



# WELCOME TO



&



# kubernetes

*September*  
**meetup**



**Dreaming about  
Kubernetes for years**



# ReceiptBank™

&



# kubernetes

# from Lab to Prod



 ReceiptBank™ != Bank



# ReceiptBank™

- how we started
- where we are
- where we intend to go

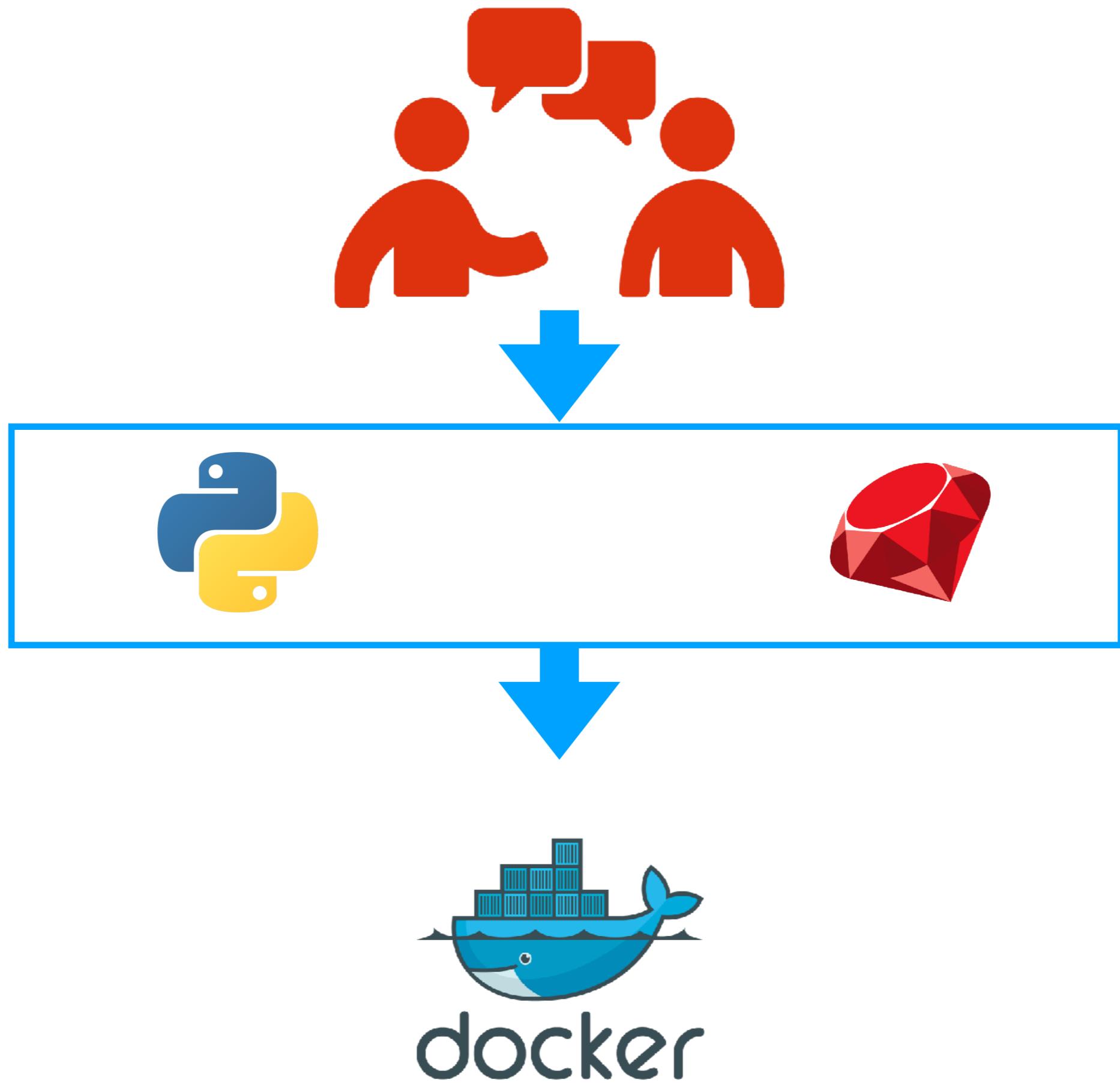


# ReceiptBank™

- Using Kubernetes for QA & Demo
- Daily dev process in/with kubernetes
- Having a production ready kubernetes



# How we started?



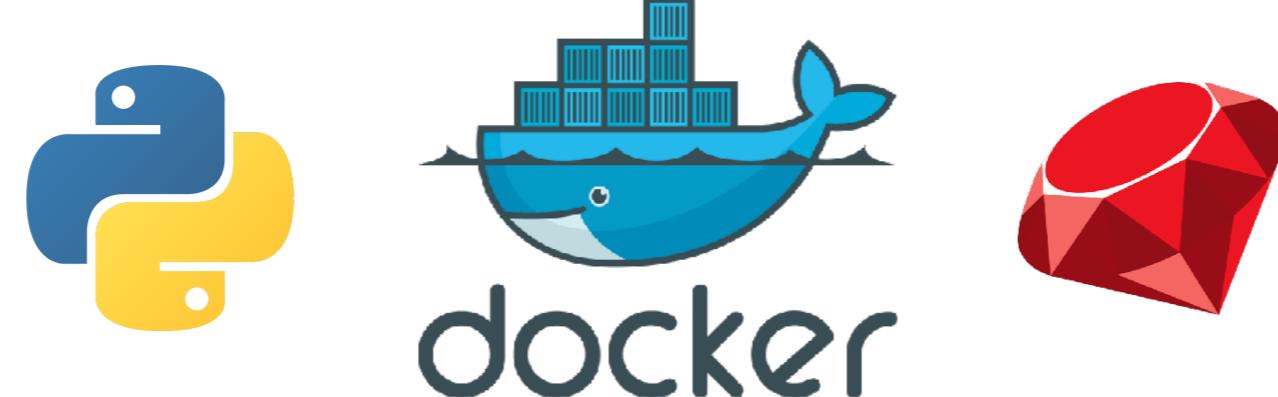


```
◆ Dockerfile ●
1  FROM ruby:2.5.1-slim
2
3  RUN apt-get update && apt-get -y install \
4      apt-transport-https \
5      ca-certificates \
6      curl \
7      gnupg \
8      lsb-release \
9      wget
10
11 RUN echo "deb http://apt.postgresql.org/pub/repos/apt/ $(lsb_release -cs)-pgdg main" > \
12     /etc/apt/sources.list.d/pgdg.list \
13 RUN wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | apt-key add - \
14 RUN wget -O- https://deb.nodesource.com/setup_10.x | bash - \
15 RUN curl -sS https://dl.yarnpkg.com/debian/pubkey.gpg | apt-key add - \
16 RUN echo "deb https://dl.yarnpkg.com/debian/ stable main" | tee /etc/apt/sources.list.d/yarn.list
17
18 RUN apt-get update && apt-get -y install \
19     build-essential \
20     bzip2 \
21     default-jre-headless \
22     expect \
23     fonts-croscore \
24     git \
25     imagemagick \
26     libcurl4-openssl-dev \
27     libmariadb-dev \
28     libpq-dev \
29     libreadline-dev \
30     libreoffice-common \
31     libreoffice-writer \
32     libssl1.0-dev \
33     nodejs \
34     openssh-client \
35     postgresql-client-10 \
36     poppler-utils \
37     readline-common \
38     tesseract-ocr \
39     unzip \
40     yarn \
41     zip \
42     zlib1g-dev
43
44 RUN mkdir /root/.ssh/
45 RUN echo 'Host github.com\n\tForwardAgent yes\n' > /root/.ssh/config
46 RUN ssh-keyscan github.com >> /root/.ssh/known_hosts
47
48 RUN gem install rack --version 1.4.7
49 RUN gem install rake --version 10.5.0
50 RUN gem install passenger --version 5.0.13
51 RUN passenger-install-nginx-module --auto --auto-download --languages ruby --prefix=/opt/nginx/
52
53 ENV APP_HOME=/opt/application
54 ENV BUNDLE_PATH=/opt/gems
55 ENV BUNDLE_APP_CONFIG=/opt/application/.bundle
56
57 WORKDIR $APP_HOME
58
```



