WELCOME

TO

CLOUD NATIVE
COMPUTING FOUNDATION

&

kubernetes

September Meetup

Dreaming about Kubernetes for years

ReceiptBank™

&

kubernetes

from Lab to Prod

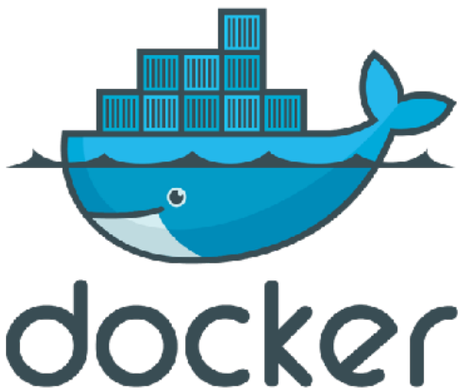
ReceiptBank™ != Bank

ReceiptBank™

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

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

How we started?

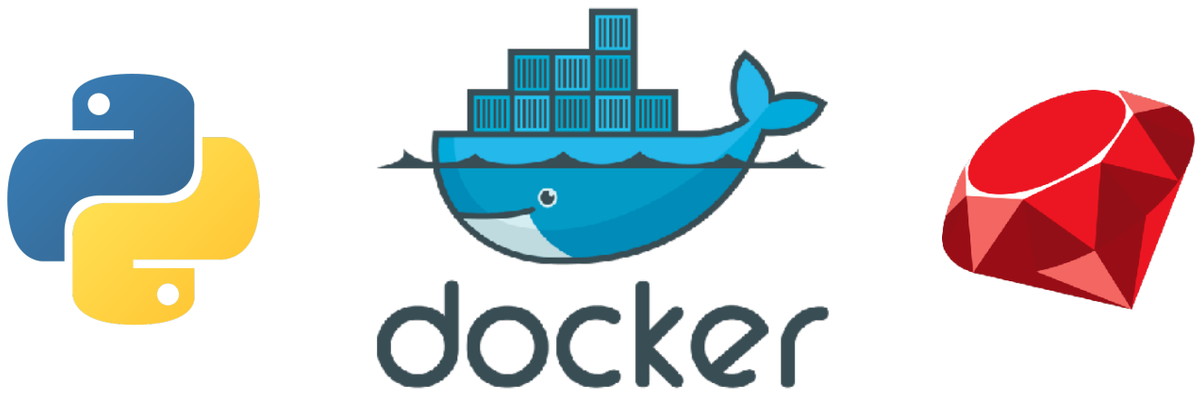

docker

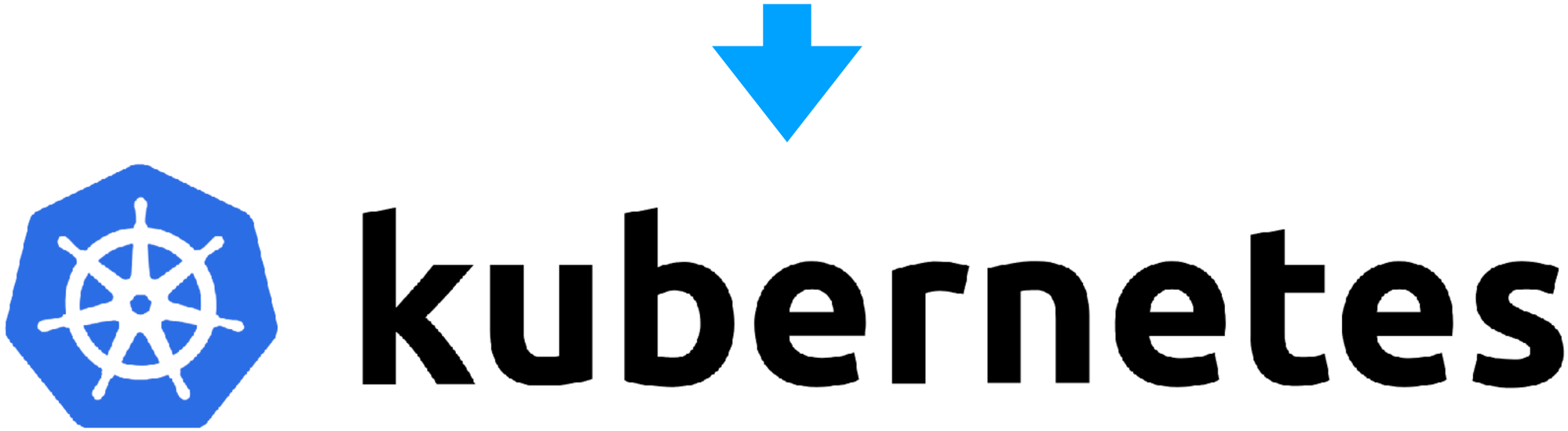
Dockerfile

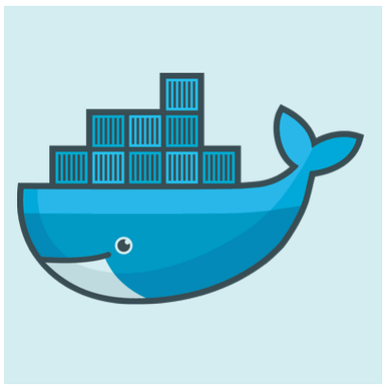
```
FROM ruby:2.5.1-slim

RUN apt-get update && apt-get -y install \
    apt-transport-https \
    ca-certificates \
    curl \
    gnupg \
    lsb-release \
    wget

RUN echo "deb http://apt.postgresql.org/pub/repos/apt/ $(lsb_release -cs)-pgdg main" > \
    /etc/apt/sources.list.d/pgdg.list \
RUN wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | apt-key add - \
RUN wget -O- https://deb.nodesource.com/setup_10.x | bash - \
RUN curl -sS https://dl.yarnpkg.com/debian/pubkey.gpg | apt-key add - \
RUN echo "deb https://dl.yarnpkg.com/debian/ stable main" | tee /etc/apt/sources.list.d/yarn.list

RUN apt-get update && apt-get -y install \
    build-essential \
    bzip2 \
    default-jre-headless \
    expect \
    fonts-croscore \
    git \
    imagemagick \
    libcurl4-openssl-dev \
    libmariadbd-dev \
    libpq-dev \
    libreadline-dev \
    libreoffice-common \
    libreoffice-writer \
    libssl1.0-dev \
    nodejs \
    openssh-client \
