Google Cloud

NEXT

Next'19 Extended Hanoi – VietKubers Meetup #2

Kubernetes:
Upgrading your cluster without downtime practice

Nguyen Hai Truong <truongnh@vn.fujitsu.com>

Nguyen Phuong An <annp@vn.fujitsu.com>

Agenda

1. whoarewe
2. Deploying multi-master nodes (High Availability) K8s cluster
3. Running a replicated stateful application
4. Upgrading kubeadm HA cluster
5. Load-testing client fortio

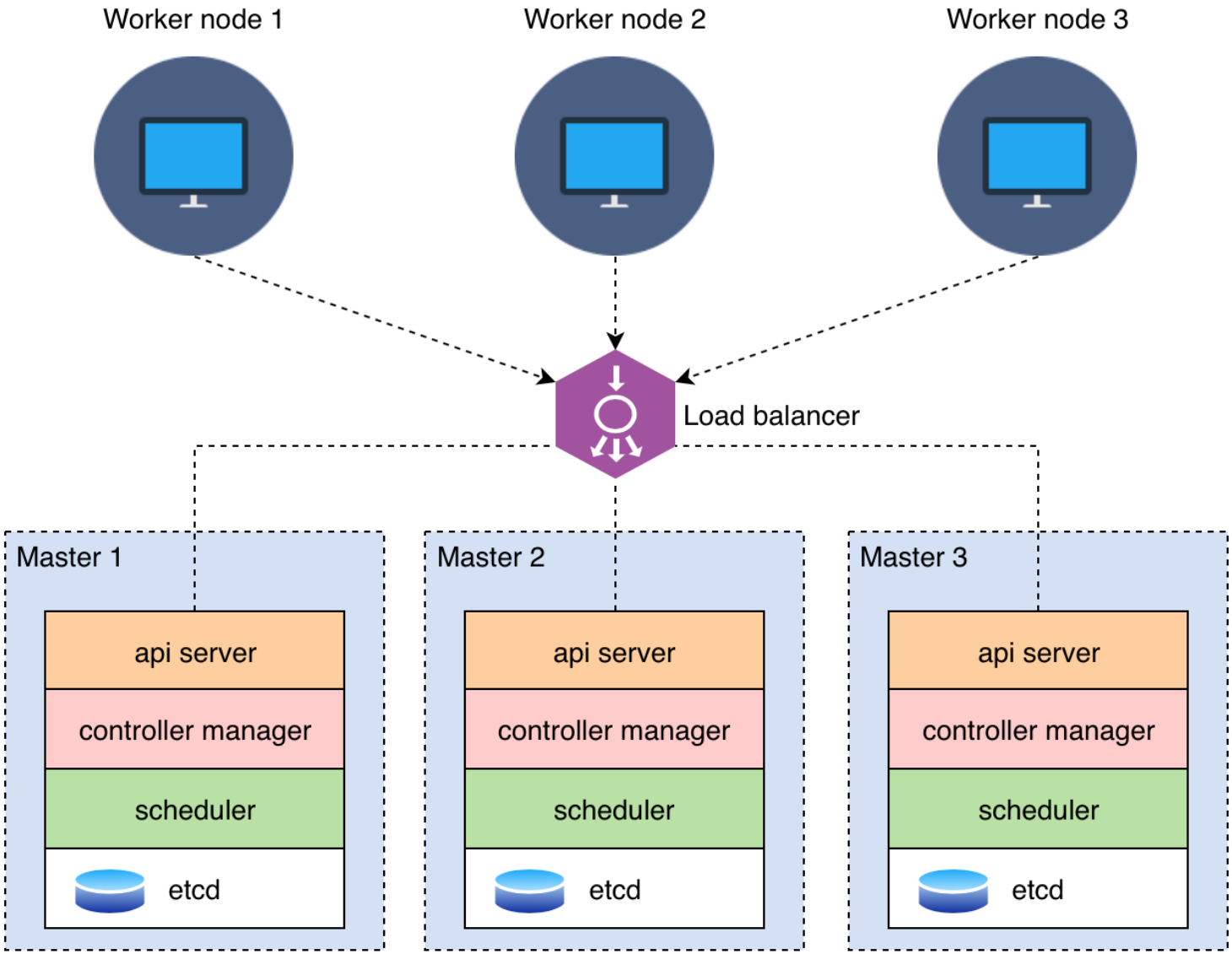
whoarewe

- Software engineers
- Upstream contribute to OpenStack Networking
- Now, we’re moving to K8s and CNCF
- Organizers of VietKubers

