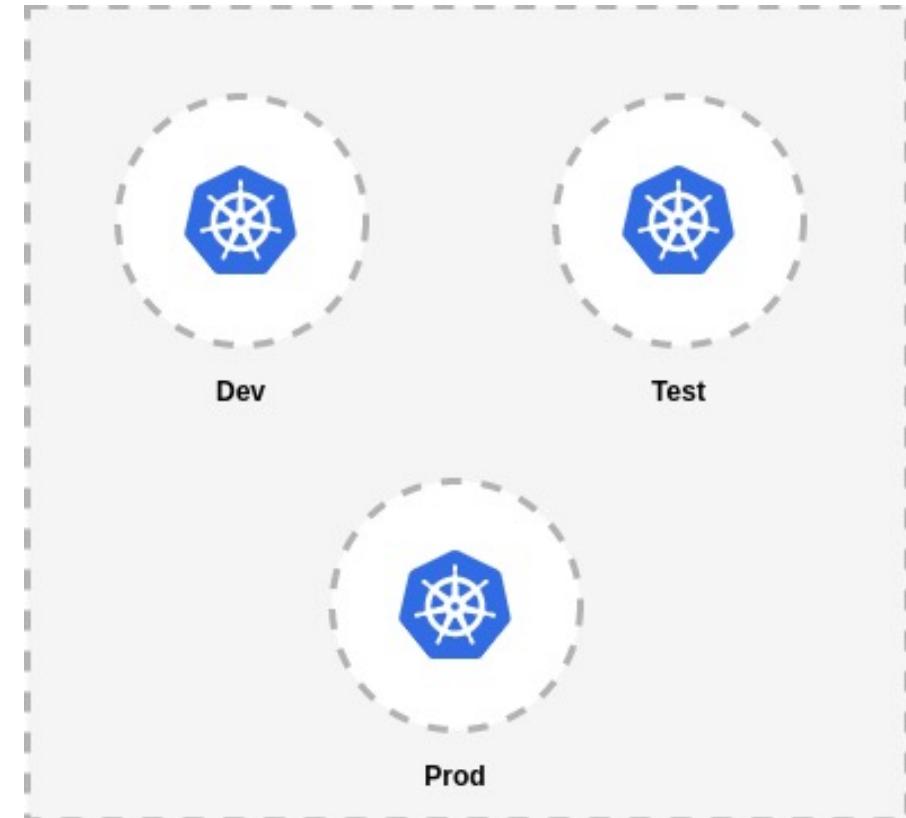


Multi-Cluster Models

Cluster per application

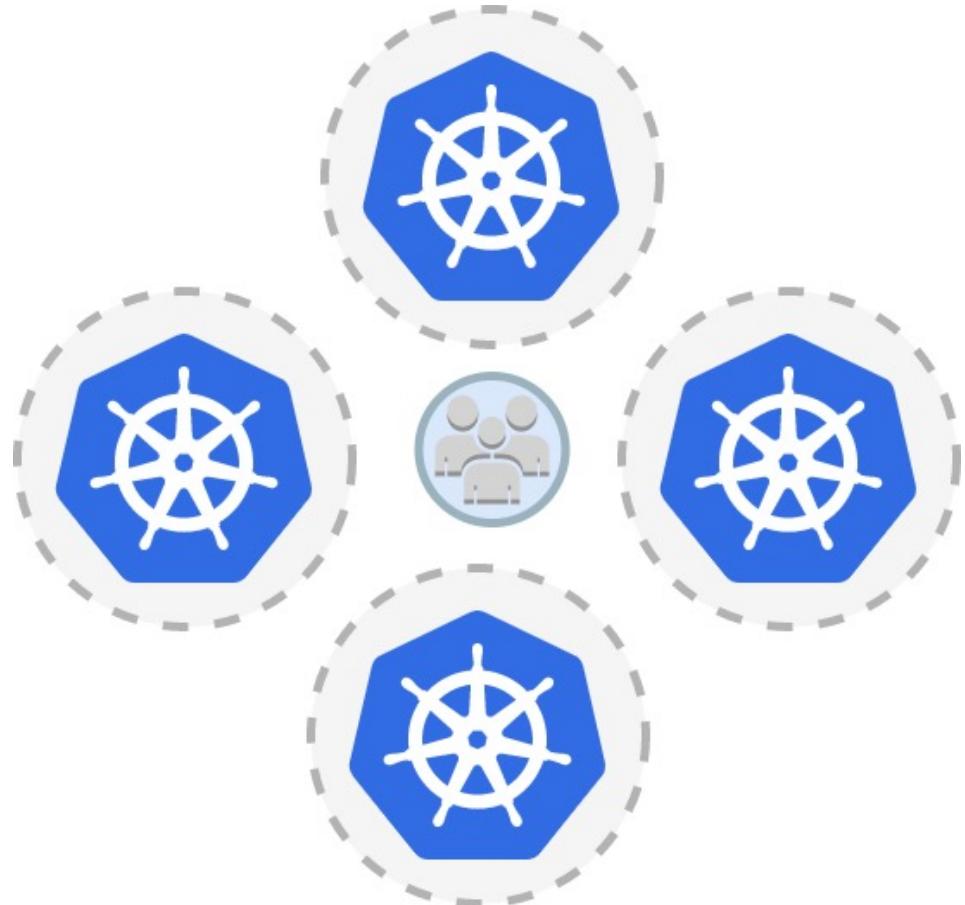


Cluster per environment



Optimizing Multi-Cluster Management

- Security – Cluster Personas & RBAC
- Organization – Namespaces
- Visibility – Monitoring, Auditing, Resource Utilization
- Node Configuration – Scaling Configurations
- Templating for consistency
- LFCM of Hosted Clusters – EKS, GKE, AKS, etc.
- Great platform experience

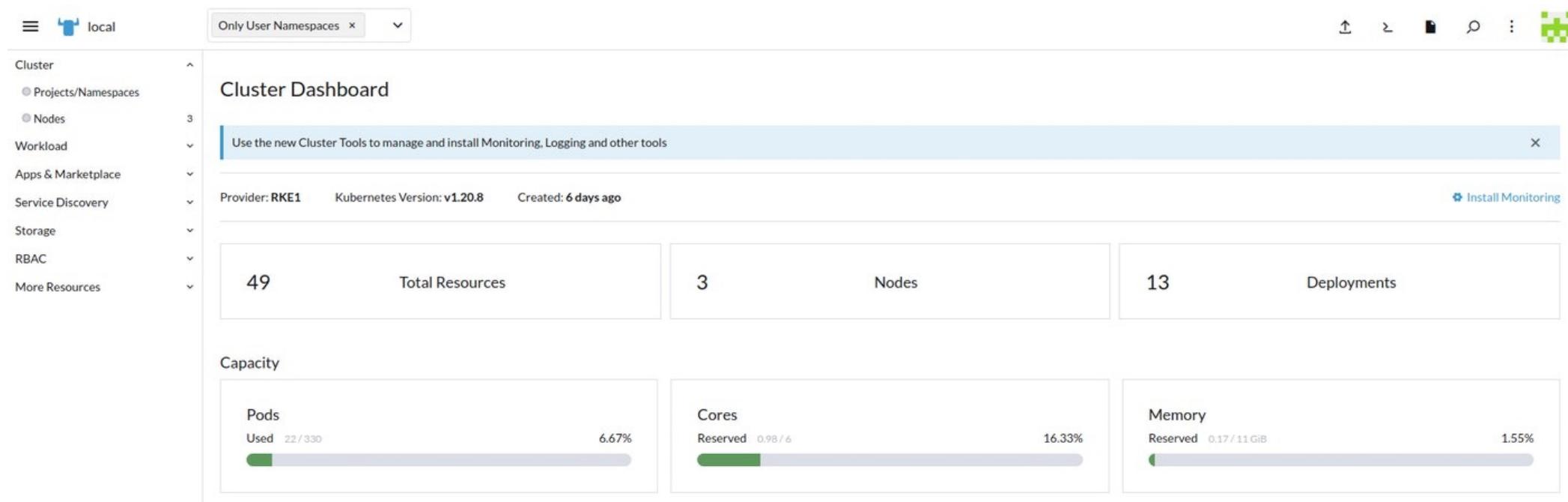


Why use Rancher?

- Rancher can help organizations maintain visibility into its Kubernetes infrastructure.
- Application teams can work in their own environments with minimal organizational constraints.
- Rancher's CD can manage deployments to multiple clusters for simultaneous release.
- Multi-cloud and Hybrid Cloud Clusters: Rancher can manage clusters in AWS, Azure, GCP, Digital Ocean, as well as clusters in data centres.
- Rancher can enforce security controls on all clusters to ensure compliance requirements are met.
- Rancher can manage clusters at edge locations like satellite offices, oil rigs, and Internet of Things (IoT) devices.

A Quick Look at Rancher

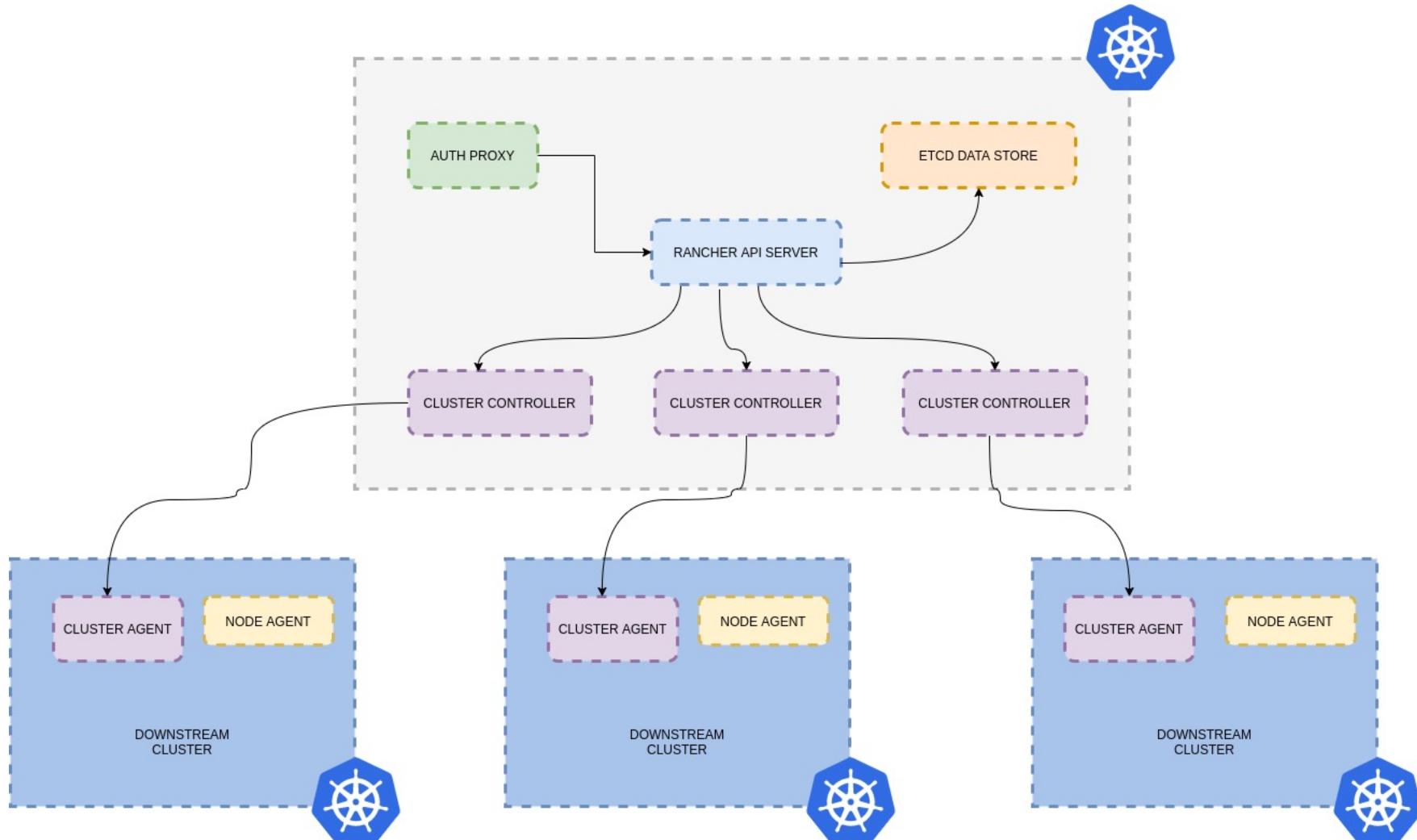
Let's take a look at Rancher.





