6 MAY 2020

Tanzu vSphere 7 with Kubernetes on NSX-T 3.0 VDS Install Part 5: Testing, Demo Apps

In this section, we will perform some testing on the TKG clusters.

Step 1 – Login to the TKG Cluster and switch to the tkg cluster context.

```
kubectl vsphere login server 10.30.10.1 vsphere-username admi
```

```
root@ubuntu:~# kubectl vsphere login --server 10.30.10.1 --vsphere-username administrator@vs
phere.local --insecure-skip-tls-verify --tanzu-kubernetes-cluster-name tkg-cluster --tanzu-k
ubernetes-cluster-namespace demo-app-01

Password:
Logged in successfully.

You have access to the following contexts:
```

```
kubectl config use-context tkg-cluster

kubectl get nodes
```

```
root@ubuntu:~# kubectl config use-context tkg-cluster
Switched to context "tkg-cluster".
root@ubuntu:~# kubectl get nodes
NAME                                        STATUS   ROLES    AGE   VERSION
tkg-cluster-control-plane-fh7rm             Ready    master   17d   v1.16.8+vmware.1
tkg-cluster-workers-lknmk-8777c64f7-69bbf   Ready    <none>   17d   v1.16.8+vmware.1
tkg-cluster-workers-lknmk-8777c64f7-zgzsw   Ready    <none>   17d   v1.16.8+vmware.1
```

Step 2 – Apply ClusterRole Security Policies.

```
kubectl apply -f https://raw.githubusercontent.com/dstamen/Kul
```

```
root@ubuntu:~# kubectl apply -f https://raw.githubusercontent.com/dstamen/Kubernetes/master/
demo-applications/allow-runasnonroot-clusterrole.yaml
clusterrole.rbac.authorization.k8s.io/psp:privileged unchanged
clusterrolebinding.rbac.authorization.k8s.io/all:psp:privileged unchanged
```

I used the hipster shop demo.

```
kubectl create ns hipster

kubectl apply -f https://raw.githubusercontent.com/dstamen/Kul
```

```
root@ubuntu:~# kubectl create ns hipster
namespace/hipster created
root@ubuntu:~# kubectl apply -f https://raw.githubusercontent.com/dstamen/Kubernetes/master/
demo-applications/demo-hipstershop.yaml -n hipster
deployment.apps/emailservice created
service/emailservice created
deployment.apps/checkoutservice created
service/checkoutservice created
deployment.apps/recommendationservice created
service/recommendationservice created
deployment.apps/frontend created
service/frontend created
service/frontend-external created
deployment.apps/paymentservice created
service/paymentservice created
deployment.apps/productcatalogservice created
service/productcatalogservice created
deployment.apps/cartservice created
service/cartservice created
deployment.apps/loadgenerator created
deployment.apps/currencyservice created
service/currencyservice created
deployment.apps/shippingservice created
service/shippingservice created
deployment.apps/redis-cart created
service/redis-cart created
deployment.apps/adservice created
service/adservice created
```

Step 4 – Access the application.

Find out the virtual IP which is being load-balanced by NSX-T LB being assigned.

