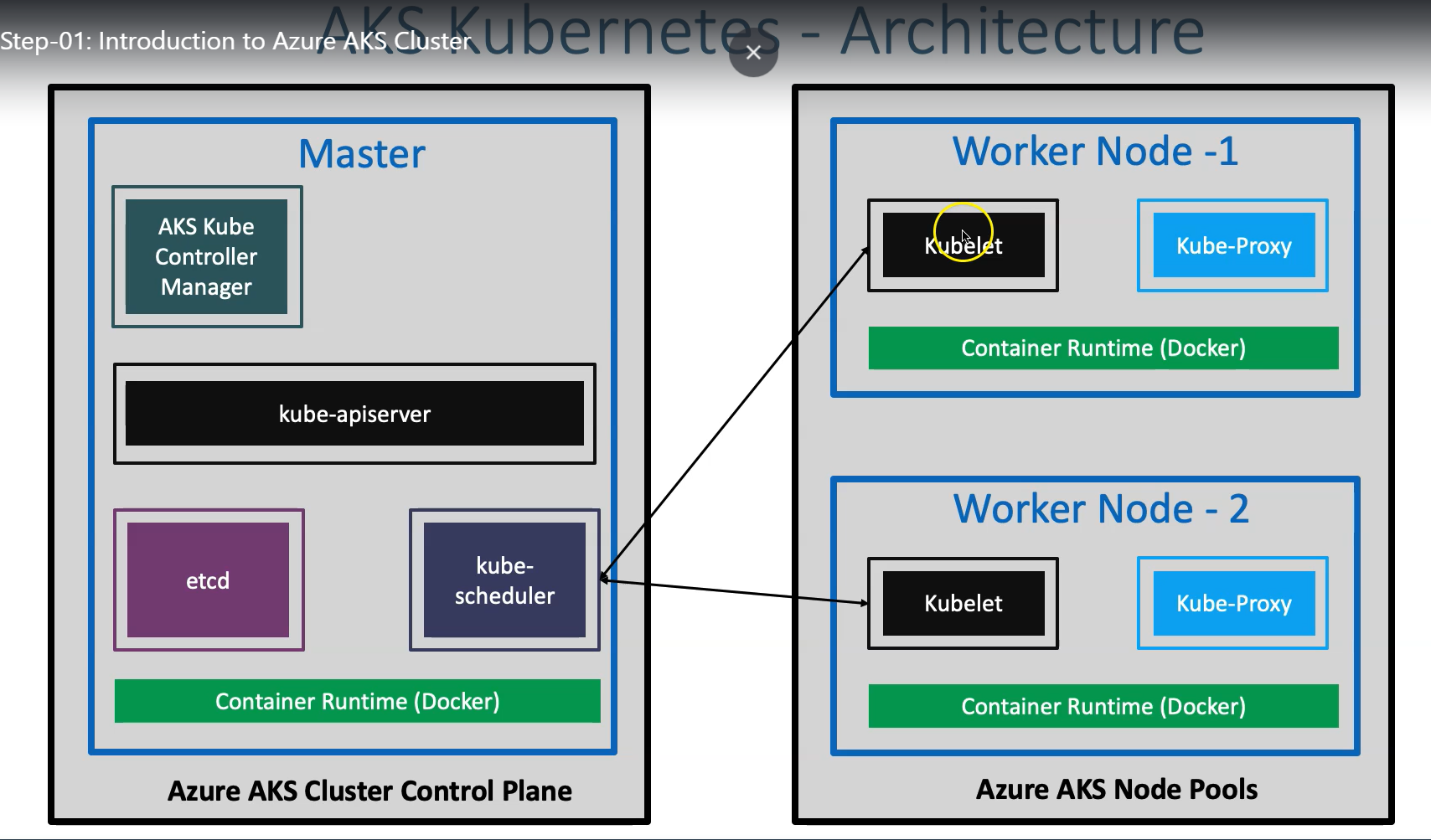
# Thành phần AKS



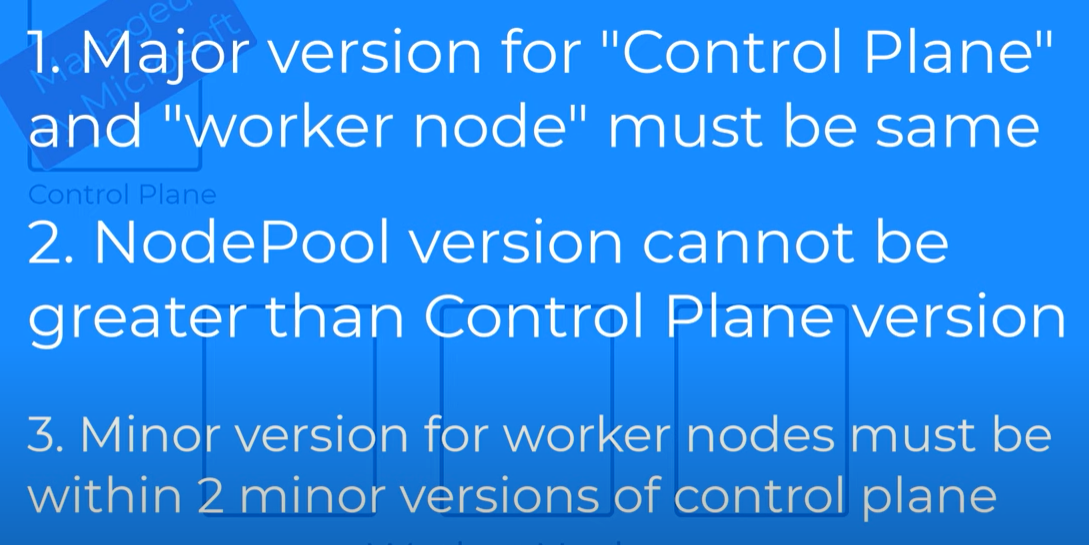
* Kubernetes version upgrade
* NodeImage version upgrade

Kubernetes version upgrade: 1.12.2

1: minor version

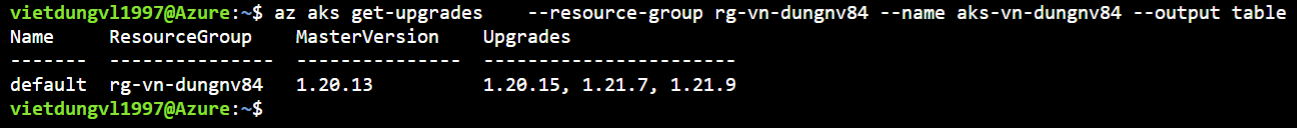
12: major version

2: patch version



* Cluster control plane version

# az aks get-upgrades --resource-group rg-vn-dungnv84 --name aks-vn-dungnv84 --output table

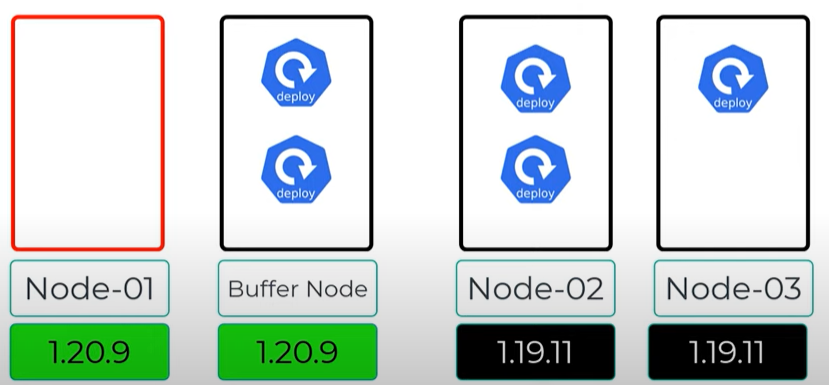


* Node pool véin

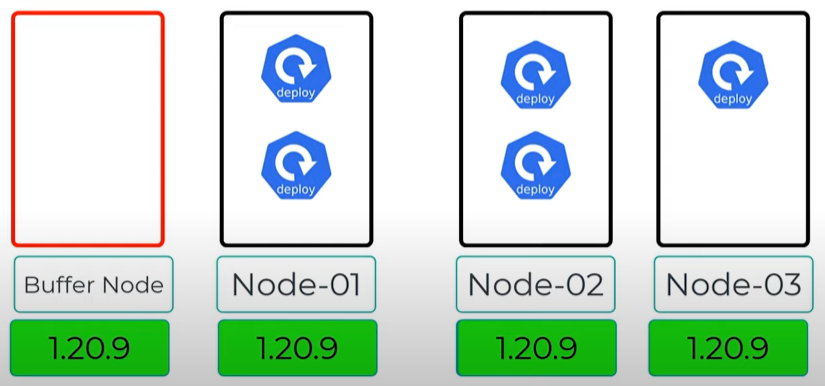
# az aks nodepool list --resource-group rg-vn-dungnv84 --cluster-name aks-vn-dungnv84 --query "[].{Name:name,k8version:orchestratorVersion}" --output table