The Rancher Architecture

The Rancher Architecture



Authentication Proxy

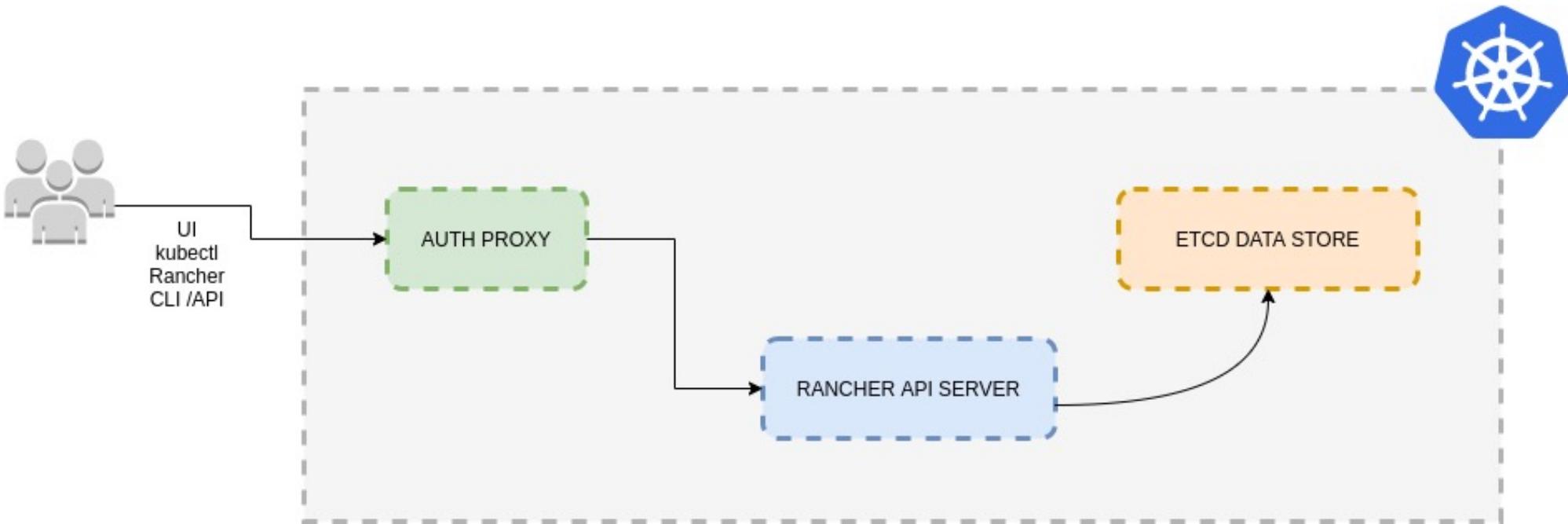


Authentication Proxy

It is known as the gate keeper. It receives requests from users who want to do something.

- Authenticates the user.
- Forwards the request on behalf of the user to the Rancher API server and in turn to the downstream cluster's API server.
- The downstream cluster API server runs an authorization step on that request and if it passes, then the control plane will carry out the request.

Rancher API Server



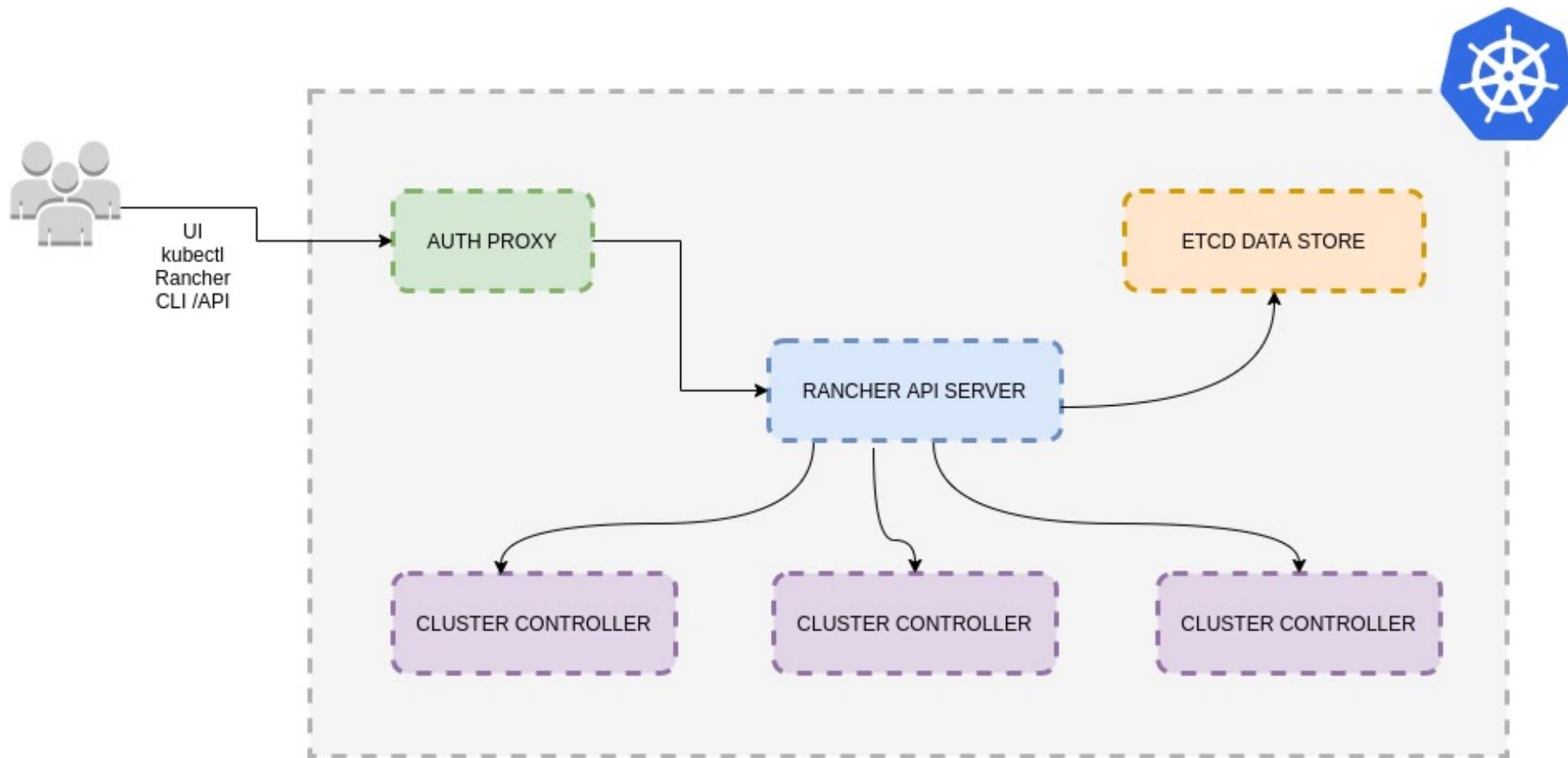
Rancher API Server

The Rancher API Server is built on top of the Kubernetes API server and an etcd database.

Some core functions include:

- User Management – Manages user identities that correspond with external auth providers like ActiveDirectory, Azure AD, Google, Okta, Ping Identity, GitHub, etc.
- Authorization – Manages access control and security policies.
- Cluster Provisioning – The Rancher API Server is the engine behind Rancher's cluster provisioning capabilities with existing nodes.
- Project Management – Projects in Rancher are a group of namespaces.

Cluster Controller



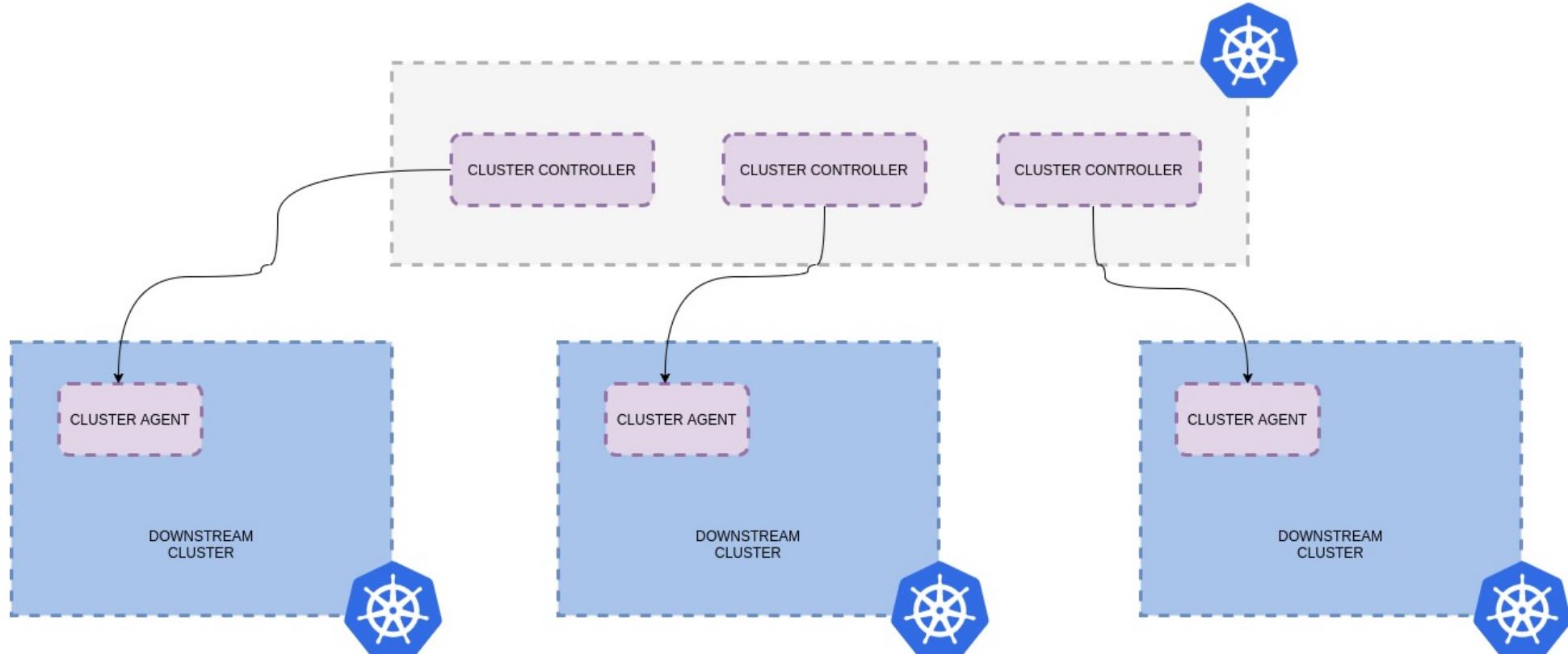
Cluster Controller

This component runs in the Rancher management cluster (or Rancher server). There is a cluster controller for every downstream cluster being managed by Rancher.

