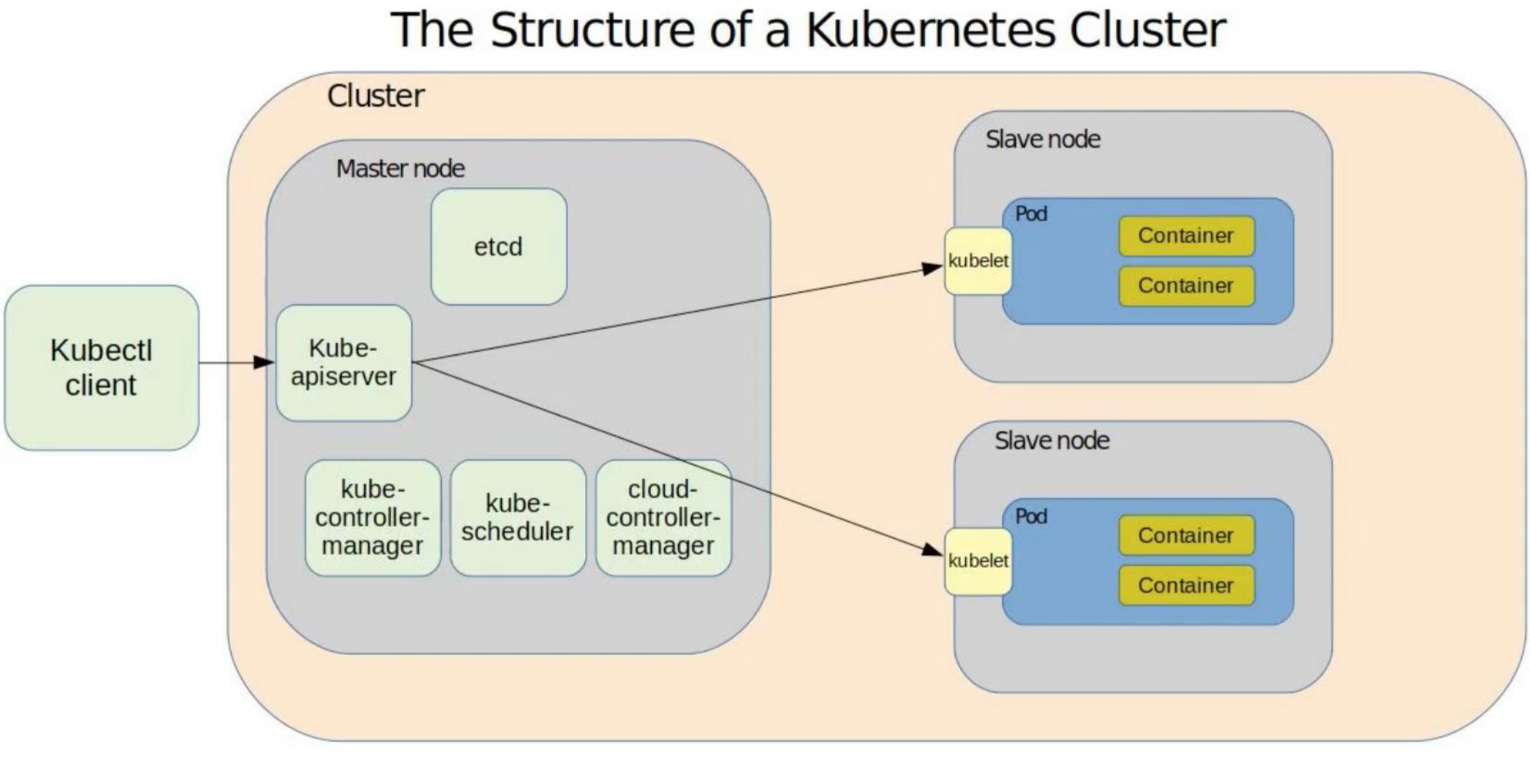
Orchestrating Kubernets vs. Docker Swarm

Amazon EKS:
Container Orchestration Using Kubernetes

David Clinton
SYSTEM ADMINISTRATOR
bootstrap-it.com/docker4aws | @davidbclinton | linkedin.com/in/dbclinton

Orchestrators

Kubernetes	Docker Swarm
Requires hypervisor backend	Lighter footprint
Complicated setup	Intuitive interface
Seamless integration	Less mature ecosystem
Robust monitoring	


