
AWS
re:Invent

CON224

# Improving usability for Amazon EKS clusters

Jesse Butler

Senior Developer Advocate

AWS Container Services

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

# Agenda

A new console experience for Amazon EKS

Extending lifecycle management with EKS Add-Ons

What’s next?



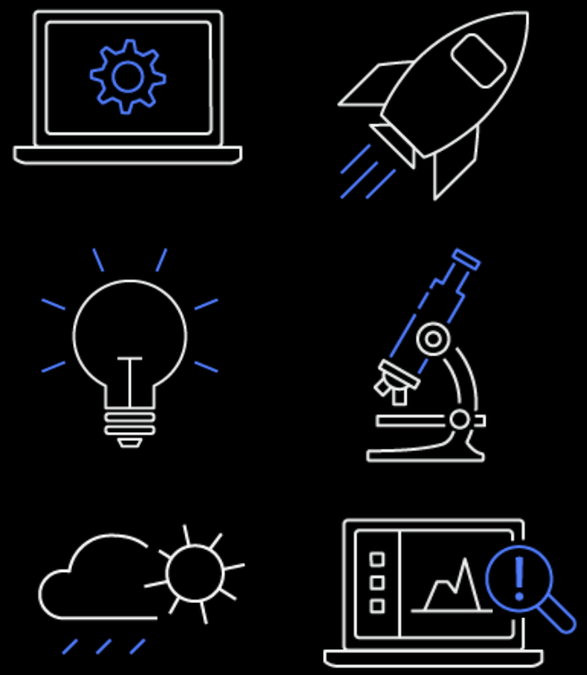
AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

# Amazon EKS is Kubernetes

Amazon EKS

- A native Kubernetes experience delivered as a service by AWS
- Users leverage an amazing ecosystem of open source solutions
- Amazon EKS provides native, well-integrated solutions into the AWS cloud



# “We consider user experience to be our top feature, understanding that simplicity is powerful…”

AWS Container Services Priorities



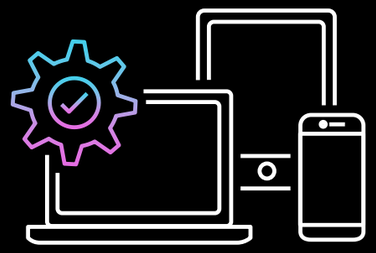
AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

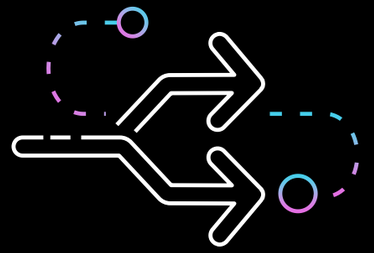
# Evolution of Amazon EKS

Amazon EKS

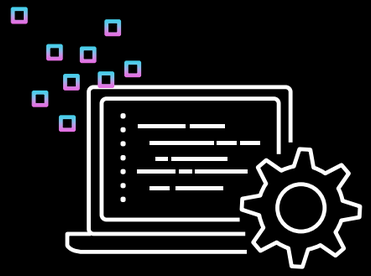
- Security, reliability, and availability
- Managed control plane, managed nodes, and AWS Fargate
- Lifecycle management of your clusters and their resources
- Visualizing your clusters, their resources, and their workloads



**Production ready**



**Native and upstream**

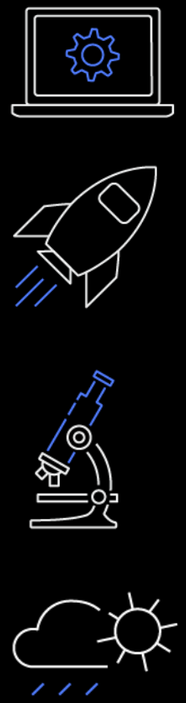


**OSS contribution**

aws

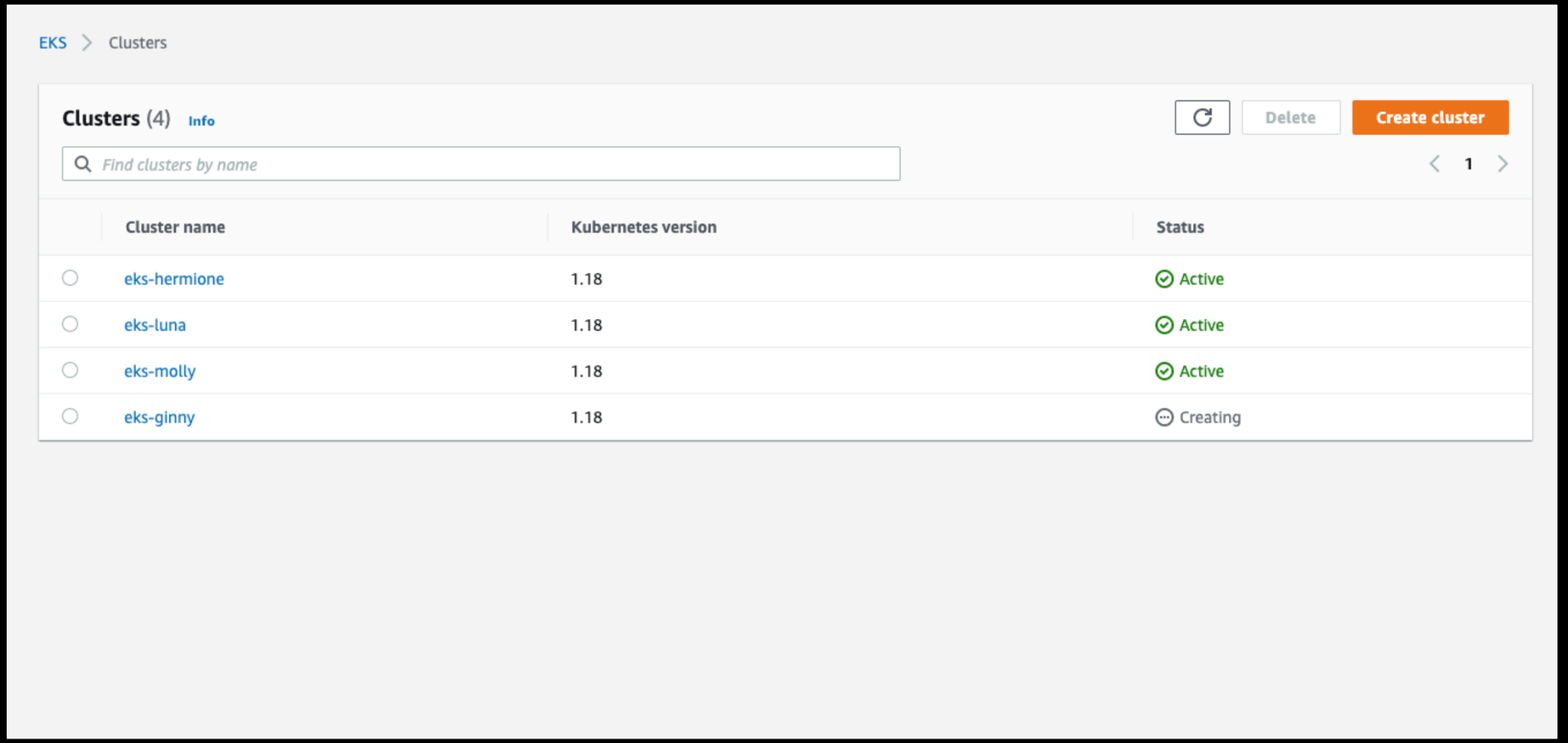
**Seamless integrations**

# Introducing: Next-generation EKS console

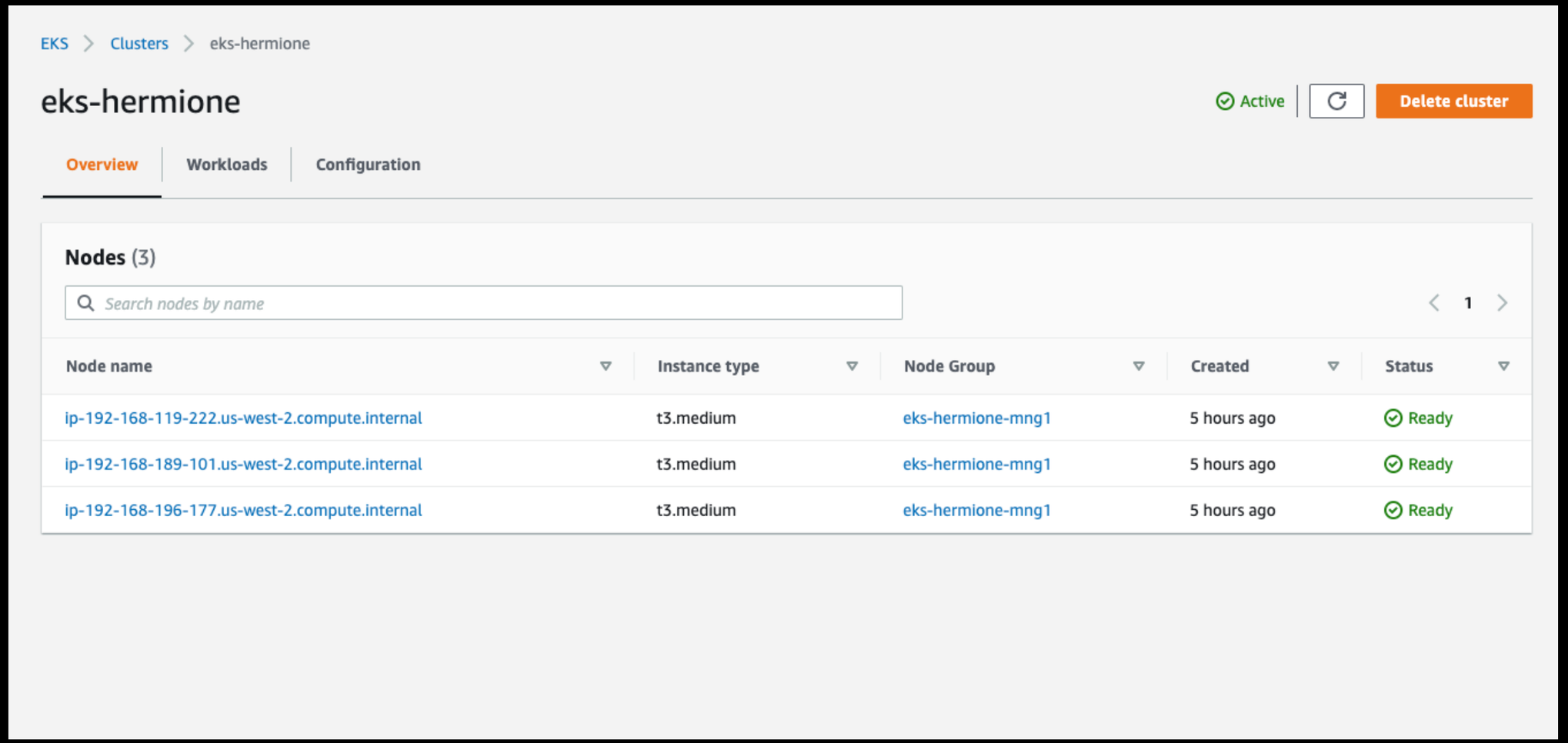


- Integrated Kubernetes object visualization
- Unified and intuitive connections between Kubernetes clusters and AWS cloud resources
- New Kubernetes node views
- Intuitive workflows for visualizing both cluster resources and in-cluster workloads

