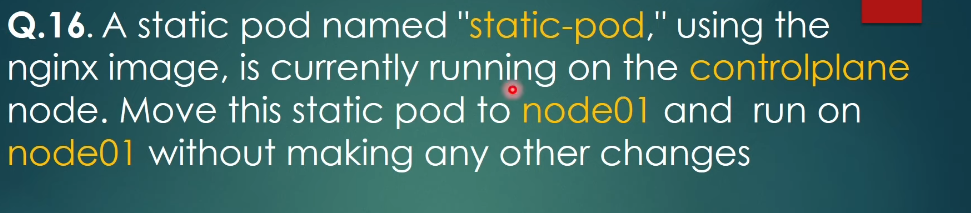
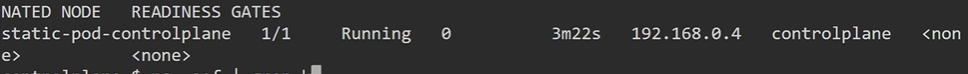
WORKLOADS and SCHEDULING

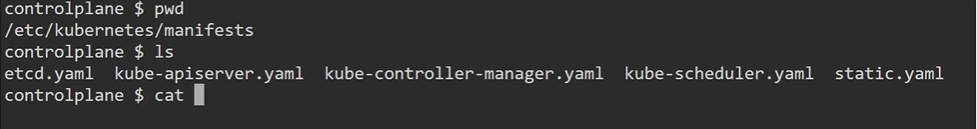
Câu 1:



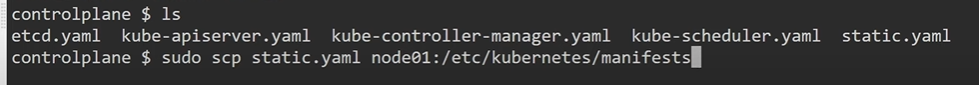
Lời giải: Hiện tại có node static-pod đang chạy trên controlplane, giờ chúng ta cần tìm thư mục static của node01 và chuyển file manifest sang đó.



* Tìm file manifest trong controlplane



* Coppy file static.yaml sang /etc/kubernetes/manifests của node01



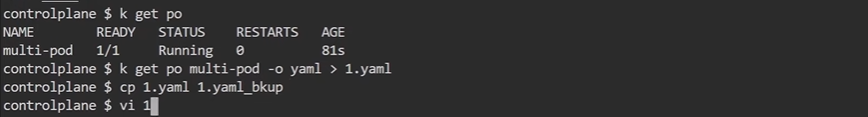
* Kết quả



Câu 2:



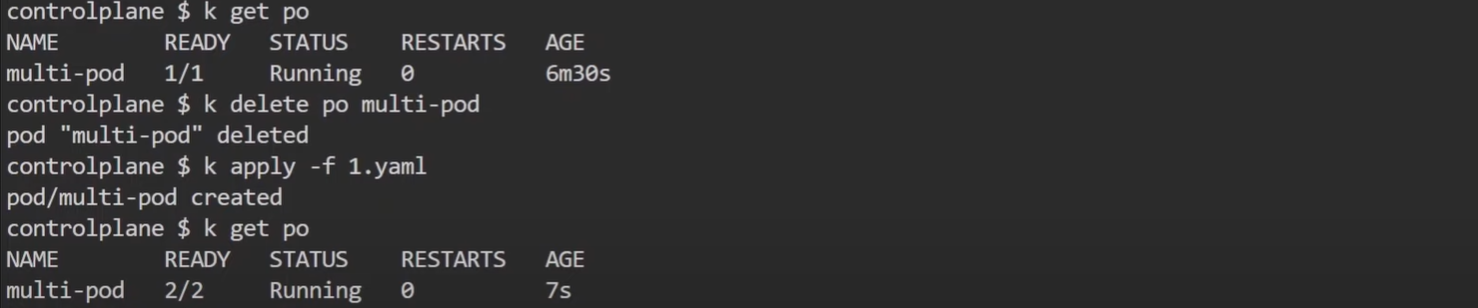
Lời giải: Ta cần cấu hình lại file manifest có sẵn để thành 1 pod chạy 2 container như yêu cầu đề bài.