The Structure of a Kubernetes Cluster
Cluster
Master node
etcd
Kube-
apiserver
kube-
controller-
manager
kube-
scheduler
cloud-
controller-
manager
Kubectl
client
Slave node
Pod
kubelet
Container
Container
Slave node
Pod
kubelet
Container
Container

Module Overview

Install eksctl

Build and provision cluster

Install and configure kubectl

Configure YAML definitions

Explore Kubernetes monitoring

aws configure settings

aws (cli)

```
$ aws configure
AWS Access Key ID [None]: AKIA5MDFWAM----YVQN7
AWS Secret Access Key [None]: YGGDh81vP1BzNrz+L5nN2y------0n9q8KFAdJcj
Default region name [None]: us-east-1
Default output format [None]:
```

```
$ aws configure
AWS Access Key ID [****************VQN7]:
AWS Secret Access Key [****************dJcj]:
Default region name [us-east-1]:
Default output format [None]:
```

```
$ cat /Users/luigicavuoti/.ecs/credentials
version: v1
default: default
ecs_profiles:
  default:
    aws_access_key_id: AKIA5MDFWA-----YVQN7
    aws_secret_access_key: YGGDh81vP1BzNrz+L5nN2ytz----0n9q8KFAdJcj
  ec2-test-App:
    aws_access_key_id: AKIA5MDFWAM----YVQN7
```

ecs-cli profile saved

```
    aws_secret_access_key: YGGDh81vP1BzNrz+L5nN2yt------0n9q8KFAdJcj

$ cat /Users/luigicavuoti/.ecs/config
version: v1
default: ec2-test-App
clusters:
  ec2-test-App:
    cluster: ec2-test-App
    region: us-east-1
    default_launch_type: EC2
```

EKS (elastiq kubernetes stack - cli)

https://docs.aws.amazon.com/eks/latest/userguide/getting-started.html

eksctl.io

Using Docker on AWS with the Command Line

Installing the eksctl and kubectl Tools

https://eksctl.io

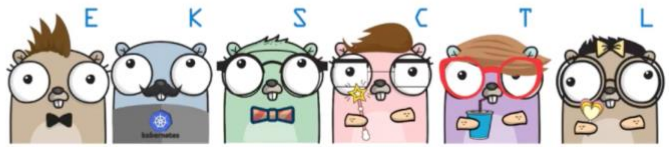
Star 1,345

sponsored by weaveworks and built by on

`eksctl` - a CLI for Amazon EKS

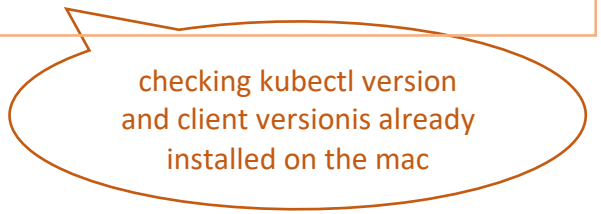
`eksctl` is a simple CLI tool for creating clusters on EKS - Amazon's new managed Kubernetes service for EC2. It is written in Go, and uses CloudFormation.

You can create a cluster in minutes with just one command – **`eksctl create cluster`**!



Need help? Join Weave Community Slack.

```
$ kubectl version
Client Version: version.Info{Major:"1", Minor:"19", GitVersion:"v1.19.7", GitCommit:"1dd5338295409edcfff11505e7bb246f0d325d15",
GitTreeState:"clean", BuildDate:"2021-01-13T13:23:52Z", GoVersion:"go1.15.5", Compiler:"gc", Platform:"darwin/amd64"}
The connection to the server localhost:8080 was refused - did you specify the right host or port?
MacBook-Pro-von-Luigi-Cavuoti:Luigi luigicavuoti$ kubectl version --short --client
Client Version: v1.19.7
```



install eksctl on the mac

```
brew tap weaveworks/tap


brew install weaveworks/tap/eksctl

$ eksctl version
0.49.0
```