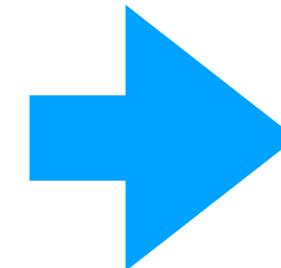
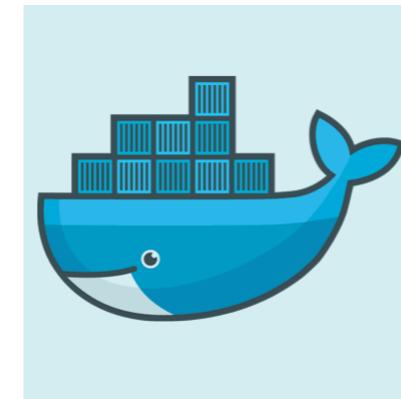
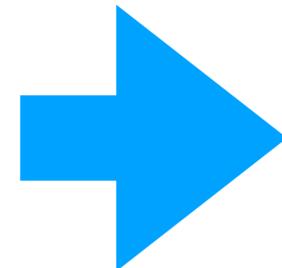
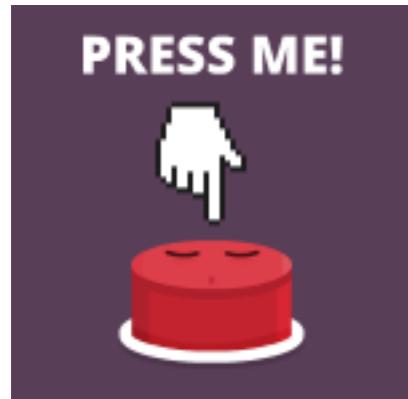
## ◀ Dockerfile ●

```
1 FROM python:3.6.2-slim
2
3 RUN apt-get update && apt-get install -y \
4     gfortran \
5     git \
6     hdf5-tools \
7     libatlas-base-dev \
8     libatlas-dev \
9     libatlas3-base \
10    libhdf5-dev \
11    libxml2-dev \
12    libxslt1-dev \
13    openssh-client \
14    python3-dev \
15    python3-pip \
16    unzip \
17    zip \
18
19 RUN pip install --disable-pip-version-check virtualenv==15.1.0
20
21 RUN mkdir /root/.ssh/
22 RUN echo 'Host github.com\n\tForwardAgent yes\n' > /root/.ssh/config
23 RUN ssh-keyscan github.com >> /root/.ssh/known_hosts
24
25 ENV APP_HOME=/opt/application
26 ENV REQUIREMENTS_PATH=/opt/requirements
27
28 WORKDIR $APP_HOME
29 |
```



# kubernetes





## namespace: Gosh0

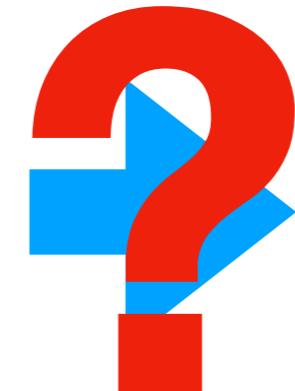
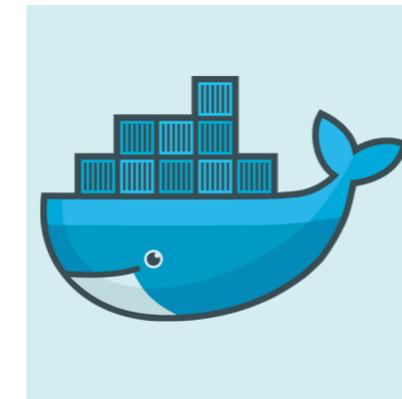
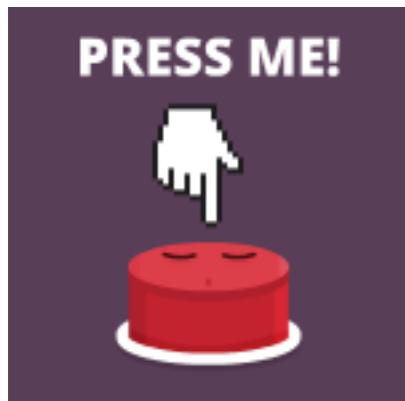
- App1 (branch4)
- App2 (branch8)
- App3 (master)
- App4 (staging)
- App5 (branch12)
- App6 (master)
- App7 (master)
- App8 (branch2)
- App9 (branch7)
- App10 (master)