    postgresql-client-10 \
    poppler-utils \
    readline-common \
    tesseract-ocr \
    unzip \
    yarn \
    zip \
    zlib1g-dev

RUN mkdir /root/.ssh/
RUN echo 'Host github.com\n\tForwardAgent yes\n' > /root/.ssh/config
RUN ssh-keyscan github.com >> /root/.ssh/known_hosts

RUN gem install rack --version 1.4.7
RUN gem install rake --version 10.5.0
RUN gem install passenger --version 5.0.13
RUN passenger-install-nginx-module --auto --auto-download --languages ruby --prefix=/opt/nginx/

ENV APP_HOME=/opt/application
ENV BUNDLE_PATH=/opt/gems
ENV BUNDLE_APP_CONFIG=/opt/application/.bundle

WORKDIR $APP_HOME
```

Dockerfile

```
FROM python:3.6.2-slim

RUN apt-get update && apt-get install -y \
    gfortran \
    git \
    hdf5-tools \
    libatlas-base-dev \
    libatlas-dev \
    libatlas3-base \
    libhdf5-dev \
    libxml2-dev \
    libxslt1-dev \
    openssh-client \
    python3-dev \
    python3-pip \
    unzip \
    zip

RUN pip install --disable-pip-version-check virtualenv==15.1.0

RUN mkdir /root/.ssh/
RUN echo 'Host github.com\n\tForwardAgent yes\n' > /root/.ssh/config
RUN ssh-keyscan github.com >> /root/.ssh/known_hosts

ENV APP_HOME=/opt/application
ENV REQUIREMENTS_PATH=/opt/requirements

WORKDIR $APP_HOME

```


docker


kubernetes



namespace: Gosho

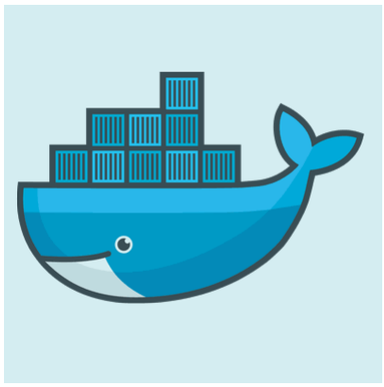
- App1 (branch4)
- App2 (branch8)
- App3 (master)
- App4 (staging)
- App5 (branch12)
- App6 (master)
- App7 (master)
- App8 (branch2)
- App9 (branch7)
- App10 (master)