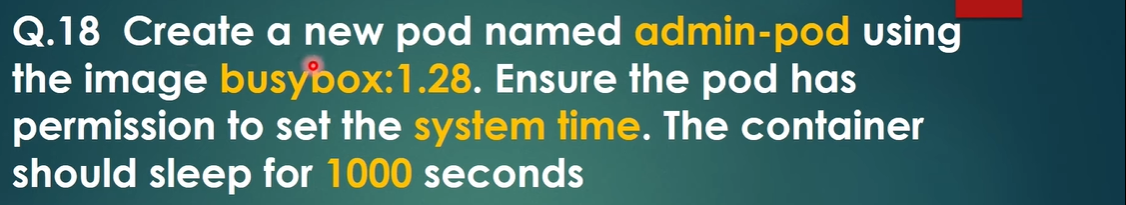
* Sửa file manifest thành như sau:



* Tiếp theo xoá bỏ pod cũ đang chạy với 1 container rồi chạy lại file mới 1.yaml và cung xem kết quả:



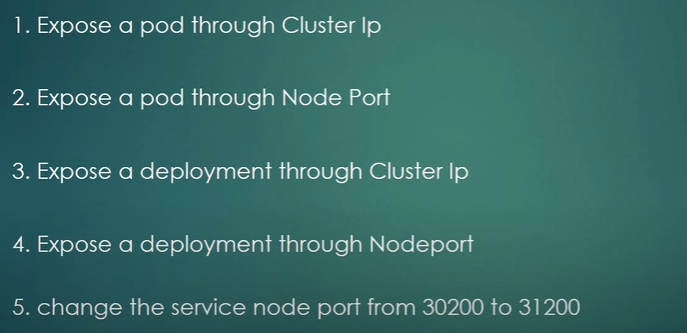
Câu 3:



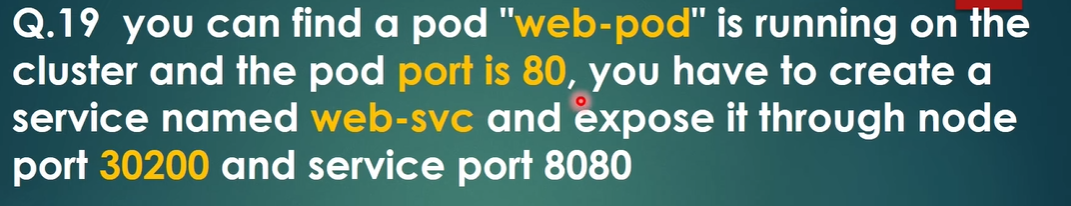
Lời giải: Ta sẽ tạo 1 manifest có phần “capabilities” như sau:



SERVICE



Câu 1:



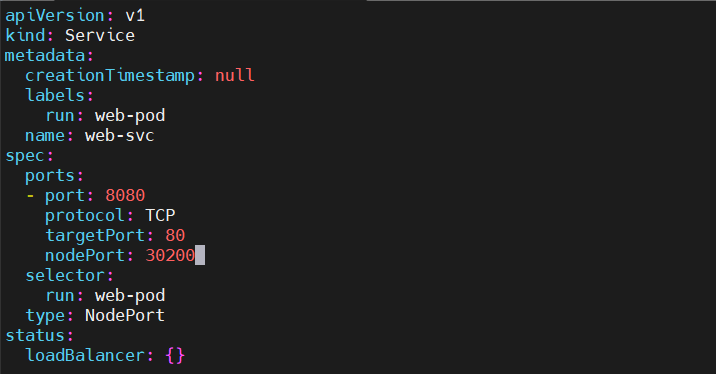
Lời giải : Tạo 1 pod bằng lệnh “kubectl run web-pod –-image=nginx”