# Các kiểu upgrade AKS

* Upgrade tất cả cùng 1 lúc
* Chỉ upgrade control plane
* Chỉ chọn nodepool



* Tạo 1 node mới với version mới
* Node cũ sẽ bị cordon và drain và node cũ này sẽ lại được upgrade lên version mới để trở thành node với version mới
* Node 1 thành 1.20.9, node 2 lại đẩy resource sang node 1
* Tương tự đén node 3 cũng thế



Và cuối cùng buffer node bị xóa

az aks get-versions --location

* Upgrade tất cả cùng lúc

az aks upgrade --resource-group <ResourceGroupName> --name <AKSClusterName> --no-wait --kubernetes-version <KubernetesVersion>

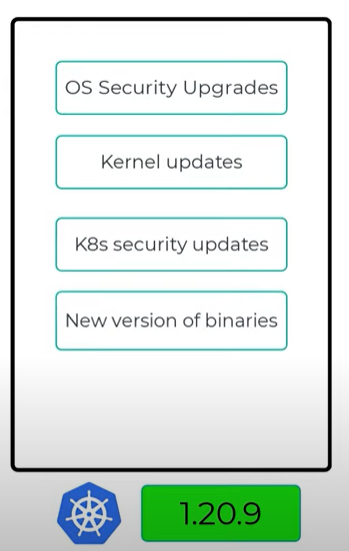
* upgrade control plane (master của K8s)

az aks upgrade --resource-group <ResourceGroupName> --name <AKSClusterName> --control-plane-only --no-wait --kubernetes-version <KubernetesVersion>

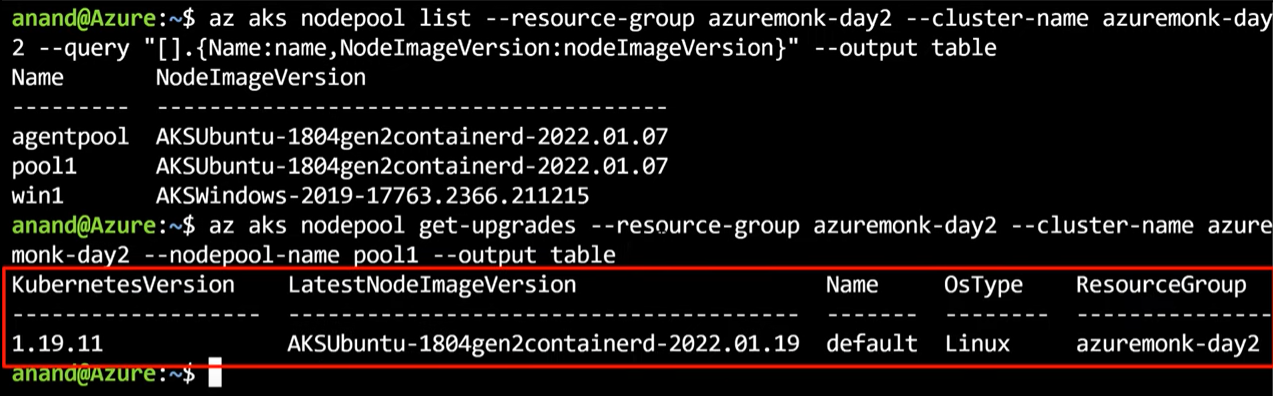
* upgrade cluster trên 1 nodepool

az aks nodepool upgrade --resource-group <ResourceGroupName> --cluster-name <AKSClusterName> --name NodePoolName --no-wait --kubernetes-version <KubernetesVersion>

* upgrade nodeImage version



az aks nodepool list --resource-group ResourceGroupName --cluster-name AKSClusterName --query "[].{Name:name,NodeImageVersion:nodeImageVersion}" --output table



the following command to list the available latest nodeimage version for us

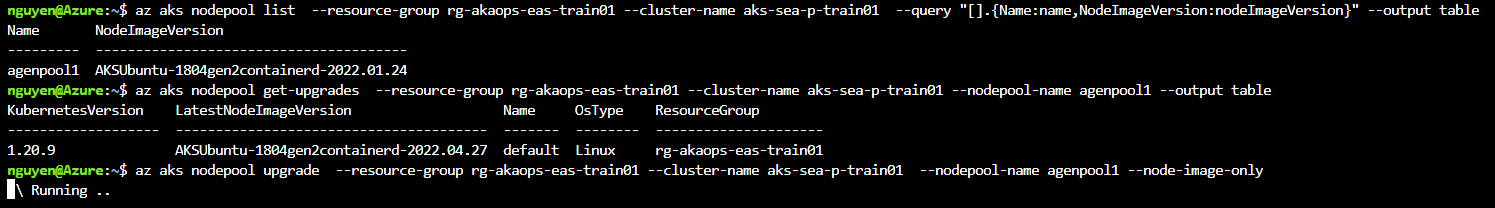
az aks nodepool get-upgrades --resource-group ResourceGroupName --cluster-name AKSClusterName --nodepool-name NodePoolName --output table

Upgrade nodeimage version for specific nodepool

In order to upgrade the nodeimage vefsion for a specific nodepool we run the following command :

az aks nodepool upgrade \ --resource-group myResourceGroup \ --cluster-name myAKSCluster \ --name mynodepool \ --node-image-only

* upgrade nodepool OS



Cái đầu là image OS node cũ, cái sau là image OS mới upgrade lên

az aks nodepool list --resource-group rg-akaops-eas-train01 --cluster-name aks-sea-p-train01 --query "[].{Name:name,NodeImageVersion:nodeImageVersion}" --output table

az aks nodepool get-upgrades --resource-group rg-akaops-eas-train01 --cluster-name aks-sea-p-train01 --nodepool-name agenpool1 --output table

az aks nodepool upgrade --resource-group rg-akaops-eas-train01 --cluster-name aks-sea-p-train01 --nodepool-name agenpool1 --node-image-only

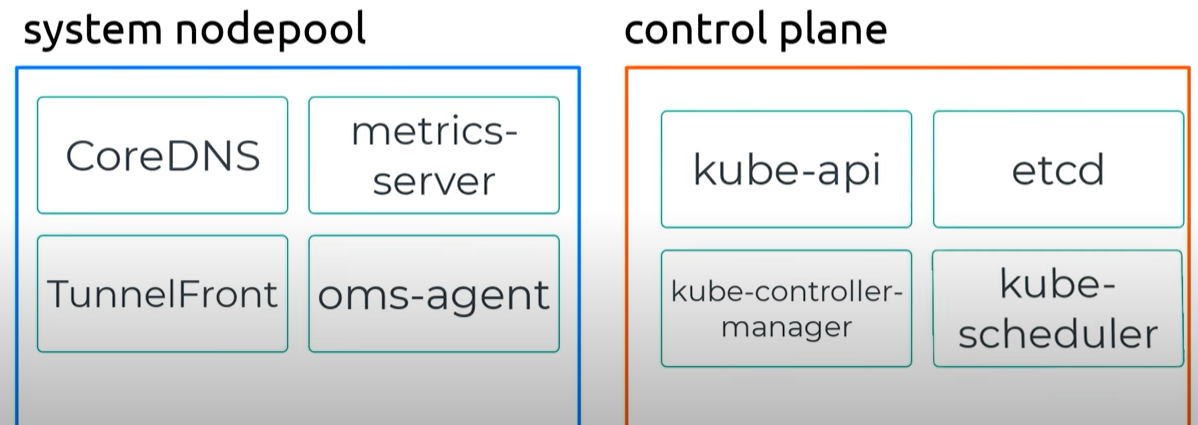
az aks nodepool show \ --resource-group myResourceGroup \ --cluster-name myAKSCluster \ --name mynodepool

## Upgrade nodeimage version for all nodepools at once

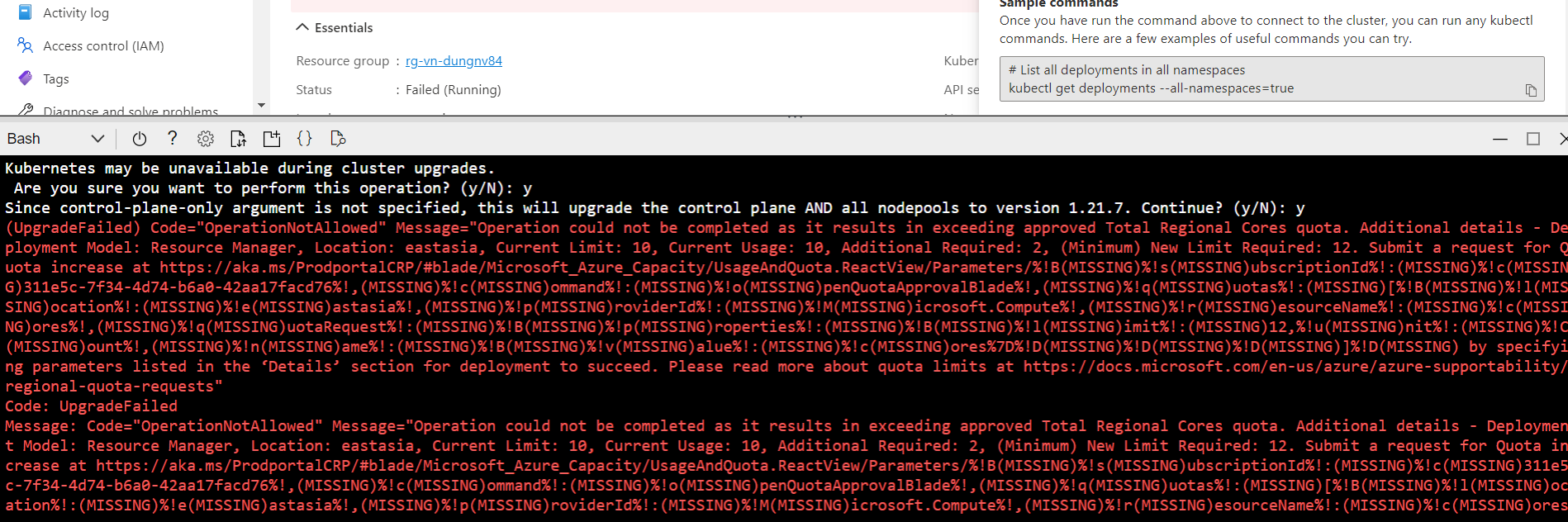
az aks upgrade \ --resource-group myResourceGroup \ --name myAKSCluster \ --node-image-only