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)



namespace: Gosho

- 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)

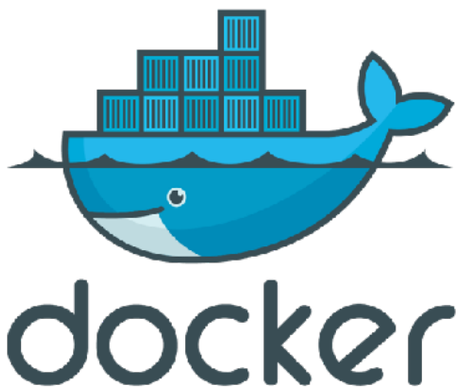

+
=
kuby

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

- ami: k8s-1.11-debian-stretch
- docker: 17.09.0
- k8s: v1.12-beta1
- etcd: v3.3.9
- kops 1.10.0

Cluster 2

Production

- ami: k8s-1.10-debian-jessie
- docker: 17.03.2
- k8s: v1.11.0
- etcd: v3.1.11
- kops 1.9.2

Cluster 3..?

On-demand

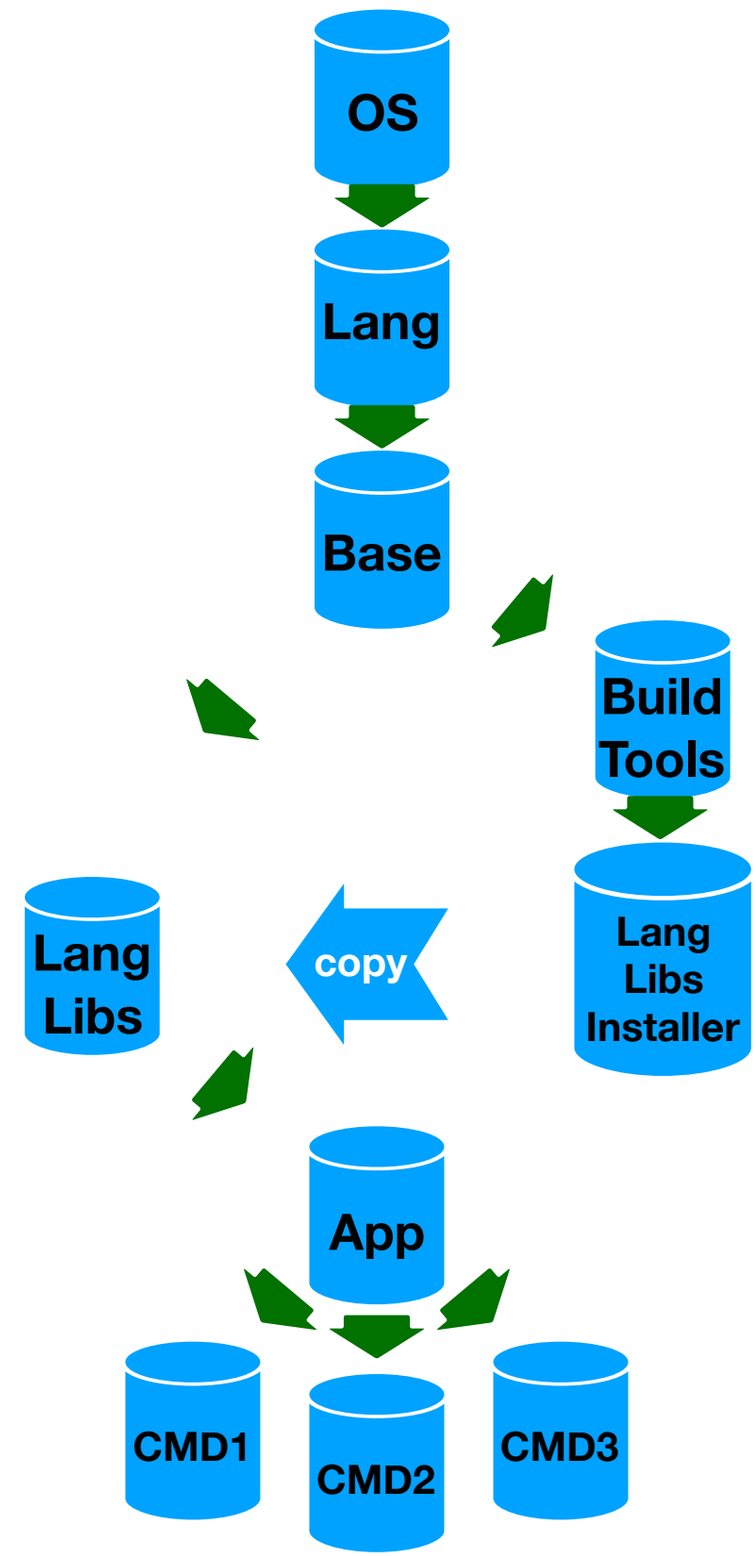
- ami: k8s-1.11-debian-strech
- docker: 17.03.2
- k8s: v1.11.4
- etcd: v3.1.11
- kops 1.10.0