## namespace: Tosh0

- App1 (master)
- App2 (branch2)
- App3 (master)
- App4 (master)
- App5 (master)
- App6 (staging)
- App7 (branch7)
- App8 (branch1)
- App9 (branch9)
- App10 (staging)

## namespace: Mosho

- App1 (staging)
- App2 (master)
- App3 (master)
- App4 (master)
- App5 (master)
- App6 (master)
- App7 (master)
- App8 (master)
- App9 (master)
- App10 (master)



## namespace: Gosh0

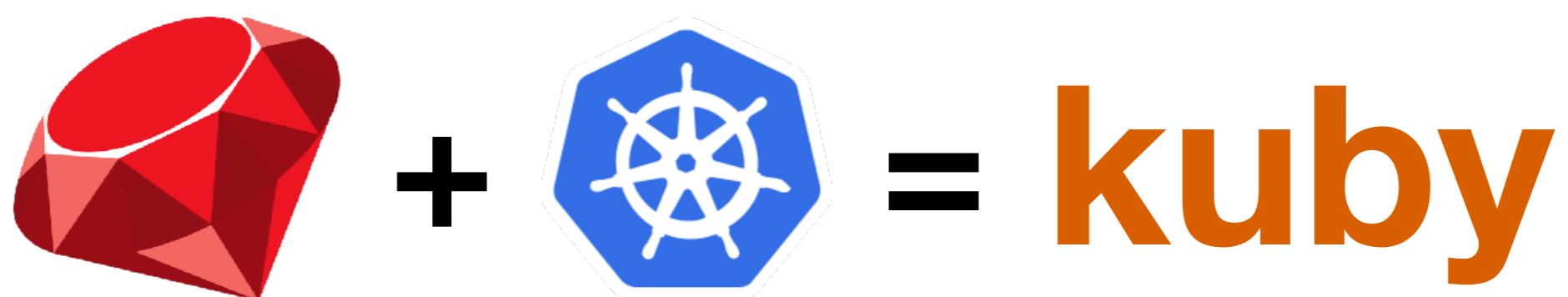
- App1 (branch4)
- App2 (branch8)
- App3 (master)
- App4 (staging)
- App5 (branch12)
- App6 (master)
- App7 (master)
- App8 (branch2)
- App9 (branch7)
- App10 (master)

## namespace: Tosho

- App1 (master)
- App2 (branch2)
- App3 (master)
- App4 (master)
- App5 (master)
- App6 (staging)
- App7 (branch7)
- App8 (branch1)
- App9 (branch9)
- App10 (staging)

## namespace: Mosho

- App1 (staging)
- App2 (master)
- App3 (master)
- App4 (master)
- App5 (master)
- App6 (master)
- App7 (master)
- App8 (master)
- App9 (master)
- App10 (master)





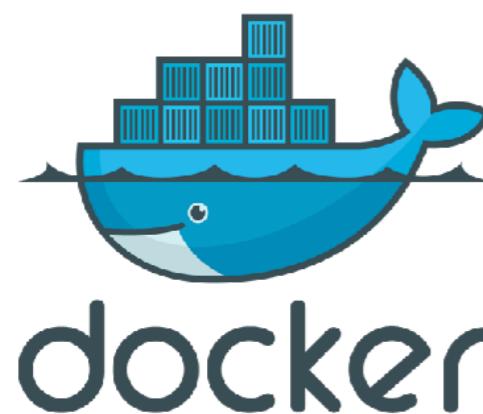
# WHO / WHAT IS KUBY?



# kuby

```
replica_set :web do |version, revision, namespace|
  name "companies-web-#{version}"
  labels app: 'companies', component: 'web', revision: revision, version: version
  ports 3000
  image "receiptbank/companies/web:#{revision}"
  readiness_probe get: '/_health_checks_/database', port: 3000, delay: 15, period: 5
  memory '300Mi'
end
```

docker build



API call





# How did we run Kubernetes?



# kOps



**Cluster 1****Staging****Cluster 2****Production****Cluster 3..?****On-demand**

- |  |   |  |
|--|---|--|
| <ul style="list-style-type: none"><li>- ami: k8s-1.11-debian-stretch</li><li>- docker: 17.09.0</li><li>- k8s: v1.12-beta1</li><li>- etcd: v3.3.9</li><li>- kops 1.10.0</li></ul> | <ul style="list-style-type: none"><li>- ami: k8s-1.10-debian-jessie</li><li>- docker: 17.03.2</li><li>- k8s: v1.11.0</li><li>- etcd: v3.1.11</li><li>- kops 1.9.2</li></ul> | <ul style="list-style-type: none"><li>- ami: k8s-1.11-debian-strech</li><li>- docker: 17.03.2</li><li>- k8s: v1.11.4</li><li>- etcd: v3.1.11</li><li>- kops 1.10.0</li></ul> |
|--|---|--|