- It is responsible for watching resource changes in the downstream cluster.
- Controls downstream cluster state making sure it matches the desired state.
- Configuring access control policies.
- Used by Rancher API Server to provision clusters.

By default the Cluster Controller communicates with the cluster agent. If the cluster agent is unavailable, it can use a node agent as a fail back channel.

Cluster Agent

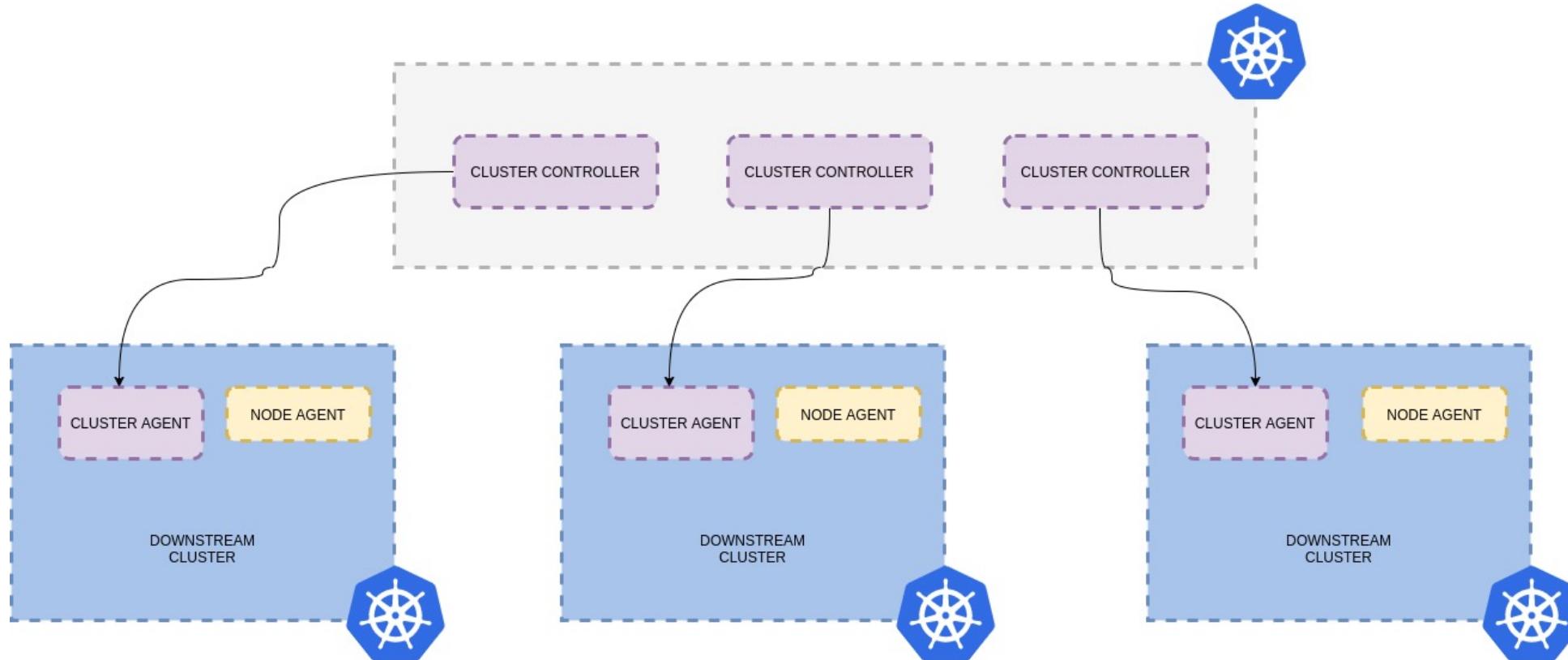


Cluster Agent

It handles various things such as connecting to the Kubernetes API of the downstream clusters that Rancher launched or imported.

- Manages workloads within each cluster.
- Applying the roles and bindings defined in each cluster's global security policy.
- Communicating through the tunnel between the Rancher server and the cluster.

Node Agent



Node Agent

This runs as a DaemonSet with one pod in every node of the cluster. Its primary function is to function with node specific functions like upgrading K8s or restoring etcd snapshots.

It can also fulfil the function of the cluster agent in the case that it is unavailable. It will establish a tunnel back to the Rancher Server.



Q & A



Thank You

© 2020 SUSE LLC. All Rights Reserved. SUSE and the SUSE logo are registered trademarks of SUSE LLC in the United States and other countries. All third-party trademarks are the property of their respective owners.



Up & Running: Rancher

Installing Rancher & Initial Setup Part 1

Instructor



Luke Mwila

I'm a Senior Technical Evangelist at SUSE and I'm an AWS Container Hero. I specialize in cloud and DevOps engineering and cloud-native technologies.

I'm passionate about sharing knowledge through various mediums and engaging with the developer community at large.

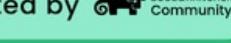
Related Events

Presented by  SUSE & RANCHER Community

Introduction to Kubernetes



Learn the basics of Kubernetes in this one-hour session.

Presented by  SUSE & RANCHER Community

Bring It On... with Luke!

Thursdays at 8am PT

→ Bring your questions—whether they are from this week's Master Class or general questions about Kubernetes and Cloud Native. With your host, Luke Mwila, Principal Technical Evangelist in the SUSE & Rancher Community.



Course Session Format

1. All classes will be live sessions held on Crowdcast every Wednesdays at 8am PT (for the next 8 weeks).
2. Each session will be between 1hr and 1 hr 15 minutes long.



Agenda

1. Developing & Testing with Rancher
2. Installing Rancher with Docker
3. Installing Rancher with K3D



Developing & Testing with Rancher



Kubernetes & Rancher

Both Kubernetes and Rancher are infrastructure agnostic.

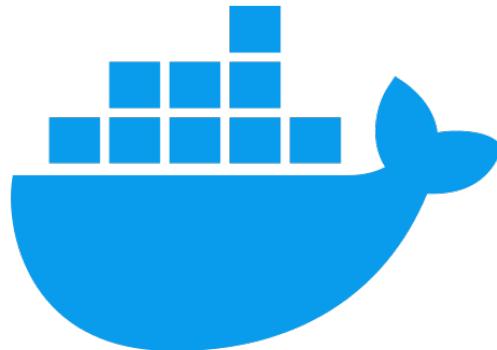


Developing & Testing with Rancher

When you get started with Rancher, you'll want to go through the typical lifecycle of having a safe space test and break things in an isolated environment.

It's part of the journey towards optimization and your desired configuration.

You can either set up your development Rancher server locally or in the cloud.



Installing Rancher with Docker



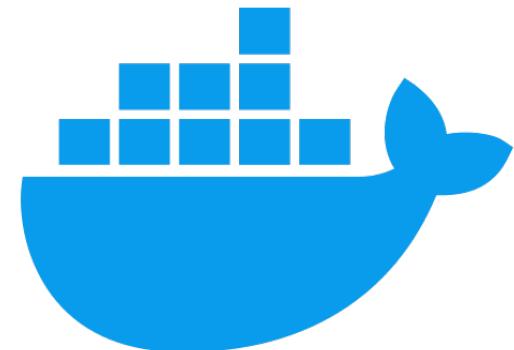
Installing Rancher with Docker

Similar to K3d, a single node K8s cluster is set up and runs inside a Docker container. In a production environment, you should be making use of a *Highly Available K8s cluster for Rancher*.

Rancher as an container-based application is deployed on the single node cluster.

So you end up with *containers within a container*.

```
docker run -d --restart=unless-stopped \
-p 80:80 -p 443:443 \
--privileged \
rancher/rancher:latest
```

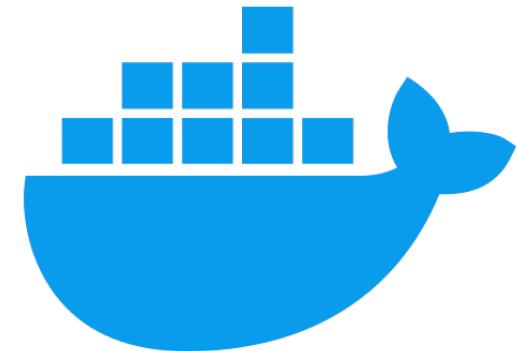
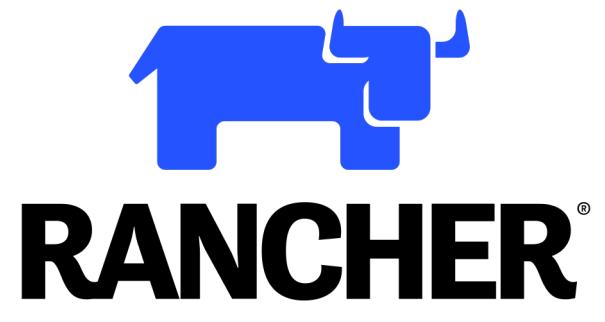


SSL Options for Rancher

SSL (Secure Sockets Layer) is required when using Rancher. This will secure all communication that users have with Rancher and interactions with the cluster.

- Rancher generated self-signed certificate (default)
- Bring your own self-signed certificate
- Bring your own certificate signed by recognized CA
- Let's Encrypt certificate

You can mount your certs into the containers using Docker's volumes feature.





Installing Rancher with K3D

Steps for Installing Rancher

1. Rancher relies on [cert-manager](#) to issue certificates from Rancher's own generated CA or to request Let's Encrypt certificates.
2. Create the `cattle-system` namespace.
3. Add Helm repository and install Rancher with Helm.
4. Update your local host resolution configuration.





Thank You

© 2020 SUSE LLC. All Rights Reserved. SUSE and the SUSE logo are registered trademarks of SUSE LLC in the United States and other countries. All third-party trademarks are the property of their respective owners.



Up & Running: Rancher

Installing Rancher & Initial Setup Part 2

Instructor



Luke Mwila

I'm a Senior Technical Evangelist at SUSE and an AWS Container Hero. I specialize in cloud and DevOps engineering and cloud-native technologies.

I'm passionate about sharing knowledge through various mediums and engaging with the developer community at large.

Related Events

Presented by  SUSE & RANCHER Community

Introduction to Kubernetes



Learn the basics of Kubernetes in this one-hour session.

Presented by  SUSE & RANCHER Community

Bring It On... with Luke!

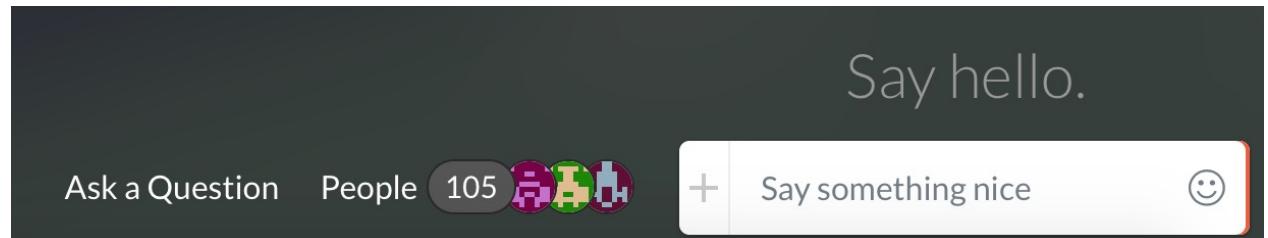
Thursdays at 8am PT

→ Bring your questions—whether they are from this week's Master Class or general questions about Kubernetes and Cloud Native. With your host, Luke Mwila, Principal Technical Evangelist in the SUSE & Rancher Community.