Where we are?

- FROM: (repository)
- Single RUN
- Change RUN only if you need to use cached docker layer
- Delete everything unnecessarily

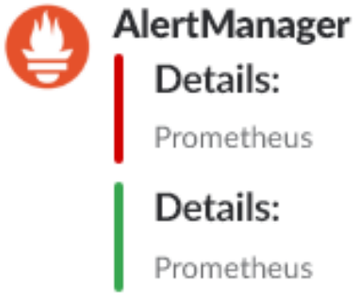
Dockerfile

```
ARG DOCKER_REPO
FROM $DOCKER_REPO/receiptbank/ruby:2.5.1

RUN apt-get update \
  && apt-get -y install --no-install-recommends \
    gnupg \
    lsb-release \
  && echo "deb http://apt.postgresql.org/pub/repos/apt/ $(lsb_release -cs)-pgdg main" > \
    /etc/apt/sources.list.d/pgdg.list \
  && curl -sL https://www.postgresql.org/media/keys/ACCC4CF8.asc | apt-key add - \
  && curl -sL https://deb.nodesource.com/setup_10.x -o setup_10.x \
  && echo "3a9e17ee8454b83b2fa2cb5a30de369ffafeff25 setup_10.x" | sha1sum -c - \
  && bash setup_10.x && rm -f setup_10.x \
  && curl -sS https://dl.yarnpkg.com/debian/pubkey.gpg | apt-key add - \
  && echo "deb https://dl.yarnpkg.com/debian/ stable main" | tee /etc/apt/sources.list.d/yarn.list \
  && buildDeps=' \
    gcc \
    g++ \
    make \
    libcurl4-gnutls-dev \
    libssl1.0-dev \
    zlib1g-dev \
  ' \
  && apt-get update \
  && apt-get -y install --no-install-recommends \
    $buildDeps \
    bzip2 \
    default-jre-headless \
    expect \
    fonts-croscore \
    imagemagick \
    libreoffice-common \
    libreoffice-writer \
    nodejs \
    postgresql-client-10 \
    poppler-utils \
    readline-common \
    tesseract-ocr \
    unzip \
    yarn \
    zip \
  && gem install passenger --version 5.0.13 \
  && passenger-install-nginx-module --auto --auto-download --languages ruby --prefix=/opt/nginx/ \
  && apt-get purge -y --auto-remove $buildDeps \
  && rm -rf /var/lib/apt/lists/*

WORKDIR $APP_HOME