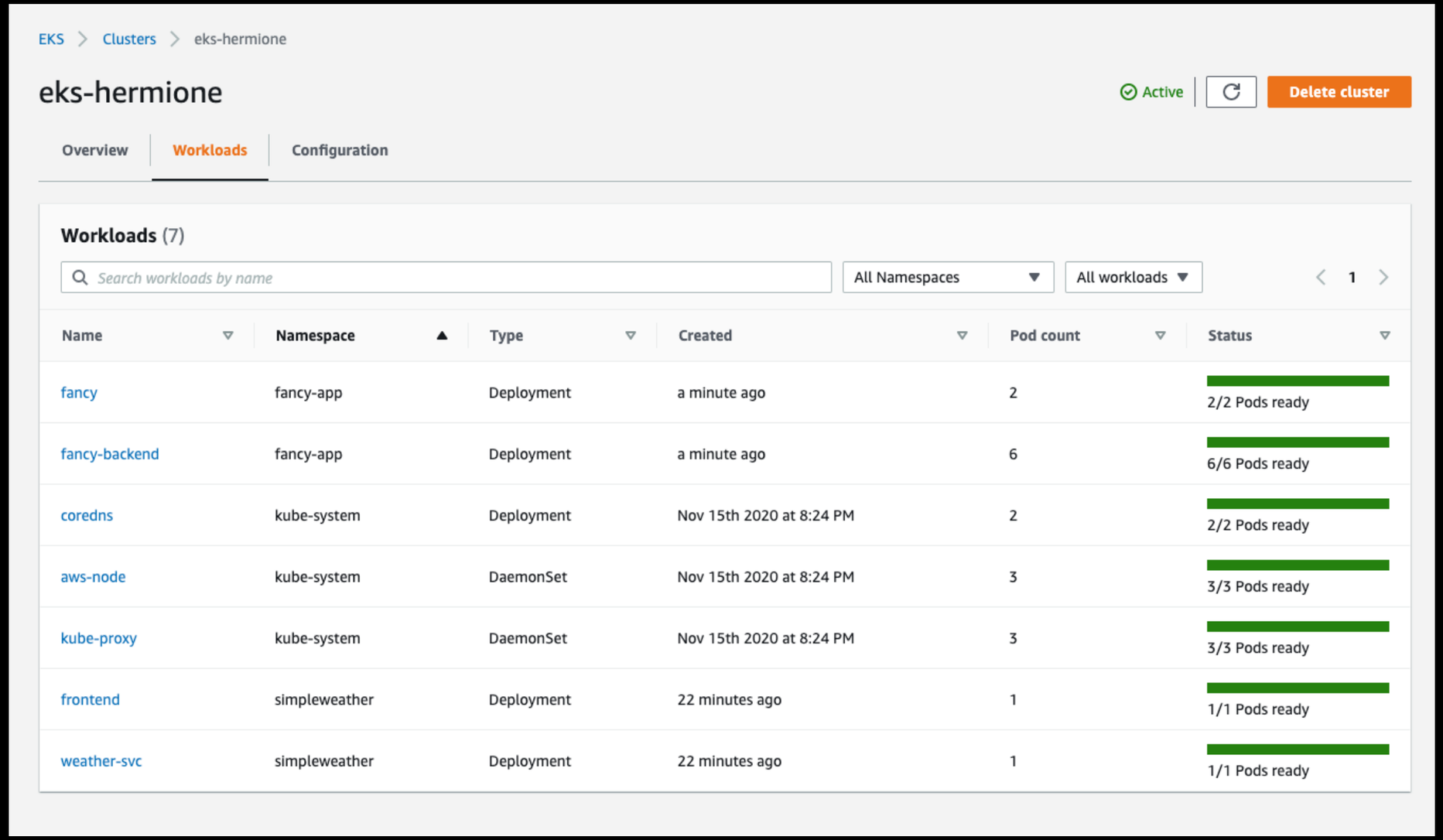
# Next-generation Amazon EKS console



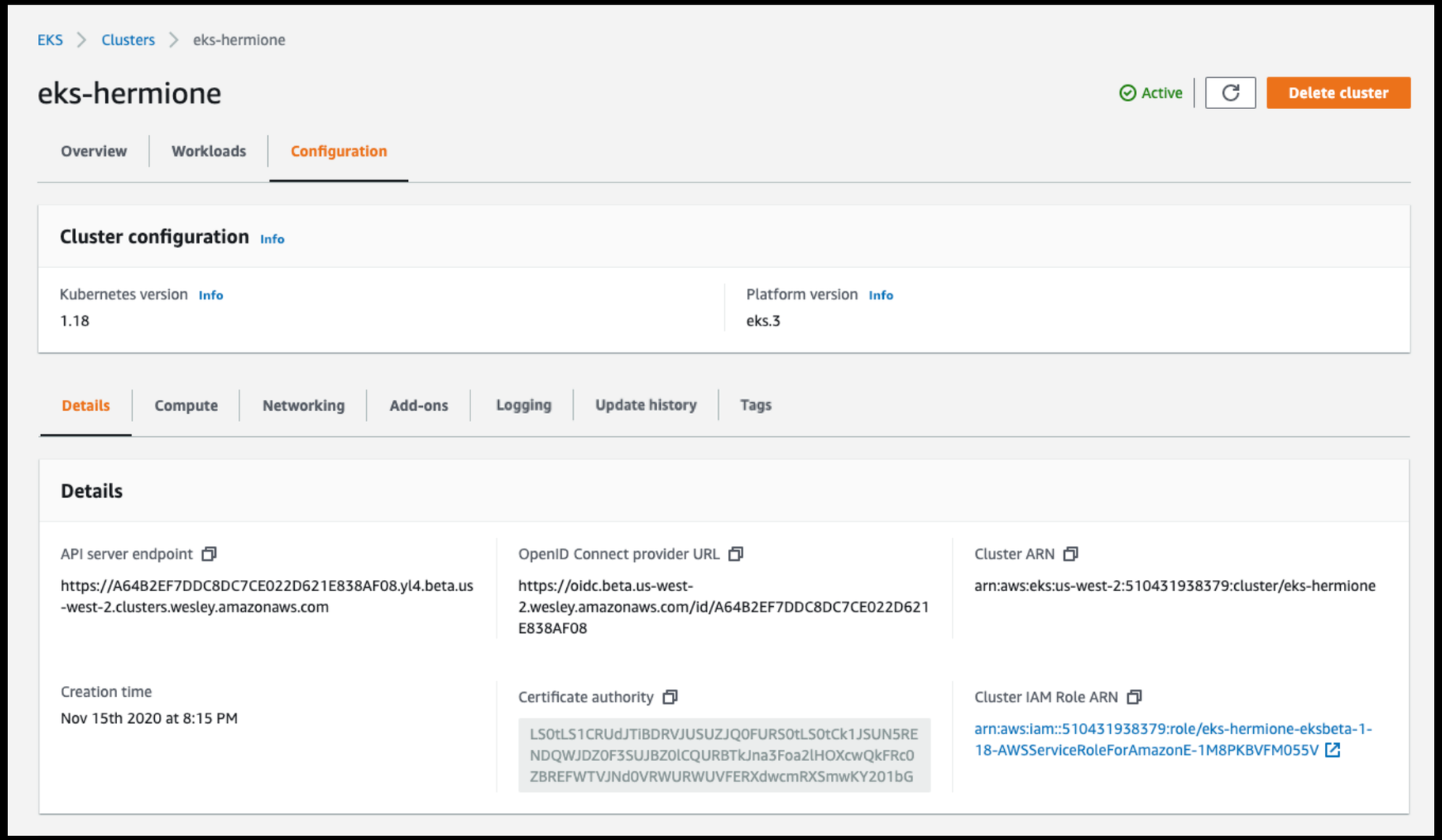
# Overview


EKS > Clusters > eks-hermione
eks-hermione
Active
Delete cluster
Overview
Workloads
Configuration
Nodes (3)
Search nodes by name
1
Node name
Instance type
Node Group
Created
Status
ip-192-168-119-222.us-west-2.compute.internal
t3.medium
eks-hermione-mng1
5 hours ago
Ready
ip-192-168-189-101.us-west-2.compute.internal
t3.medium
eks-hermione-mng1
5 hours ago
Ready
ip-192-168-196-177.us-west-2.compute.internal
t3.medium
eks-hermione-mng1
5 hours ago
Ready

# Workloads

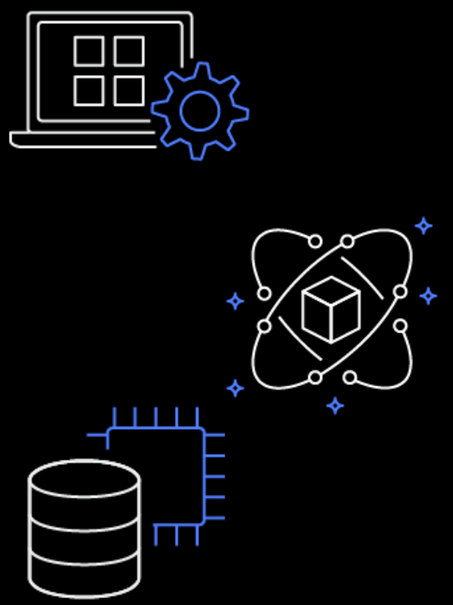

EKS > Clusters > eks-hermione
eks-hermione
Active
Delete cluster
Overview
Workloads
Configuration
Workloads (7)
Search workloads by name
All Namespaces
All workloads
1
Name
Namespace
Type
Created
Pod count
Status
fancy
fancy-app
Deployment
a minute ago
2
2/2 Pods ready
fancy-backend
fancy-app
Deployment
a minute ago
6
6/6 Pods ready
coredns
kube-system
Deployment
Nov 15th 2020 at 8:24 PM
2
2/2 Pods ready
aws-node
kube-system
DaemonSet
Nov 15th 2020 at 8:24 PM
3
3/3 Pods ready
kube-proxy
kube-system
DaemonSet
Nov 15th 2020 at 8:24 PM
3
3/3 Pods ready
frontend
simpleweather
Deployment
22 minutes ago
1
1/1 Pods ready
weather-svc
simpleweather
Deployment
22 minutes ago
1
1/1 Pods ready

# Configuration

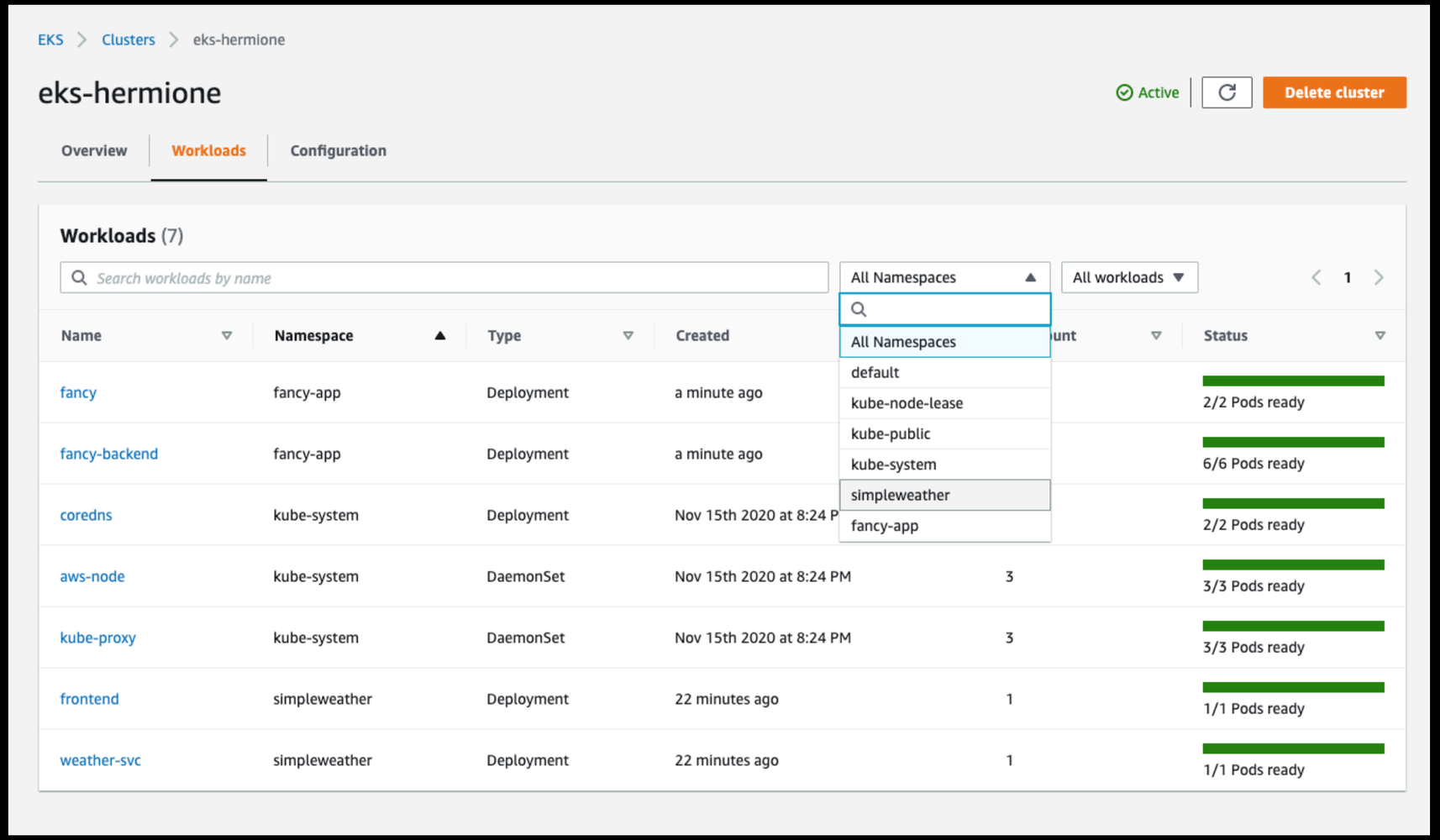


# A deeper look at workloads

- Curated views of relevant workload info, in context
- Kubernetes API is relatively flat by design
- Tooling does some heavy lifting here to build useful information and context
- Intuitive, correlative views in an otherwise flat world



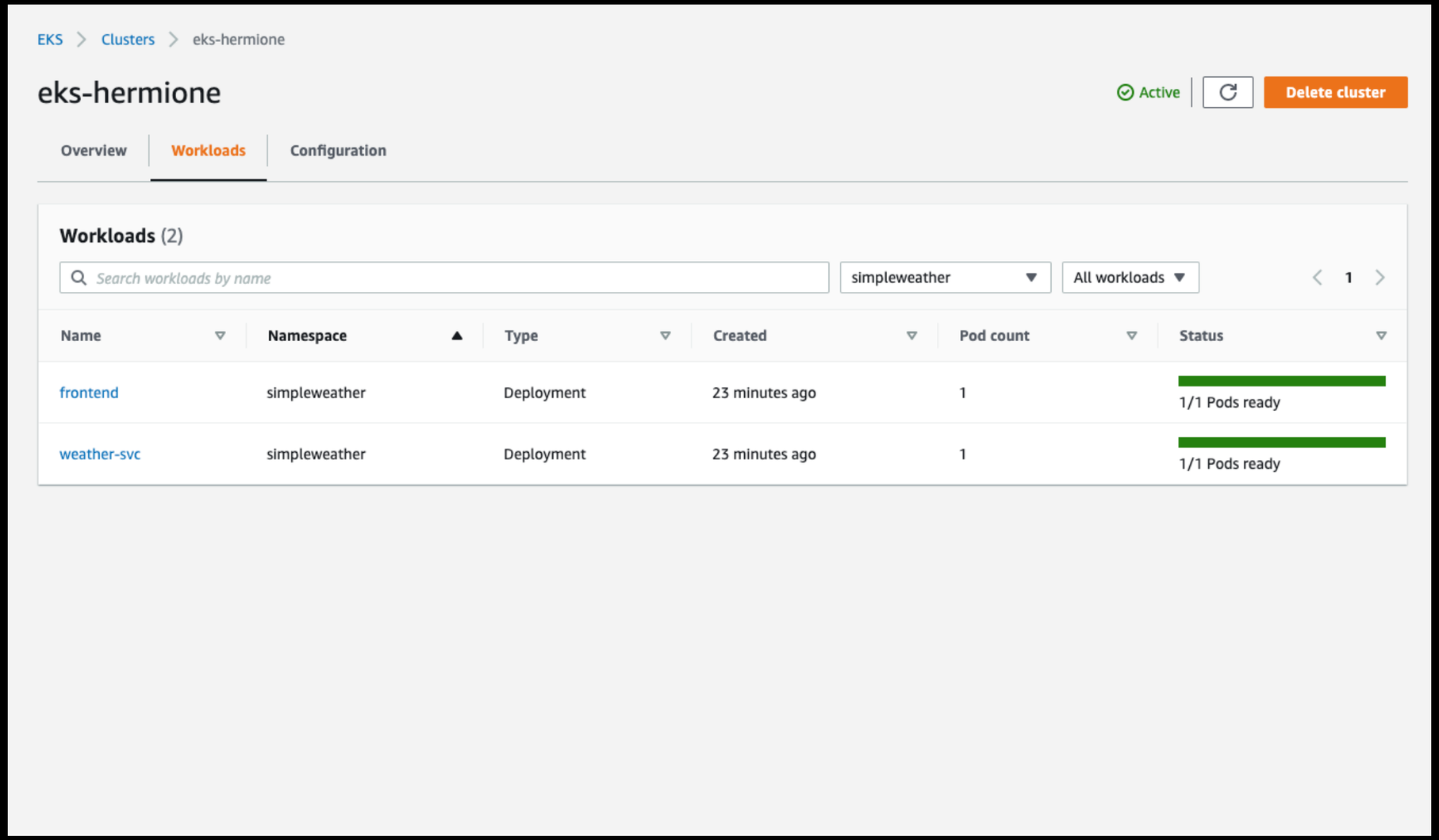
# Workloads by namespace



# Kubernetes and AWS auth integration

- Viewing cluster objects requires a bridge between AWS and Kubernetes
- New IAM Policy to access the Kubernetes API, eks:AccessKubernetesAPI
- Roles require this to access in-cluster object information
- Works in cooperation with in-cluster RBAC authorization
- Users see in the console only what their role has access to
- Seamless and intuitive experience, no surprises