# Where we are?



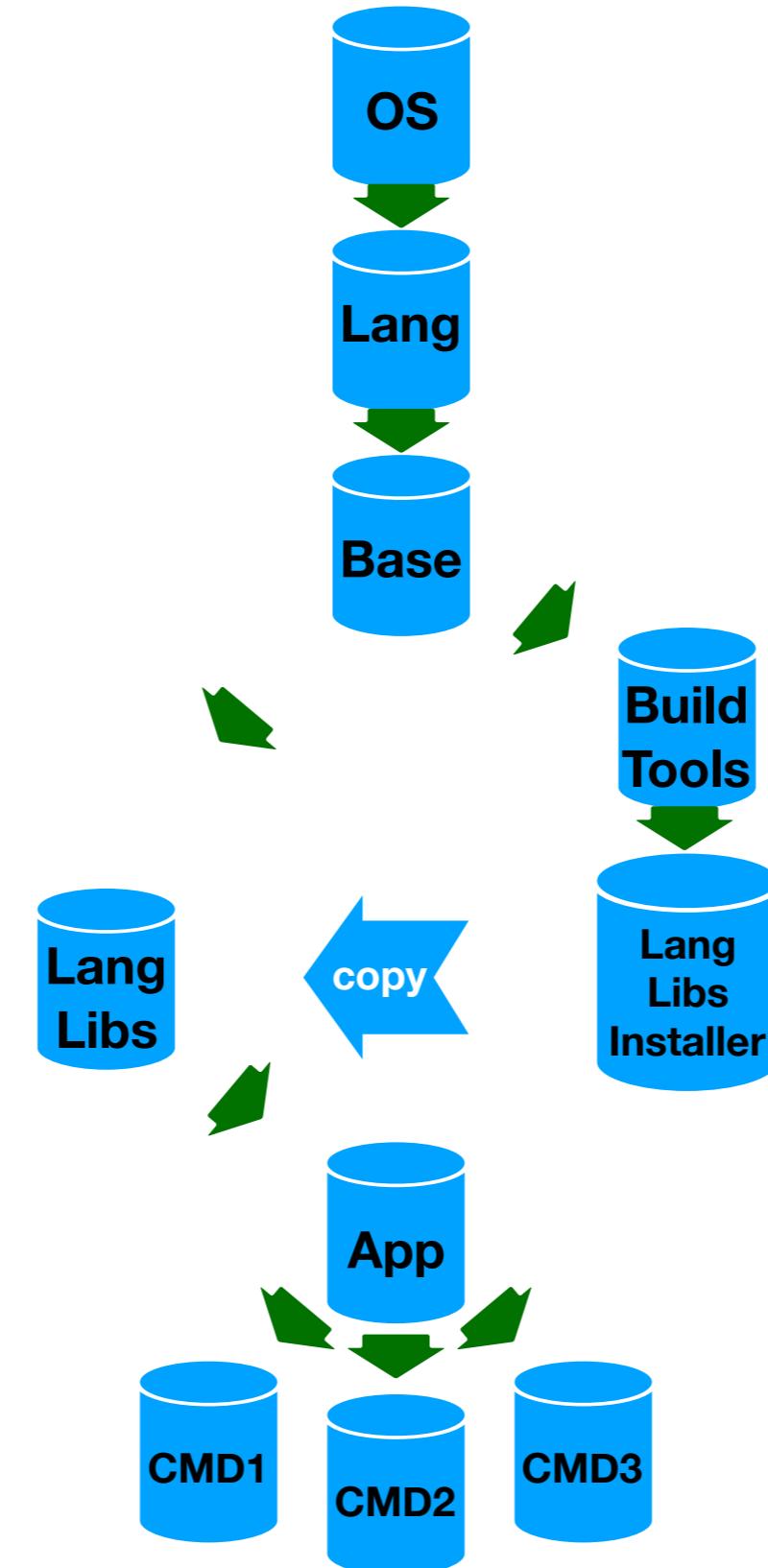
- FROM: (repository)

- Single RUN

- Change RUN only if you need to use cached docker layer

- Delete everything unnecessarily

```
◆ Dockerfile ×
1 ARG DOCKER_REPO
2 FROM $DOCKER_REPO/receiptbank/ruby:2.5.1
3
4 RUN apt-get update \
5     && apt-get -y install --no-install-recommends \
6     gnupg \
7     lsb-release \
8     && echo "deb http://apt.postgresql.org/pub/repos/apt/ $(lsb_release -cs)-pgdg main" > \
9     /etc/apt/sources.list.d/pgdg.list \
10    && curl -sL https://www.postgresql.org/media/keys/ACCC4CF8.asc | apt-key add - \
11    && curl -sL https://deb.nodesource.com/setup_10.x -o setup_10.x \
12    && echo "3a9e17ee8454b83b2fa2cb5a30de369ffafeff25 setup_10.x" | sha1sum -c - \
13    && bash setup_10.x && rm -f setup_10.x \
14    && curl -sS https://dl.yarnpkg.com/debian/pubkey.gpg | apt-key add - \
15    && echo "deb https://dl.yarnpkg.com/debian/ stable main" | tee /etc/apt/sources.list.d/yarn.list \
16    && buildDeps=' \
17    gcc \
18    g++ \
19    make \
20    libcurl4-openssl-dev \
21    libssl1.0-dev \
22    zlib1g-dev \
23    ' \
24    && apt-get update \
25    && apt-get -y install --no-install-recommends \
26    $buildDeps \
27    bzip2 \
28    default-jre-headless \
29    expect \
30    fonts-croscore \
31    imagemagick \
32    libreoffice-common \
33    libreoffice-writer \
34    nodejs \
35    postgresql-client-10 \
36    poppler-utils \
37    readline-common \
38    tesseract-ocr \
39    unzip \
40    yarn \
41    zip \
42    && gem install passenger --version 5.0.13 \
43    && passenger-install-nginx-module --auto --auto-download --languages ruby --prefix=/opt/nginx/ \
44    && apt-get purge -y --auto-remove $buildDeps \
45    && rm -rf /var/lib/apt/lists/*
46
47 WORKDIR $APP_HOME
48
```





