inovex

Put your Kubernetes into Jail

Deployment and Operations on AWS China

Arnold Bechtoldt

Baseline

- K8S for microservice container orchestration (Docker)
- kops for K8S management
- Several existing products/clusters in eu-central-1 (AWS)
- Customer plans to launch **new** product(s) in China

kops

"The easiest way to get a production grade Kubernetes cluster up and running."

All you need is:

- AWS account
- S3 bucket
- kubectl

Creating a Cluster

```
$ kops create cluster \
   --name=kubernetes.example.com \
   --state=s3://kops-state-1234 \
   --zones=eu-central-1a,eu-central-1b,eu-central-1c \
   --node-count=2 \
   --yes
```

Creating a Cluster (2)

```
ubuntu@ubuntu-xenial:~/advanced-kubernetes-course/logging$ AWS_PROFILE=ward kops update cluster kubernetes.newtech.academy --yes --state=s3://kops-state-b429b
I0920 13:54:56.620226   16271 executor.go:91] Tasks: 0 done / 63 total; 34 can run
I0920 13:54:56.738120   16271 logging_retryer.go:59] Retryable error (RequestError: send request failed
caused by: Post https://ec2.eu-west-1.amazonaws.com/: EOF) from ec2/DescribeVpcs - will retry after delay of 33ms
I0920 13:55:00.176006   16271 vfs_castore.go:422] Issuing new certificate: "kubelet"
I0920 13:55:00.198325   16271 vfs_castore.go:422] Issuing new certificate: "kubecfg"
I0920 13:55:00.202337   16271 vfs_castore.go:422] Issuing new certificate: "kube-proxy"
I0920 13:55:00.505182   16271 vfs_castore.go:422] Issuing new certificate: "master"
I0920 13:55:00.652841   16271 vfs_castore.go:422] Issuing new certificate: "kube-scheduler"
I0920 13:55:00.711248   16271 vfs_castore.go:422] Issuing new certificate: "kops"
I0920 13:55:00.713381   16271 vfs_castore.go:422] Issuing new certificate: "kube-controller-manager"
I0920 13:55:08.017307   16271 executor.go:91] Tasks: 34 done / 63 total; 12 can run
I0920 13:55:12.657784   16271 executor.go:91] Tasks: 46 done / 63 total; 15 can run
I0920 13:55:18.048776   16271 launchconfiguration.go:327] waiting for IAM instance profile "nodes.kubernetes.newtech.academy" to be ready
I0920 13:55:18.152871   16271 launchconfiguration.go:327] waiting for IAM instance profile "masters.kubernetes.newtech.academy" to be ready
I0920 13:55:31.324089   16271 executor.go:91] Tasks: 61 done / 63 total; 2 can run
I0920 13:55:32.860606   16271 executor.go:91] Tasks: 63 done / 63 total; 0 can run
I0920 13:55:32.860695   16271 dns.go:152] Pre-creating DNS records
I0920 13:55:38.500652   16271 update_cluster.go:247] Exporting kubecfg for cluster
Kops has set your kubectl context to kubernetes.newtech.academy

Cluster is starting.  It should be ready in a few minutes.
```

Creating a Cluster (3)

Hooray, IT WORKS!

And China?

Easy!

sed s/eu-central-1/cn-north-1/g kops.yaml

Right?

Issue #1

AWS CN is operated by **third parties**.

-

Some *specialists* don’t even speak English.

Issue #2

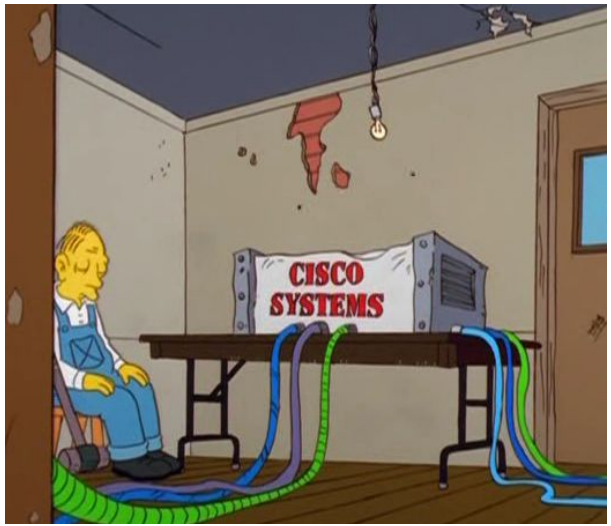
Resource names are different:

arn:aws-cn:s3:::mybucket

Issue #3

The damn great (fire)wall.™

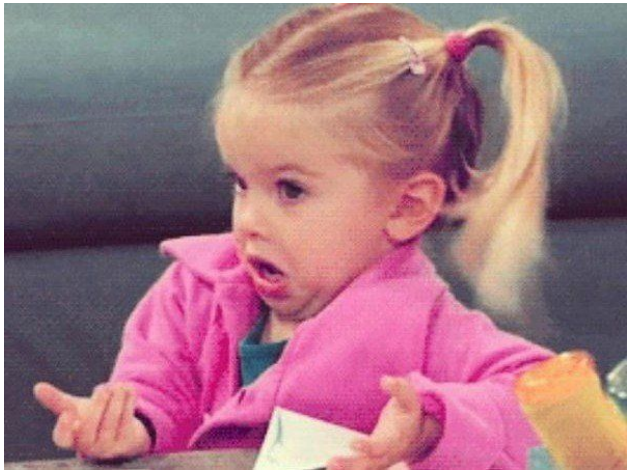
Low bandwidth - Unstable connections - Excessive Censorship

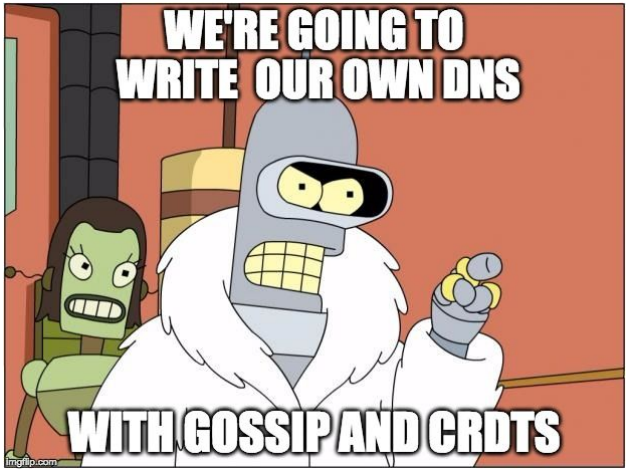


Issue #4

Route53 not available!

Global DNS potentially blocked.





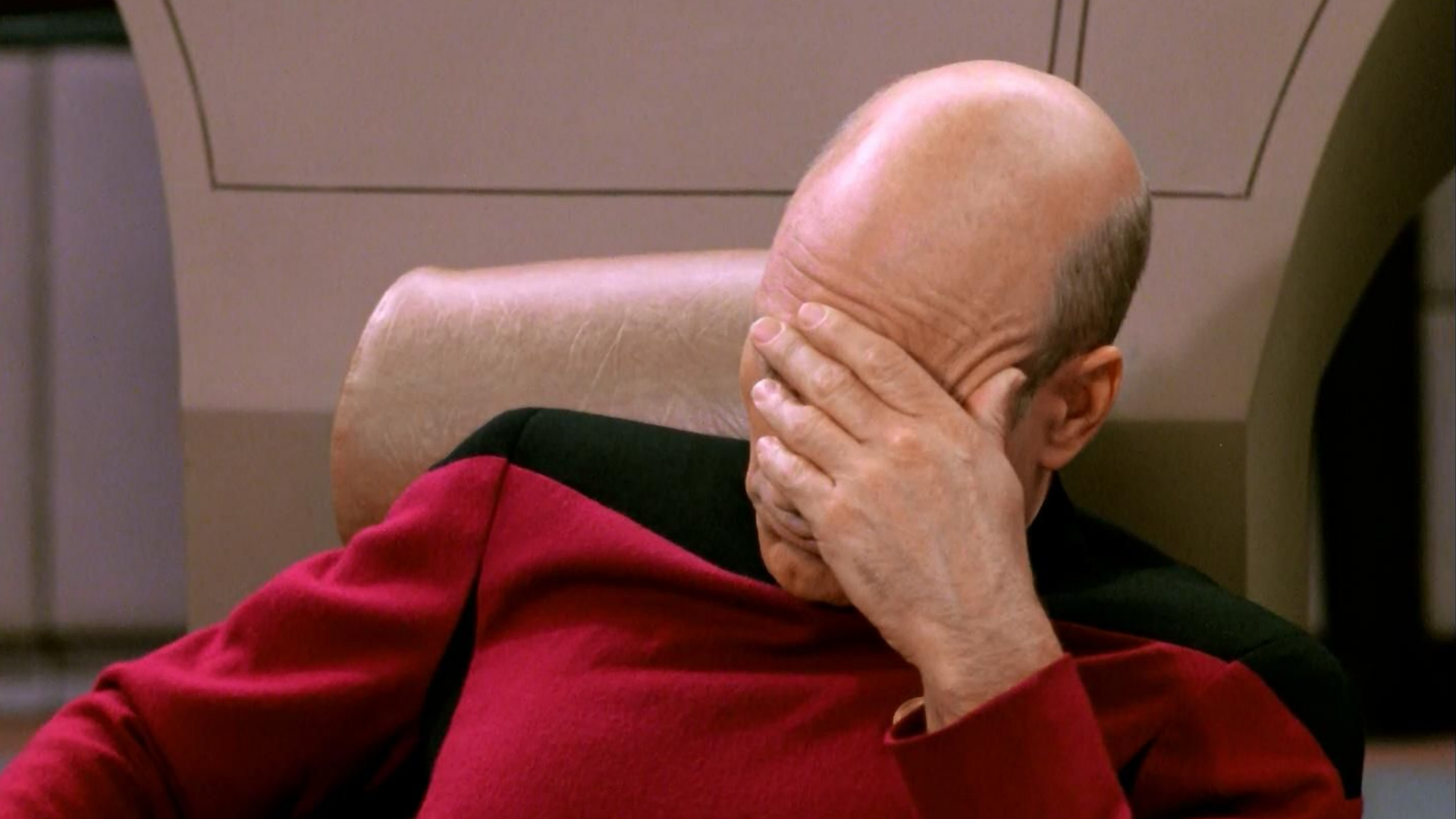
But Wait, There's More...