# Workloads by namespace



# Deployments

- Deployment info

EKS > Clusters > eks-hermione > Deployment: simpleweather/weather-svc

## weather-svc

### Info

Status
1 Desired | 1 Available | 1 Ready

Created
27 minutes ago

Namespace
simpleweather

Selector
app=weather-svc

Strategy type
RollingUpdate

### Pods (1)

1

| Name | Status | Created |
|---|---|---|
| weather-svc-66f8778754-45vxw | Running | 27 minutes ago |

### Conditions

| Name | Status | Message |
|---|---|---|
| Available | True | Deployment has minimum availability. |
| Progressing | True | ReplicaSet "weather-svc-66f8778754" has successfully progressed. |

### Labels (1)

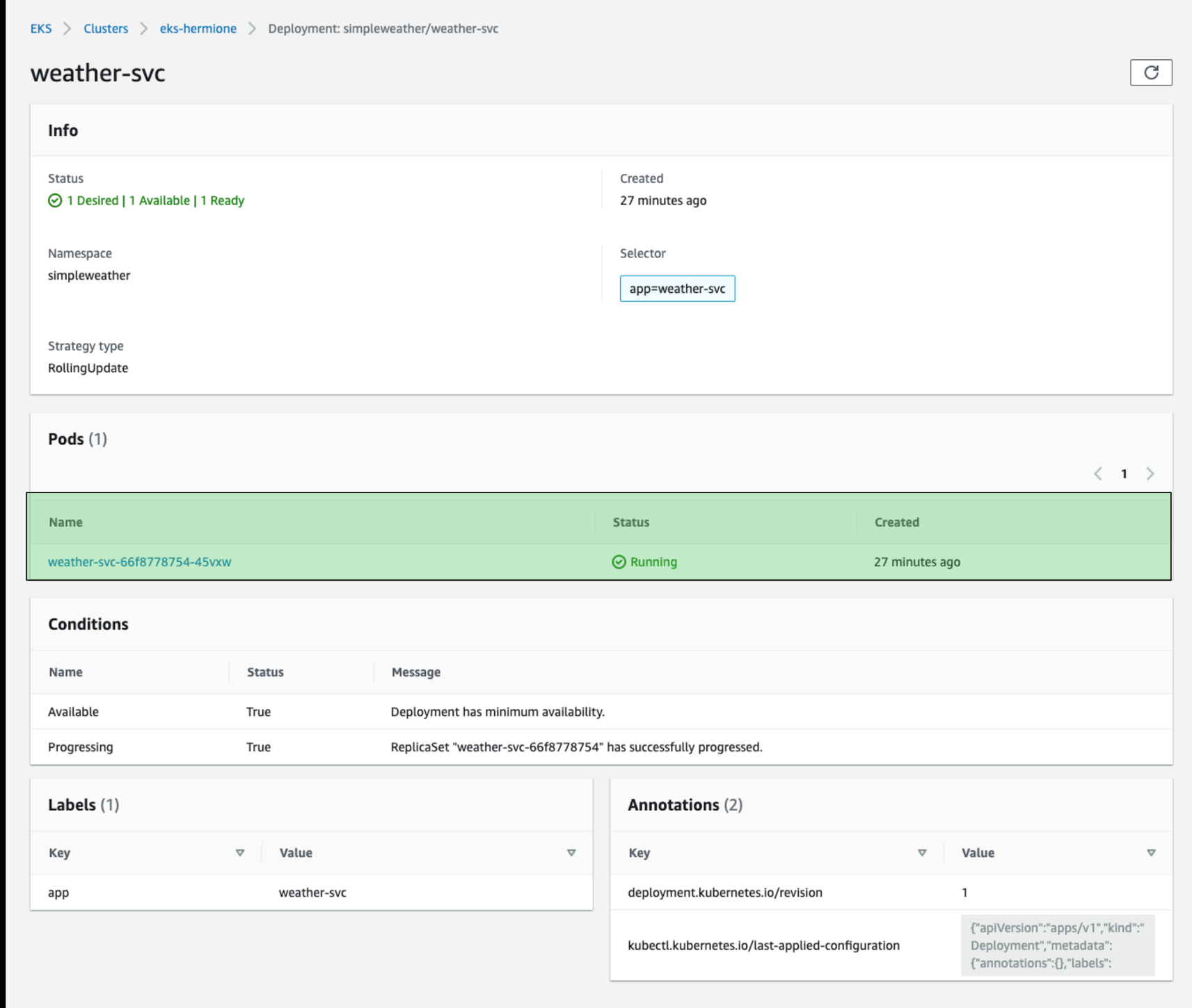
| Key | Value |
|---|---|
| app | weather-svc |

### Annotations (2)

| Key | Value |
|---|---|
| deployment.kubernetes.io/revision | 1 |
| kubectl.kubernetes.io/last-applied-configuration | {"apiVersion":"apps/v1","kind":"Deployment","metadata":{"annotations":{},"labels": |

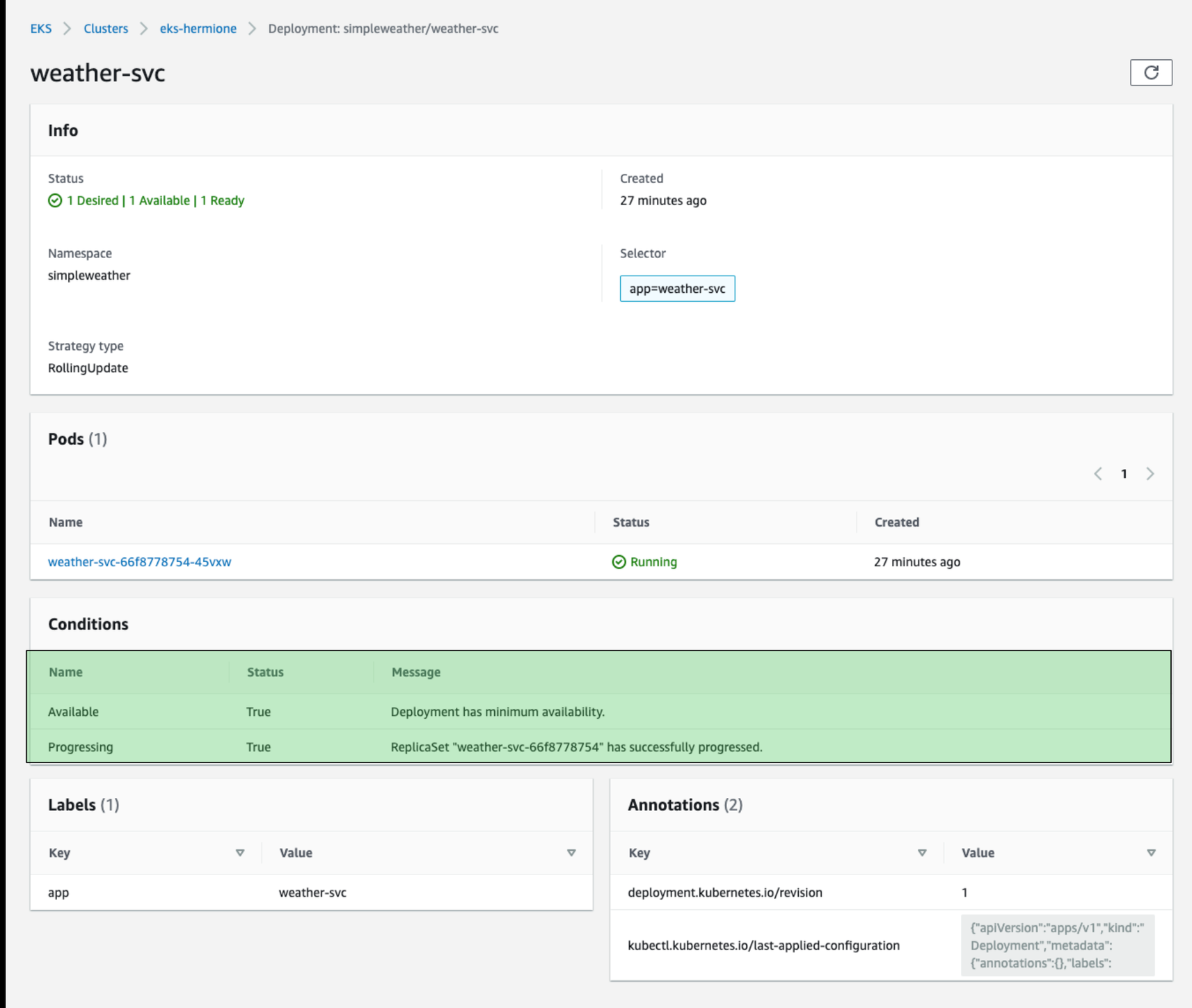
# Deployments

- Deployment info
- Pod status



# Deployments

- Deployment info
- Pod status
- Deployment conditions and state



# Pods

- Pod info, including node and ReplicaSet

EKS > Clusters > eks-hermione > Pod: simpleweather/weather-svc-66f8778754-45vxw

## weather-svc-66f8778754-45vxw

### Info

Status
Running

Created
an hour ago

Namespace
simpleweather

Controlled by
ReplicaSet/weather-svc-66f8778754

Node
ip-192-168-189-101.us-west-2.compute.internal

### Containers (2)

▶ redis

▼ weather-svc

Image
510431938379.dkr.ecr.us-east-2.amazonaws.com/simpleweather/weather-svc:0.1

Status
Running

Created
an hour ago

Ports
7001/TCP

Environment variables
OPENWEATHER_API_KEY={{undefined}}

Mounts
/var/run/secrets/kubernetes.io/serviceaccount
from default-token-tlq8n

Arguments
-

Command
-

### Labels (2)

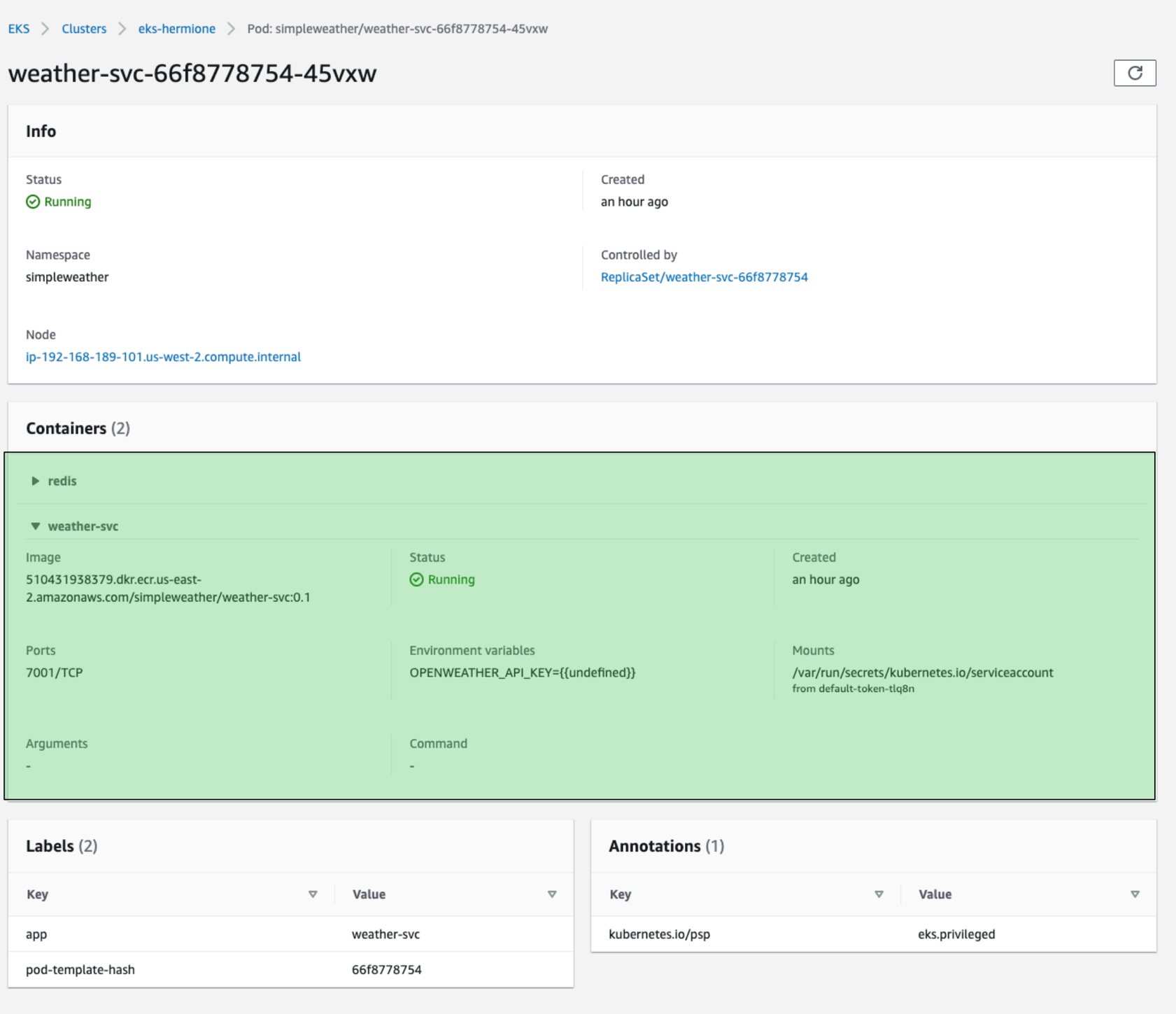
| Key | Value |
|---|---|
| app | weather-svc |
| pod-template-hash | 66f8778754 |