# We need



elasticsearch



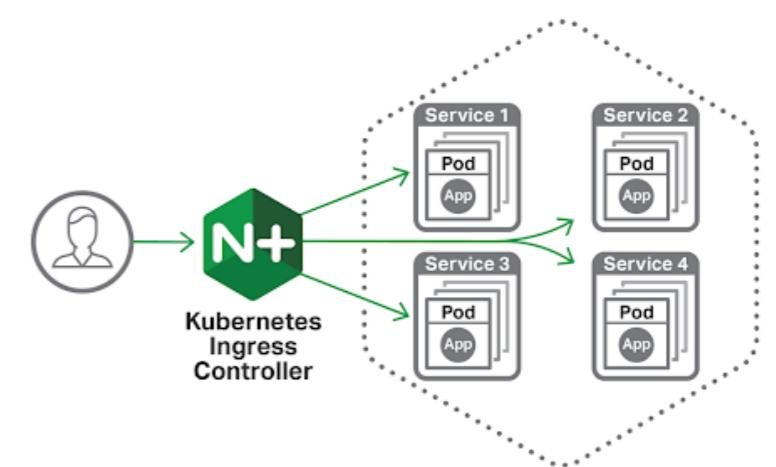
logstash

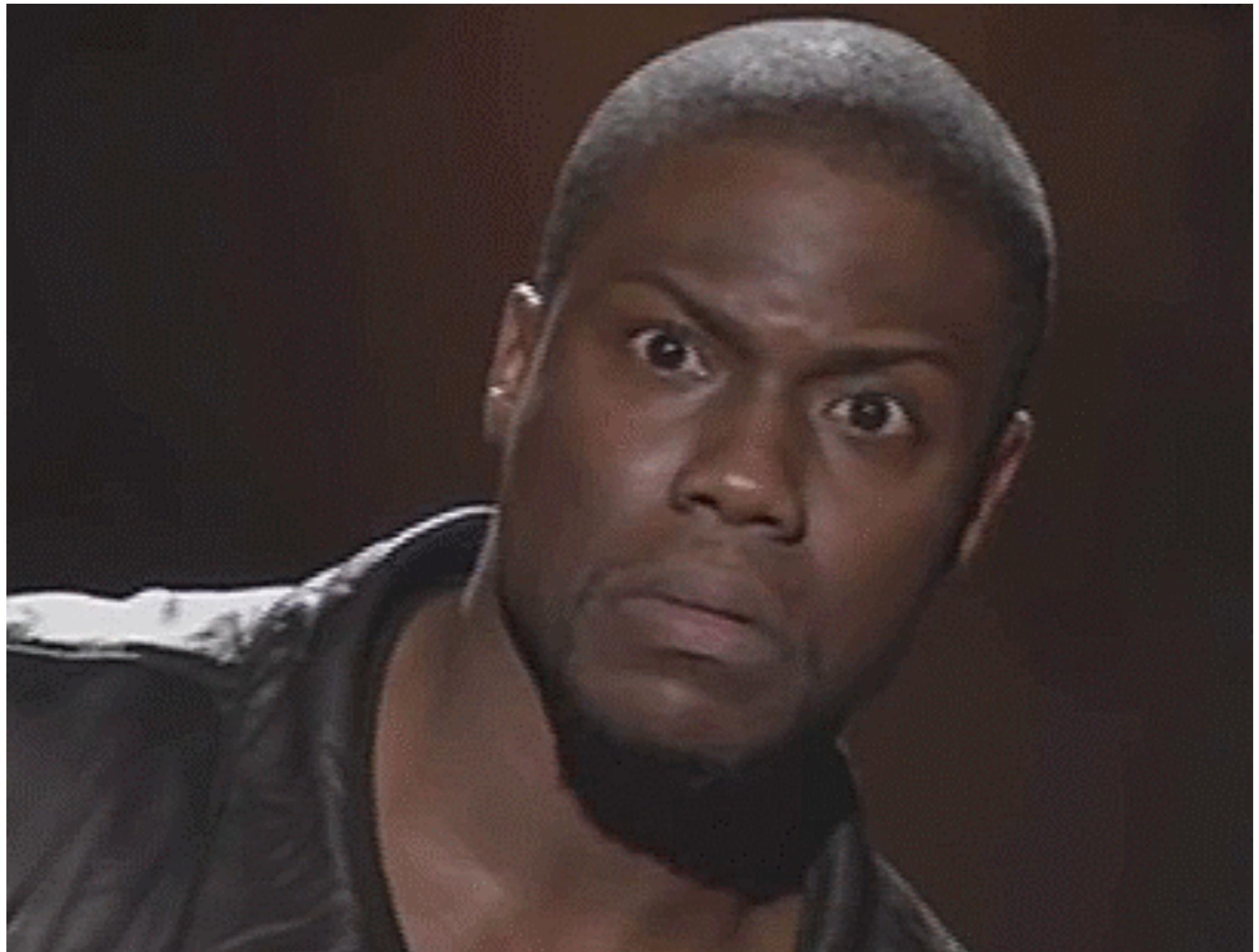


kibana



AlertManager  
Details:  
Prometheus  
  
Details:  
Prometheus







# System namespace

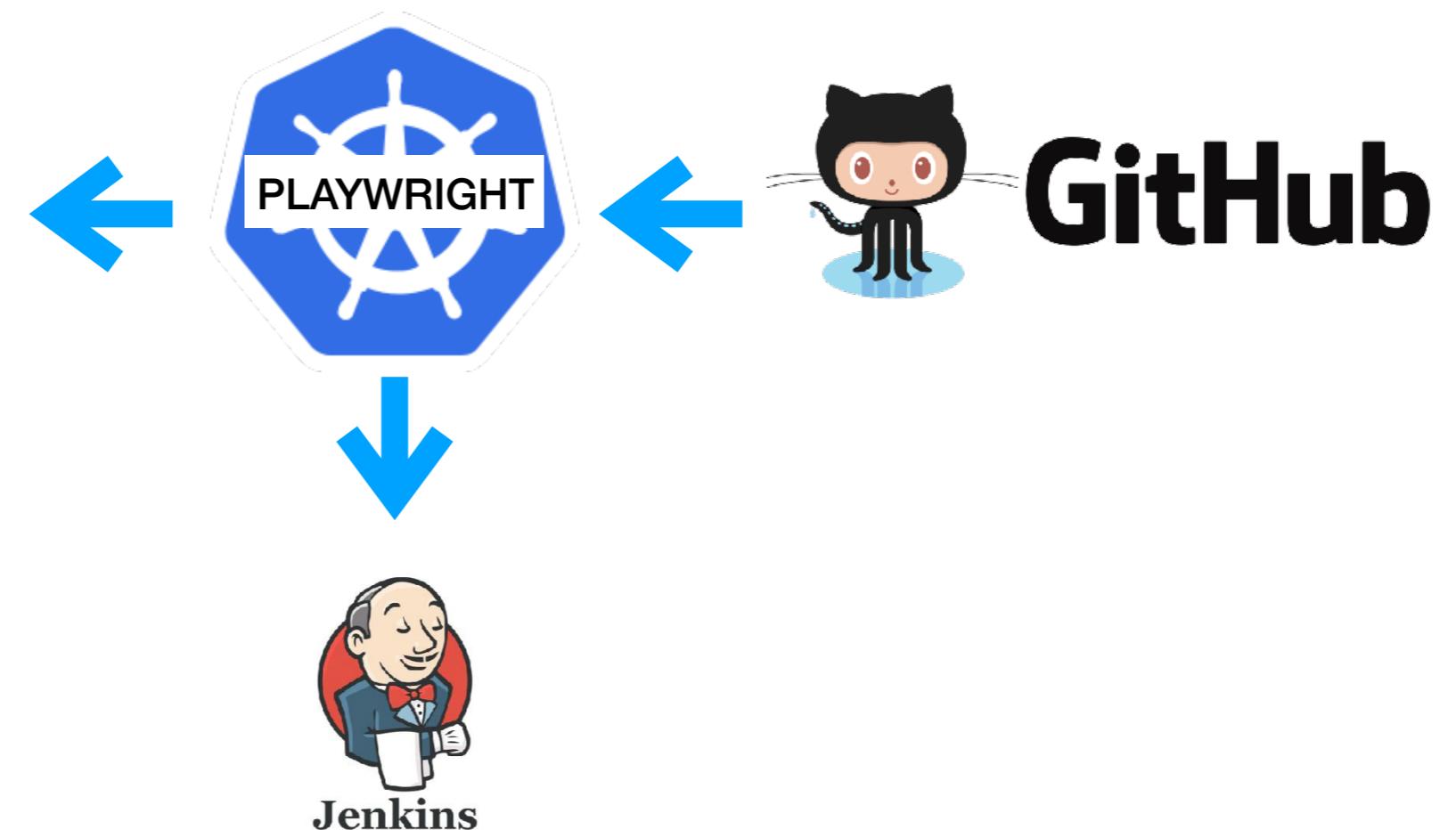
```
├── 01-namespace.yaml
├── 02-external-dns.yaml
├── 05-docker-registry.yaml
├── 06-nginx-ingress.yaml
├── 07-logging.yaml
├── 08-internal-monitoring
│   ├── 01-prometheus
│   │   ├── 01-prometheus-config.yaml
│   │   ├── 02-prometheus-rules.yaml
│   │   ├── 03-prometheus-node-exporter.yaml
│   │   ├── 04-kube-state-metrics.yaml
│   │   ├── 05-prometheus.yaml
│   │   ├── 06-elasticsearch-exporter.yaml
│   │   ├── 07-thanos-query.yaml
│   │   ├── 08-prometheus-blackbox-exporter.yaml
│   │   └── 09-kube-proxy-exporter.yaml
│   ├── 02-alertmanager
│   │   ├── 01-alertmanager-config.yaml
│   │   └── 02-alertmanager.yaml
│   └── 03-grafana
│       ├── 01-grafana-dashboards
│       │   ├── 01-grafana-config.yaml
│       │   ├── 02-home-dashboard.yaml
│       │   ├── 03-capacity-planning-dashboard.yaml
│       │   ├── 04-deployment-dashboard.yaml
│       │   ├── 05-etcd-dashboard.yaml
│       │   ├── 06-full-cluster-monitoring-dashboard.yaml
│       │   ├── 07-monolith-dashboard.yaml
│       │   ├── 08-nodes-dashboard.yaml
│       │   ├── 09-pods-dashboard.yaml
│       │   ├── 10-resource-requests-dashboard.yaml
│       │   ├── 11-statefulset-dashboard.yaml
│       │   ├── 12-elasticsearch-dashboard.yaml
│       │   ├── 13-az-memory-dashboard.yaml
│       │   └── 14-blackbox-overview-dashboard.yaml
│       ├── 02-grafana.yaml
│       └── 03-grafana-config.yaml
└── 09-external-monitoring.yaml
    ├── 10-developer-rbac.yaml
    ├── 11-playwright.yaml
    ├── 12-jenkins.yaml
    ├── 13-termination-handler.yaml
    └── 14-default-network-policies.yaml
```



# PLAYWRIGHT

- Kuby's API interface
- Working with GitHub web hooks
- Knowledge about namespaces with app workloads