Remember If you don’t run the nodeimage switch remember the whole cluster including the kubernetes version of the control plane is also upgraded.

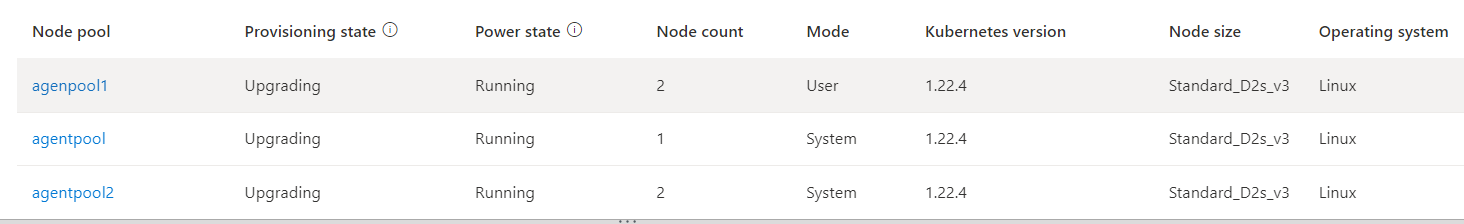
az aks show --resource-group myResourceGroup --name myAKSCluster



# Quy trình upgrade 3 node

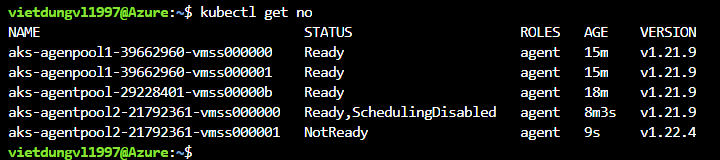


Lỗi hết resource quota trên azure, vì 3 nodepool với sizre khác nhau, cho nên trên mỗi nodepool sẽ tạo ra 1 node, vì vậy 3 node cộng lại sẽ tạo ra resource lớn làm tràn qouta (lỗi này do microsoft ko đủ resource)



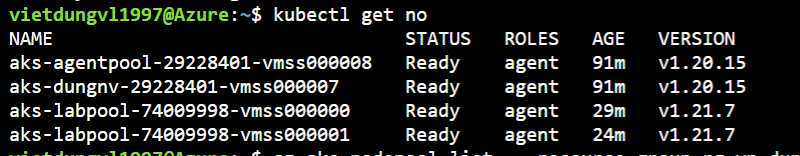
Còn nếu 3 nodepool cùng 1 size thì n sẽ chỉ tạo 1 node trên 1 nodepool, n sẽ tạo lần lượt trên các nodepool thay vì 3 node với 3 nodepool có size khác nhau

Ban đầu số node là 2-1-1, giờ thành 2-1-2

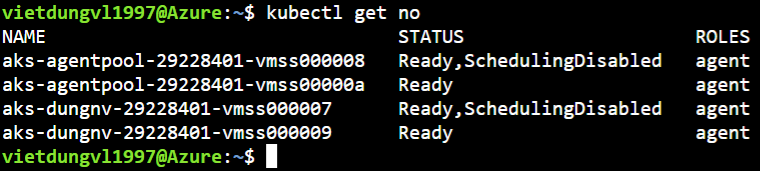


Cùng 1 size, nó sẽ chỉ tạo 1 node trên 1 nodepool

* nodepool mà có 3 size tương tự nhau thì có thể upgrade thẳng



Và bgio thằng nên thằng ko lên



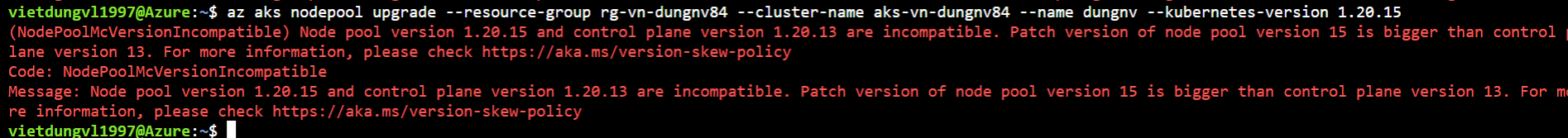
Upgrade đồng thời n sẽ tạo ra mỗi nodepool 1 node

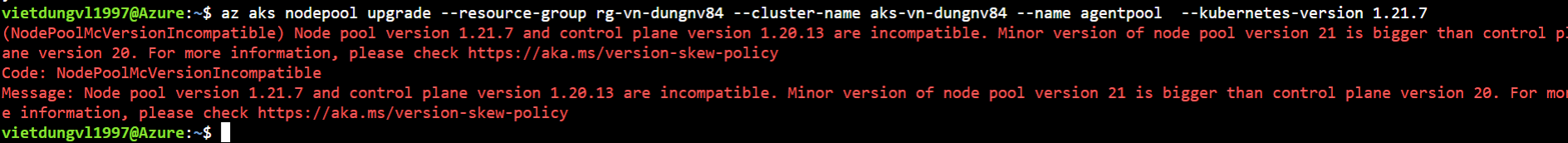
Upgrade đồng thời lỗi ngay, fix hoặc làm lại từng bước như này

1. upgrade node Image version
2. upgrade control plane
3. upgrade cluster cho từng nodepool

* phải upgrade nodepool của system trước

az aks nodepool upgrade --resource-group rg-vn-dungnv84 --cluster-name aks-vn-dungnv84 --name agentpool --kubernetes-version 1.21.7





* vì đây là upgrade nodepool version, sẽ không được, nodepool = workernode

nên phải upgrade masternode trước (control plane), sau đó mới upgrade từng nodepool rồi upgrade cả cluster

1. upgrade node Image version

pre-check:

az aks nodepool list --resource-group rg-vn-dungnv84 --cluster-name aks-vn-dungnv84 --query "[].{Name:name,NodeImageVersion:nodeImageVersion}" --output table

or

az aks nodepool get-upgrades --resource-group rg-vn-dungnv84 --cluster-name aks-vn-dungnv84 --nodepool-name dungnv --output table

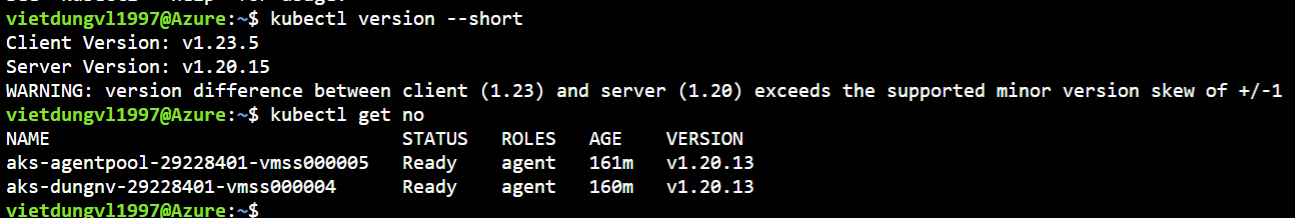
or

az aks get-upgrades --resource-group rg-vn-dungnv84 --cluster-name aks-vn-dungnv84 --output table

* upgrade nodeImage (cái này ko quan trọng, version sẽ là latest)

1. **upgrade control plane**

az aks upgrade --resource-group rg-vn-dungnv84 --name aks-vn-dungnv84 --control-plane-only --kubernetes-version 1.20.15