```


OS
Lang
Base
Build
Tools
Lang
Libs
copy
Lang
Libs
Installer
App
CMD1
CMD2
CMD3

We need


elasticsearch
logstash
kibana

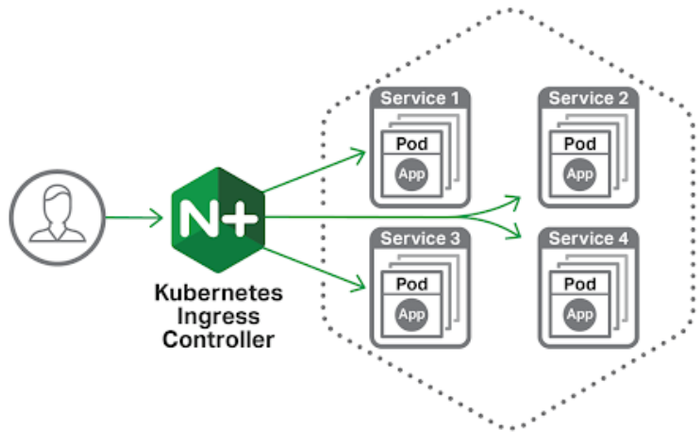

AlertManager
Details:
Prometheus
Details:
Prometheus

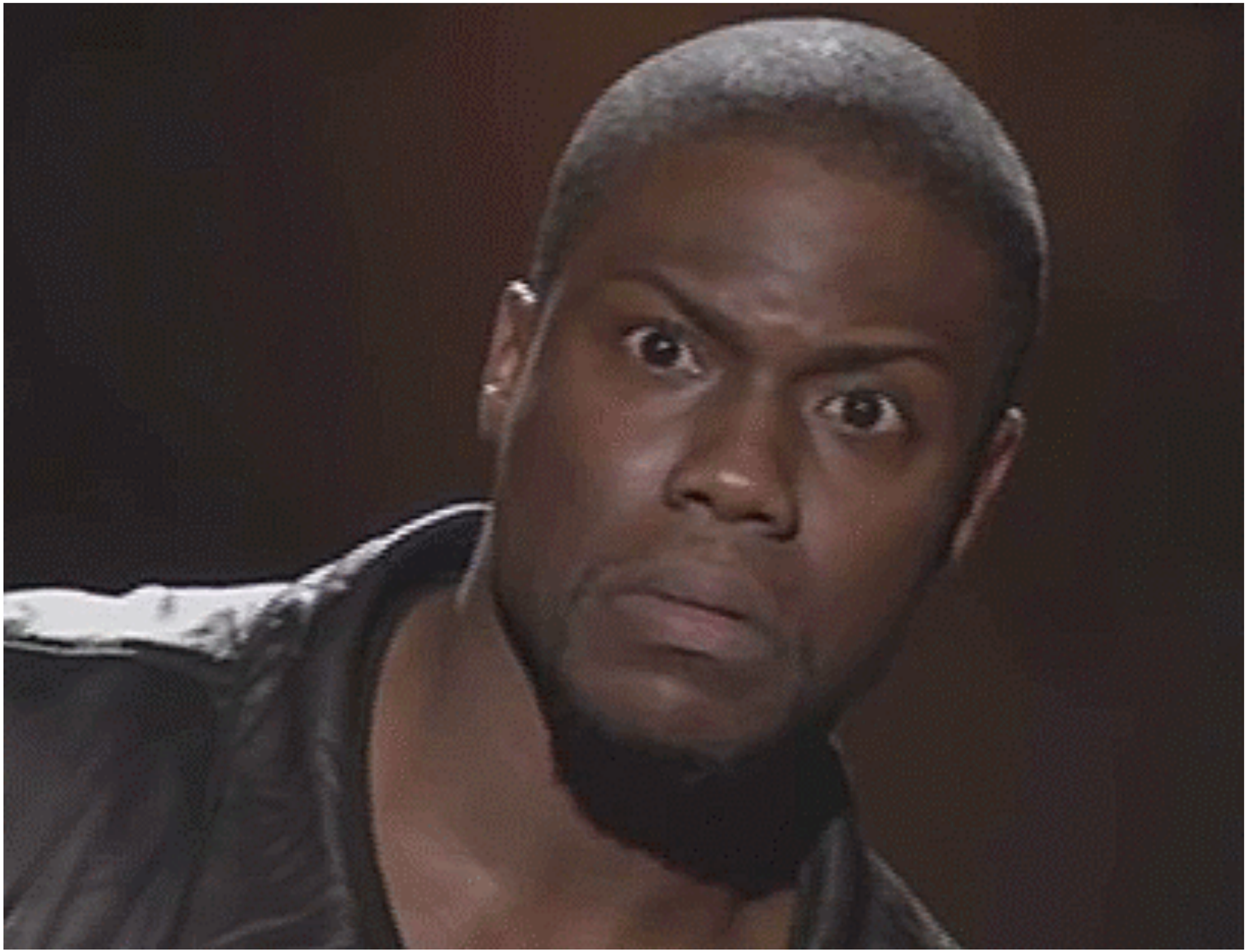

Grafana


fluentd


docker
REGISTRY


Jenkins


Service 1
Pod
App
Service 2
Pod
App
Service 3
Pod
App
Service 4
Pod
App
N+
Kubernetes
Ingress
Controller