### Annotations (1)

| Key | Value |
|---|---|
| kubernetes.io/psp | eks.privileged |

# Pods

- Pod info, including node and ReplicaSet
- Container list
  - Image
  - Ports
  - Environment variables
  - Volume mounts
- Troubleshoot specific containers



# More Kubernetes objects coming soon

Next steps in bringing you more unified and integrated experiences

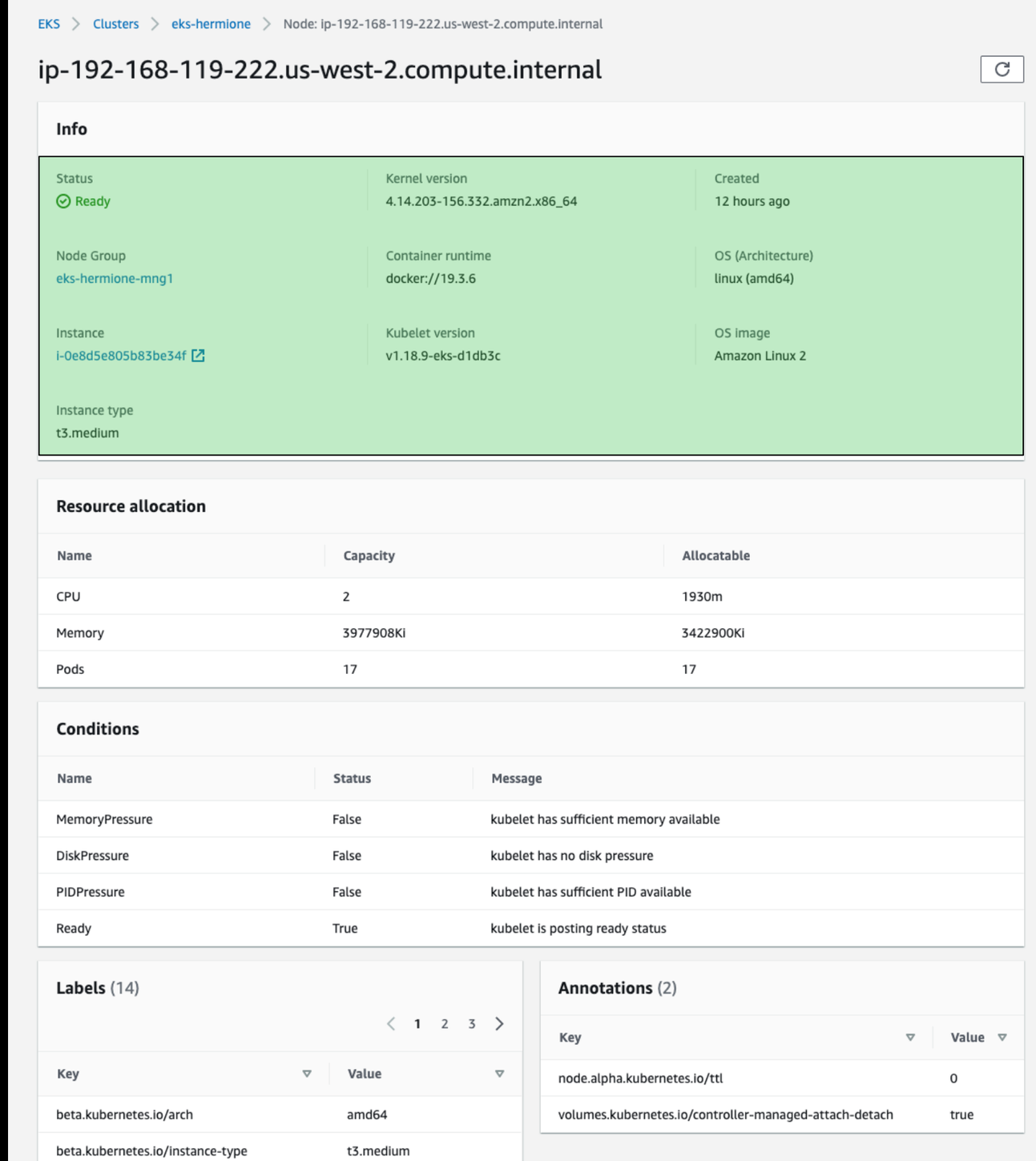
- Service
- Ingress
- Endpoint
- Secret
- Volume
- VolumeAttachment
- PersistentVolumeClaim
- and more

# Node details

- Kubernetes Node object information
- At-a-glance status, runtime, kubelet version, and more
- Resources and conditions, along with labels and annotations
- Links out to relevant resources, such as node group and EC2 instances