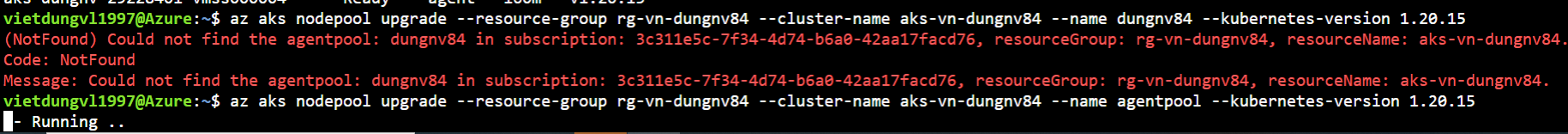


oke control plane đã lên version 1.20.15

1. **upgrade cluster cho từng nodepool**

ở trường hợp này control plan phải được lên trước, nếu ko ko thể upgrade được nodepool, sẽ lỗi ngay

* sau đó upgrade từng nodepool 1, nhớ là upgrade node system trước,



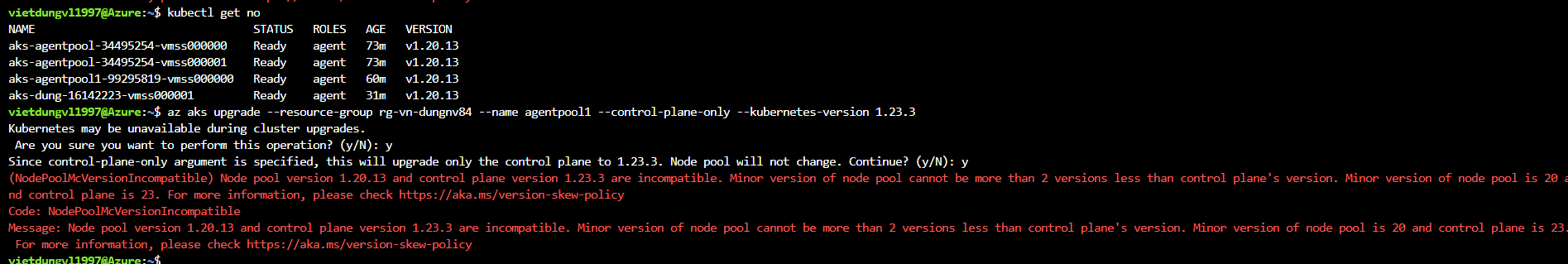
az aks nodepool upgrade --resource-group rg-vn-dungnv84 --cluster-name aks-vn-dungnv84 --name dungnv --kubernetes-version 1.21.7

Nếu node này lên version rồi thì chạy lại n cũng sẽ nhanh thôi, mất 30s

* az aks nodepool : --cluster-name
* az aks upgrade : --name (tên cluster)
* sau cùng upgrade cả

az aks upgrade --resource-group rg-vn-dungnv84 --name aks-vn-dungnv84 --kubernetes-version 1.21.7

lưu ý nhỏ



Node pool version 1.20.13 and control plane version 1.23.3 are incompatible. Minor version of node pool cannot be more than 2 versions less than control plane's version. Minor version of node pool is 20 and control plane is 23

Nodepool version 1.20.13

Control plane version: 1.22.6

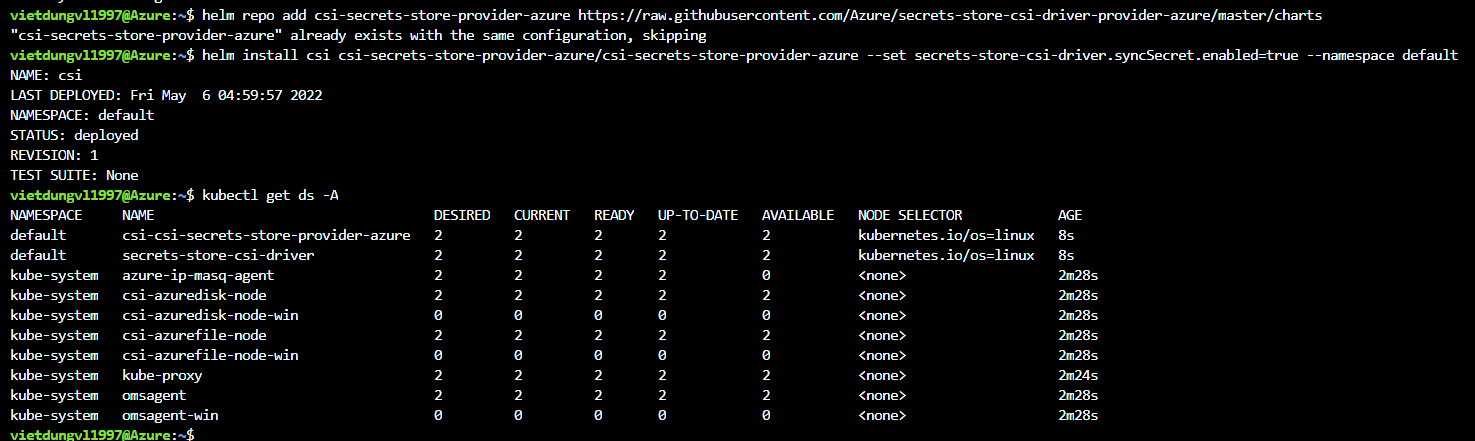
* thỏa mãn

nodepool version 1.20.13 - control plane version 1.23.3 => không thỏa mãn

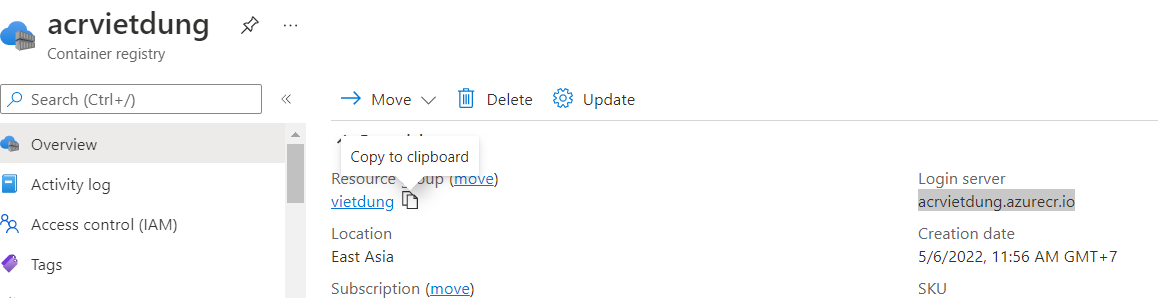
# Install CSI

helm repo add csi-secrets-store-provider-azure <https://raw.githubusercontent.com/Azure/secrets-store-csi-driver-provider-azure/master/charts>

helm install csi csi-secrets-store-provider-azure/csi-secrets-store-provider-azure --set secrets-store-csi-driver.syncSecret.enabled=true --namespace default



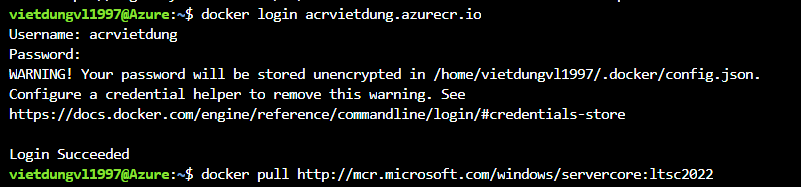
az acr import --name acrgo02easpaiaim01.azurecr.io --source acrgo02easpshared01.azurecr.io/original/ingress-nginx/controller:v0.50.0 --image original/ingress-nginx/controller:v0.50.0