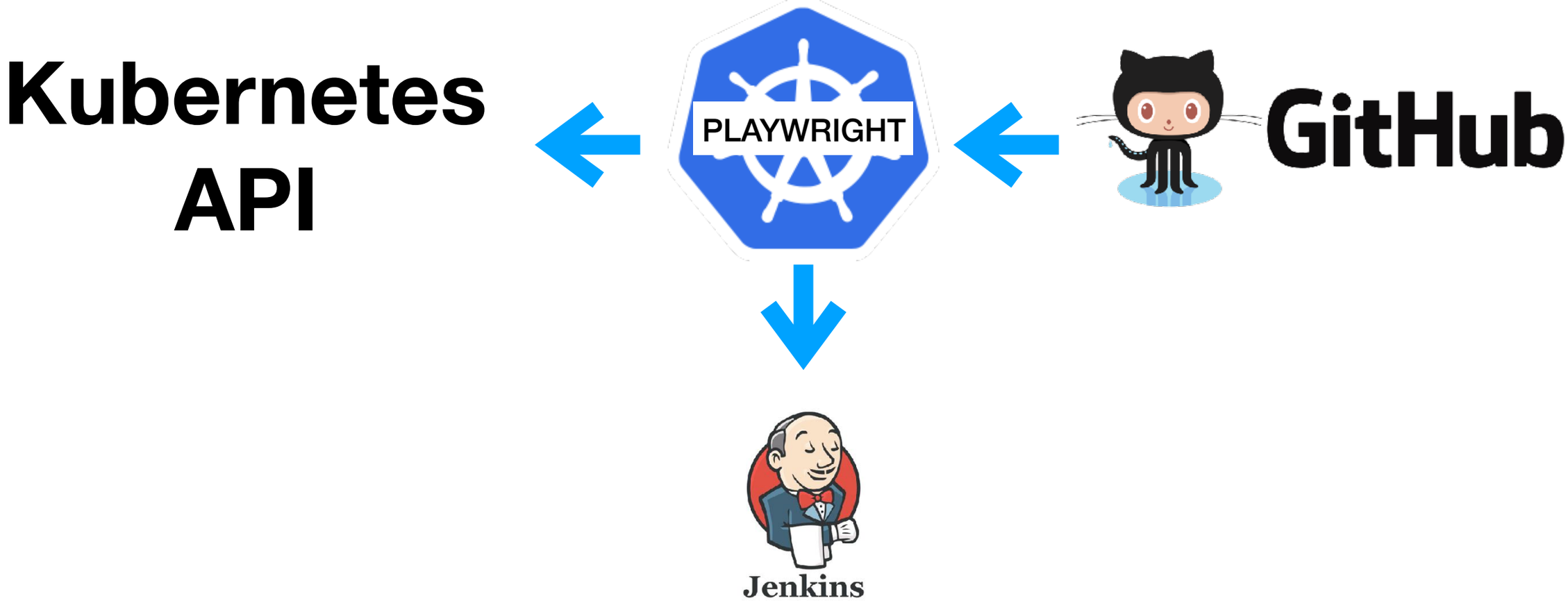


System namespace

```
.
├── 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



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/9a9fc01af8fb5d98b8eec0740716226fadb3735c/contrib/mkimage/debootstrap
```

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

..