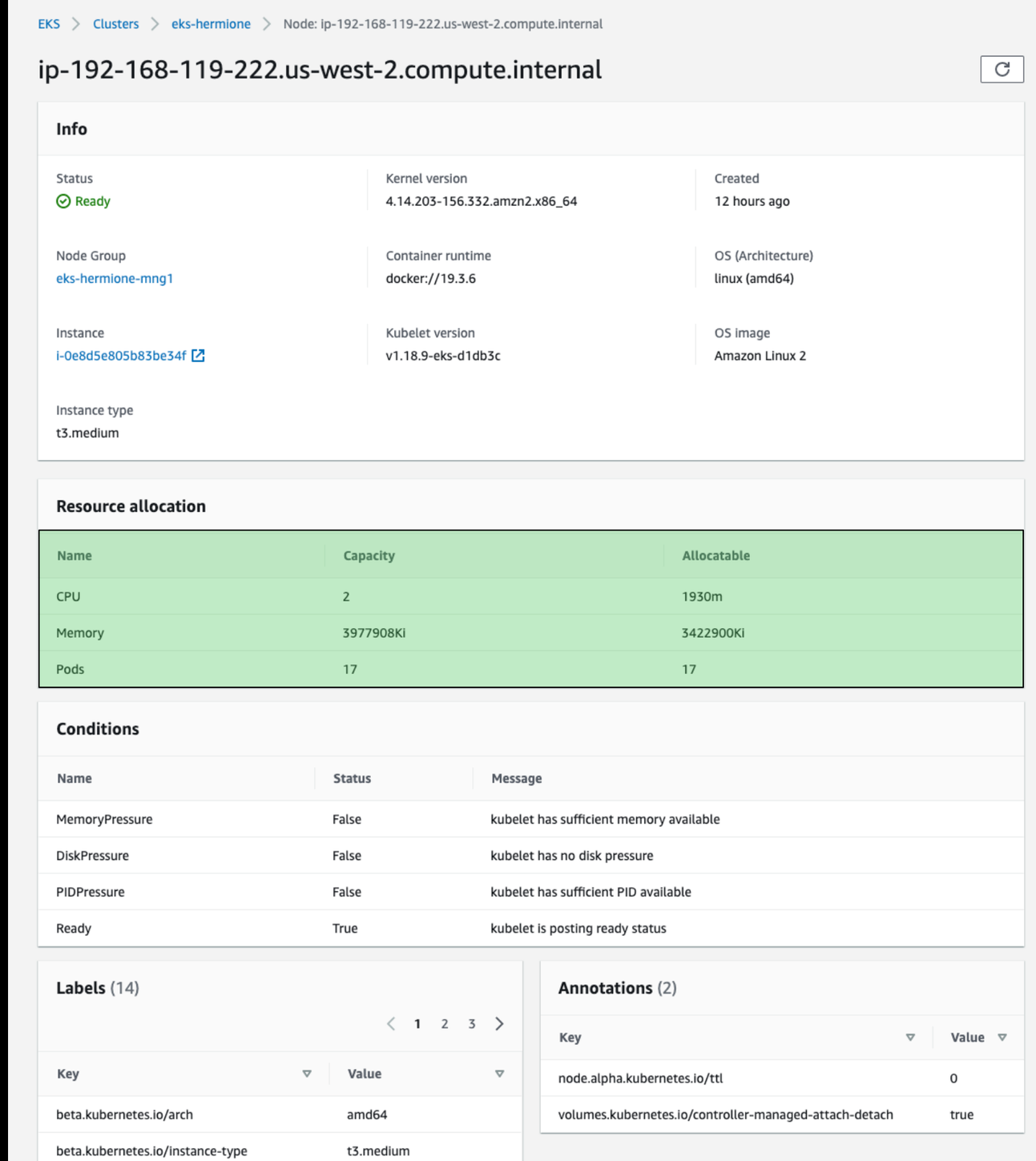
# Node details

- Node status and information at a glance



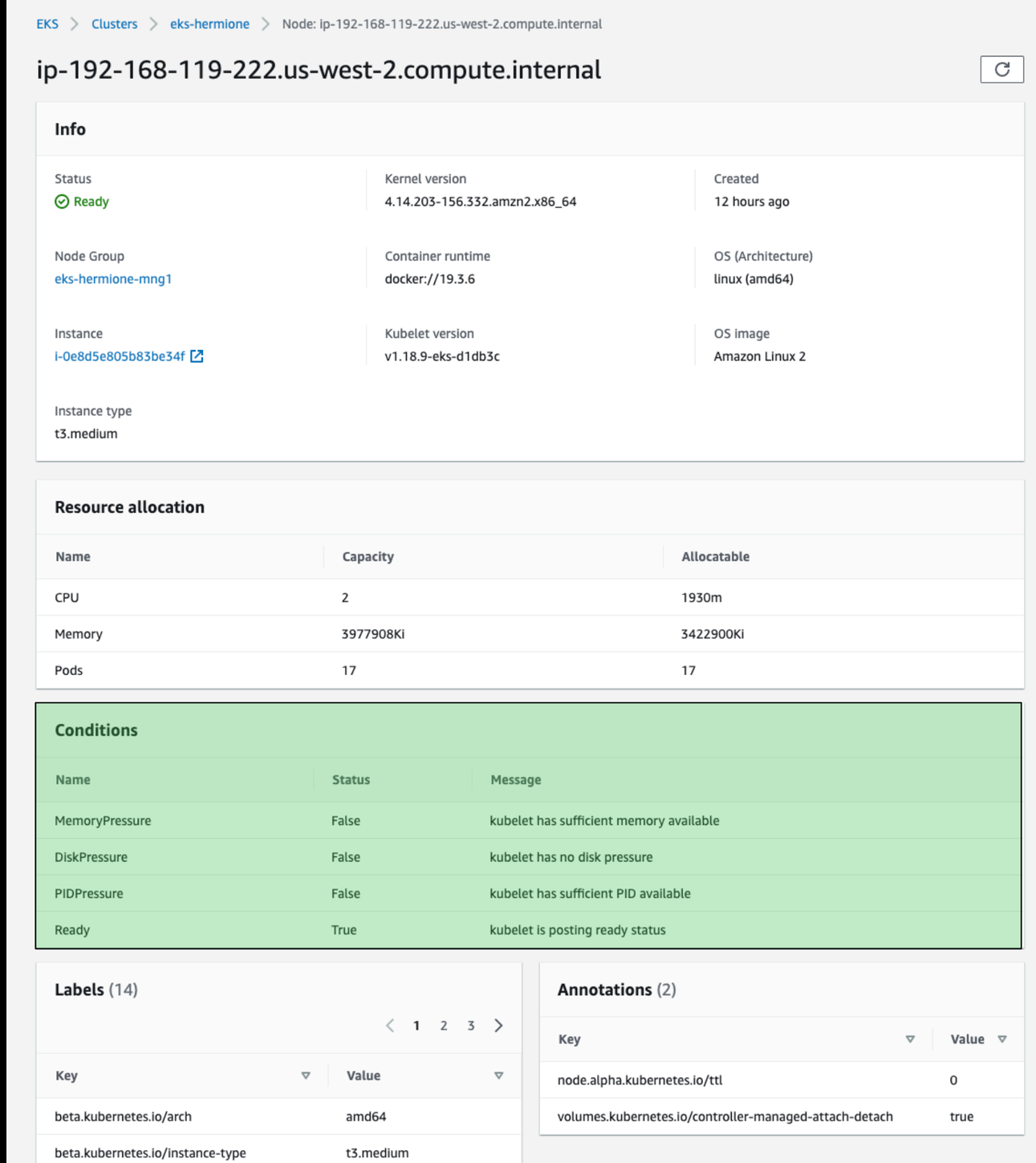
# Node details

- Node status and information at a glance
- Resource capacities

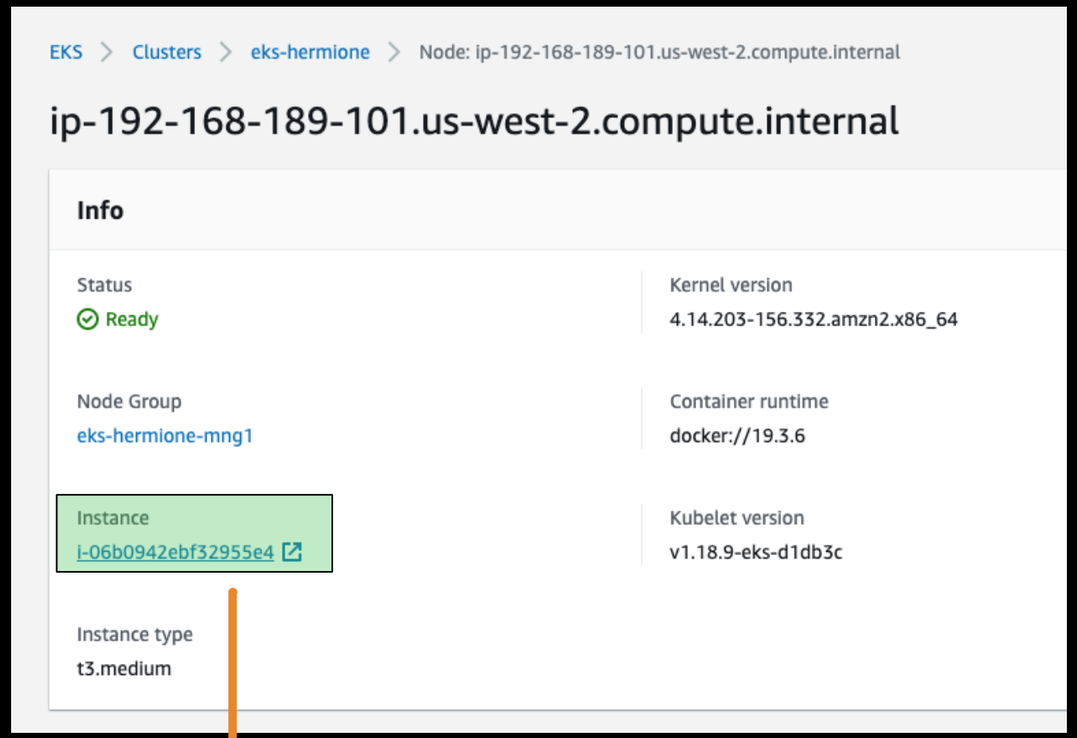


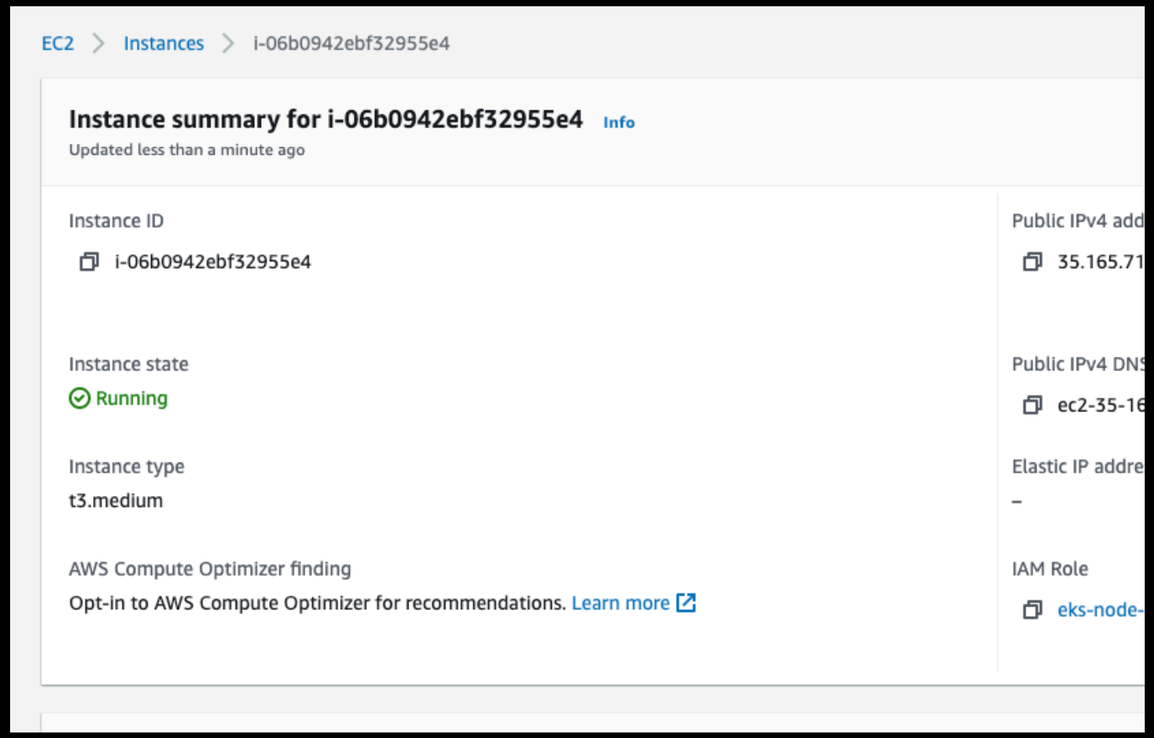
# Node details

- Node status and information at a glance
- Resource capacities
- Node conditions, labels, and annotations