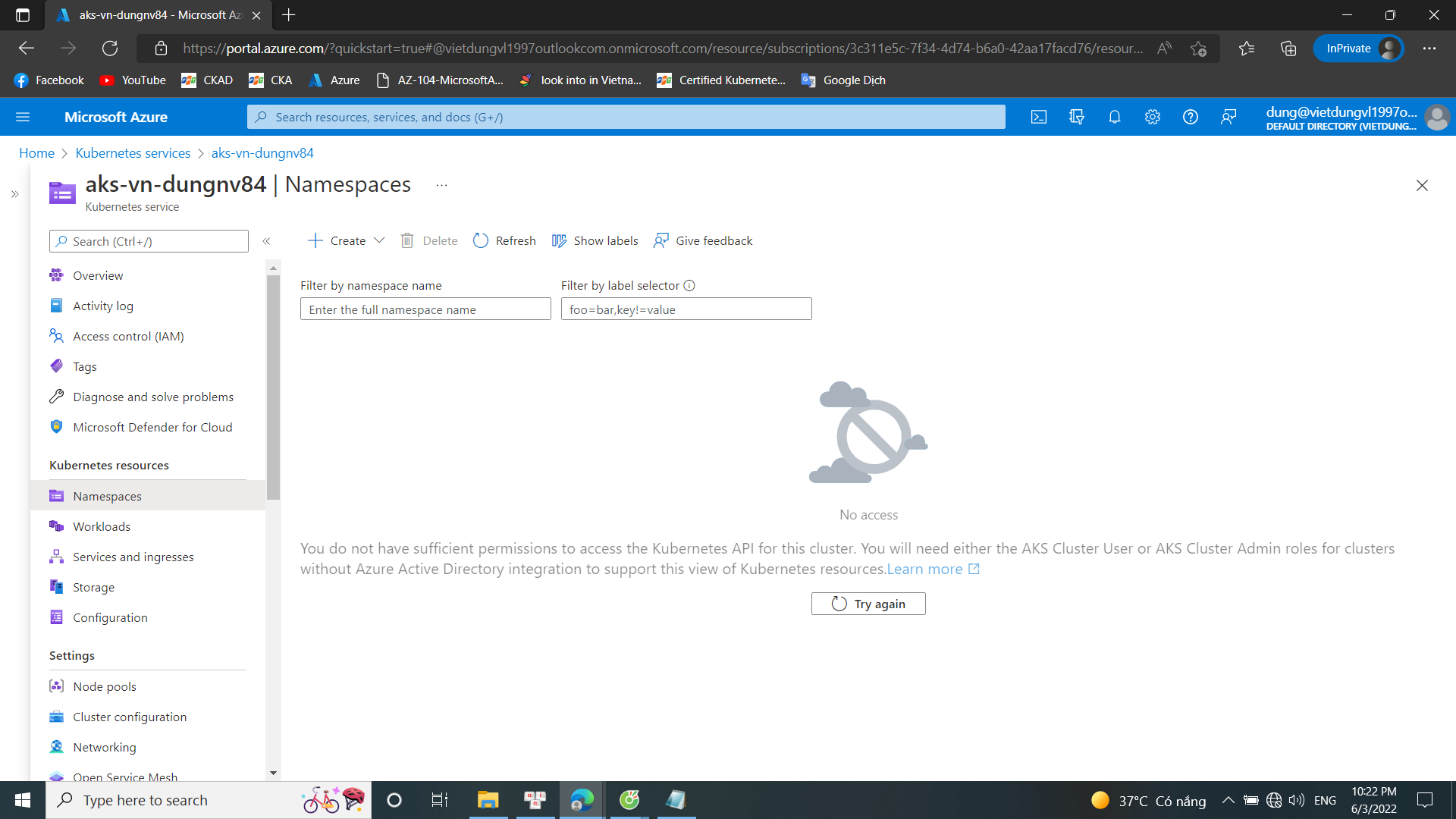
Server của container registry

## login docker



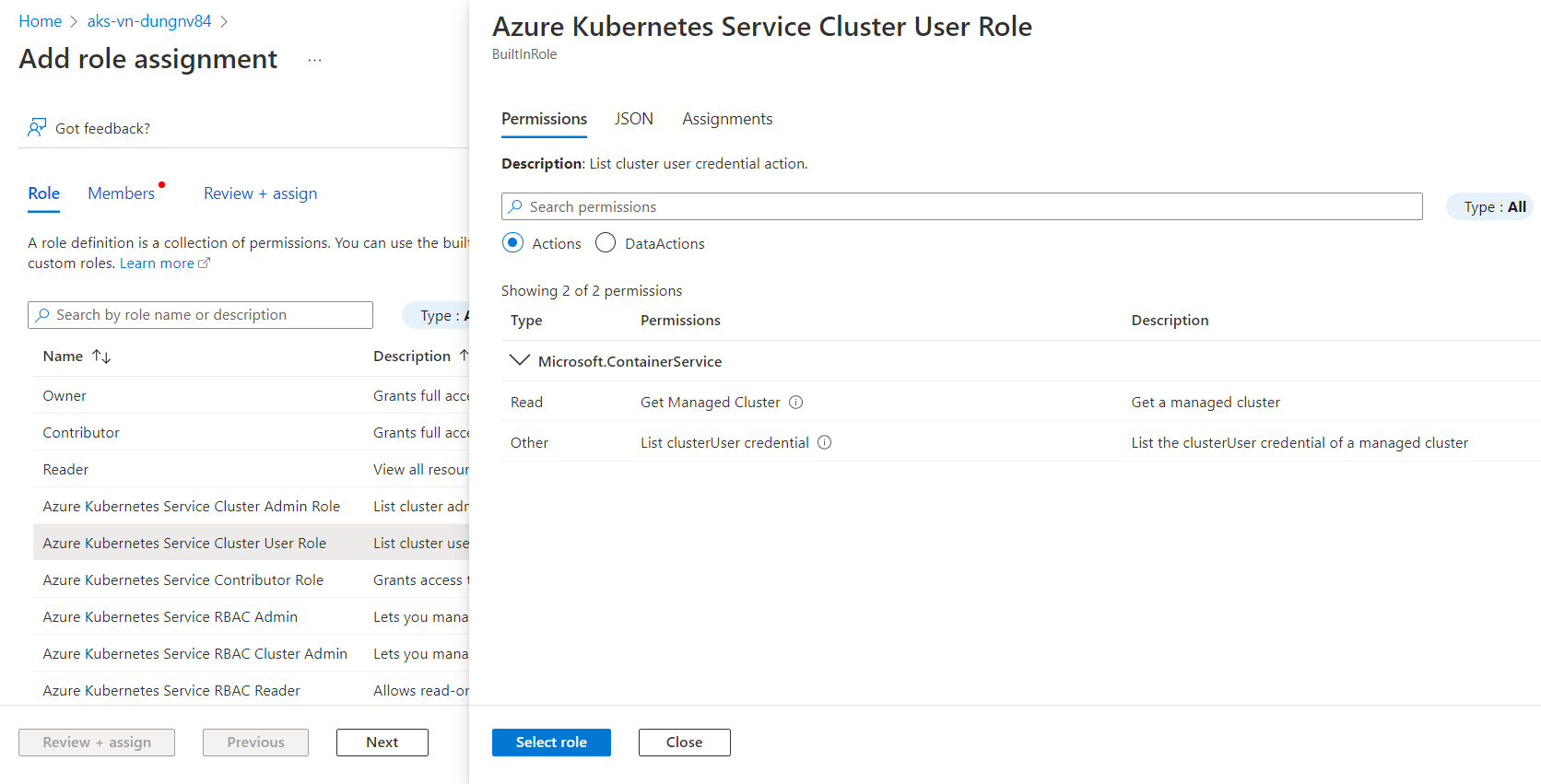


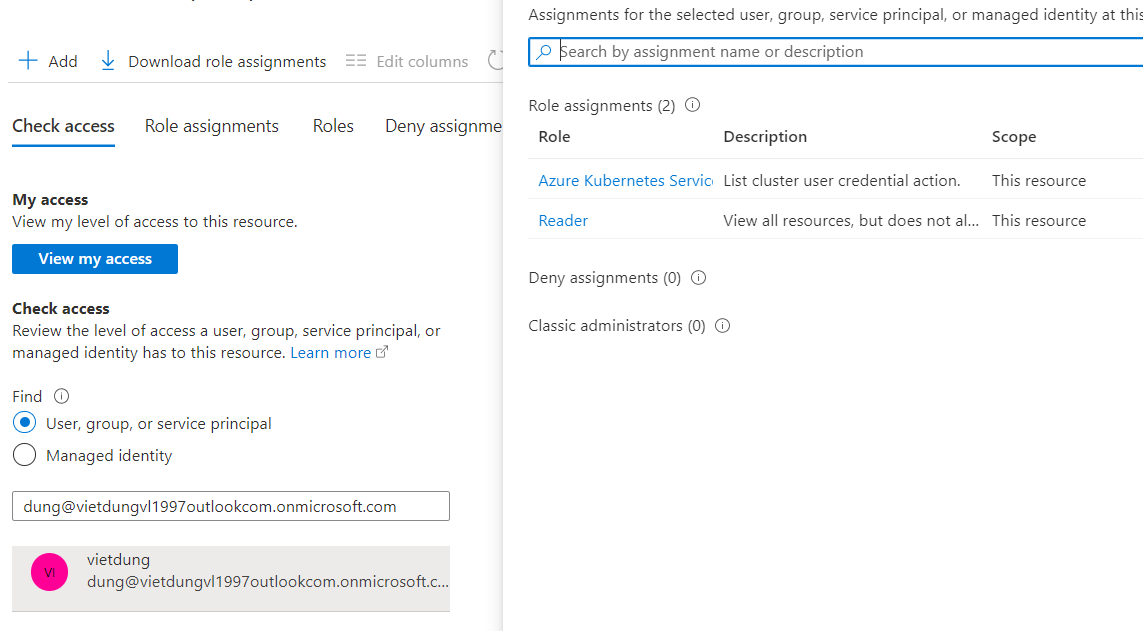
# Quyền xem resource của cluster trên portal