Crowdcast

- Ask questions and upvote
- Look out for polls
- Say something nice in the chat
- Session is recorded



Course Session Format

1. All classes will be live sessions held on Crowdcast every Wednesdays at 8am PT (for the next 8 weeks).
2. Each session will be between 1hr and 1 hr 15 minutes long.



Agenda

1. HA Rancher Install with RKE
2. Centralized RBAC and Auth Providers
3. Rancher Backup & Restore
4. Backup & Restore
5. Q & A



HA Rancher Install with RKE



Kubernetes & Rancher

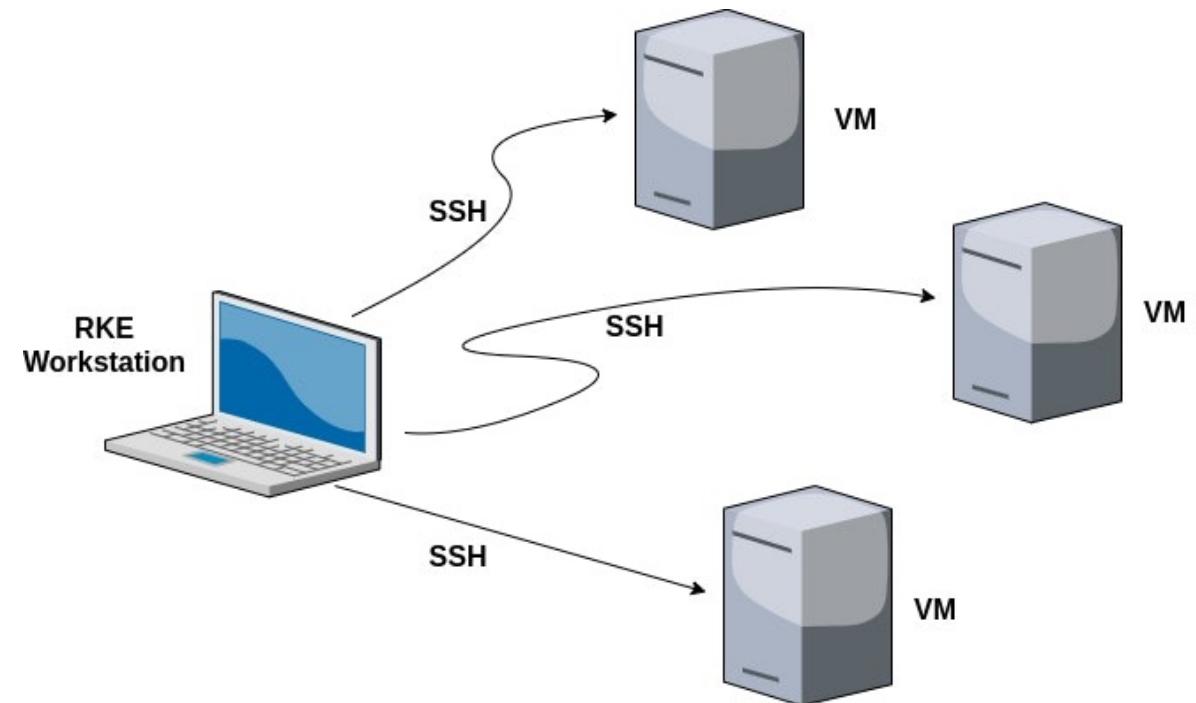
In a production environment, you should be making use of a *Highly Available K8s cluster for Rancher* to avoid a single point of failure.



Creating a K8s Cluster with RKE

Our machine will fulfil the role of the RKE workstation, which will connect to each of the nodes (VMs) via SSH to establish a tunnel to access the docker socket.

When your cluster has been provisioned, Rancher will be installed using Helm.



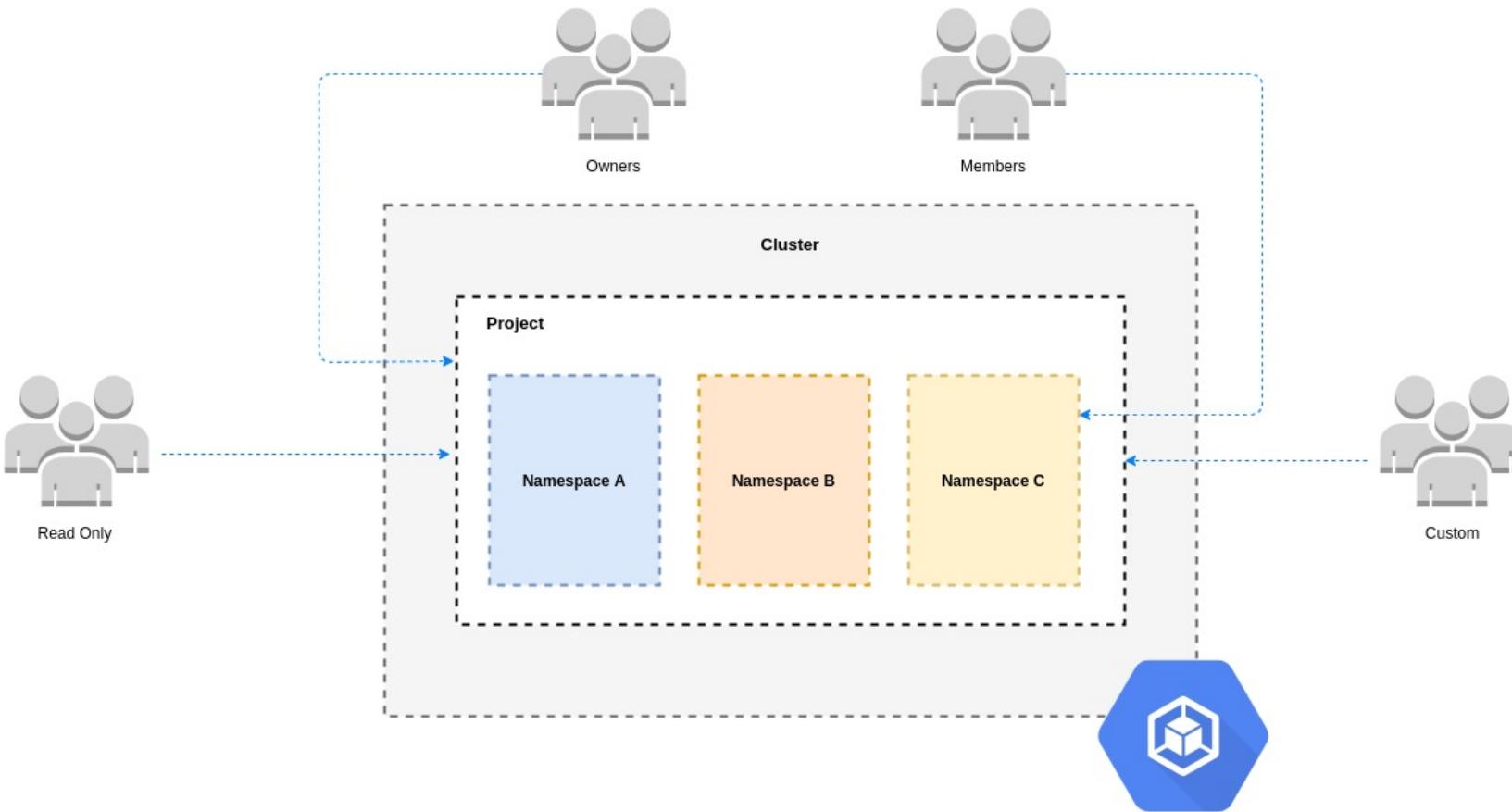
Centralized RBAC and Auth Providers

Authentication & RBAC

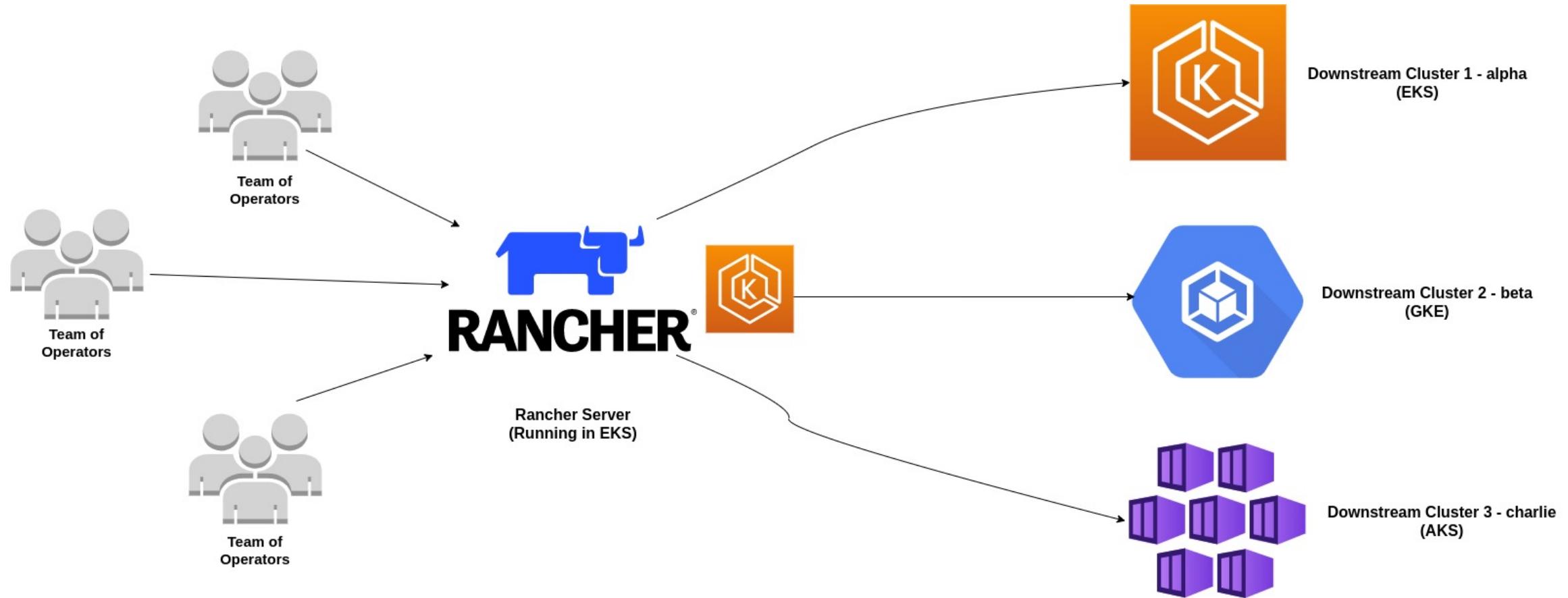
RBAC authorization uses the `rbac.authorization.k8s.io` [API group](#) and consists of four API objects.

1. Role – Used to determine which operations can be carried out on which resources in a specific namespace.
2. ClusterRole – Used to determine which operations can be carried out on which resources in the scope of the entire cluster.
3. RoleBinding – Used to determine which users or service accounts are authorized to carry out operations on resources in a given namespace as specified in a Role.
4. ClusterRoleBinding – Used to determine which users or service accounts are authorized to carry out operations on resources across the scope of the cluster as specified in a ClusterRole.

Projects & Namespaces



Security with Centralized RBAC



Backup & Restore



Backup Rancher

The rancher-backup operator is used to backup and restore Rancher on any Kubernetes cluster.

This application is a Helm chart, and it can be deployed through the Rancher Apps & Marketplace page, or by using the Helm CLI.