@annp87 @truongnh92

annp1987 truongnh1992

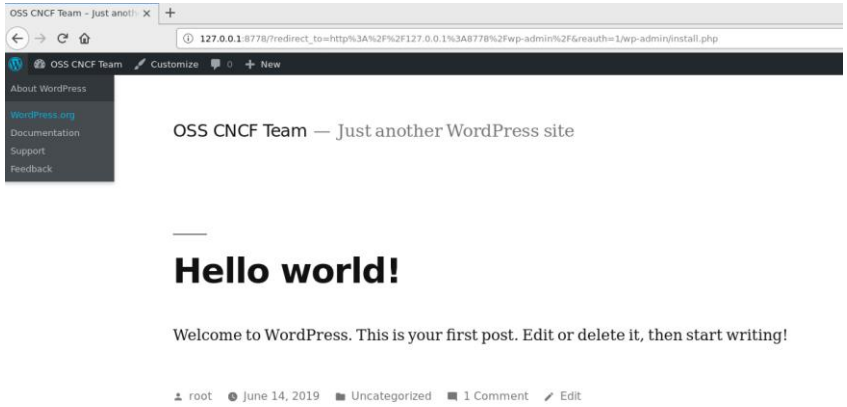
Deploying multi-master nodes (High Availability) K8s cluster



Running a replicated stateful application

Deploying a highly available WordPress application in Kubernetes

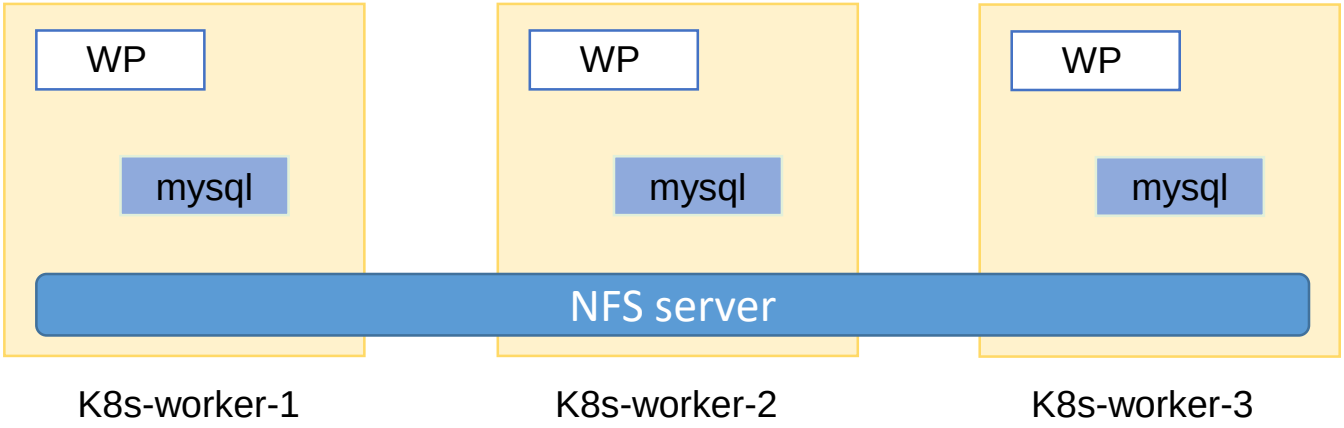
```
NAME                                READY   STATUS    RESTARTS   AGE   IP           NODE          NOMINATED NODE   READINESS GATES
pod/fortio-deploy-cd48fb5db-j865f   1/1     Running   1          8d    10.40.0.6    k8s-worker3   <none>           <none>
pod/wordpress-mpv49                 1/1     Running   0          13h   10.47.0.3    k8s-worker1   <none>           <none>
pod/wordpress-mysql-0               1/1     Running   0          14h   10.39.0.2    k8s-worker2   <none>           <none>
pod/wordpress-mysql-1               1/1     Running   0          14h   10.47.0.2    k8s-worker1   <none>           <none>
pod/wordpress-mysql-2               1/1     Running   0          14h   10.40.0.8    k8s-worker3   <none>           <none>
pod/wordpress-t867z                 1/1     Running   0          13h   10.40.0.9    k8s-worker3   <none>           <none>
pod/wordpress-znn2d                 1/1     Running   0          13h   10.39.0.3    k8s-worker2   <none>           <none>

NAME                      TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE     SELECTOR
service/fortio            ClusterIP   10.108.5.175    <none>        8080/TCP         2d21h   app=fortio
service/kubernetes        ClusterIP   10.96.0.1       <none>        443/TCP          23d     <none>
service/wordpress         NodePort    10.98.81.179    <none>        80:31530/TCP     13h     name=wordpress
service/wordpress-mysql   ClusterIP   None            <none>        3306/TCP         14h     app=wordpress,tier=mysql
```