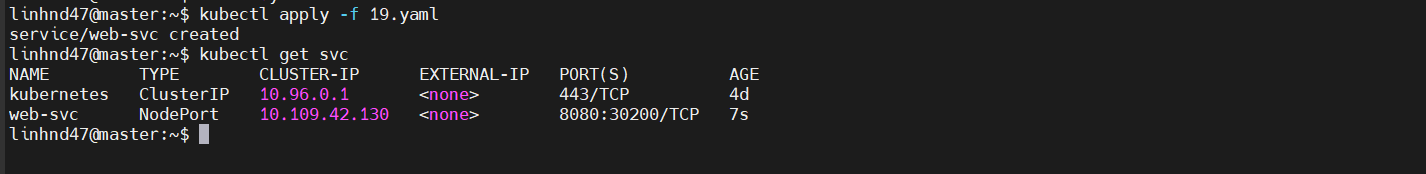
* Expose pod bằng NodePort qua nodeport 30200

kubectl expose pod web-pod --name=web-svc --type=NodePort --port=8080 --dry-run=client -oyaml > 19.yaml

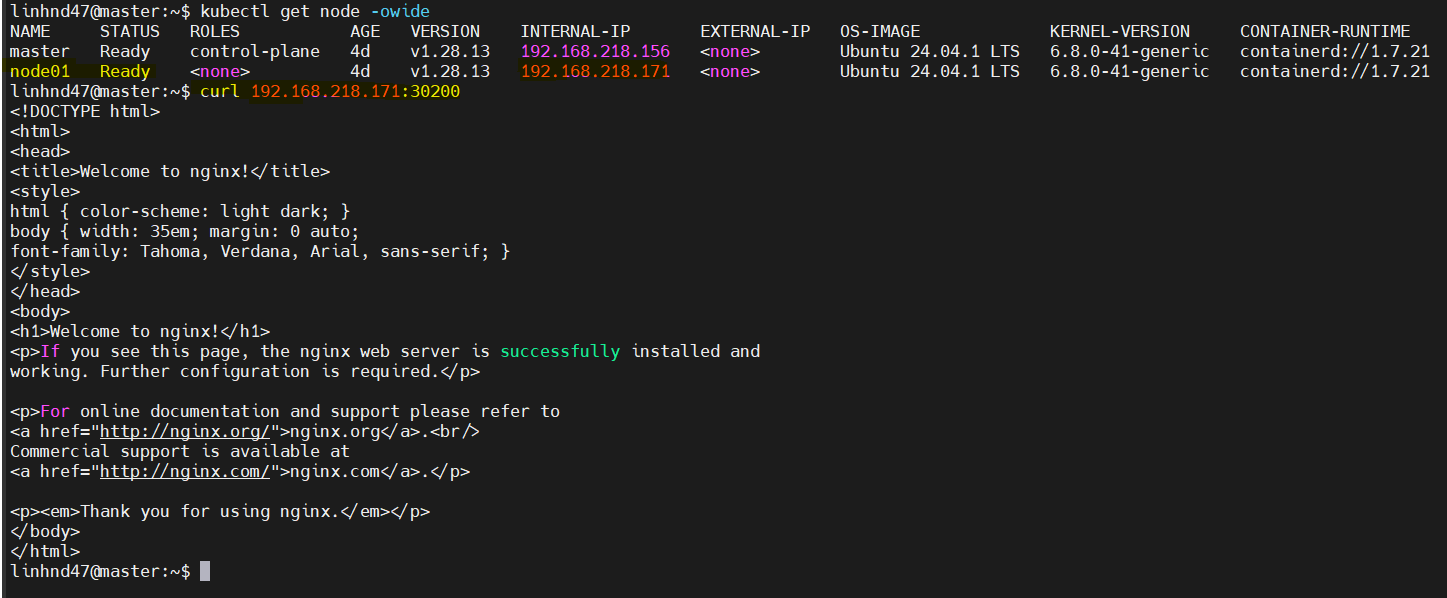
* Vào trong file 19.yaml edit như sau:



* Thực thi file svc này:



* Kiểm tra bằng lệnh curl tới dịch vụ:



Câu 2:



Lời giải:



* Expose deployment với tên là web-deploy-svc

kubectl expose deploy nginx-deployment --name=web-deploy-svc --port=8080

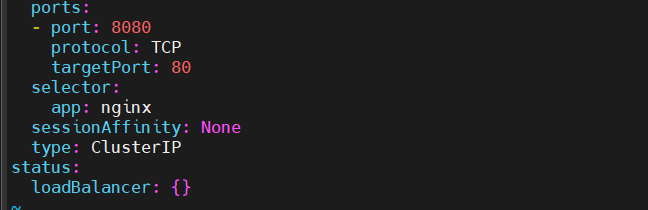
* Expose thành công:



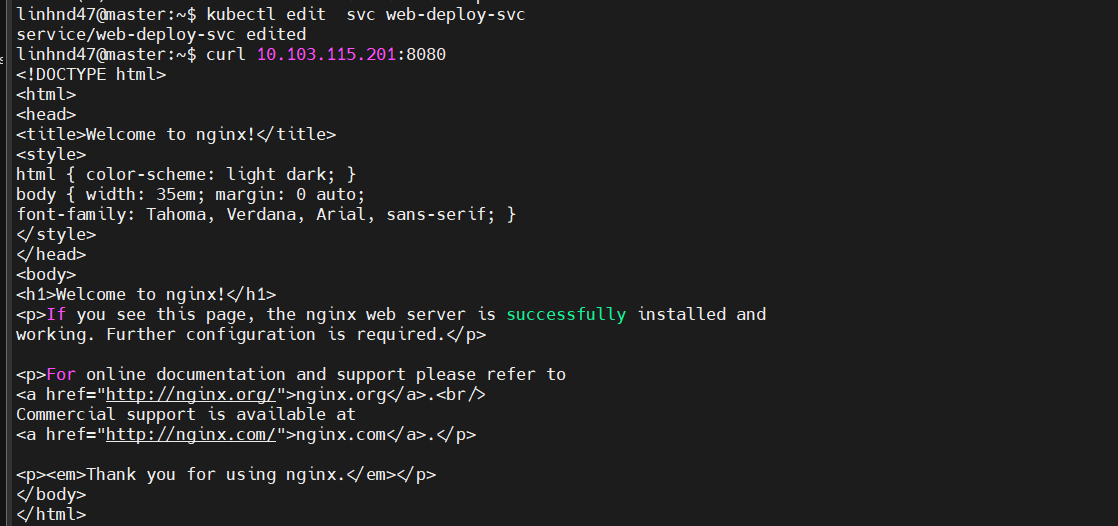
* Khi curl thì sẽ không thành công vì port trên container là 80 mà targetport của svc là 8080 nên chưa match với nhau:



* Ta cần vào edit lại trên svc phần targetport từ 8080 thành 80:



* Thực hiện curl lại:

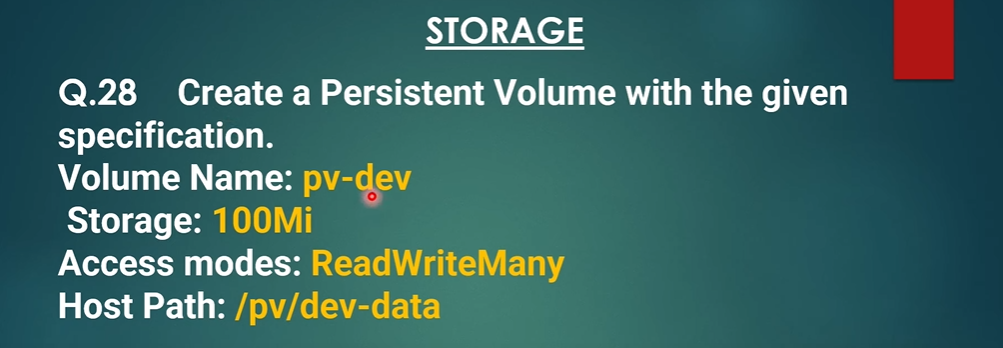


**Cách 2: là chạy luôn lệnh này mà không cần edit mất time:**

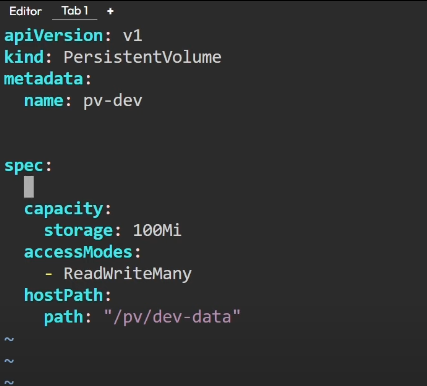
kubectl expose deploy nginx-deployment --name=web-deploy-svc --port=8080 –-target-port=80

🡪 Nice

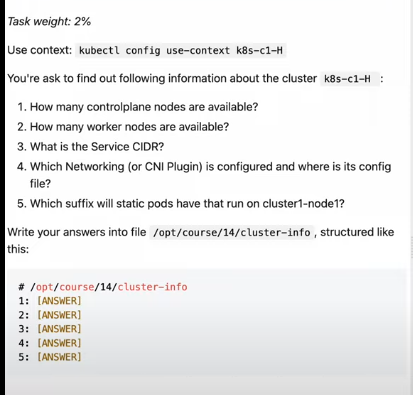
Câu 4:



Lời giải: Tạo manifest như sau:

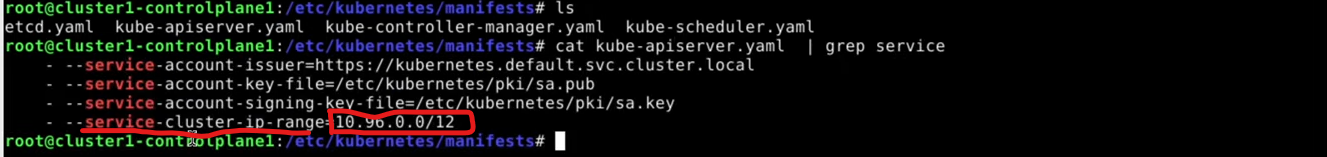


Câu 5:

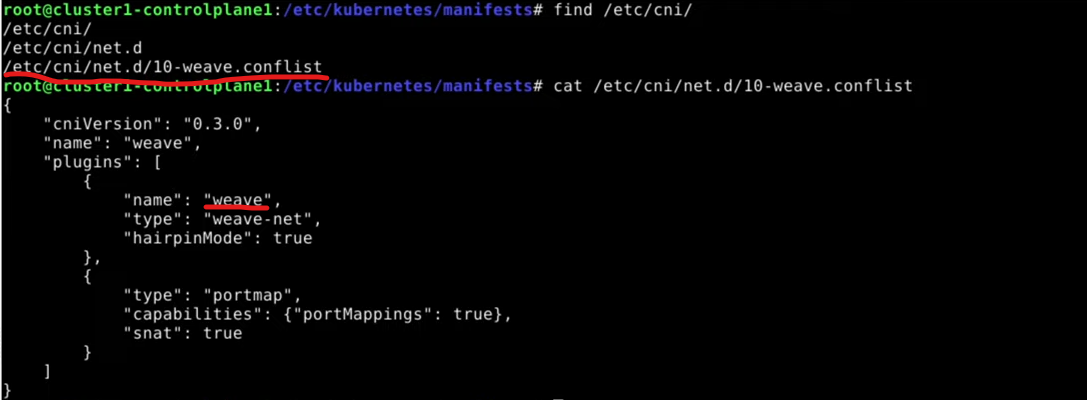


Lời giải:

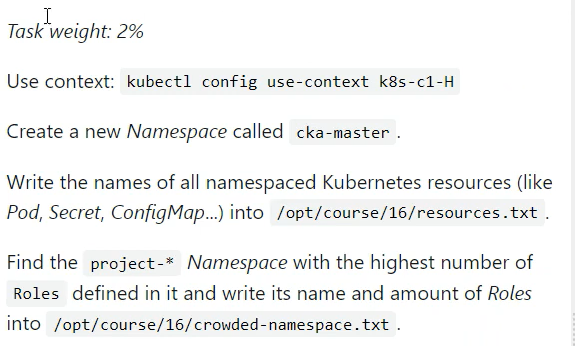
3:



4:



Câu 6:



Lời giải:

1:

Kubectl create ns cka-master

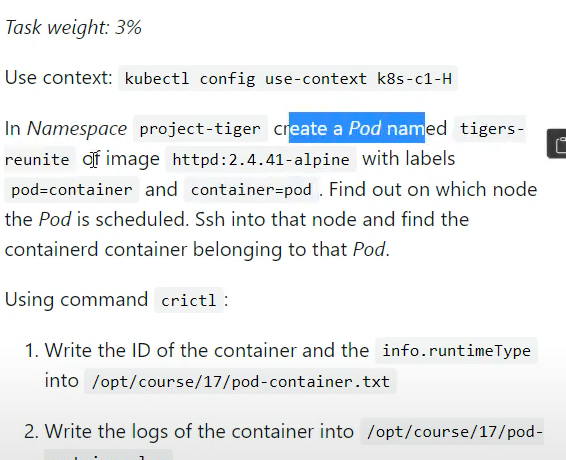
2:

Kubectl api-resource –namespace -o name > /opt/course/16/resources.txt

3:



Câu 7:

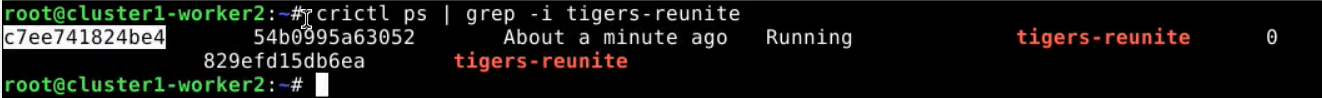


Lời giải:

1: Tạo pod trong namespace project-tiger

Kubectl run -n project-tiger tigers-reunite –image=httpd:2.4.41-alpine –labels=”pod=container,container=pod”

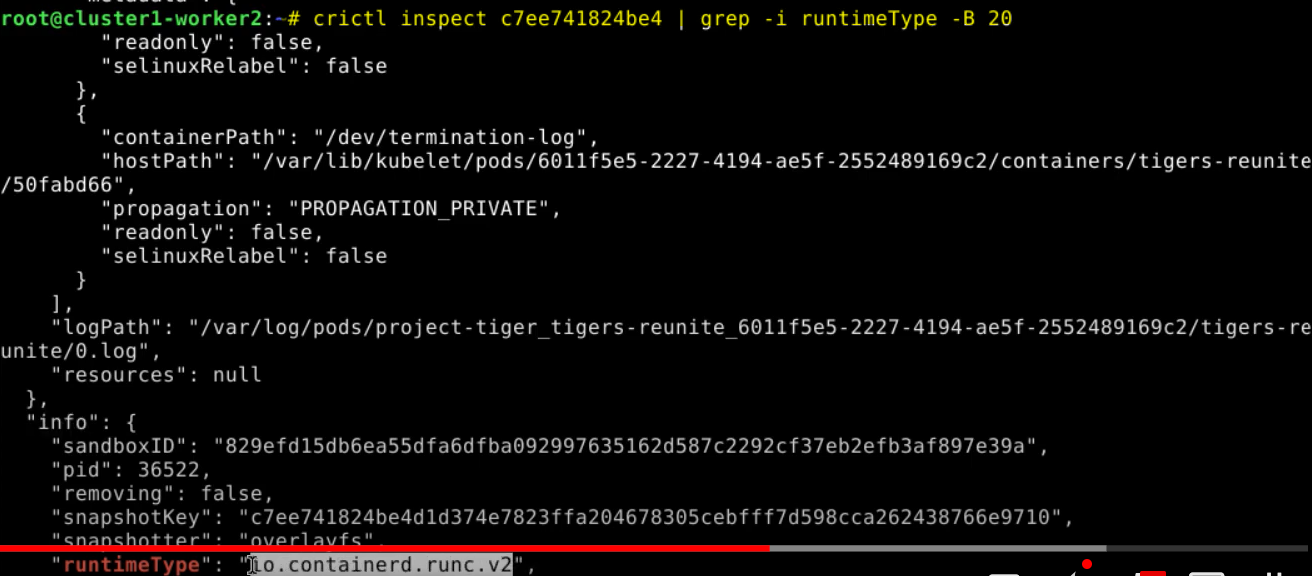
2: Vào trong node đang chạy container đó chạy lệnh crictl ps | grep -i tiger- reunite