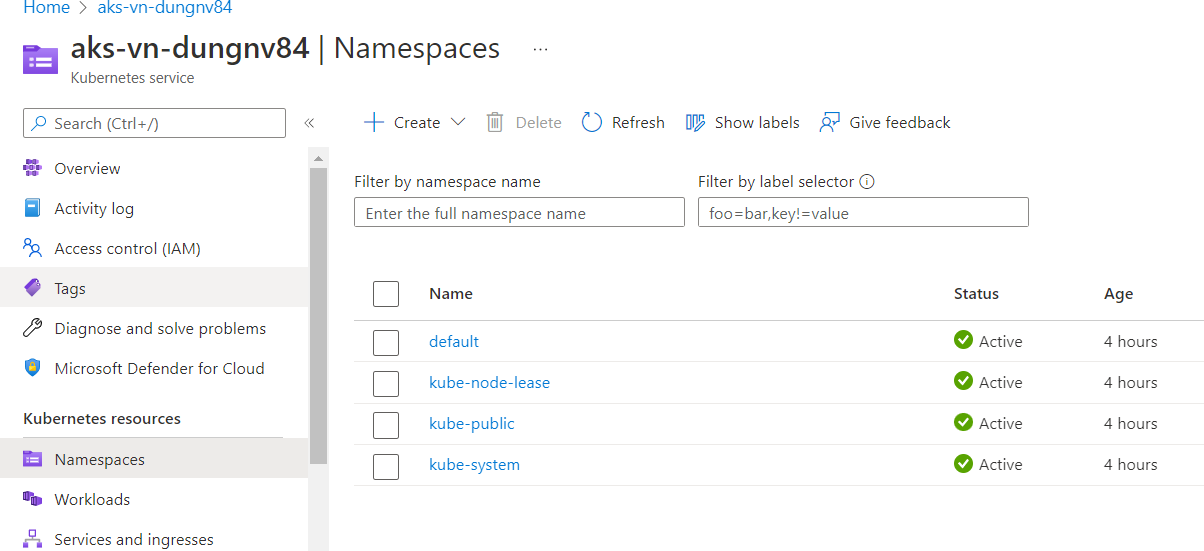


Chỉ cần thằng này là đủ, thằng trên mạnh hơn





Oke 2 quyền rồi

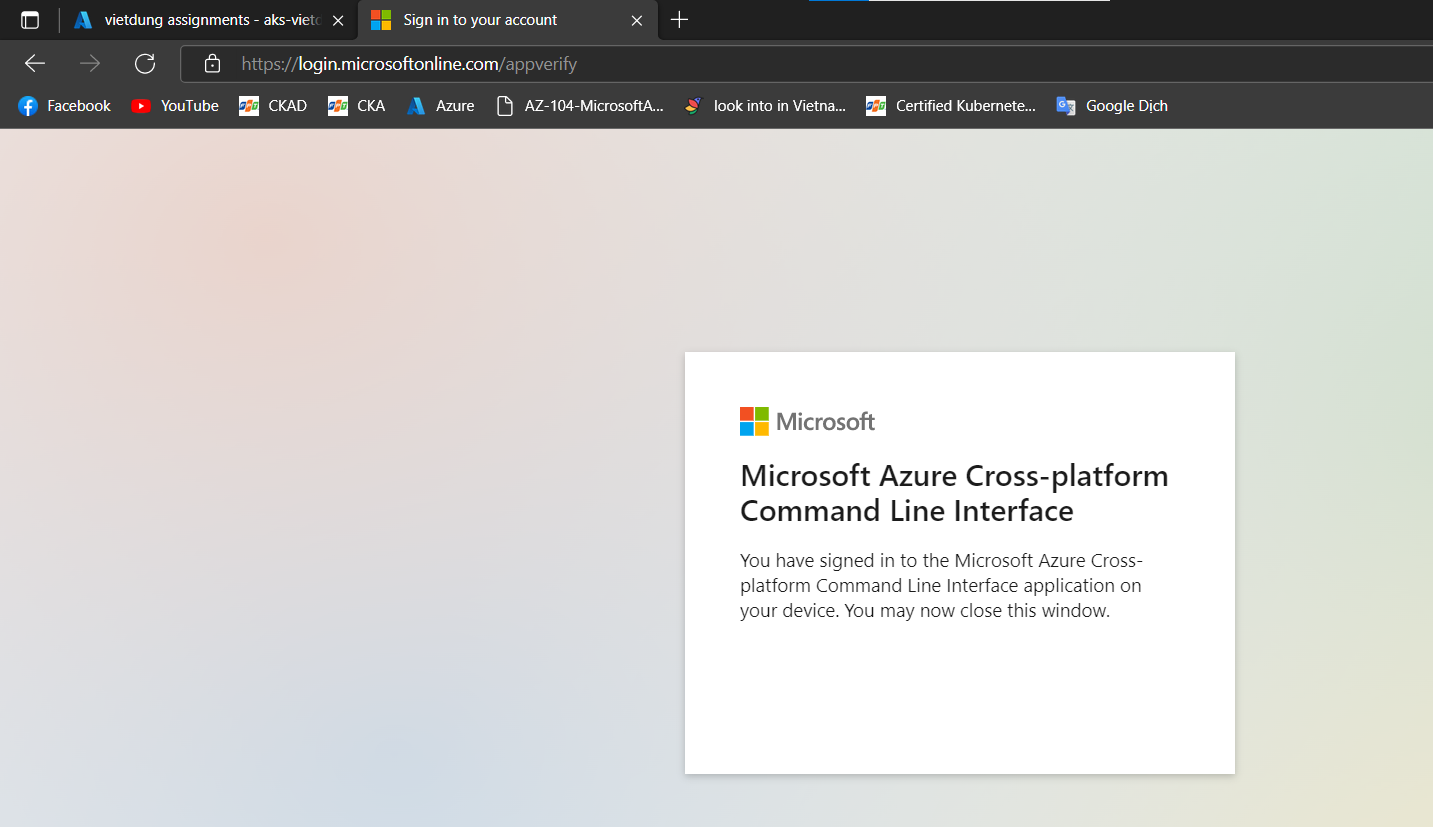


Vì RBAC là quyền cho CLI, view trên portal là quyền khác

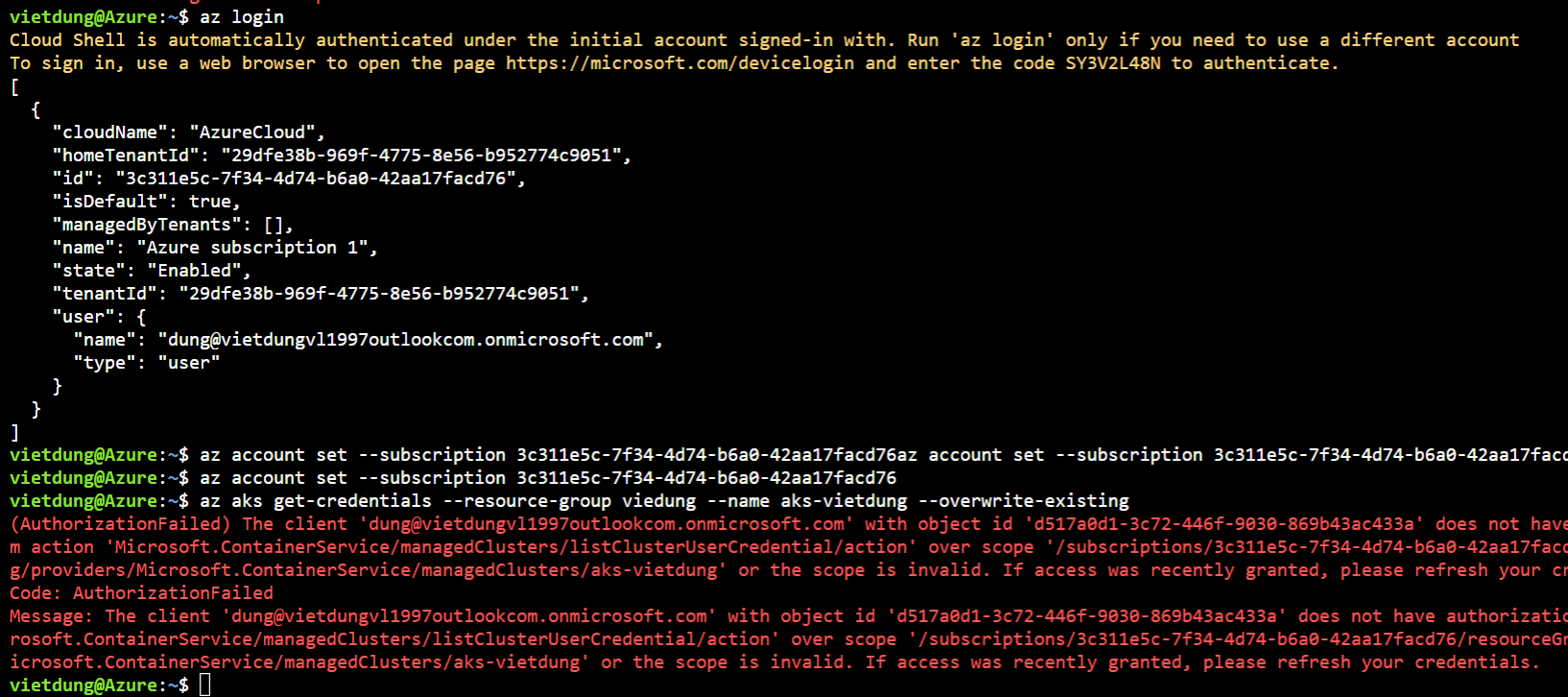
# Ko access được đến cluster

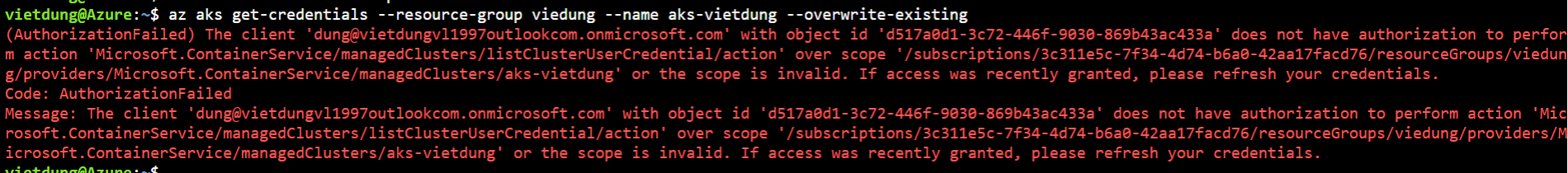
Az logout

Az login

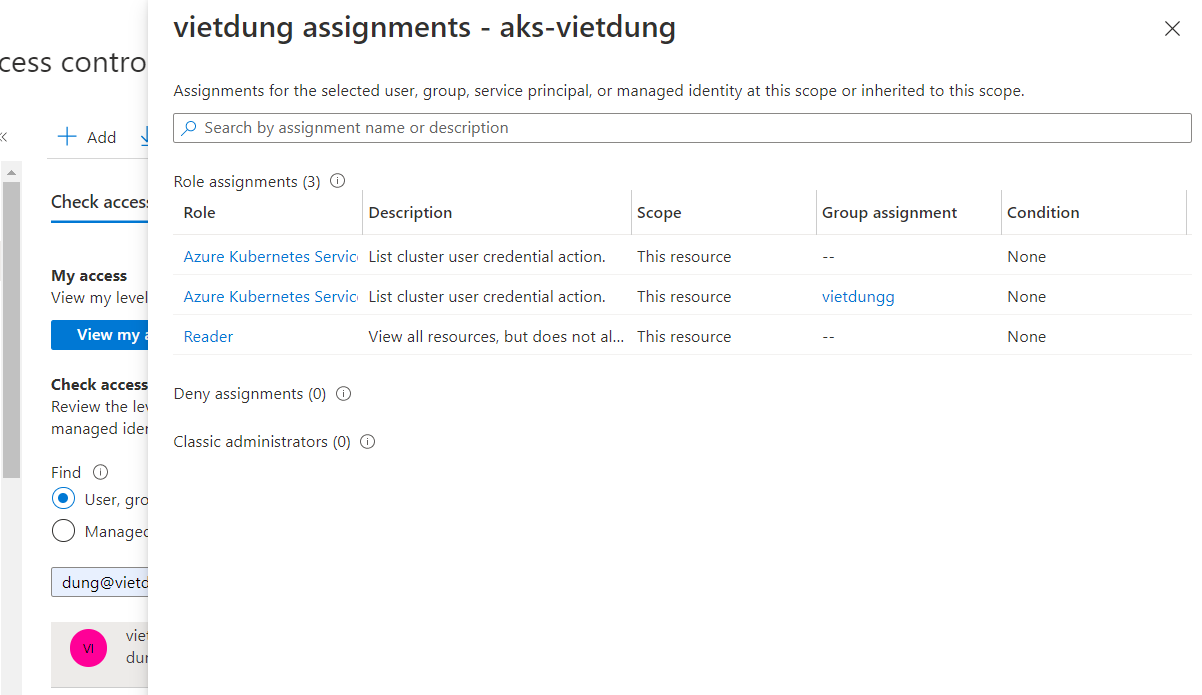