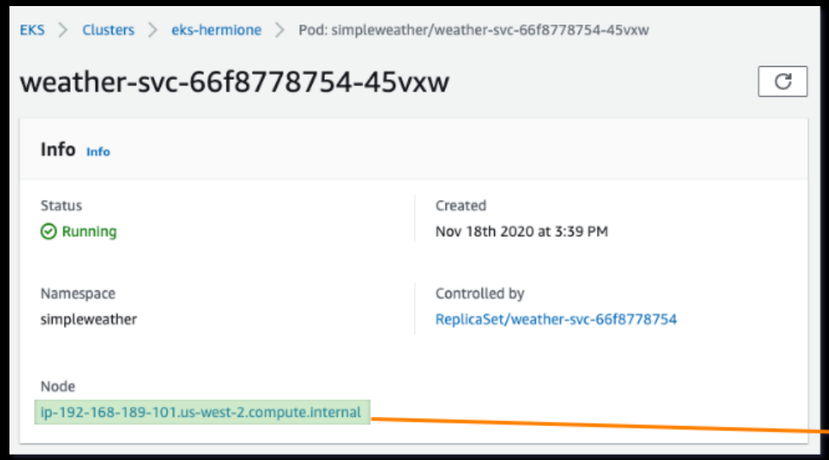


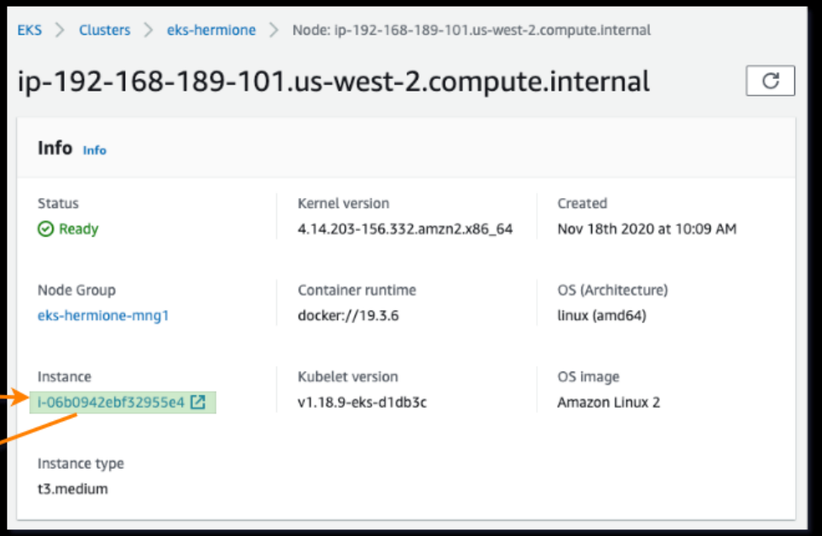
# Cloud resources in context

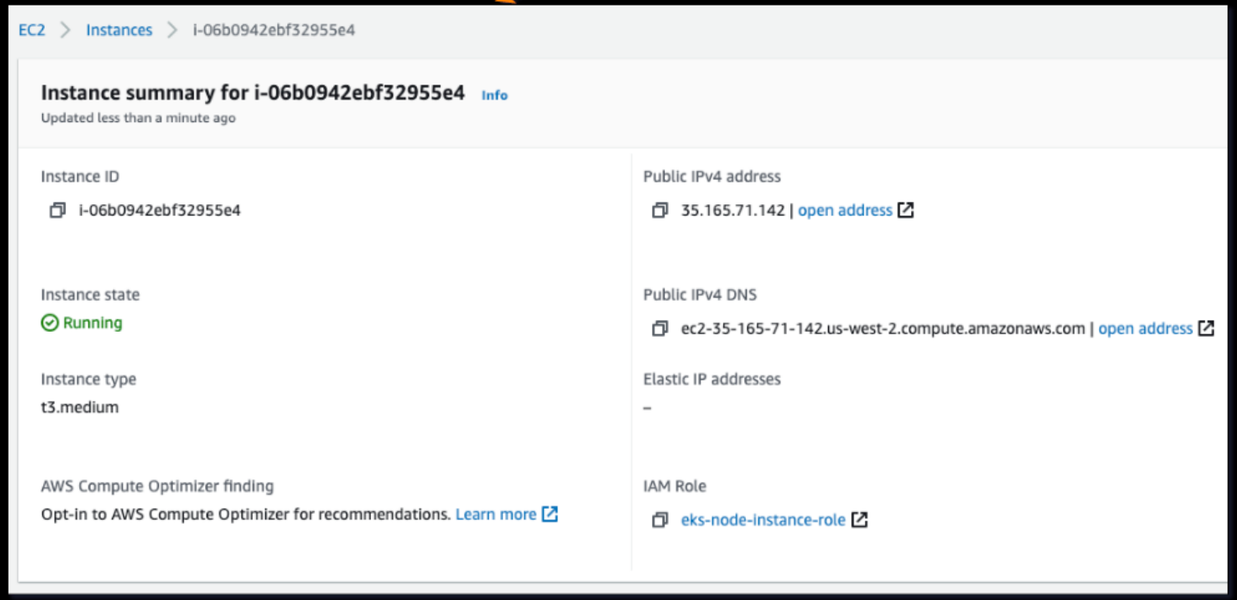


# Multiple paths to resources



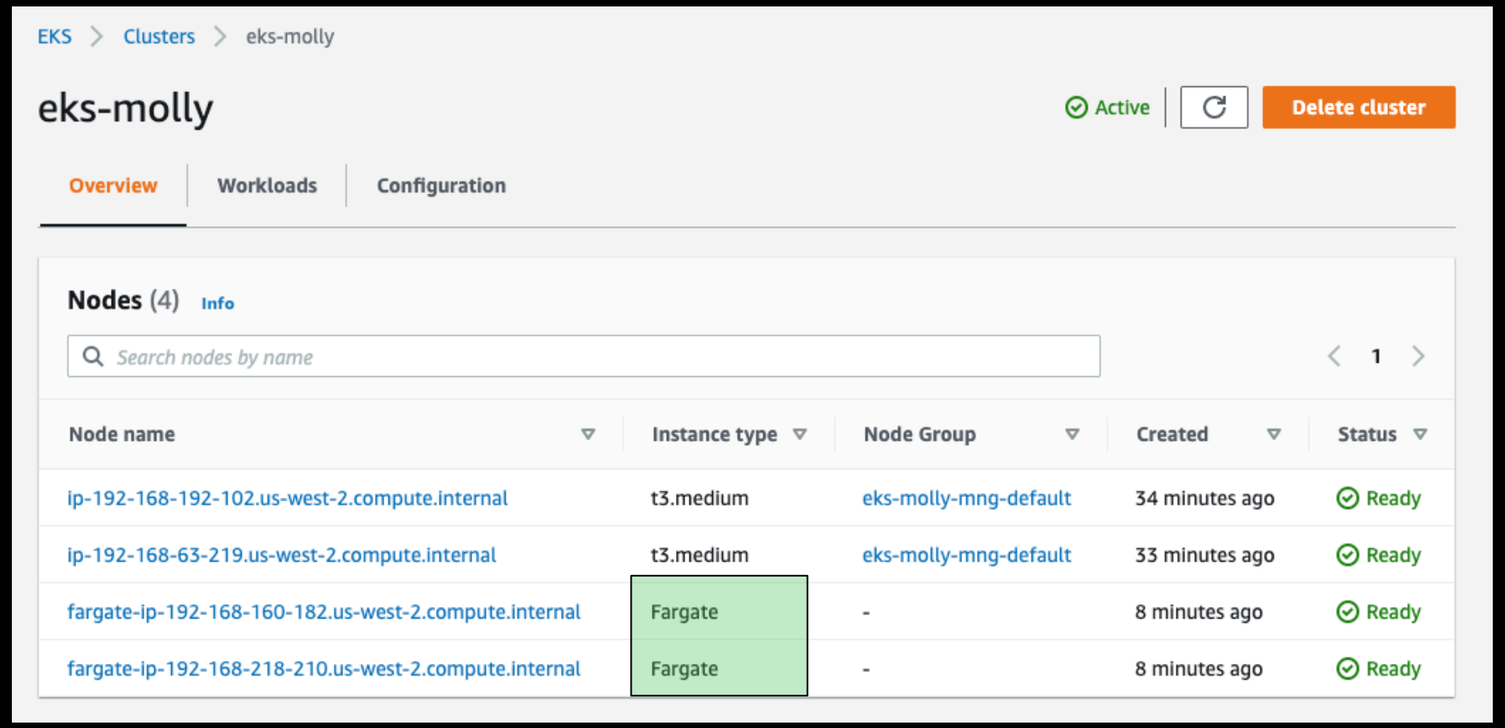




- Intuitive workflows, predictable views
- AWS cloud resources in context

# Fargate nodes

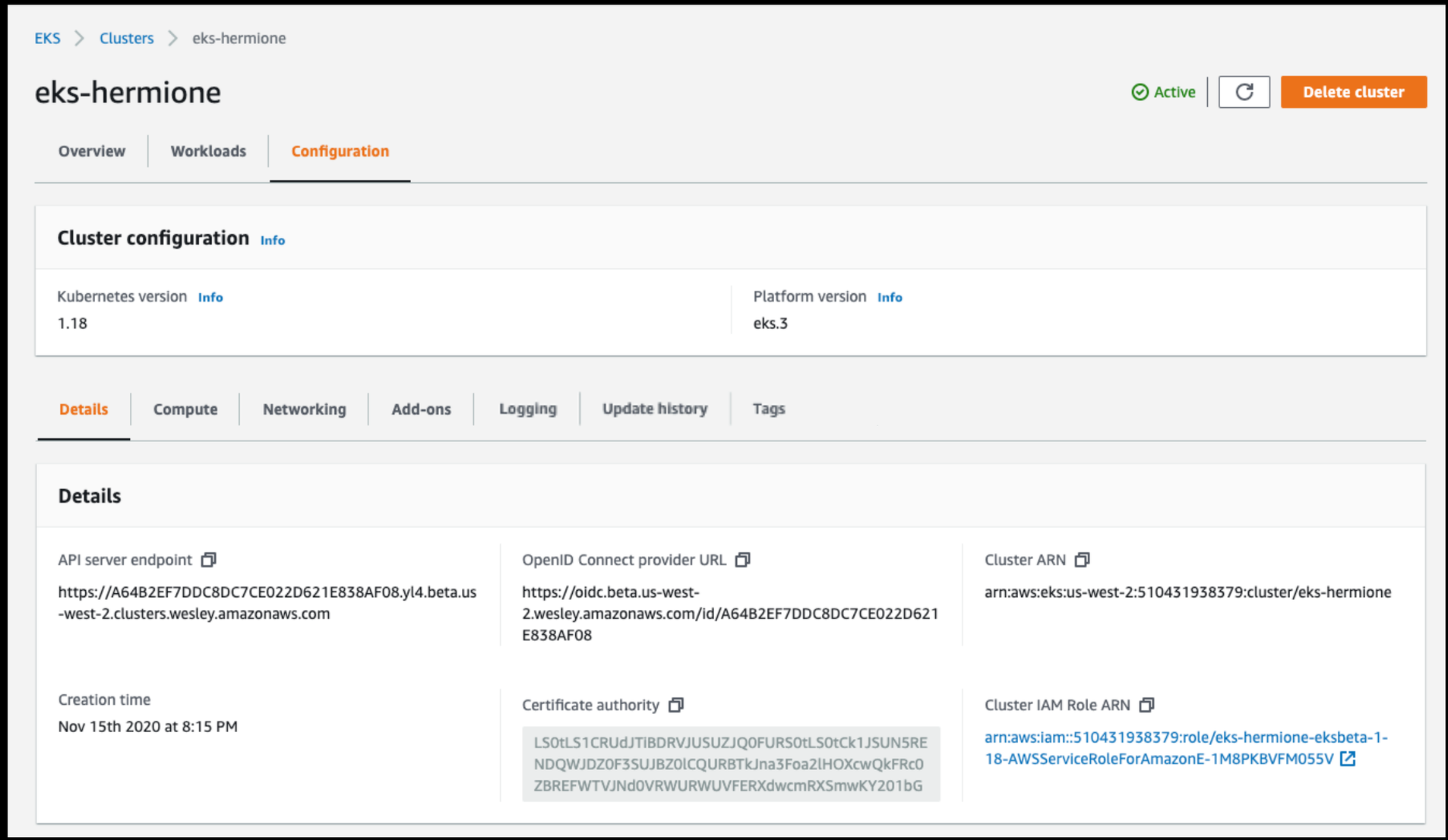
- Note, Fargate instances appear in the nodes list
- This may seem odd, but working as designed



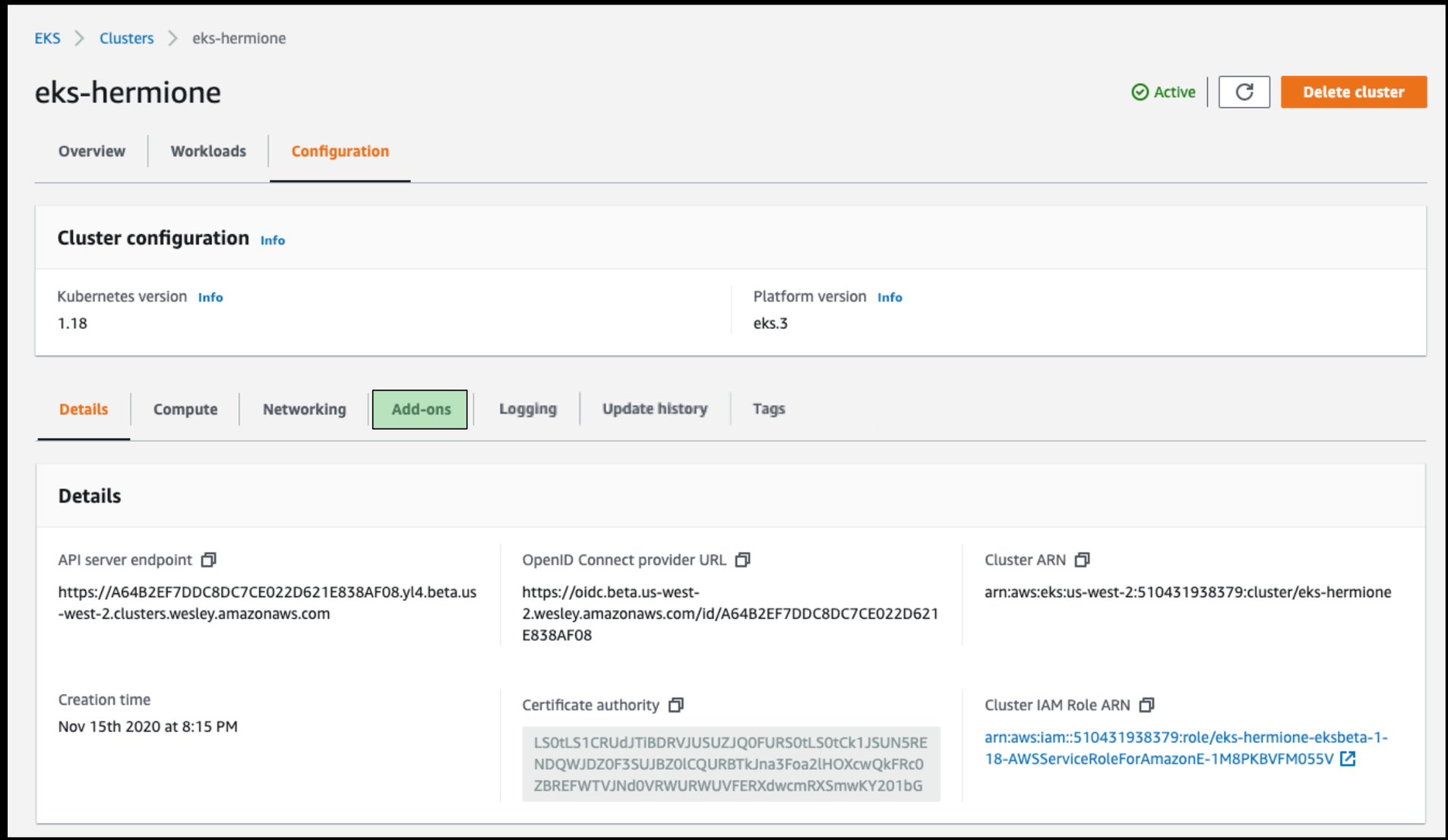
# Cluster configuration

- Familiar cluster details and configuration are now under the new Configuration tab
- Managed node group and Fargate profile configuration
- Logging management, update history, tags

# Cluster configuration details



# Cluster configuration details



# Introducing: EKS Add-ons



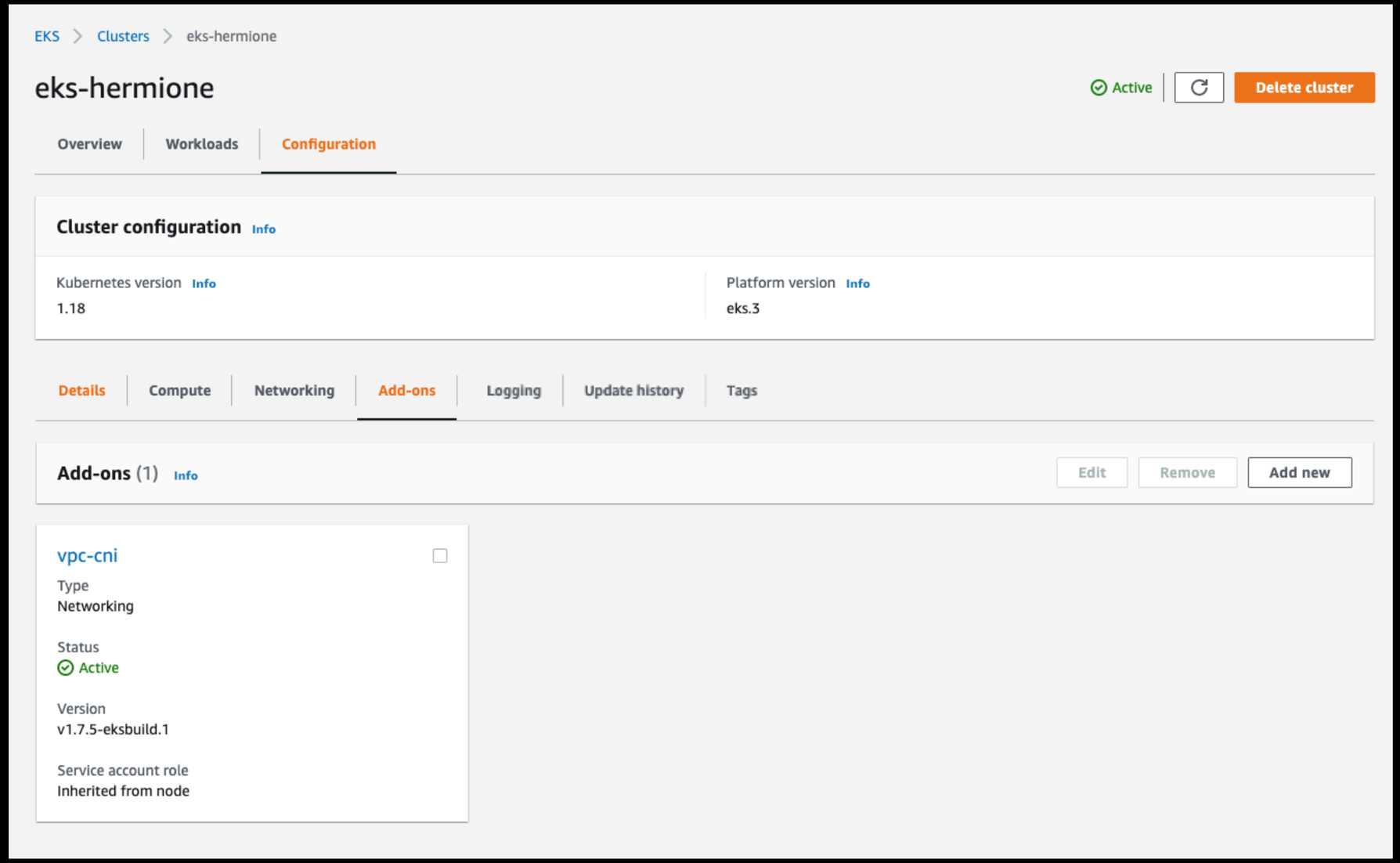
- Curated list of operational cluster software
- Built and tested together to work with Amazon EKS
- Enable during cluster creation, or any time
- Single operation enable, disable, and update
- Uses worker node IAM role by default
- IAM role for service accounts for finer granularity
- Requires minimum Kubernetes v1.18 and eks.3

# Extending Amazon EKS lifecycle management



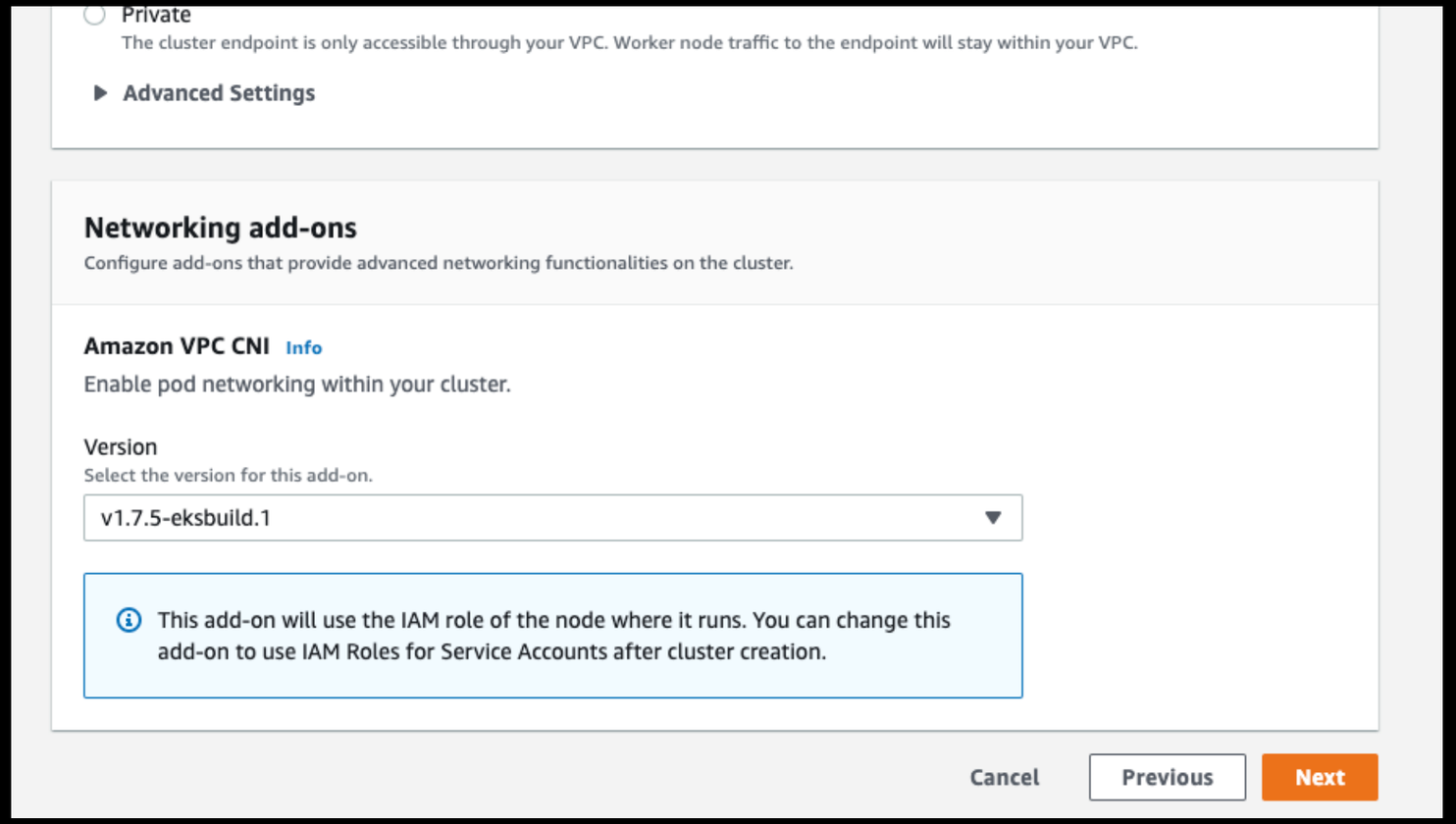
- Managed control plane, cluster lifecycle
- Managed data plane, cluster node management
- Lifecycle management for your operational software