Backup file is saved as a tar file.

rancher/backup-restore-operator



11

Contributors

25

Issues

35

Stars

23

Forks







Thank You

© 2020 SUSE LLC. All Rights Reserved. SUSE and the SUSE logo are registered trademarks of SUSE LLC in the United States and other countries. All third-party trademarks are the property of their respective owners.



Up & Running: Rancher Cluster Operations Part 1

Instructor



Lukonde (Luke) Mwila

I'm a Senior Technical Evangelist at SUSE. I specialize in cloud and DevOps engineering and cloud-native technologies.

I'm passionate about sharing knowledge through various mediums and engaging with the developer community at large.

Other Events

Presented by SUSE & RANCHER Community

Introduction to Kubernetes



Nigel Poulton:
Ask Me Anything
About Getting Started
with Kubernetes!

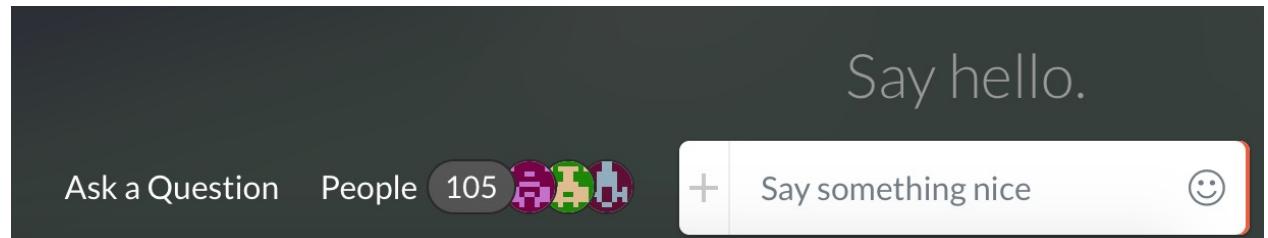
→ TWO FREE SESSIONS:
Oct 27 at 12am PT/7am GMT AND
Oct 28 at 9am PT

Presented by SUSE & RANCHER Community

Hands on
QUICK START
KUBERNETES
2021 Edition nigelpoulton

Crowdcast

- Ask questions and upvote
- Look out for polls
- Say something nice in the chat
- Session is recorded



Agenda

1. Importing Kubernetes Clusters
2. Creating Kubernetes Clusters
3. Modifying Kubernetes Clusters
4. The kubectl Shell
5. Working with the Rancher CLI/API
6. Q & A



Importing Kubernetes Clusters

Creating Kubernetes Clusters



Modifying Kubernetes Clusters



Working with the Rancher CLI/API





Thank You

© 2020 SUSE LLC. All Rights Reserved. SUSE and the SUSE logo are registered trademarks of SUSE LLC in the United States and other countries. All third-party trademarks are the property of their respective owners.



Up & Running: Rancher Cluster Operations Part 2

Instructor



Luke Mwila

I'm a Senior Technical Evangelist at SUSE and an AWS Container Hero. I specialize in cloud and DevOps engineering and cloud-native technologies.

I'm passionate about sharing knowledge through various mediums and engaging with the developer community at large.

Other Related Events

Presented by  SUSE & RANCHER Community

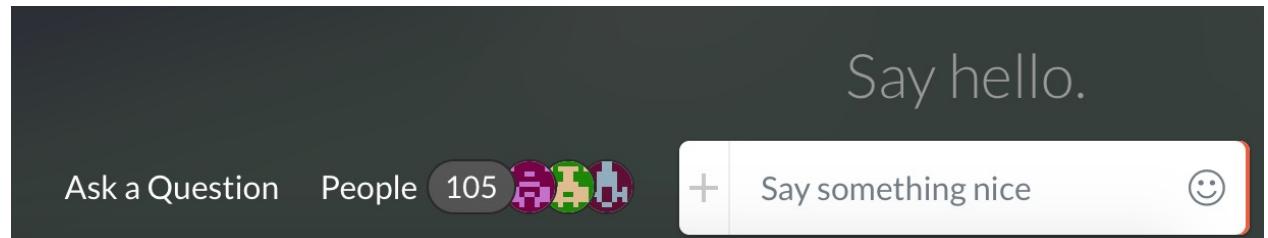
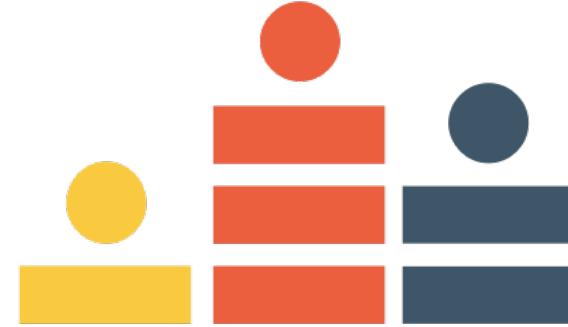
Introduction to Kubernetes



Learn the basics of
Kubernetes in this free
one-hour free class

Crowdcast

- Ask questions and upvote
- Look out for polls
- Say something nice in the chat
- Session is recorded



Agenda

1. Downstream Cluster Backup, Restore and Upgrade
2. Certificate Management
3. CIS Scanning
4. Q & A



Downstream Cluster Backup, Restore and Upgrade



Cluster Backup, Restore and Upgrade

Before upgrading the Kubernetes version on your downstream cluster, make sure to [back up your etcd database](#).

If something goes wrong, restore the snapshot you took prior to initiating an upgrade.

RKE will upgrade and restart your cluster components.



Certificate Management



Securing the API Server & Cluster Components

The API server is the central component of the control plane.

To secure the API server, you would make use of x509 PKI certificates for authentication over TLS.

The goal is to ensure all the cluster components have the required keys and certificates for secure communication across the board.

Securing the API Server & Cluster Components

Kubernetes requires PKI certs for the following operations:

- Client certificates for the kubelet to authenticate to the API server
- Server certificate for the API server endpoint
- Client certificates for administrators of the cluster to authenticate to the API server
- Client certificates for the API server to talk to the kubelets
- Client certificates for the API server to talk to etcd
- Client certificate/kubeconfig for the controller manager to talk to the API server
- Client certificate/kubeconfig for the scheduler to talk to the API server
- Client and server certificates for the front-proxy

Securing the Kubernetes Datastore

The K8s datastore is where all the cluster configuration and desired state is stored.

Security of your datastore will look very similar to security of your cluster components.

etcd has security features based on PKI keys and certificates.

The main goal is to ensure all data in transit between peers (other etcd instances) is encrypted with TLS, and has another set of credentials for secure communication with the API server.



CIS Scanning

CIS Kubernetes Benchmark

You should test whether the cluster is deployed and configured according to the Kubernetes benchmark developed by the CIS (Center for Internet Security).

To execute this test, you will use **kube-bench**, which is a tool that is used to run CIS Kubernetes benchmark checks.

The `rancher-cis-benchmark` app leverages [kube-bench](#), an open-source tool from Aqua Security, to check clusters for CIS Kubernetes Benchmark compliance. Also, to generate a cluster-wide report, the application utilizes [Sonobuoy](#) for report aggregation.







Thank You

© 2020 SUSE LLC. All Rights Reserved. SUSE and the SUSE logo are registered trademarks of SUSE LLC in the United States and other countries. All third-party trademarks are the property of their respective owners.



Up & Running: Rancher Cluster Operations Part 3

Instructor



Luke Mwila

I'm a Senior Technical Evangelist at SUSE and an AWS Container Hero. I specialize in cloud and DevOps engineering and cloud-native technologies.

I'm passionate about sharing knowledge through various mediums and engaging with the developer community at large.

Other Related Events

Presented by  Community

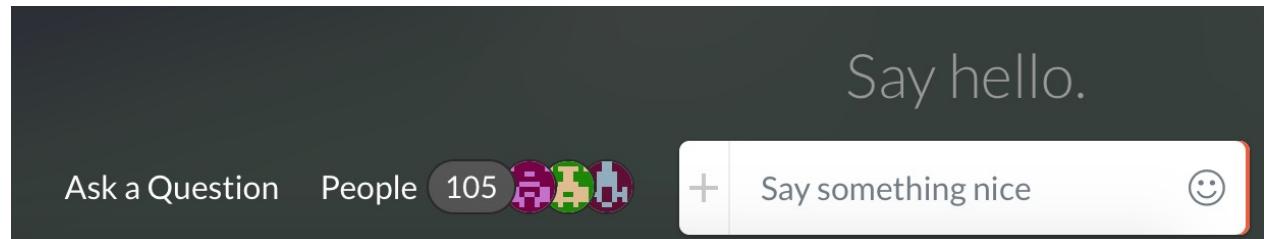
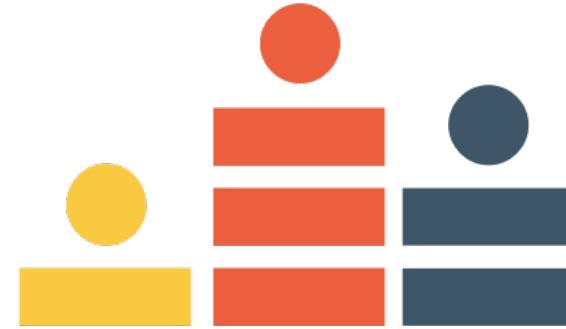
Introduction to Kubernetes



Learn the basics of
Kubernetes in this free
one-hour free class

Crowdcast

- Ask questions and upvote
- Look out for polls
- Say something nice in the chat
- Session is recorded



Agenda

1. Logging
2. Monitoring
3. Q & A



Logging



Understanding Logging

Logging allows you to keep a record of data input, output and processes in your applications and the underlying infrastructure.

Rancher can be configured with a logging system and a destination system to send log information.



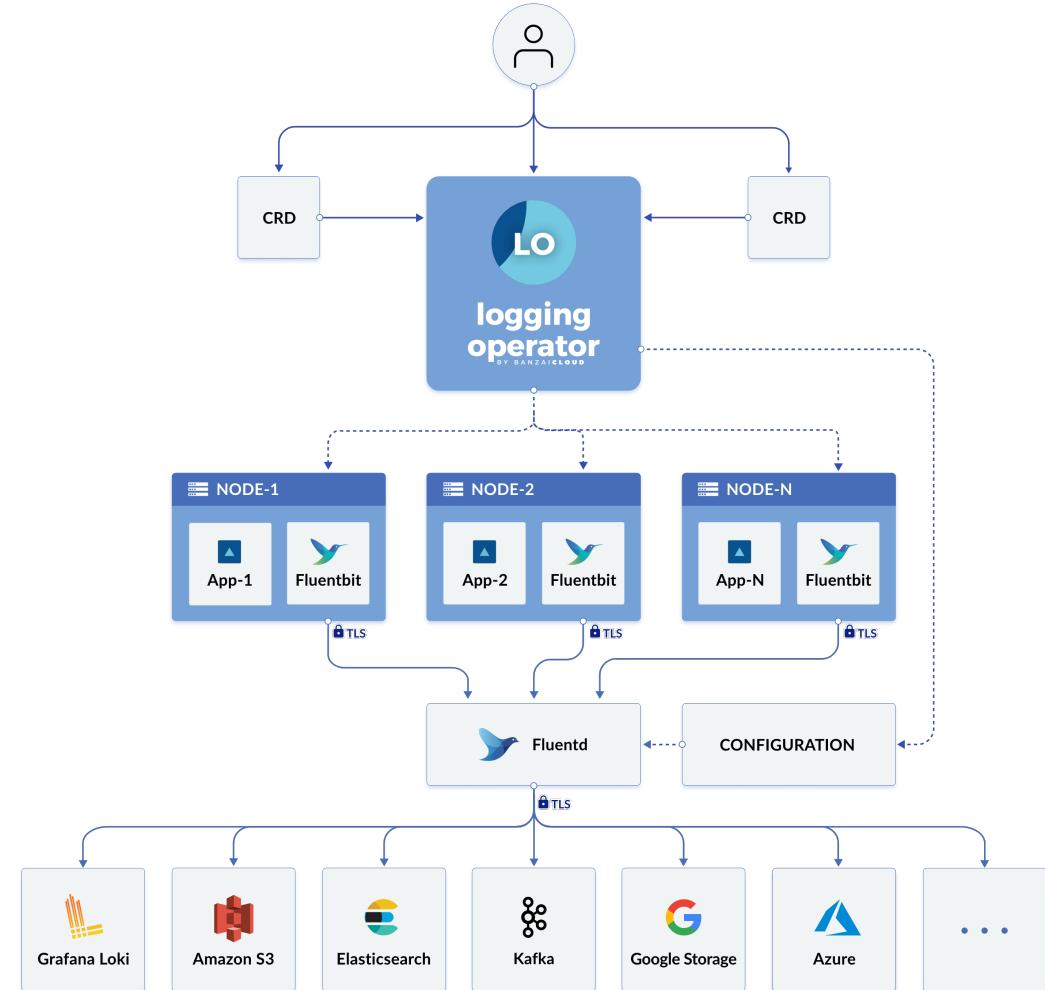
Benefits of Logging

Logging allows administrators to:

- Capture and analyse the state of the cluster
- Save the logs to a safe location outside of the cluster nodes
- Stay informed of events like a container crashing, a Pod eviction, or a Node dying
- Helps with debugging and troubleshooting problems

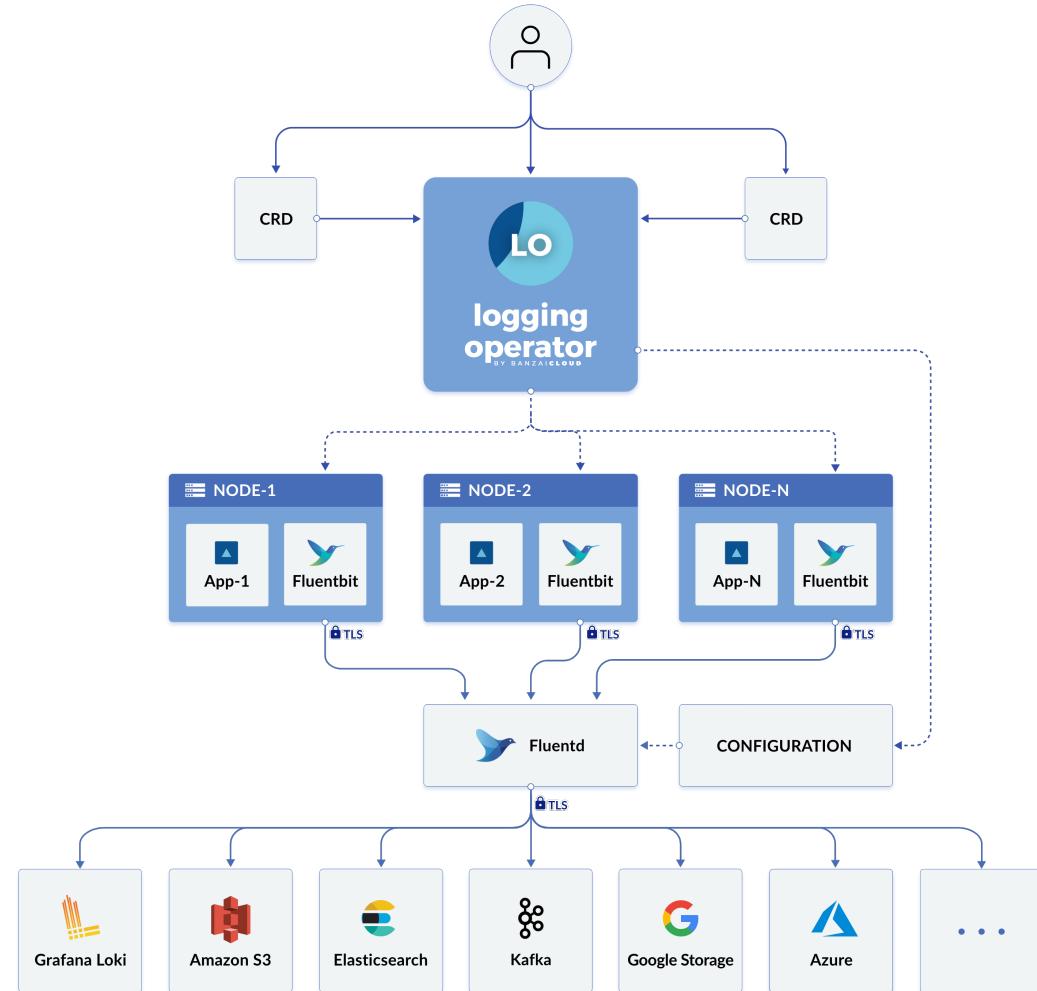
Logging Architecture

- Uses Banzai Cloud Logging operator
- Provides fine-grained logging for administrators and users
- Ability to write logs to multiple outputs
- Lots of configuration options
- Log filtering



Output Configuration

- Output destination for logging must be configured
- Flow must be configured



Monitoring

Visibility & Insight with Monitoring

Getting visibility and insight into clusters and workloads is accomplished with monitoring.

Benefits of this include:

- Troubleshooting
- Optimize cluster performance
- Manage costs
- Security

Rancher monitoring stack consists of Prometheus, Grafana, and Alertmanager



Grafana

Monitoring Stack: Prometheus

Prometheus is a pull based monitoring system that accesses specific HTTP endpoints and scrapes the metrics from the endpoints.

Scraping is just a request that is made to the HTTP endpoint. Then the response to the scrape is parsed and ingested into storage.



Monitoring Stack: Prometheus

Prometheus Client Libraries

Before you can monitor your application, you need to add instrumentation to your code. Typically software teams use a client library that is specific to their code.

These libraries gather metrics on your application and format it in such a way that Prometheus can understand it.

You will have to set up a metrics endpoint for your application. This is the endpoint that Prometheus is going to scrape.



Monitoring Stack: Prometheus

Prometheus Exporters

Exporters are used to instrument applications that we don't have the source code for. You can install an exporter and it will live next to the application that you want to get metrics from.

There are different types of exporters.

You can also collect hardware and software metrics and this is where the Node exporter comes in. Typically, the Node exporter will go and live on the host server where we want to gather metrics.



Monitoring Stack: Prometheus

Prometheus allows the administrator to:

- Monitor the state and processes of the cluster nodes, Kubernetes components, and software deployments
- Collect precomputed time series based on metrics
- Define alerts based on the collected metrics
- Notifications can be sent to multiple destinations by Alertmanager



Monitoring Stack: Prometheus

You can also set up rules and alerting. There are two types of rules and alerting.

- Alerting rules – allow you to create an expression that when the condition is met, you will get a notification.
- Recording rules – allow you to use PromQL to create an expression. The expression will be evaluated and the results of this expression will be ingested into the storage engine.



Monitoring Stack: Prometheus

- Routes – The route configuration is the section of the Alertmanager custom resource that controls how the alerts fired by Prometheus are grouped and filtered before they reach the receiver.
- Receivers – The [Alertmanager Config](#) Secret contains the configuration of an Alertmanager instance that sends out notifications based on alerts it receives from Prometheus.



Monitoring Stack: Notifiers

These are services that inform you of alert events. They're configured at a cluster level and are then available at a cluster and project level for users to connect to the specific alerts.

Rancher integrates with Slack, Email OpsGenie, Webhook, PagerDuty



Monitoring Stack: Grafana

Grafana allows the administrator to:

- See the current state or collected metrics as graphical representation
- Create custom dashboards to suit the needs



Grafana



Q & A



Thank You

© 2020 SUSE LLC. All Rights Reserved. SUSE and the SUSE logo are registered trademarks of SUSE LLC in the United States and other countries. All third-party trademarks are the property of their respective owners.



Up & Running: Rancher Application Management

Instructor



Luke Mwila

I'm a Senior Technical Evangelist at SUSE and an AWS Container Hero. I specialize in cloud and DevOps engineering and cloud-native technologies.

I'm passionate about sharing knowledge through various mediums and engaging with the developer community at large.

Other Related Events

Presented by  SUSE & RANCHER Community

Introduction to Kubernetes



Presented by  SUSE & RANCHER Community

App Modernization: When and How Far to Modernize

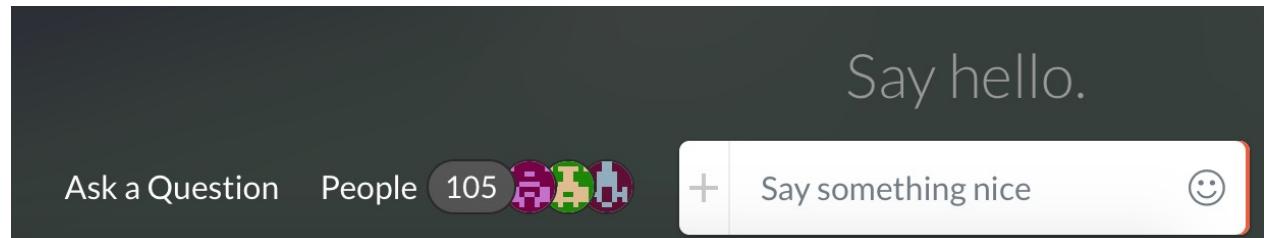
Hosted by: Robert Sirchia

NOVEMBER 18TH AT 8AM PT



Crowdcast

- Ask questions and upvote
- Look out for polls
- Say something nice in the chat
- Session is recorded



Agenda

1. Project & Namespace Management
2. Deploying Workloads
3. Service Discovery
4. Persistent Storage
5. Continuous Delivery with Rancher Fleet
6. Q & A