Rồi vào đường link điền mã là được, n sẽ hiện





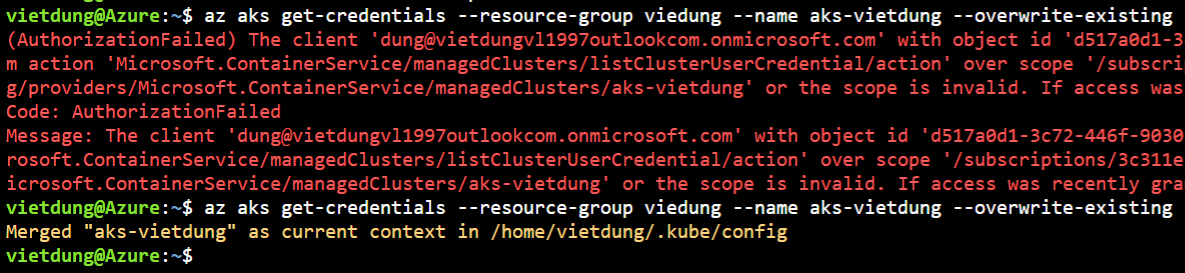
Cũng assign quyền cho kia thôi, assign cho user hoặc group



## **Permission: Azure Kubernetes Service Cluster User Role**

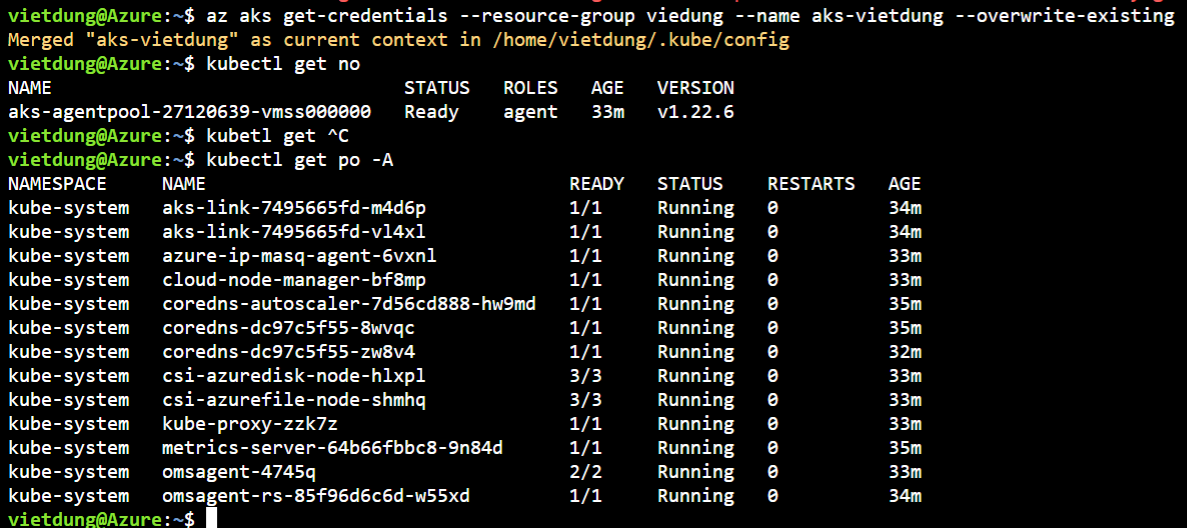
Nếu assign cho group, thì khi mình search user n cũng ra quyền, và quyền n assign cho group **vietdungg** kia