# EKS Add-ons in console

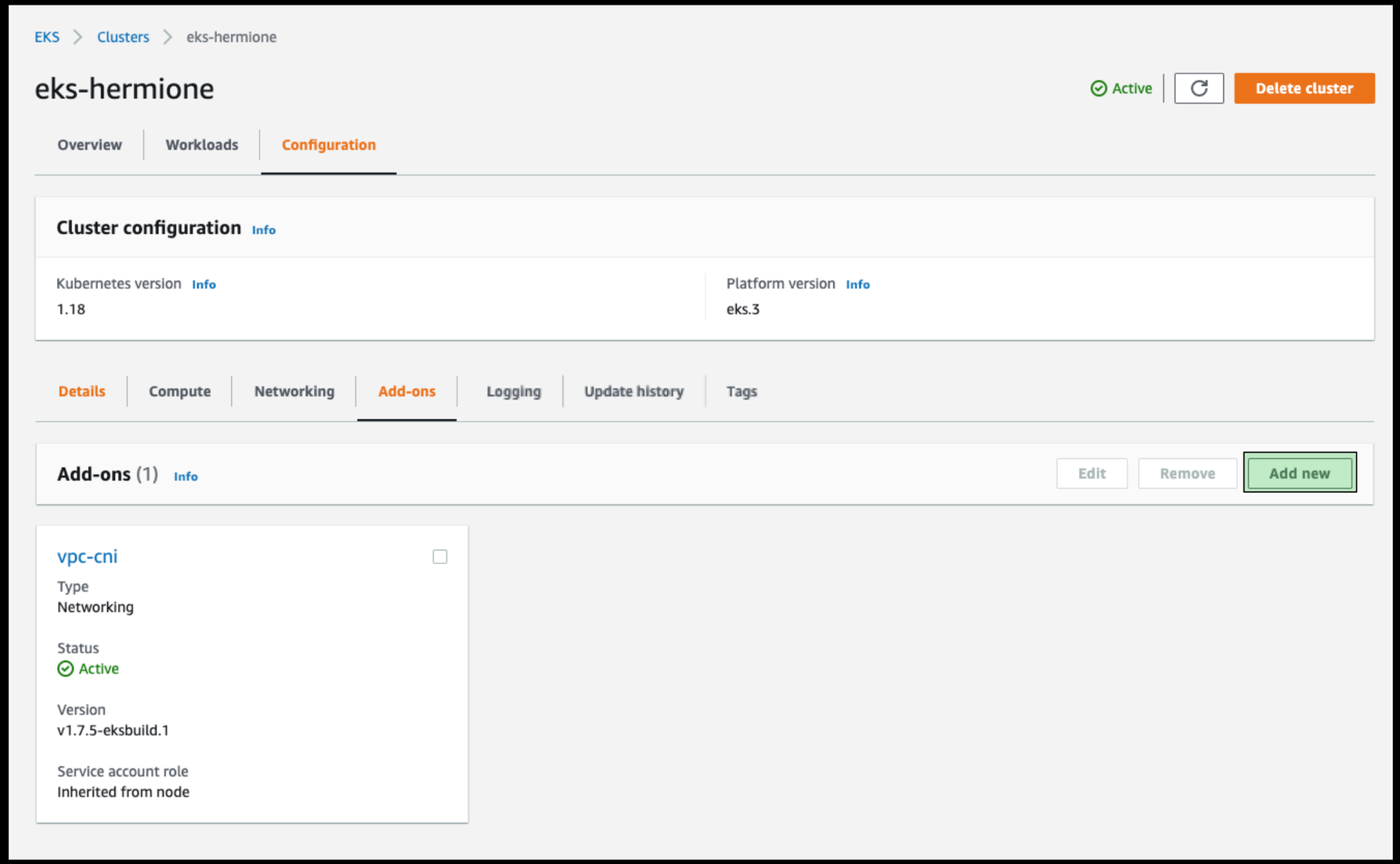


# EKS Add-ons creation during cluster creation

- Version selection during cluster creation
- Inherits IAM role of the node, can be updated later

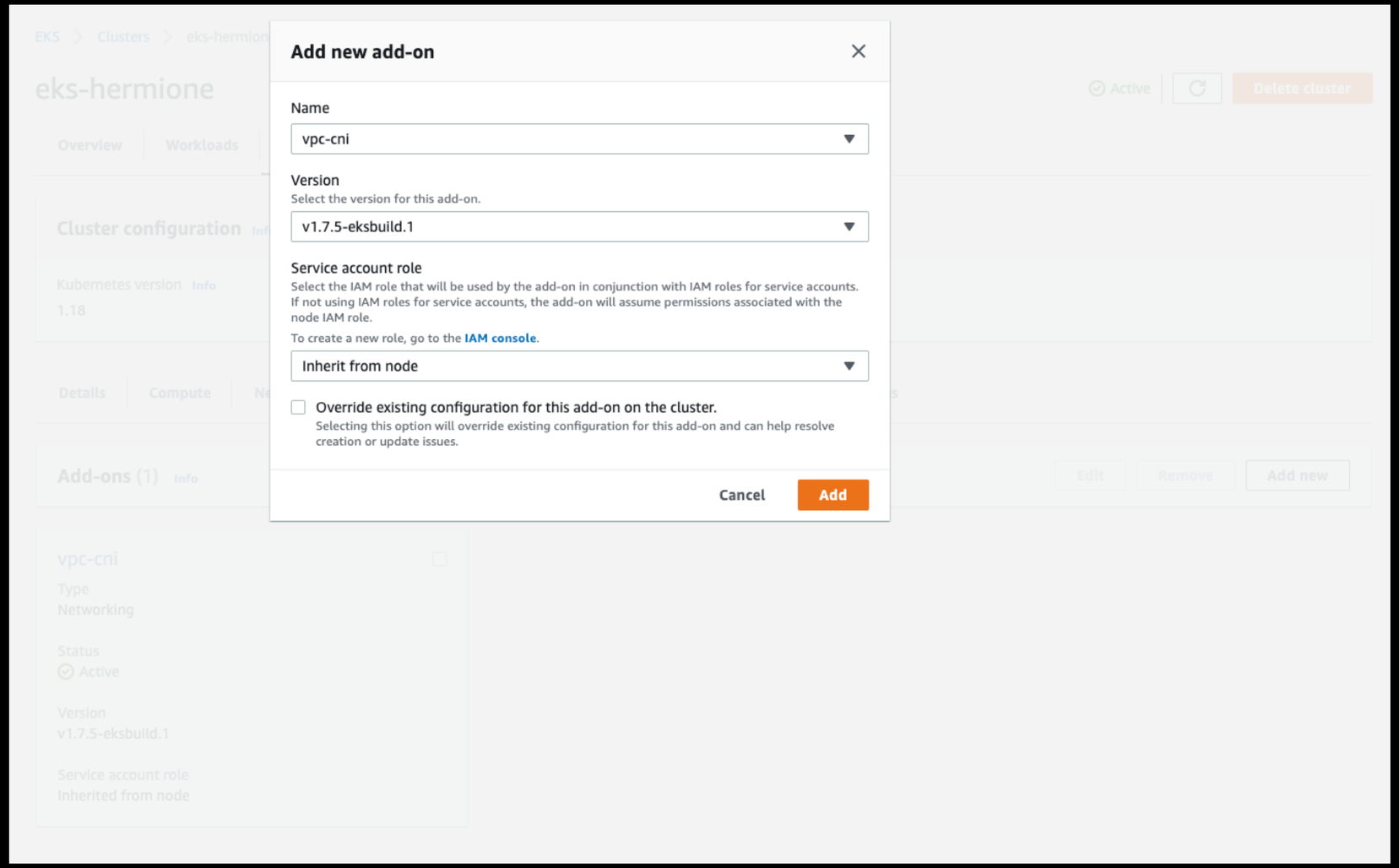


# EKS Add-ons creation, any time

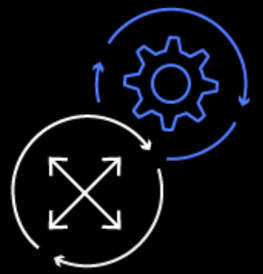


# EKS Add-ons creation, any time

- Selection for add-on type and version
- Choose to inherit node role or specify a custom role

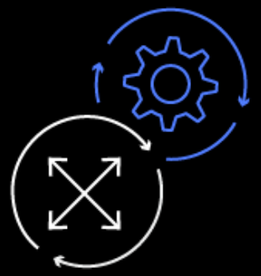


# Single-operation updates



- Single-operation updates and role changes
- Rolling updates, no outages
- Configuration customization is preserved
- Optionally resolve conflicts during upgrade

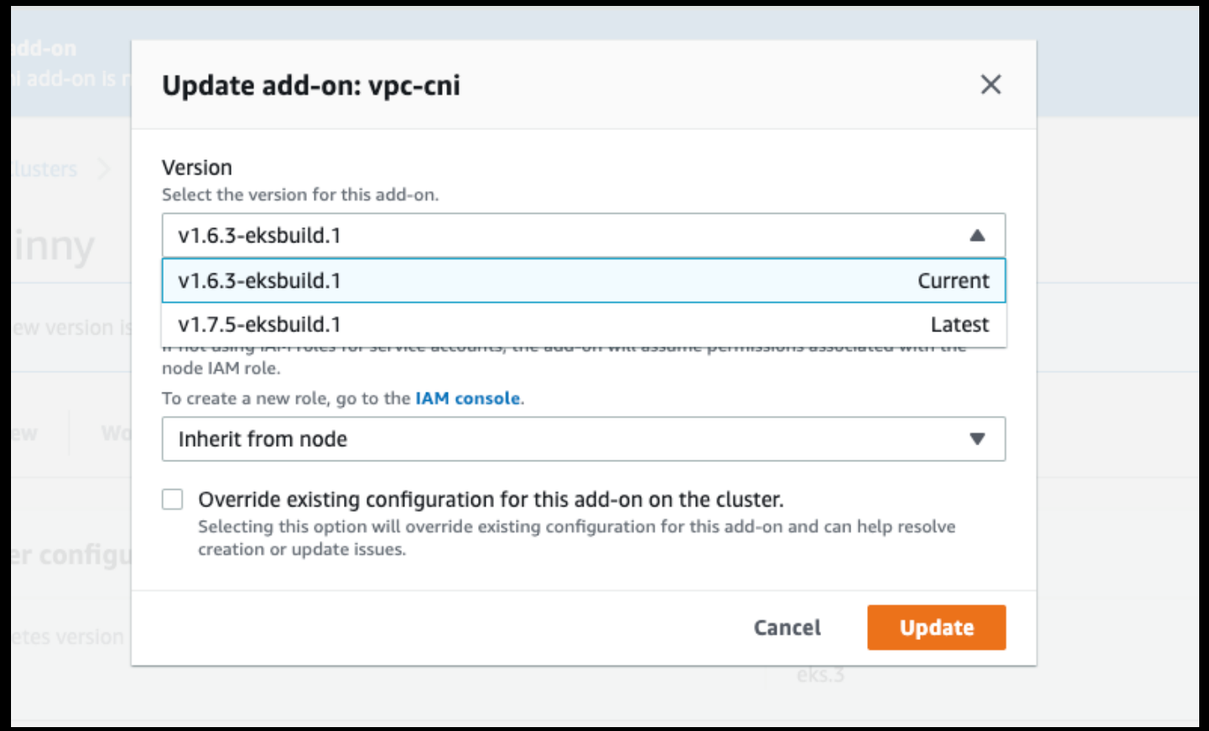
# Lifecycle management and custom configuration



- EKS Add-ons uses the new Server-side apply feature in Kubernetes (v1.18)
- New merging algorithm as well as tracking of field ownership and changes over time
- Results in server-side reporting of configuration conflicts and improved conflict resolution
- Add-on configuration customization is preserved across lifecycle management operations

# EKS Add-ons updates

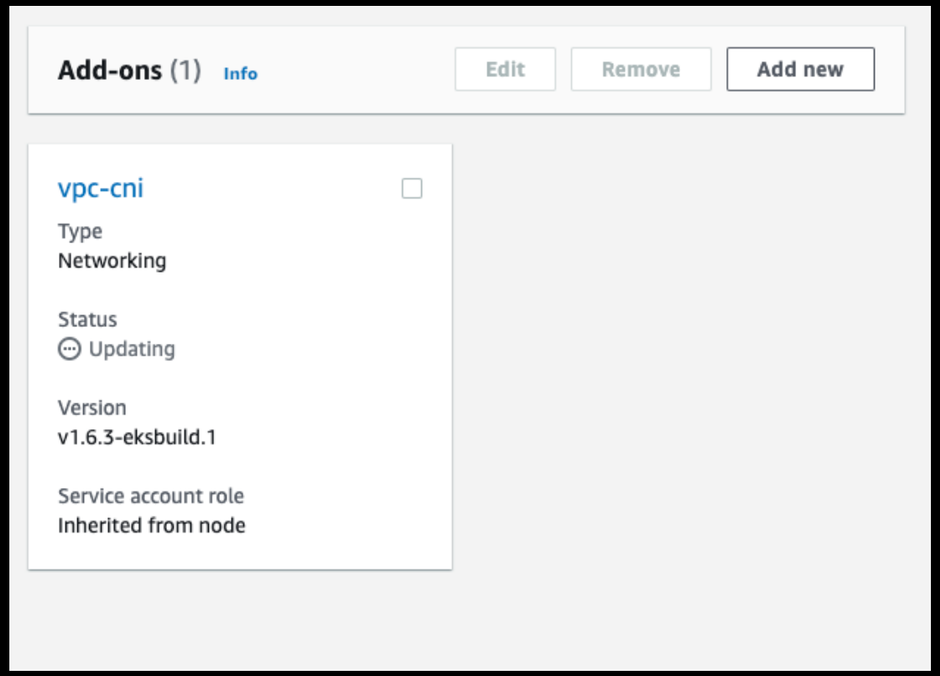
- Select desired version from the drop-down
- Choose to inherit node role or specify a custom one
- Optionally override existing configuration if there are conflicts