- Speaking with Kubernetes API server

Kubernetes  
API





# ReceiptBank

**Demo video is not available**



# It's Working!@#\$%





Start spending more time on



**kubernetes**



## We need to control:

- 1. AMI**
- 2. Infrastructure**
- 3. Bootstrap**



# Building AMI

[tianon / docker-brew-ubuntu-core](#) Watch 45 Star 294 Fork 249

Code Issues 5 Pull requests 0 Projects 0 Insights

Tree: 59aa7dfef1 → [docker-brew-ubuntu-core / bionic / Dockerfile](#) Find file Copy path

**docker-library-bot** Update to 20180125 for amd64 (amd64) 85822fe on Jan 25

1 contributor

49 lines (43 sloc) | 2.77 KB Raw Blame History

```
1 FROM scratch
2 ADD ubuntu-bionic-core-cloudimg-amd64-root.tar.gz /
3
4 # a few minor docker-specific tweaks
5 # see https://github.com/docker/docker/blob/9a9fc01af8fb5d98b8eec0740716226fadbd3735c/contrib/mkimage/debootstrap
6
7
```

[tianon / docker-brew-ubuntu-core](#) Watch 45 Star 294 Fork 249

Code Issues 5 Pull requests 0 Projects 0 Insights

Tree: 59aa7dfef1 → [docker-brew-ubuntu-core / bionic /](#) Create new file Upload files Find file History

**docker-library-bot** Update to 20180821 for amd64 (amd64) ... Latest commit 59aa7df 16 days ago

..

File	Description	Updated
Dockerfile	Update to 20180821 for amd64 (amd64)	16 days ago
MD5SUMS	Update to 20180821 for amd64 (amd64)	16 days ago
MD5SUMS.gpg	Update to 20180821 for amd64 (amd64)	16 days ago
SHA1SUMS	Update to 20180821 for amd64 (amd64)	16 days ago
SHA1SUMS.gpg	Update to 20180821 for amd64 (amd64)	16 days ago
SHA256SUMS	Update to 20180821 for amd64 (amd64)	16 days ago
SHA256SUMS.gpg	Update to 20180821 for amd64 (amd64)	16 days ago
alias	Add bionic	10 months ago
build-info.txt	Update to 20180821 for amd64 (amd64)	16 days ago
ubuntu-bionic-core-cloudimg-amd64-root.tar.gz	Update to 20180821 for amd64 (amd64)	16 days ago
ubuntu-bionic-core-cloudimg-amd64.manifest	Update to 20180821 for amd64 (amd64)	16 days ago