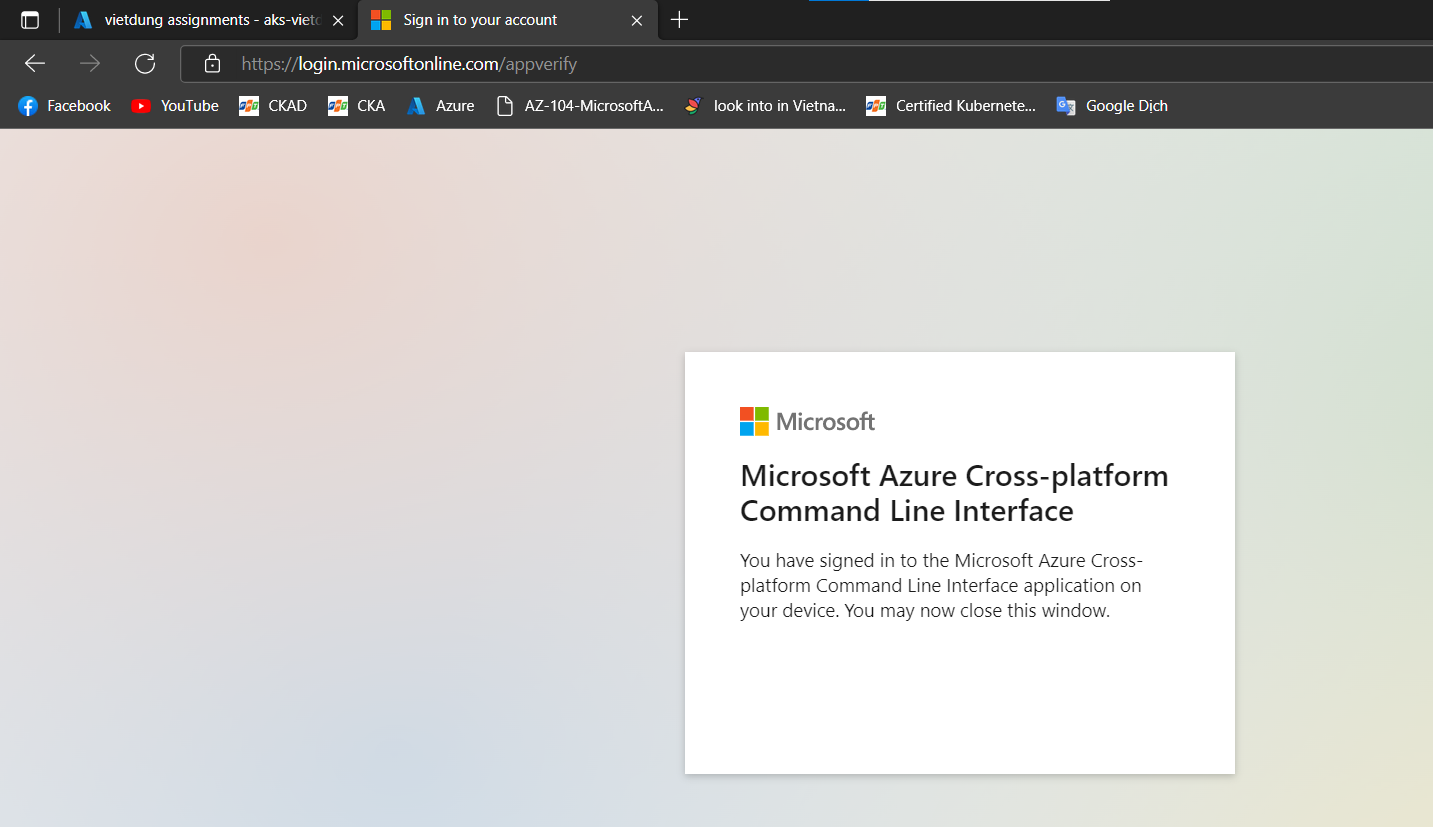
Khi assign xog là access dk



* nhưng lưu ý 1 điều rằng là grant quyền này là cấp cluster, vì vậy ko cần add Azure RBAC

ak đâu vẫn cần, vì cái trên là chỉ được list trên cả cluster thôi, còn get, delete, create thì vẫn cần phải RBAC





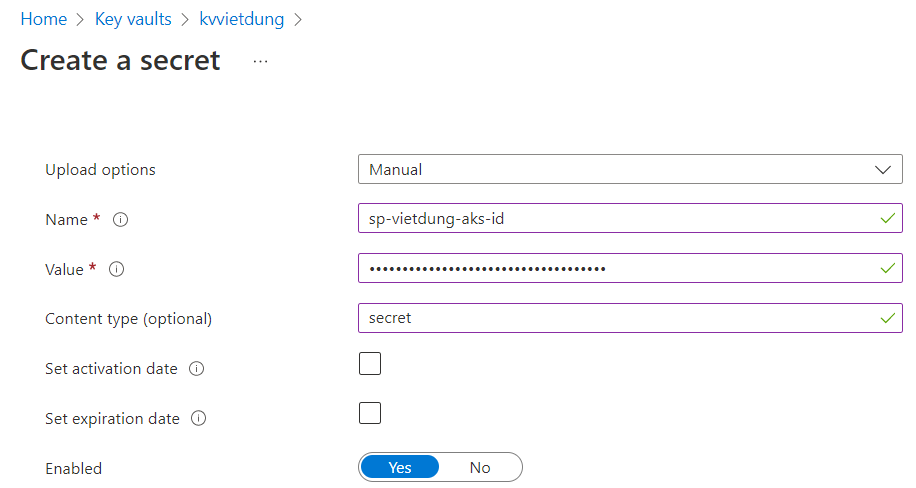
# Thực hành SPC

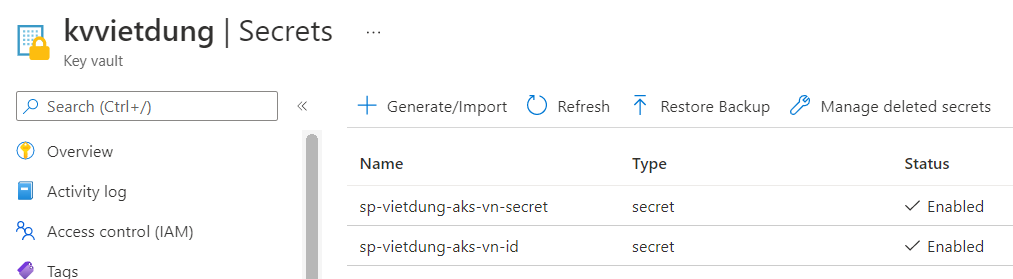
* Install CSI

helm repo add csi-secrets-store-provider-azure <https://raw.githubusercontent.com/Azure/secrets-store-csi-driver-provider-azure/master/charts>

helm install csi csi-secrets-store-provider-azure/csi-secrets-store-provider-azure --set secrets-store-csi-driver.syncSecret.enabled=true --namespace default

* Tạo 2 SP: 1 SP để làm KV
* Grant quyền cho 1 SP kv: quyền access policy : GET là đủ





Có 2 secret, 1 cái là ID của SP, 1 cái là password SP

* Tạo SPC

apiVersion: secrets-store.csi.x-k8s.io/v1alpha1

kind: SecretProviderClass

metadata:

  name: spc-vietdung

  namespace: ns-vietdung

spec:

  parameters:

    keyvaultName: kvvietdung # tên KV

    objects: |

      array:

        - |

          objectName: sp-vietdung-aks-vn-id # tên lưu trong KV

          objectType: secret # type là secret

        - |

          objectName: sp-vietdung-aks-vn-secret

          objectType: secret

    tenantId: f01e930a-b52e-42b1-b70f-a8882b5d043b

    usePodIdentity: "false"

    useVMManagedIdentity: "false"

    userAssignedIdentityID: a1b00a03-0642-47a3-a1f1-717e324b9132

  provider: azure

  secretObjects:

  - data:

    - key: sp-id      # đặt thế nào cũng dk, để gọi trong secretRef

      objectName: sp-vietdung-aks-vn-id # cái object này vs cái trên phải giống nhau

    - key: sp-secret

      objectName: sp-vietdung-aks-vn-secret

    secretName: intl-app-secret

    type: Opaque

* Tạo deployment

apiVersion: apps/v1

kind: Deployment

metadata:

  name: aks-helloworld

  namespace: ns-vietdung

spec:

  replicas: 1

  selector:

    matchLabels:

      app: aks-helloworld

  template:

    metadata:

      labels:

        app: aks-helloworld

    spec:

      containers:

      - name: aks-helloworld

        image: mcr.microsoft.com/azuredocs/aks-helloworld:v1

        ports:

        - containerPort: 80

        env: # cái Env này ko quan trọng, n vẫn pull hết secret về

        - name: TITLE

          value: "Welcome to Azure Kubernetes Service (AKS)"

        - name: "DB\_ID" # naming ENV variable inside the container

          valueFrom:

            secretKeyRef:

              name: "intl-app-secret" # name of k8s secret (sync secret)

              key: "sp-id" # key secretObject trong SPC

        - name: "DB\_secret" # naming ENV variable inside the container

          valueFrom:

            secretKeyRef:

              name: "intl-app-secret"

              key: "sp-secret"

        volumeMounts:

        - name: secrets-store-inline

          mountPath: "/mnt/secrets-store"

          readOnly: true

      volumes:

        - name: secrets-store-inline

          csi:

            driver: secrets-store.csi.k8s.io

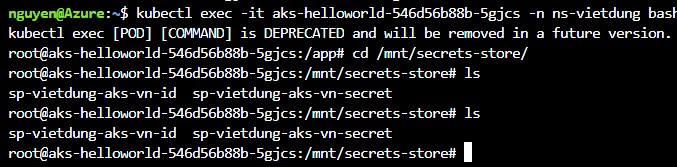
            readOnly: true

            volumeAttributes:

              secretProviderClass: "spc-vietdung" # name SecretProviderClass

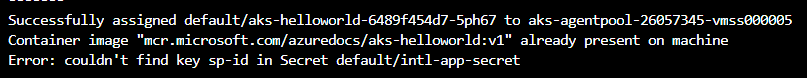
            nodePublishSecretRef:

              name: "kvcred-vietdung-aks" # name of secret created to access keyvault

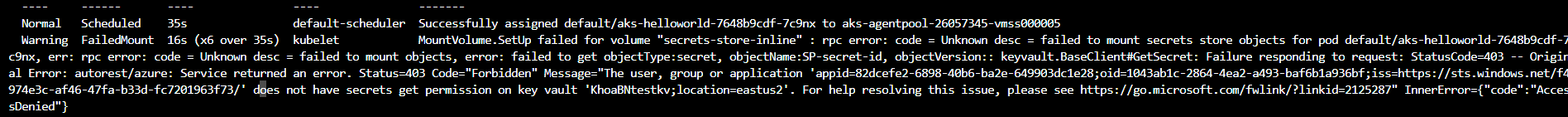


Oke đã pull được

Sync secret n sẽ tự tạo



Lỗi này ko xóa sync secret cũ

lỗi này ko add permission kvcreds