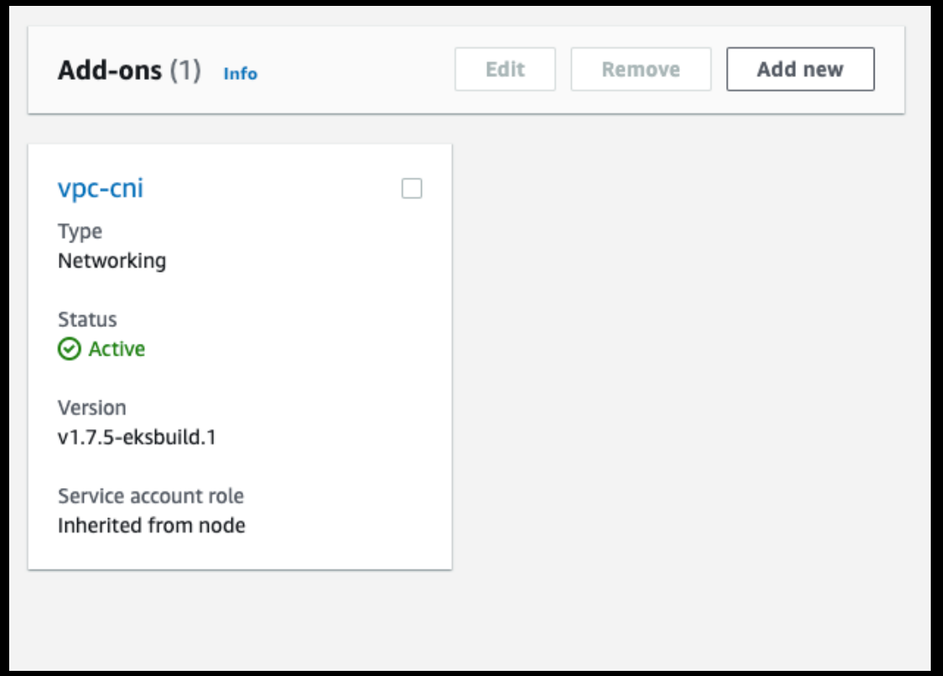


# EKS Add-ons updates

- Kubernetes objects are gracefully cycled
- Add-on is active at its new version once complete

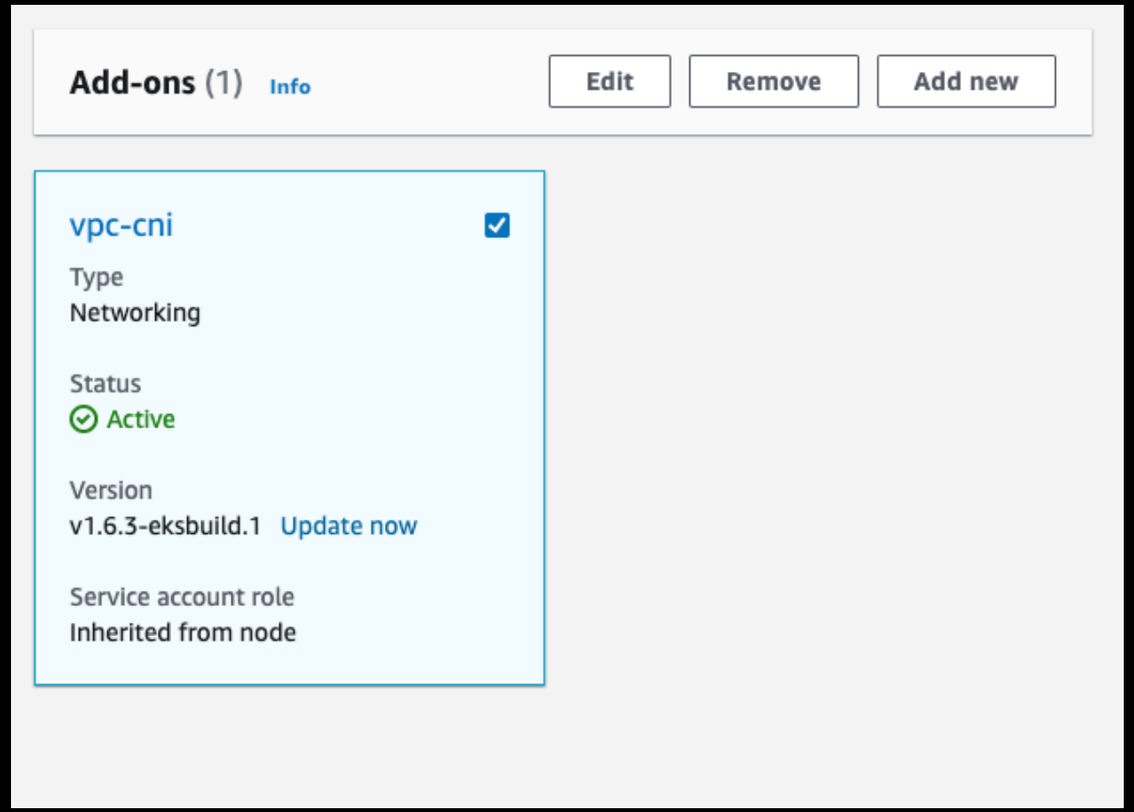
```
Events:
  Type    Reason            Age    From                  Message
  ----    ------            ----   ----                  -------
  Normal  SuccessfulDelete  17m    daemonset-controller  Deleted pod: aws-node-vxqrb
  Normal  SuccessfulCreate  17m    daemonset-controller  Created pod: aws-node-27w8f
  Normal  SuccessfulDelete  16m    daemonset-controller  Deleted pod: aws-node-ndx7f
  Normal  SuccessfulCreate  16m    daemonset-controller  Created pod: aws-node-gcj5m
  Normal  SuccessfulDelete  15m    daemonset-controller  Deleted pod: aws-node-vdvbr
```



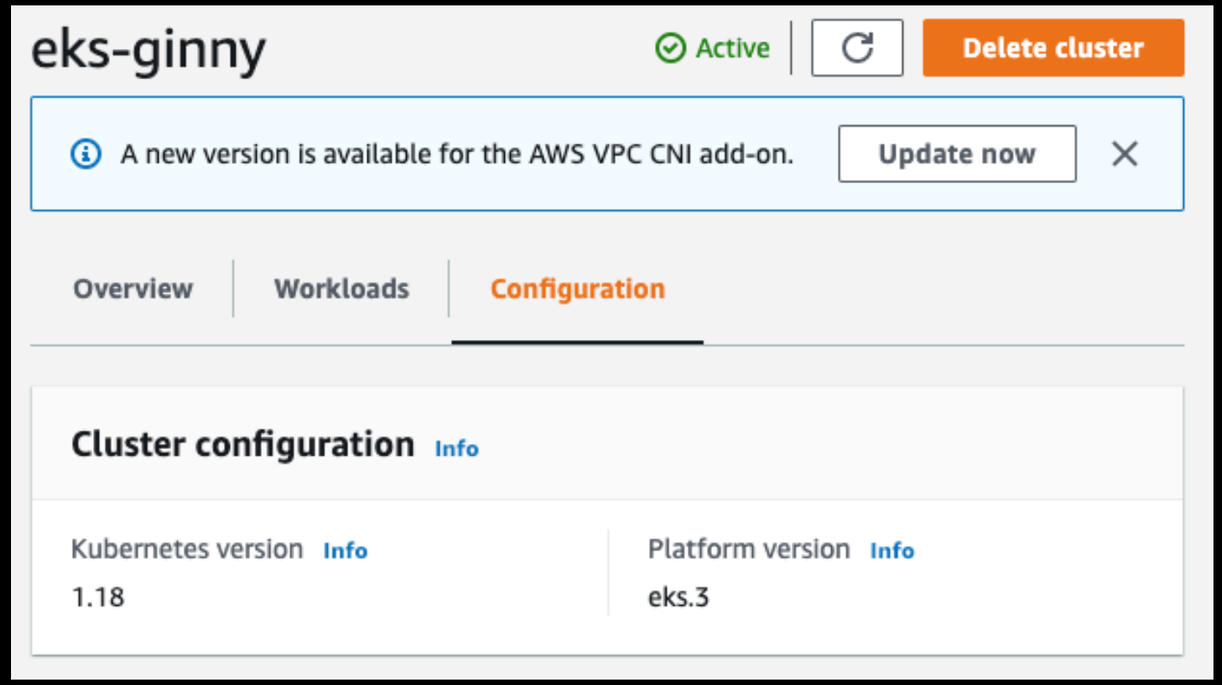
# EKS Add-ons updates

- Edit add-on at any time to its modify version
- Down-rev add-ons have an ‘Update now’ action



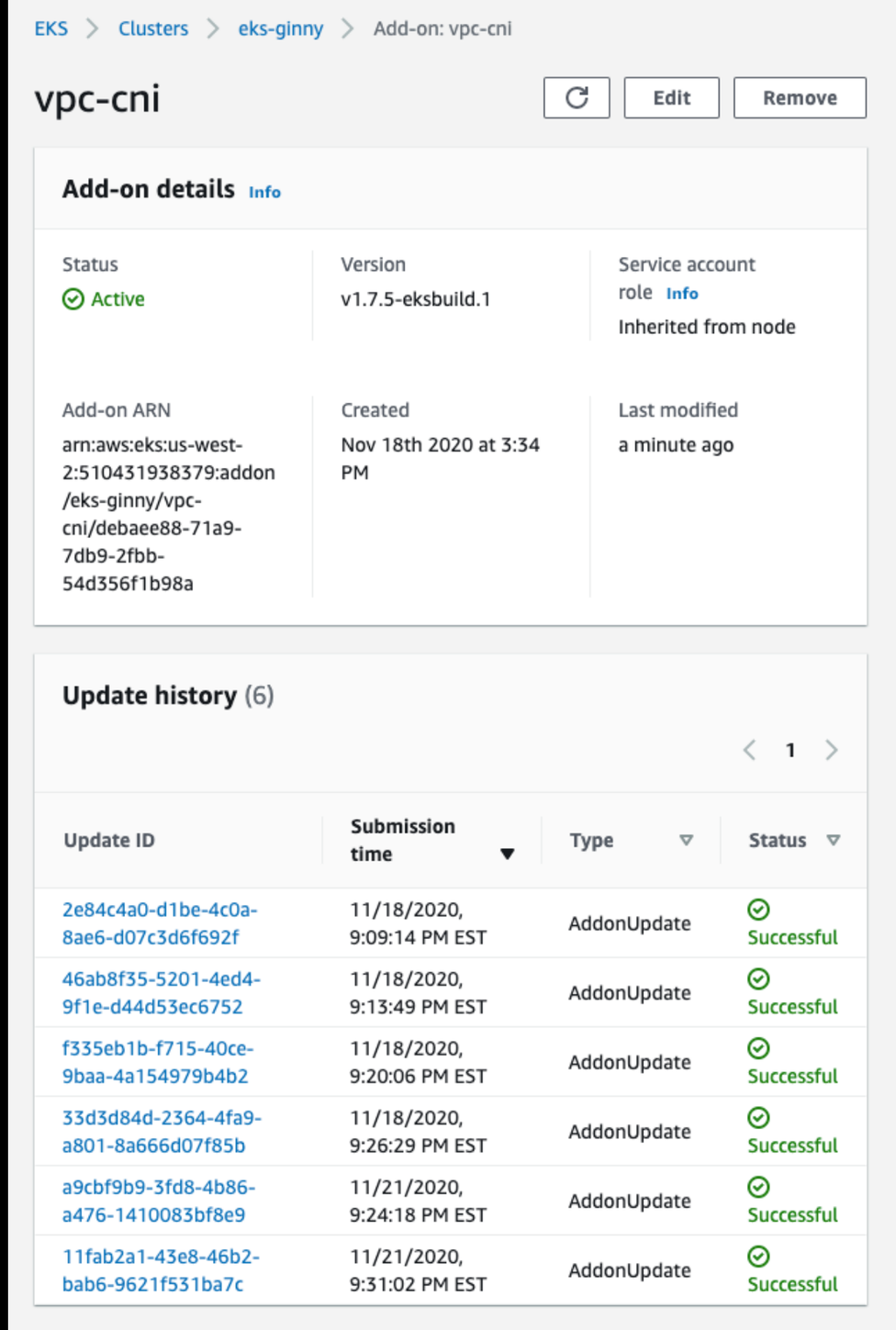
# EKS Add-ons updates

- Additionally, down-rev add-ons are seen in a banner
- Updates can be invoked from here as well



# EKS Add-ons updates

- As with cluster and node group updates, full update history is available
- Each update entry has detailed information, such as version, state, and conflicts resolved



# EKS Add-ons with the AWS CLI

Of course, if you prefer `aws` there are new commands

```
$ aws eks list-addons --cluster-name eks-luna
$ aws eks create-addon --cluster-name eks-luna --addon-name vpc-cni --addon-version 1.6.3
$ aws eks describe-addon --cluster-name eks-luna --addon-name vpc-cni
$ aws eks update-addon --cluster-name eks-luna --addon-name vpc-cni --addon-version 1.7.5
```

Optionally provide service account ARN, default inherits

Optionally override configuration and resolve conflicts

# EKS Add-ons with eksctl

Also, support in `eksctl`

```
$ eksctl get addons --cluster eks-molly
$ eksctl create addon --name vpc-cni --version 1.6.5 --cluster eks-molly
$ eksctl update addon --name vpc-cni -version 1.7.4 --cluster eks-molly
```

Optionally provide service account ARN, default inherits

Optionally override configuration and resolve conflicts

# EKS Add-ons with eksctl

```
$ eksctl create cluster -f config.yaml

apiVersion: eksctl.io/v1alpha5
kind: ClusterConfig

metadata:
  name: basic-cluster
  region: eu-north-1

iam:
  withOIDC: true

addons:
  - name: vpc-cni
    version: 1.7.5
    serviceAccountRoleARN: role-arn
```

If `withOIDC` is true without a policy specified, a role will be created for the add-on with the required policy

# EKS Add-ons Roadmap

- VPC CNI available now



**Coming soon**

- CoreDNS
- Kube-proxy
- AWS Load Balancer Controller
- AWS App Mesh Controller
- CloudWatch agent
- Metrics server
- Flux GitOps controller
- and more

# Connect with us

As always, our work is guided by you.

Visit the AWS Container Services Roadmap on GitHub

 https://github.com/aws/containers-roadmap

# Thank you!

Jesse Butler

@jlb13

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.



# Please complete the session survey

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.