Follow this guide to deploy: https://vietkubers.github.io/2019-06-17-deploying-stateful-wordpress.html

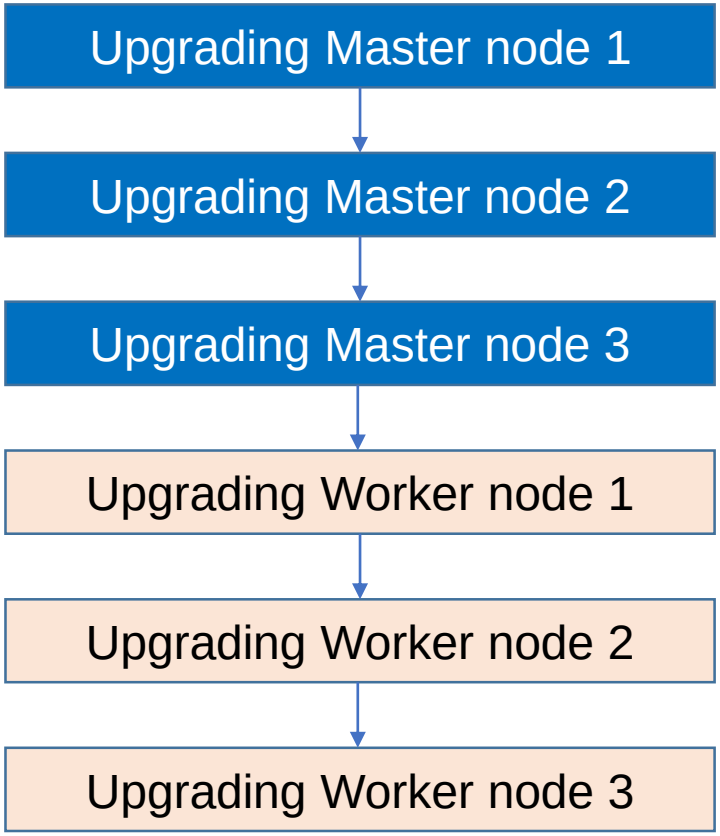
Running a replicated stateful application (2/2)

- WordPress is a stateful application relies on 2 persistence back ends
  - Persistent volume storage
  - MySQL database
- Installing and configuring NFS server



- Deploying MySQL
- Deploying WordPress

Upgrading kubeadm HA cluster (1/2)



The detail steps could be reached out:
https://vietkubers.github.io/2019-05-24-upgrading-ha-k8s-cluster.html

Upgrading kubeadm HA cluster (2/2)

```
master1@k8s-master1:~$ sudo kubectl get node -o wide
NAME          STATUS   ROLES    AGE     VERSION   INTERNAL-IP      EXTERNAL-IP
k8s-master1   Ready    master   35m     v1.13.0   10.164.178.161   <none>
k8s-master2   Ready    master   27m     v1.13.0   10.164.178.162   <none>
k8s-master3   Ready    master   25m     v1.13.0   10.164.178.163   <none>
k8s-worker1   Ready    <none>   13m     v1.13.0   10.164.178.233   <none>
k8s-worker2   Ready    <none>   8m24s   v1.13.0   10.164.178.234   <none>
k8s-worker3   Ready    <none>   2m40s   v1.13.0   10.164.178.235   <none>
```

```
master1@k8s-master1:~$ sudo kubectl get node
NAME          STATUS   ROLES    AGE   VERSION
k8s-master1   Ready    master   23d   v1.14.0
k8s-master2   Ready    master   23d   v1.14.0
k8s-master3   Ready    master   23d   v1.14.0
k8s-worker1   Ready    <none>   16h   v1.14.0
k8s-worker2   Ready    <none>   23d   v1.14.0
k8s-worker3   Ready    <none>   23d   v1.14.0
```

Load-testing client fortio

Fortio lets you control the number of connections, concurrency and delays for outgoing HTTP calls.

https://github.com/fortio/fortio

fortio load [-json -] -t [duration] http://your.app/

```
Sockets used: 4 (for perfect keepalive, would be 4)
Jitter: false
Code 200 : 40 (100.0 %)
Response Header Sizes : count 40 avg 75 +/- 0 min 75 max 75 sum 3000
Response Body/Total Sizes : count 40 avg 75 +/- 0 min 75 max 75 sum 3000
All done 40 calls (plus 4 warmup) 0.378 ms avg, 8.0 qps
```

Thank you