# Building AMI



HashiCorp  
**Packer**

+

**ubuntu-bionic-core-cloudimg-amd64-root.tar.gz**

+

**Grub**

+

**sysctls**

+

**Docker**



# ReceiptBank

**Demo video is not available**



# Building Infrastructure



# Terraform

## Custom TF Modules

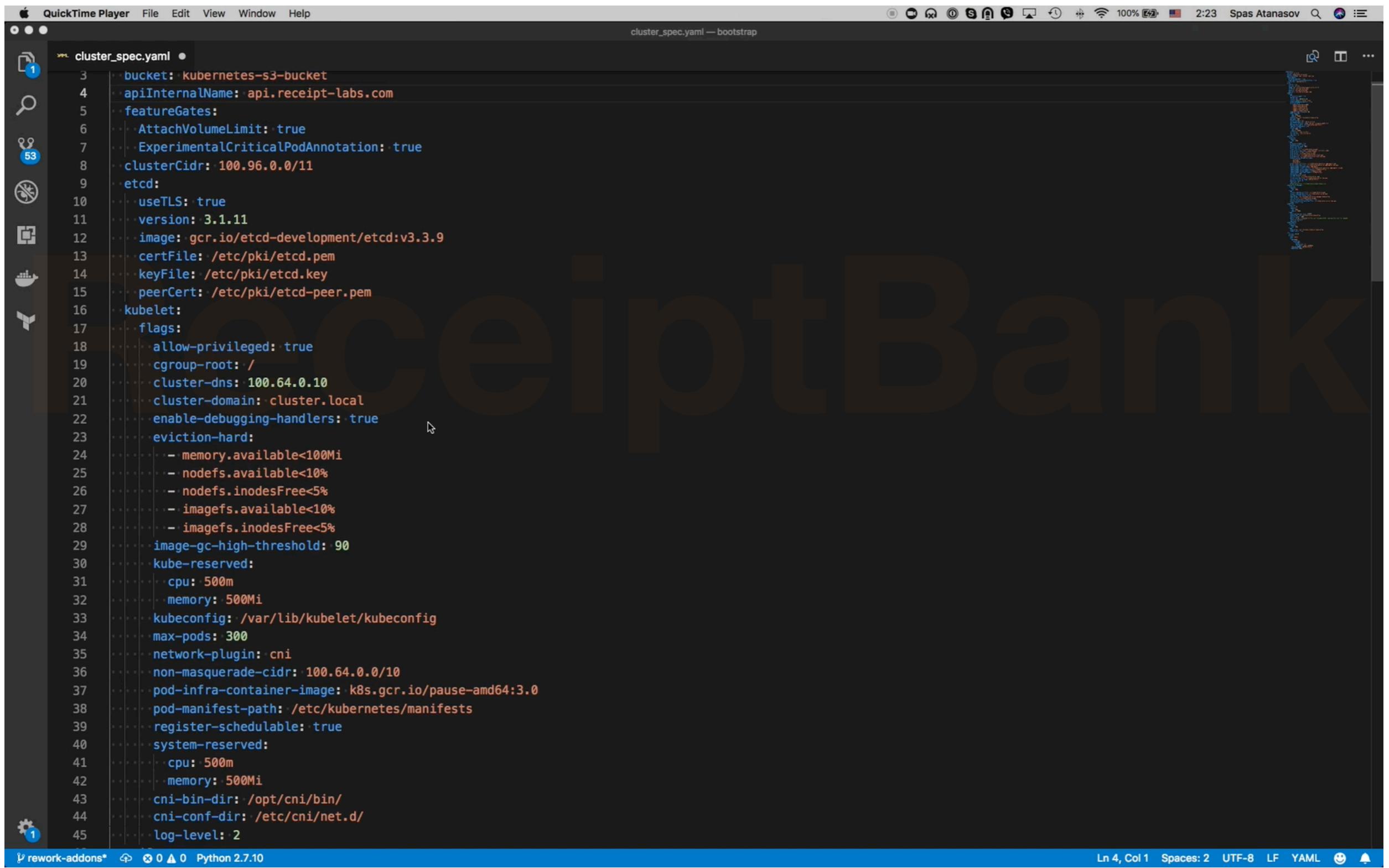
- Network
- Autoscaling
- ELB
- IAM
- Security Groups

## Separate TF States

- Base (s3, dynamodb)
- Network
- Per Masters
- ETCD Volumes
- Per Nodes
- Everything Shared
- Security Groups



# Bootstrap + Cluster spec



The screenshot shows a terminal window titled "cluster\_spec.yaml — bootstrap" in a dark-themed interface. The window contains a large amount of YAML configuration code for a Kubernetes cluster. The code defines various parameters such as bucket, apiInternalName, featureGates, etcd, kubelet, kube-reserved, and system-reserved resources. It also specifies network configurations like clusterCidr, cluster-dns, and non-masquerade-cidr. The terminal window includes standard OS X icons for file operations and a status bar at the bottom.

```
cluster_spec.yaml
1 bucket: kubernetes-s3-bucket
2 apiInternalName: api.receipt-labs.com
3 featureGates:
4   AttachVolumeLimit: true
5   ExperimentalCriticalPodAnnotation: true
6 clusterCidr: 100.96.0.0/11
7 etcd:
8   useTLS: true
9   version: 3.1.11
10  image: gcr.io/etcd-development/etcd:v3.3.9
11  certFile: /etc/pki/etcd.pem
12  keyFile: /etc/pki/etcd.key
13  peerCert: /etc/pki/etcd-peer.pem
14 kubelet:
15   flags:
16     allow-privileged: true
17     cgroup-root: /
18   cluster-dns: 100.64.0.10
19   cluster-domain: cluster.local
20   enable-debugging-handlers: true
21   eviction-hard:
22     - memory.available<100Mi
23     - nodefs.available<10%
24     - nodefs.inodesFree<5%
25     - imagefs.available<10%
26     - imagefs.inodesFree<5%
27   image-gc-high-threshold: 90
28 kube-reserved:
29   cpu: 500m
30   memory: 500Mi
31 kubeconfig: /var/lib/kubelet/kubeconfig
32 max-pods: 300
33 network-plugin: cni
34 non-masquerade-cidr: 100.64.0.0/10
35 pod-infra-container-image: k8s.gcr.io/pause-amd64:3.0
36 pod-manifest-path: /etc/kubernetes/manifests
37 register-schedulable: true
38 system-reserved:
39   cpu: 500m
40   memory: 500Mi
41   cni-bin-dir: /opt/cni/bin/
42   cni-conf-dir: /etc/cni/net.d/
43 log-level: 2
```