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

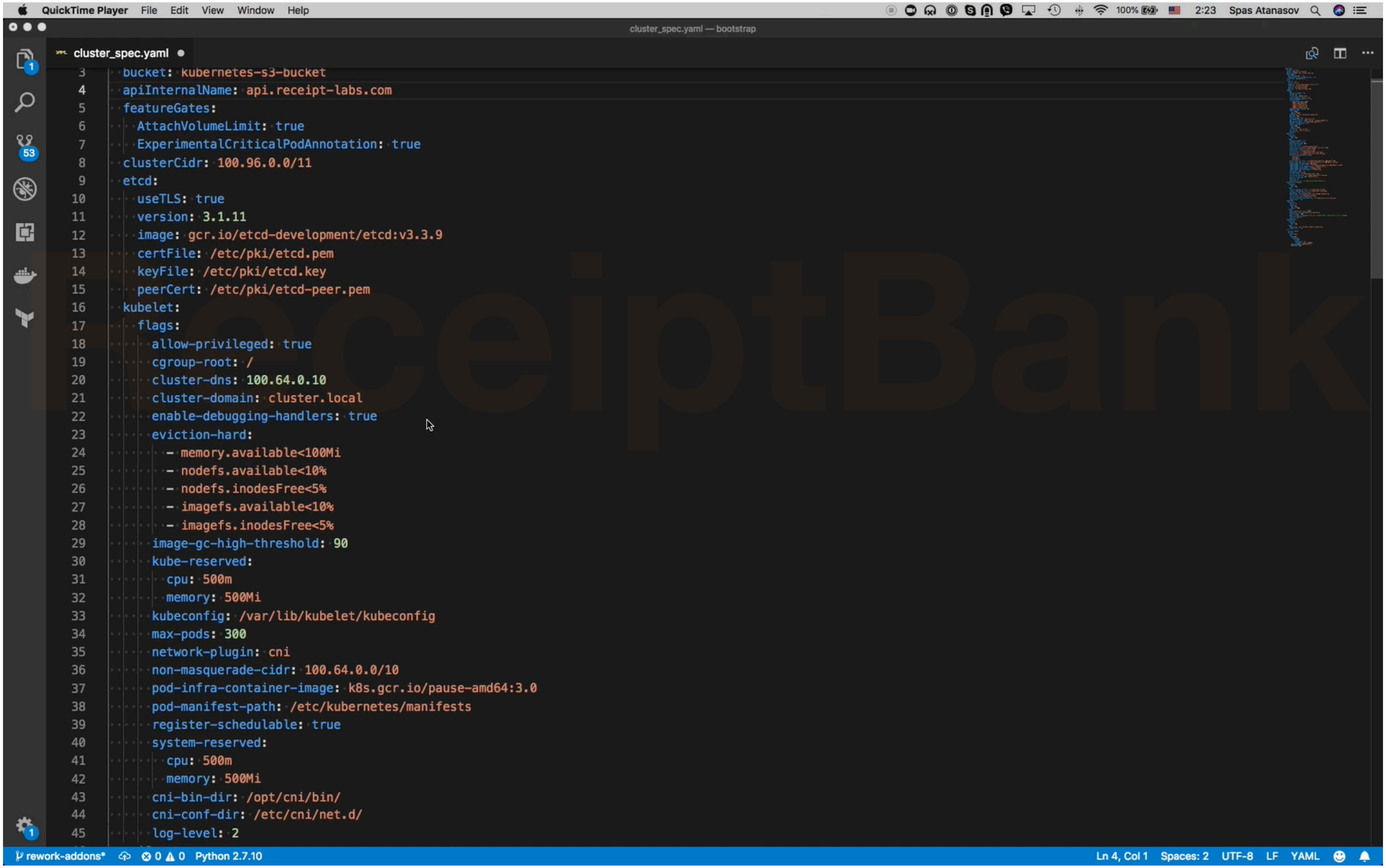
Custom TF Modules

- 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



ReceiptBank

Demo video is not available

Where we intend to go?