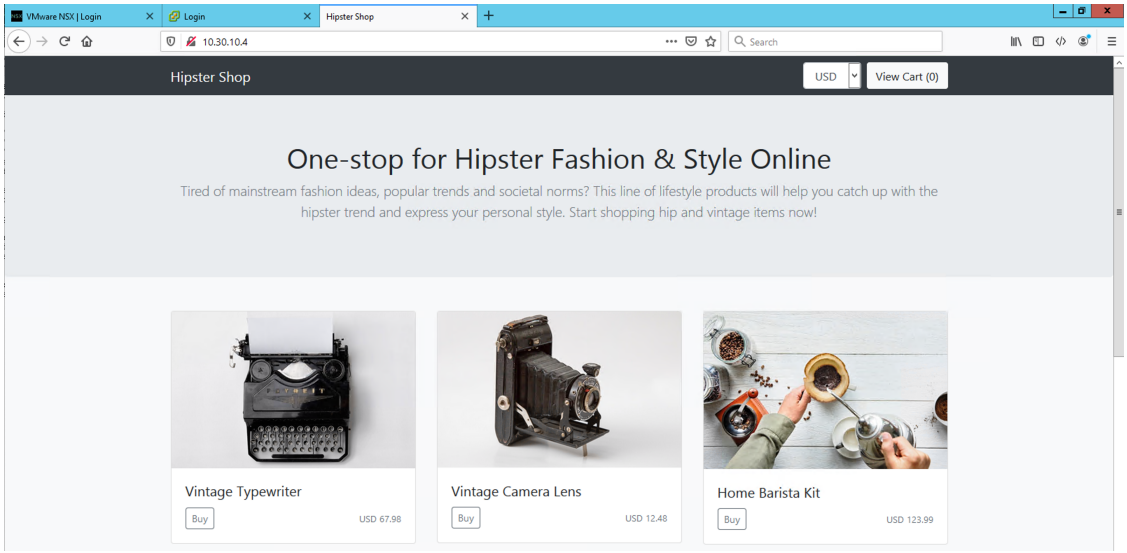
```
kubectl get pod -n hipster

kubectl get svc -n hipster
```

```
root@ubuntu:~# kubectl get pod -n hipster
NAME                                     READY   STATUS    RESTARTS   AGE
adservice-746fffd4db-zdxlt               1/1     Running   0          7m44s
cartservice-6b8c4fc7bc-l6skx             1/1     Running   0          7m45s
checkoutservice-766b67945b-g7brm         1/1     Running   0          7m45s
currencyservice-789cc856bd-v4t8h         1/1     Running   0          7m44s
emailservice-5887cb476b-zch78            1/1     Running   0          7m45s
frontend-745ff8f75d-dh4mj                1/1     Running   0          7m45s
loadgenerator-665744bfcb-bx7xn           1/1     Running   0          7m45s
paymentservice-68bb6d779d-lrjgw          1/1     Running   0          7m45s
productcatalogservice-7596dc684b-vf74m   1/1     Running   0          7m45s
recommendationservice-5d6f9b896b-cnk2c   1/1     Running   0          7m45s
redis-cart-55d9fc49fb-6znqj              1/1     Running   0          7m44s
shippingservice-844874bc76-9qzjv         1/1     Running   0          7m44s
root@ubuntu:~# kubectl get svc -n hipster
NAME                    TYPE           CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
adservice               ClusterIP      198.58.30.11     <none>        9555/TCP       7m57s
cartservice             ClusterIP      198.58.85.67     <none>        7070/TCP       7m58s
checkoutservice         ClusterIP      198.51.203.111   <none>        5050/TCP       7m58s
currencyservice         ClusterIP      198.57.124.89    <none>        7000/TCP       7m57s
emailservice            ClusterIP      198.49.138.24    <none>        5000/TCP       7m58s
frontend                ClusterIP      198.50.38.166    <none>        80/TCP         7m58s
frontend-external       LoadBalancer   198.55.129.211   10.30.10.4    80:31949/TCP   7m58s
paymentservice          ClusterIP      198.48.94.51     <none>        50051/TCP      7m58s
productcatalogservice   ClusterIP      198.48.193.88    <none>        3550/TCP       7m58s
recommendationservice   ClusterIP      198.53.94.55     <none>        8080/TCP       7m58s
redis-cart              ClusterIP      198.60.61.101    <none>        6379/TCP       7m57s
shippingservice         ClusterIP      198.49.17.222    <none>        50051/TCP      7m57s
```

Viola! Now you can try to access the application!

```
http://10.30.10.4
```



That’s it! Hopefully you find this vSphere with Kubernetes blog series

Share this

Tanzu vSphere 7 with Kubernetes on NSX-T 3.0 VDS Install

Part 1: Overview, Design, Network Topology, Hardware Used

Part 2: ESXi, vCenter, VDS Config and NSX-T Manager

Part 3: NSX-T Edges, Segments, Tier-0 Routing

Part 4: Supervisor Cluster, Content Library, TKG Clusters

Part 5: Testing, Demo Apps

Vincent Han
Your description here

— Vincent Han Blog —

NSX-T 2.1 Installation using ovftool (GA ver)

Kubernetes with NSX-T Container Plugin Demo List

How to install AMKO on Tanzu Kubernetes Clusters(TKC) / TKG-Service

How to install AMKO on Tanzu Kubernetes Clusters(TKC) / TKG-Service ***Disclaimer: If you are using TKG-Service with NSX-T network with AMKO, this solution is not supported by VMware at the moment. It would be good for POC / Testing only. ***Update (May 2021): I have tested this on vSphere 7.0U2, NSX-T 3.1 and AMKO 1.4.1, below steps are still valid. Introduction Well, let me guess. If you stumble on this blog post, I would assume you have seen the goodness and coolness of AKO (AVI Kubernetes Operator). ...

VINCENT HAN

Tanzu vSphere 7 with Kubernetes on NSX-T 3.0 VDS Install Part 4: Supervisor Cluster, Content Library, TKG Clusters

In this section, we will enable Workload Management, creation of Supervisor Cluster, enabling Content Library as well as creation of TKG Clusters also known as Guest Clusters. Step 1 – Create the VM Storage Policies. Login to vCenter if you have not. Menu -> Datastores. Click on the datastore that you like to use for vSphere with Kubernetes. Under Tags, Click on Assign. Click on ADD TAG. ...

VINCENT HAN

Share this