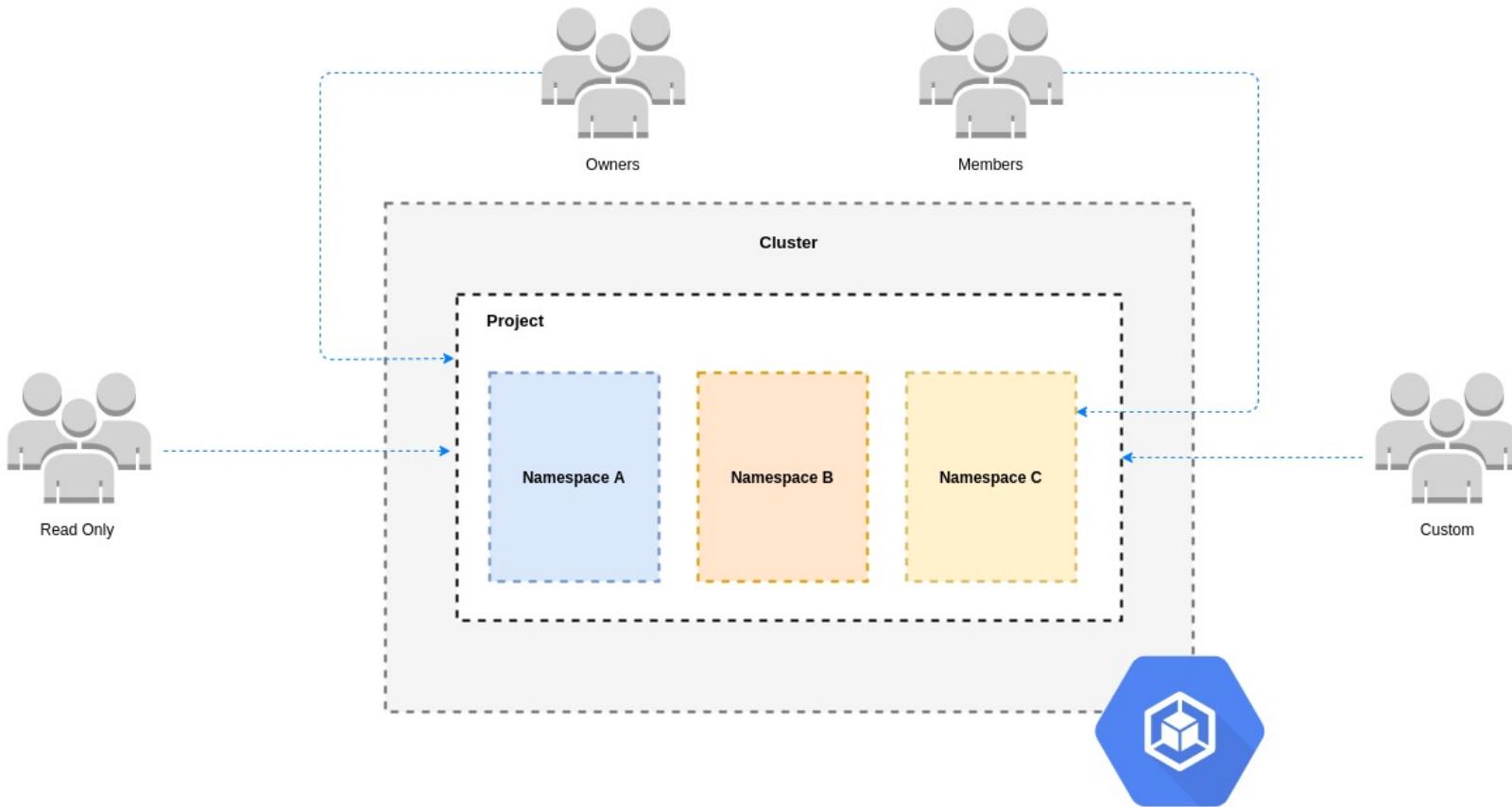
Project & Namespace Management

Projects & Namespace Management

- Namespaces – Namespaces (often referred to as virtual clusters within a cluster) allow division of access control and resource quotas.
- Projects – Custom Rancher objects used to group namespaces. Assigning resources at the project level allows each namespace in the project to use them.



Projects & Namespace Management



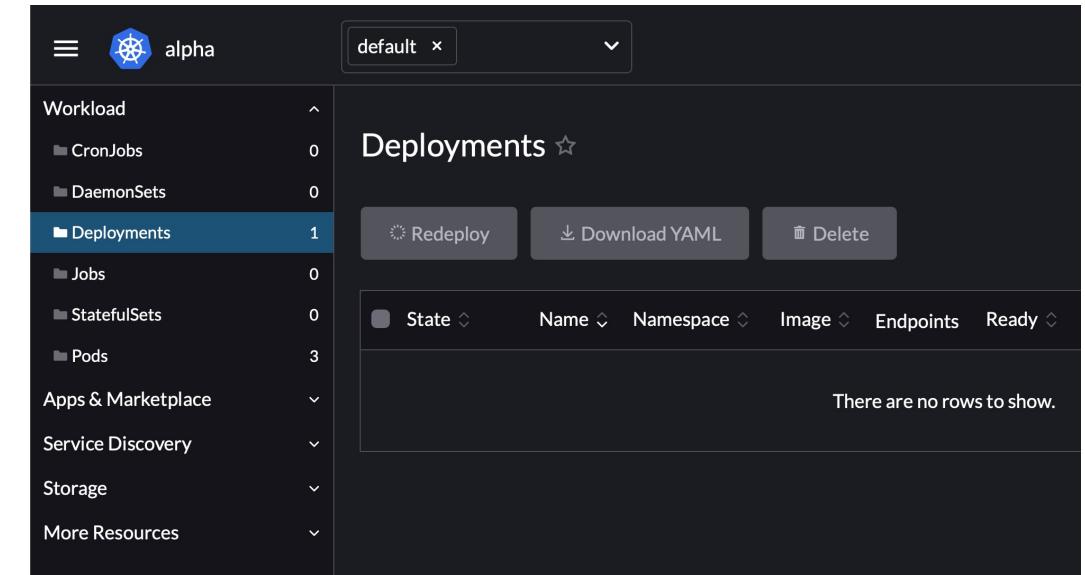
Deploying Workloads



Deploying Workloads

Rancher offers an intuitive approach to deploying and managing your workloads and the different K8s resources that they consist of.

- 1) Create workload resources by type OR
- 2) Create workloads using Import YAML



The screenshot shows the Rancher web interface. In the top left, there's a sidebar with a navigation menu:

- Workload
 - CronJobs: 0
 - DaemonSets: 0
 - Deployments: 1 (highlighted in blue)
 - Jobs: 0
 - StatefulSets: 0
 - Pods: 3
- Apps & Marketplace
- Service Discovery
- Storage
- More Resources

In the top right, there's a dropdown menu showing "alpha" and "default". Below the sidebar, the main content area has a title "Deployments" with a star icon. It contains three buttons: "Redeploy", "Download YAML", and "Delete". At the bottom of the table area, there are sorting columns for State, Name, Namespace, Image, Endpoints, and Ready, each with a downward arrow icon. A message at the bottom right of the table says "There are no rows to show."

Service Discovery and Load Balancers

Persistent Storage

Persistent Storage

The file system for containers is ephemeral and only lasts as long as the container exists.

Stateful applications need a persistent mechanism for storing data.

- Volumes – Volumes allow you to store data outside the container file system while allowing the container to access the data at runtime.
- Persistent Volumes – Persistent volumes allow you to treat storage as an abstract resource and consume it using your Pods.

Persistent Storage

1. You have a container running in a Pod.
2. You setup a Persistent Volume Claim which binds to a Persistent Volume
3. Then the Persistent Volume implements some form of external storage.

Continuous Delivery with Rancher Fleet

Continuous Delivery with Rancher Fleet

Rancher Fleet uses a GitOps is a model for Continuous Delivery.

GitOps is the deployment of immutable infrastructure with Git as the single source of truth. It requires you to describe and observe systems with declarative configurations.

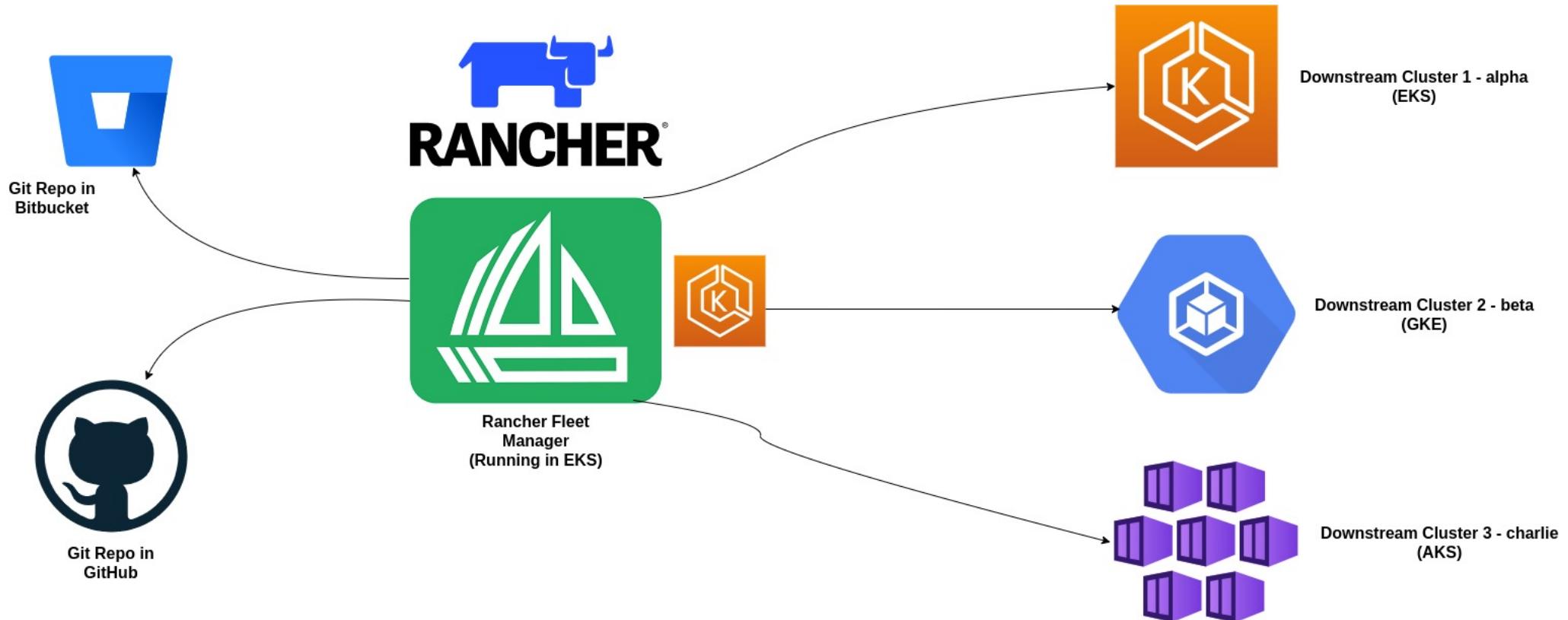
Benefits of this model include:

- Infrastructure as Code
- Code Reviews
- Declarative Paradigm
- Better Observability



FLEET

Deployments with Rancher Fleet







Thank You

© 2020 SUSE LLC. All Rights Reserved. SUSE and the SUSE logo are registered trademarks of SUSE LLC in the United States and other countries. All third-party trademarks are the property of their respective owners.



Up & Running: Rancher Where Do You Go From Here?

Instructor



Luke Mwila

I'm a Senior Technical Evangelist at SUSE and an AWS Container Hero. I specialize in cloud and DevOps engineering and cloud-native technologies.

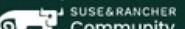
I'm passionate about sharing knowledge through various mediums and engaging with the developer community at large.