Issue #5: ICP License

"Internet Content Provider" License

- Ports 80 and 443 are closed by default
- No web service without ICP license
- Bound to a domain
- Explicit IP address whitelist

https://www.zenlayer.com/internet-content-provider-license/



Solving “Mission Impossible”

1. Third Parties: Local employees for translation & contracting
2. Different ARN: Enable kops’ terraform output and customize code
3. Bad internet connection: Local asset cloning/mirroring
4. No route53: Local DNS services providers, kops’ Gossip DNS for discovery
5. ICP: Local employees for translation & contracting, Ingress via NodePort (TF-managed)

kops

“The easiest way to get a production grade Kubernetes cluster up and running.”

kops

~~"The easiest way to get a production grade Kubernetes cluster up and running."~~

Conclusion

kops will hurt in *complex* environments!

Long Term Solution

Custom Kubernetes Deployment Tooling!*

*NIH syndrom might apply here.

Arnold Bechtoldt

inovex

arnold.bechtoldt@inovex.de

sayat.me/arbe

---

github.com/bechtoldt

youtube.com/inovexGmbH

arbe.io

inovex.de

inovex.de/blog

kops.yaml

```yaml
apiVersion: kops/v1alpha2
kind: Cluster
metadata:
  creationTimestamp: 2017-05-04T23:21:47Z
  name: k8s.example.com
spec:
  api:
    loadBalancer:
      type: Public
  authorization:
    alwaysAllow: {}
  channel: stable
  cloudProvider: aws
  configBase: s3://example-state-store/k8s.example.com
  etcdClusters:
  - etcdMembers:
    - instanceGroup: master-us-east-2d
      name: a
    - instanceGroup: master-us-east-2b
      name: b
    - instanceGroup: master-us-east-2c
      name: c
    name: main
  - etcdMembers:
    - instanceGroup: master-us-east-2d
      name: a
    - instanceGroup: master-us-east-2b
      name: b
    - instanceGroup: master-us-east-2c
      name: c
    name: events
  kubernetesApiAccess:
  - 0.0.0.0/0
  kubernetesVersion: 1.6.6
  masterPublicName: api.k8s.example.com
  networkCIDR: 172.20.0.0/16
  networkID: vpc-6335dd1a
```

```yaml
  networking:
    weave: {}
  nonMasqueradeCIDR: 100.64.0.0/10
  sshAccess:
  - 0.0.0.0/0
  subnets:
  - cidr: 172.20.32.0/19
    name: us-east-2d
    type: Private
    zone: us-east-2d
  - cidr: 172.20.64.0/19
    name: us-east-2b
    type: Private
    zone: us-east-2b
  - cidr: 172.20.96.0/19
    name: us-east-2c
    type: Private
    zone: us-east-2c
  - cidr: 172.20.0.0/22
    name: utility-us-east-2d
    type: Utility
    zone: us-east-2d
  - cidr: 172.20.4.0/22
    name: utility-us-east-2b
    type: Utility
    zone: us-east-2b
  - cidr: 172.20.8.0/22
    name: utility-us-east-2c
    type: Utility
    zone: us-east-2c
  topology:
    bastion:
      bastionPublicName: bastion.k8s.example.com
    dns:
      type: Public
    masters: private
    nodes: private
```