QuickTime Player File Edit View Window Help

cluster\_spec.yaml — bootstrap

Ln 4, Col 1 Spaces: 2 UTF-8 LF YAML



# ReceiptBank

**Demo video is not available**



**Where we  
intend to go?**



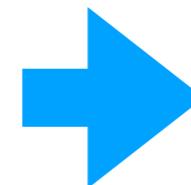
# Security

by default

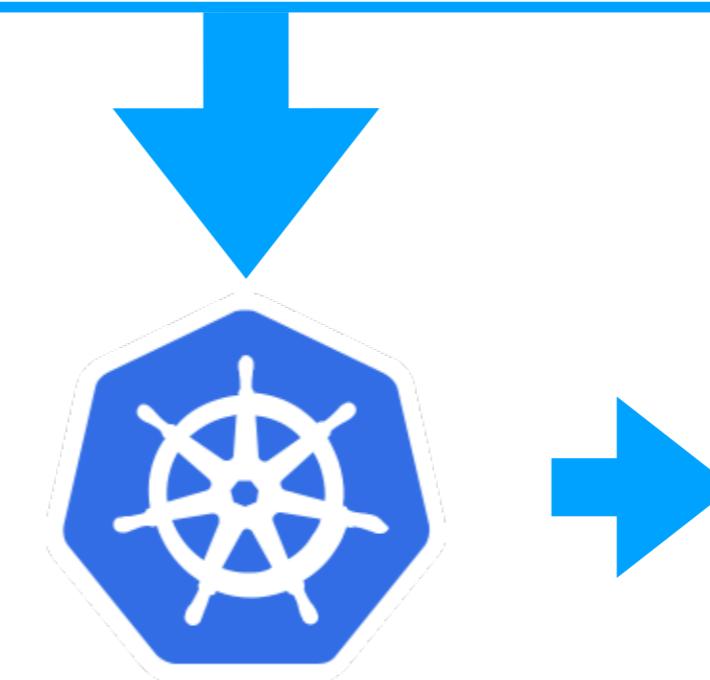
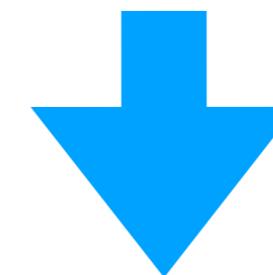
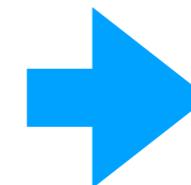


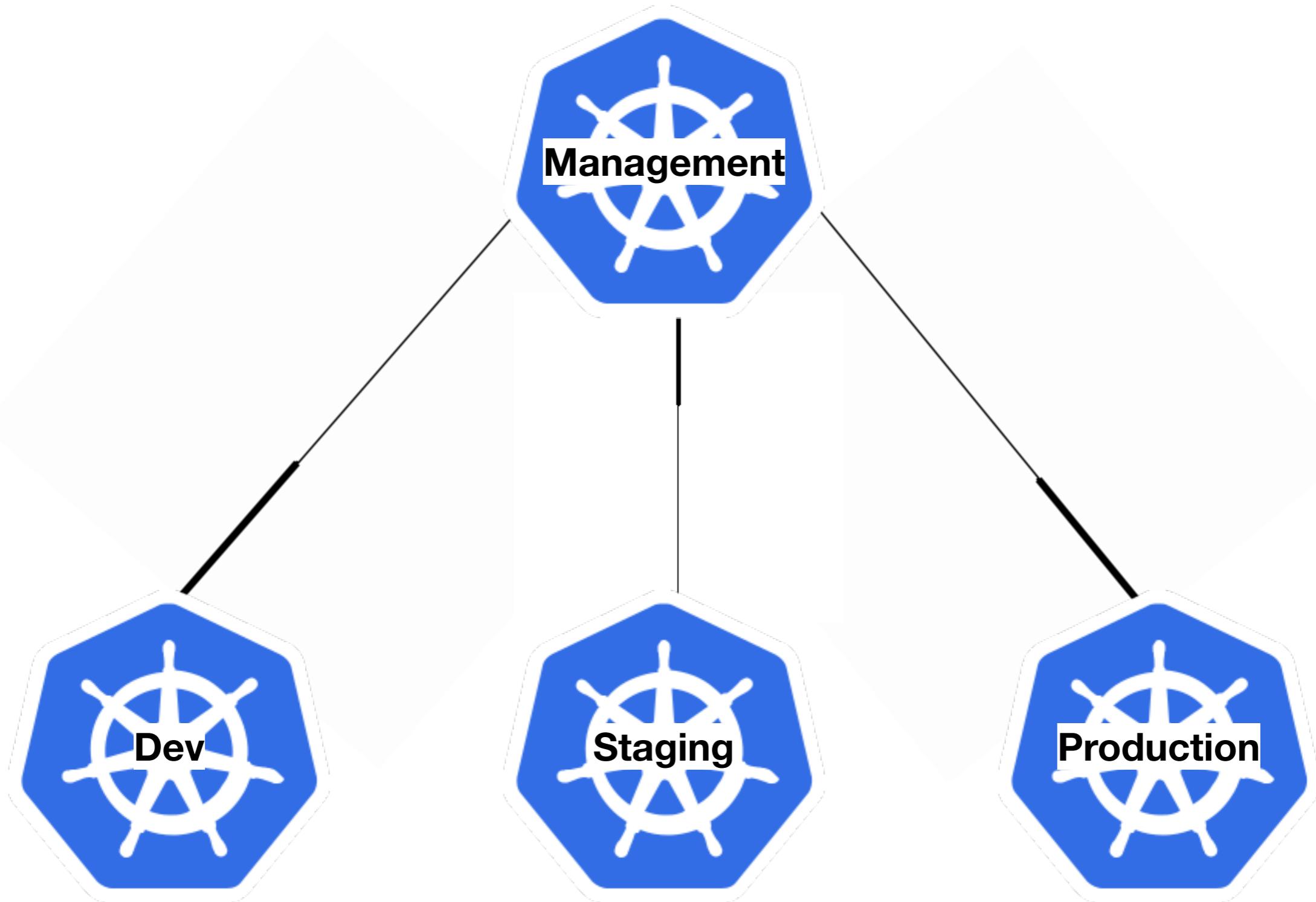
anchore





```
cluster_name: Spasterix
network_cidr: 10.10.0.0/16
docker_version: 17.09.0
kubernetes_version: 1.11.2
etcd_version: 3.3.9
cni_plugin: weave
number_of_masters: 5
number_of_workers: 15
...
...
```









# #QnA?



spasatanasov