Prepare a Cluster and an application

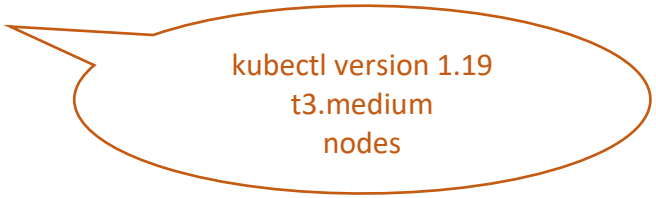
check for clusters:

```
$ eksctl get cluster
2021-05-16 21:12:07 [ℹ]  eksctl version 0.49.0
2021-05-16 21:12:07 [ℹ]  using region us-east-1
No clusters found
```

check the kubeconfig

create a cluster with a name : wp-cluster

```
eksctl create cluster \
--name wp-cluster \
--version 1.19 \
--nodegroup-name standard-workers \
--node-type t3.medium \
--nodes 3 \
--nodes-min 1 \
--nodes-max 4 \
--node-ami auto
```




```
$ eksctl create cluster \
> --name wp-cluster \
> --version 1.19 \
> --nodegroup-name standard-workers \
> --node-type t3.medium \
> --nodes 3 \
> --nodes-min 1 \
> --nodes-max 4 \
> --node-ami auto
2021-05-17 21:36:56 [ℹ]  eksctl version 0.49.0
2021-05-17 21:36:56 [ℹ]  using region us-east-1

2021-05-17 21:36:57 [ℹ]  setting availability zones to [us-east-1a us-east-1c]
2021-05-17 21:36:57 [ℹ]  subnets for us-east-1a - public:192.168.0.0/19 private:192.168.64.0/19
2021-05-17 21:36:57 [ℹ]  subnets for us-east-1c - public:192.168.32.0/19 private:192.168.96.0/19
2021-05-17 21:36:57 [!]  Custom AMI detected for nodegroup standard-workers. Please refer to
https://github.com/weaveworks/eksctl/issues/3563 for upcoming breaking changes
2021-05-17 21:36:57 [ℹ]  nodegroup "standard-workers" will use "ami-06c778c22c5e09bf7" [AmazonLinux2/1.19]
2021-05-17 21:36:57 [ℹ]  using Kubernetes version 1.19
2021-05-17 21:36:57 [ℹ]  creating EKS cluster "wp-cluster" in "us-east-1" region with un-managed nodes
2021-05-17 21:36:57 [ℹ]  will create 2 separate CloudFormation stacks for cluster itself and the initial nodegroup
2021-05-17 21:36:57 [ℹ]  if you encounter any issues, check CloudFormation console or try 'eksctl utils describe-stacks --region=us-east-1
--cluster=wp-cluster'
```

```
2021-05-17 21:36:57 [i]  CloudWatch logging will not be enabled for cluster "wp-cluster" in "us-east-1"
2021-05-17 21:36:57 [i]  you can enable it with 'eksctl utils update-cluster-logging --enable-types={SPECIFY-YOUR-LOG-TYPES-HERE (e.g.
all)} --region=us-east-1 --cluster=wp-cluster'
2021-05-17 21:36:57 [i]  Kubernetes API endpoint access will use default of {publicAccess=true, privateAccess=false} for cluster "wp-
cluster" in "us-east-1"
2021-05-17 21:36:57 [i]  2 sequential tasks: { create cluster control plane "wp-cluster", 3 sequential sub-tasks: { wait for control plane
to become ready, create addons, create nodegroup "standard-workers" } }
2021-05-17 21:36:57 [i]  building cluster stack "eksctl-wp-cluster-cluster"
2021-05-17 21:36:59 [i]  deploying stack "eksctl-wp-cluster-cluster"
```

delete a cluster


```
$ eksctl delete cluster --region=us-east-1 --name=wp-cluster
2021-05-17 21:35:34 [i]  eksctl version 0.49.0
2021-05-17 21:35:34 [i]  using region us-east-1
2021-05-17 21:35:34 [i]  deleting EKS cluster "wp-cluster"
2021-05-17 21:35:38 [✓]  kubeconfig has been updated
2021-05-17 21:35:39 [i]  1 task: { delete cluster control plane "wp-cluster" [async] }
2021-05-17 21:35:40 [i]  will delete stack "eksctl-wp-cluster-cluster"
2021-05-17 21:35:40 [✓]  all cluster resources were deleted
```


```
$ eksctl get cluster
2021-05-17 21:55:11 [i]  eksctl version 0.49.0
2021-05-17 21:55:11 [i]  using region us-east-1
```