Other Important Events

Presented by  SUSE & RANCHER Community

Introduction to Kubernetes



Presented by  SUSE & RANCHER Community

ACADEMY CLASS

Accelerate Dev Workflows

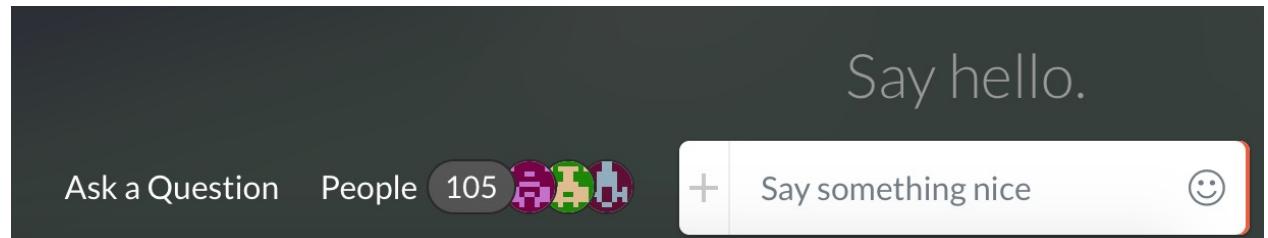
Learn to develop cloud native applications easier and faster



FREE 4-week class starts Wednesday, Dec. 1

Crowdcast

- Ask questions and upvote
- Look out for polls
- Say something nice in the chat
- Session is recorded



Agenda

1. Rancher Deployment Strategies
2. Troubleshooting Common Issues
3. Additional Learning Resources
4. FAQ
5. Q & A



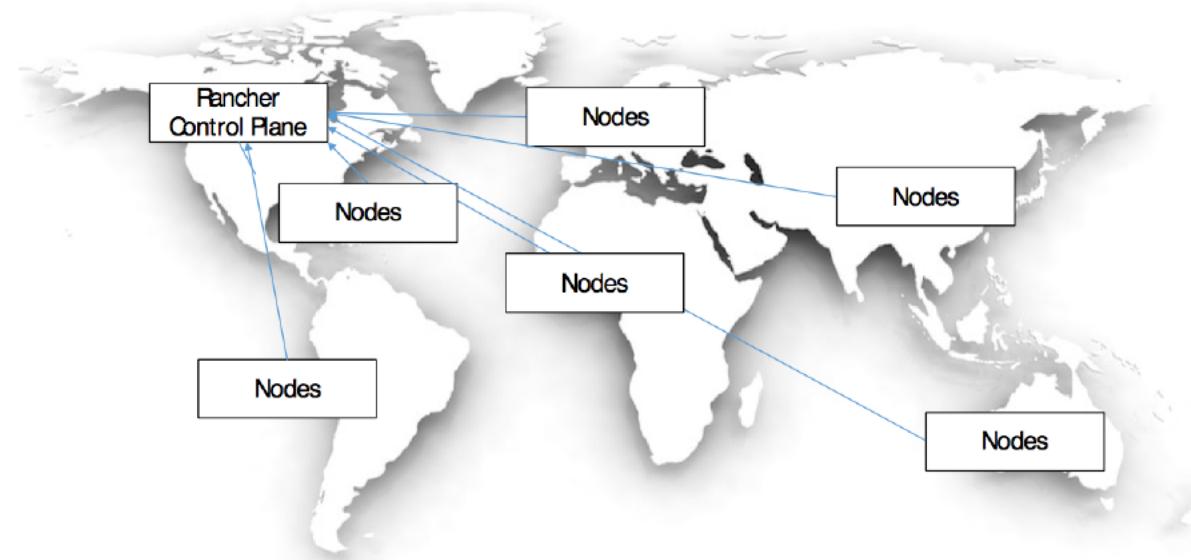
Rancher Deployment Strategy

Two recommended deployment strategies for Rancher:

- Hub & Spoke
- Regional

Hub & Spoke Deployment

In the hub and spoke model, a HA Rancher server is deployed to manage downstream clusters across the globe.



Hub & Spoke Deployment

Pros

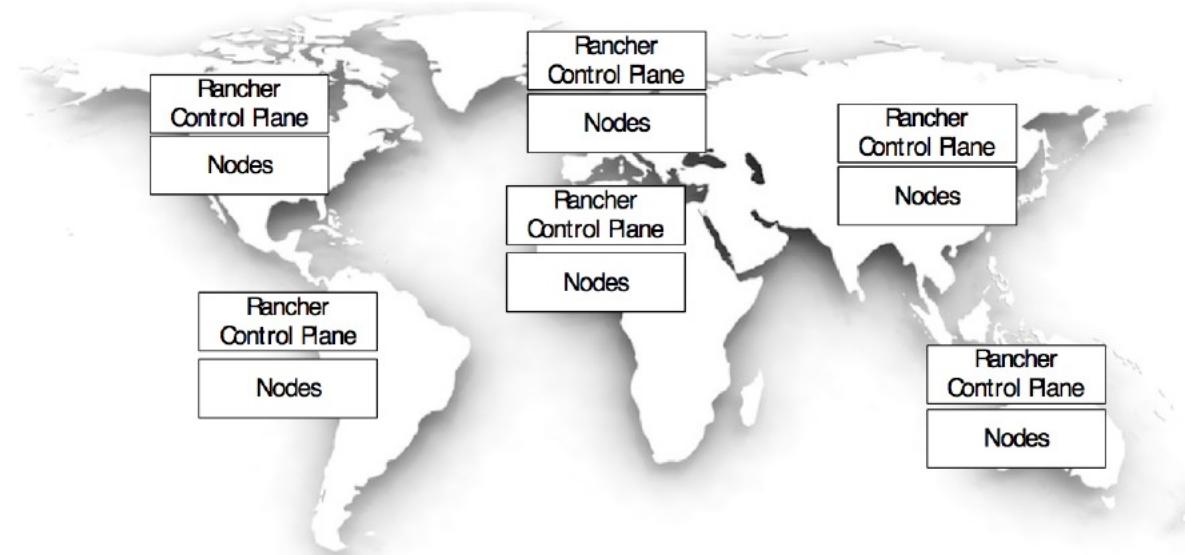
1. Environments could have nodes and network connectivity across regions.
2. Single control plane interface to view/see all regions and environments.
3. Kubernetes does not require Rancher to operate and can tolerate losing connectivity to the Rancher control plane.

Cons

1. Network latencies.
2. If the control plane goes out, global provisioning of new services is unavailable until it is restored.

Regional Deployment

In the regional deployment strategy, a Rancher control plane is deployed in close proximity to the compute nodes.



Regional Deployment

Pros

1. Rancher functionality in regions stay operational if a control plane in another region goes down.
2. Network latency is greatly reduced, improving the performance of functionality in Rancher.
3. Upgrades of the Rancher control plane can be done independently per region.

Cons

1. Overhead of managing multiple Rancher installations.
2. Visibility across global Kubernetes clusters requires multiple interfaces/panes of glass.
3. Deploying multi-cluster apps in Rancher requires repeating the process for each Rancher server.

Troubleshooting Common Issues

Important Namespaces

Troubleshooting usually happens in these namespaces:

- **cattle-system** – rancher deployment and pods.
- **ingress-nginx** – Ingress controller pods and services.
- **cert-manager** – cert-manager pods.

Troubleshooting Common Issues

1. Rancher Installation
2. Default Backend - 404
3. Registered Clusters



Rancher Installation

1. Check that the Rancher pods (3 replicas) are running.
2. Check container logs on the pods.
3. Check the namespace events.
4. Check the ingress.
5. Check the ingress logs.



Default Backend - 404

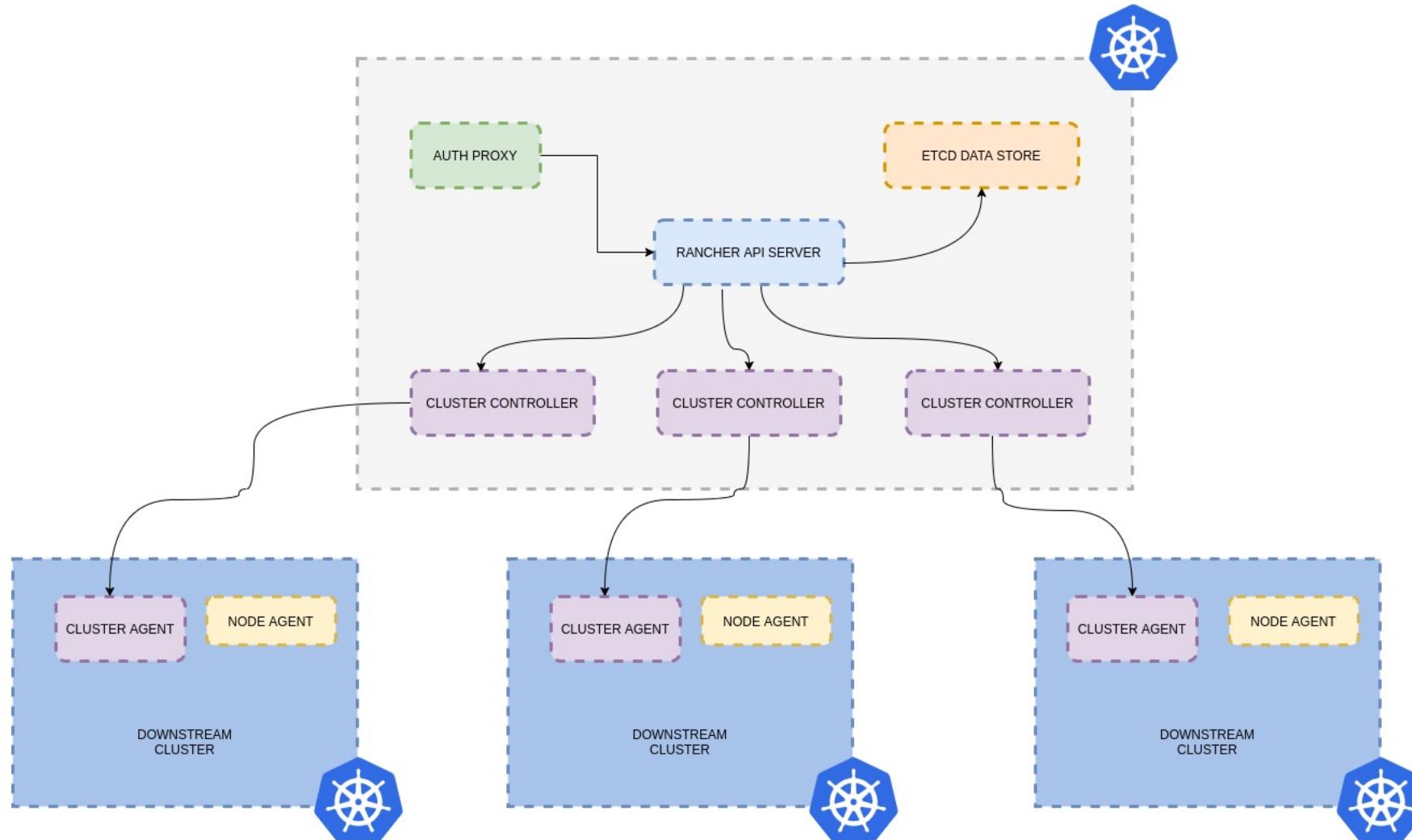
If the Ingress Controller is not forwarding traffic to your Rancher instance, check the following (to begin with):

1. Are Rancher pods running?
2. Are there issues with your SSL certificate (check browser details)?

If it's a cert issue, check all 3 parts of cert-manager:

- **cert-manager** pod in the **cert-manager** namespace
- **Issuer** object in the **cattle-system** namespace
- **Certificate** object in the **cattle-system** namespace

Registered Clusters



Registered Clusters

The Rancher server communicates and manages the downstream clusters via the Cluster Agent.

1. Check if the **cluster-agent-pod** is running or in a pending state.
2. Check the logs of the **cluster-agent-pod**.

```
kubectl -n cattle-system get pods -l app=cattle-cluster-agent -o wide
```

```
kubectl -n cattle-system logs -l app=cattle-cluster-agent
```

Additional Learning Resources



Additional Learning Resources

Accelerate your learning by collaborating with others in the Rancher community.

- SUSE & Rancher Community
- Rancher Users (Slack)
- Rancher GitHub Repository
- Rancher Documentation





FAQ

1. Do you need to know Kubernetes to use Rancher?
2. Can I import OpenShift Kubernetes clusters into v2.x?
3. If the Rancher server is deleted, how do I access my downstream clusters?
4. Can I install Rancher and deploy dedicated workloads on the same K8s cluster?
5. Can I deploy Rancher on a single node K8s cluster?





Thank You

© 2020 SUSE LLC. All Rights Reserved. SUSE and the SUSE logo are registered trademarks of SUSE LLC in the United States and other countries. All third-party trademarks are the property of their respective owners.