Security
by default

clair

anchore

CIS® Center for Internet Security®

aqua

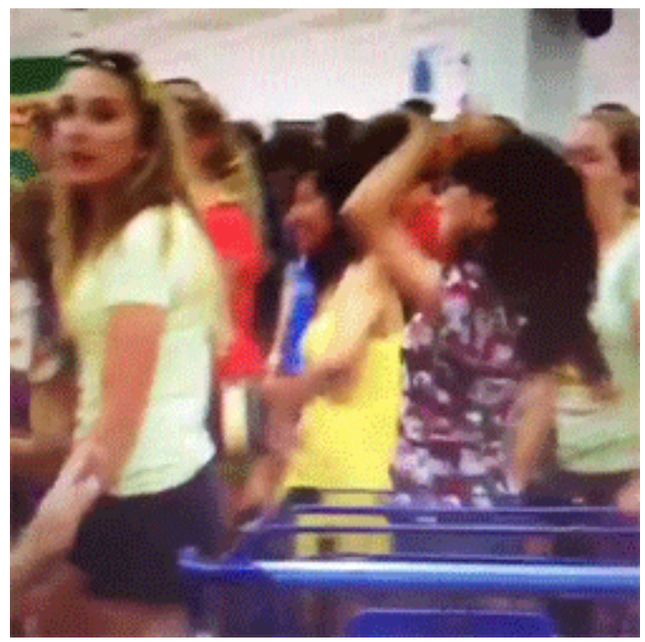
BLACKDUCK

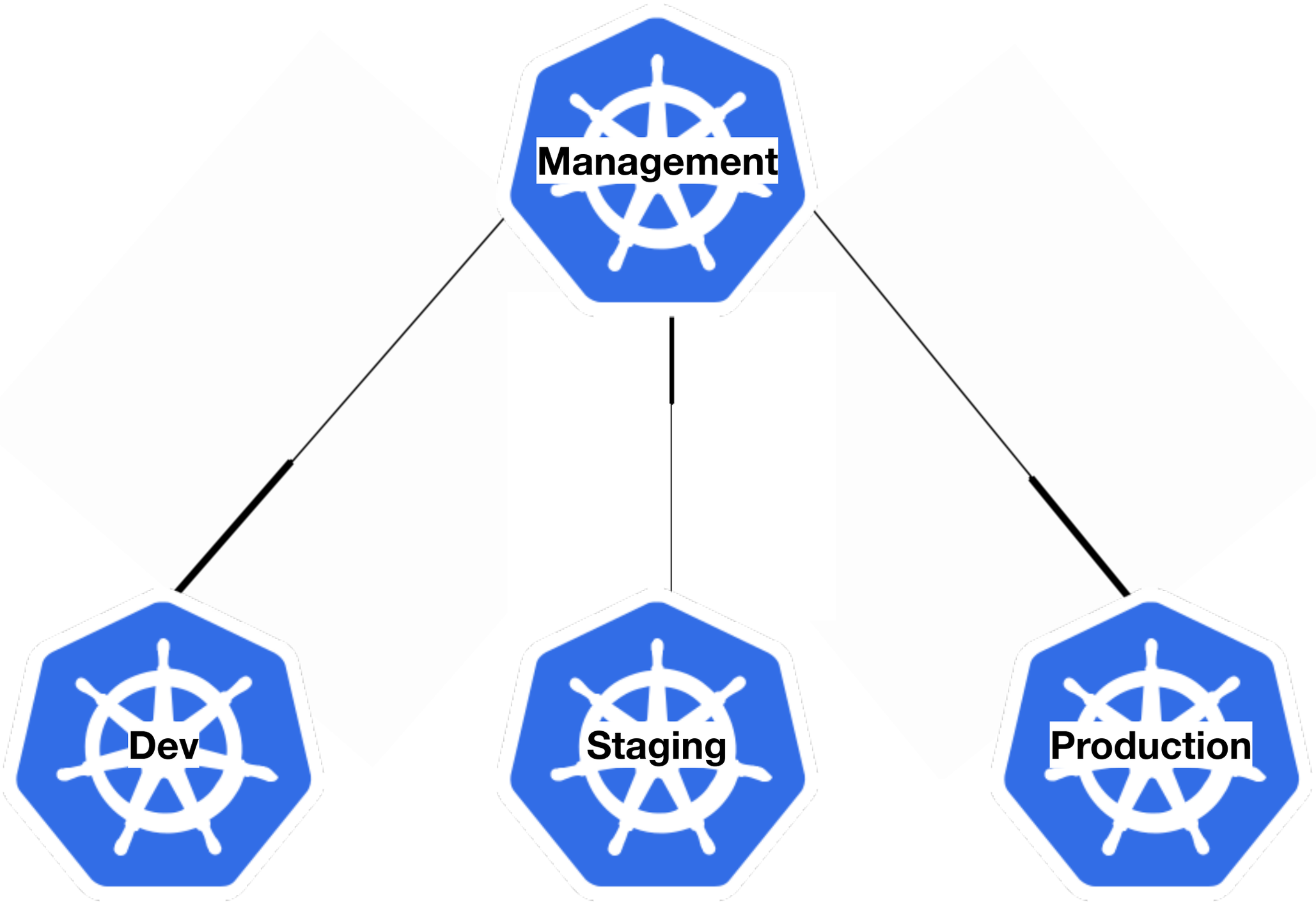
```
cluster_name: Spasterix
network_cidr: 10.10.0.0/16
docker_version: 17.09.0
kubernetes_version: 1.11.2
etcd_version: 3.3.9
cni_plugin: weave
number_of_masters: 5
number_of_workers: 15
....
```

HashiCorp Packer

Terraform

docker




Management
Dev
Staging
Production

#QnA?

spasatanasov