Rồi ghi ID kia vào file /opt/course/17/pod-container.txt

Tiếp theo lấy info bằng lệnh:

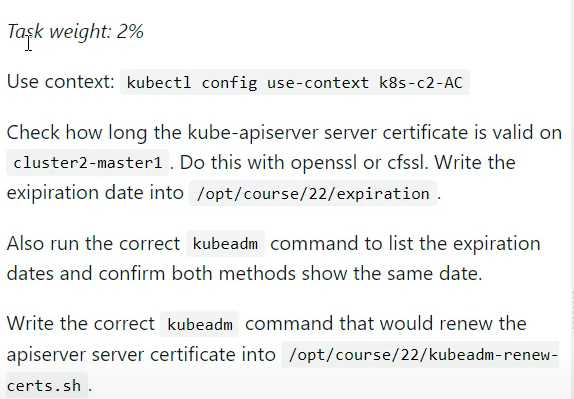
Crictl inspect c7ee741824be4 | grep -I runtimeType -B 20



3:



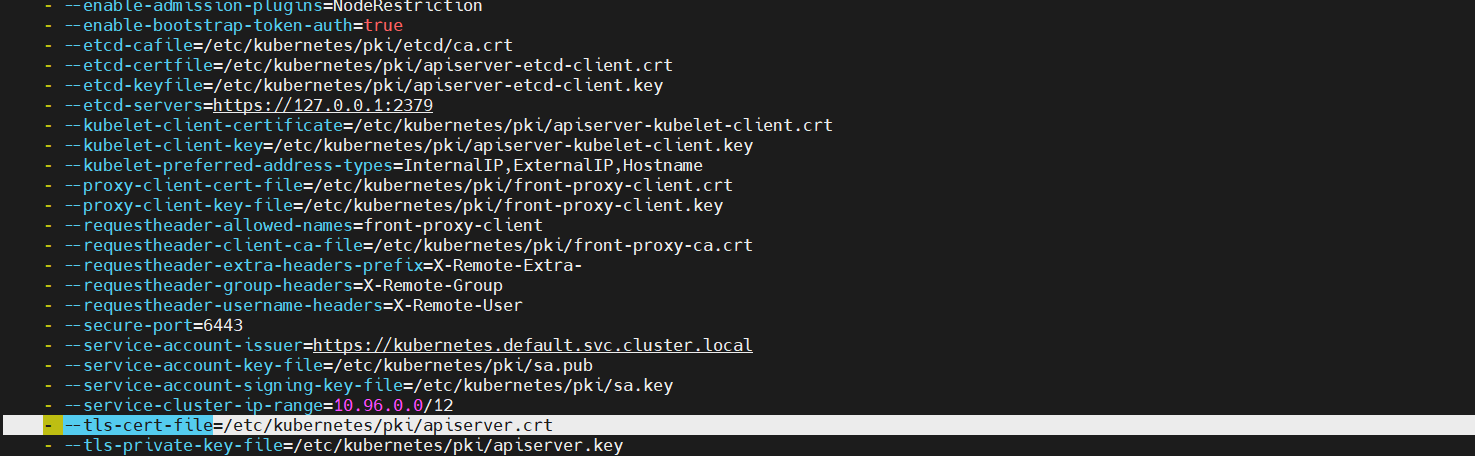
Câu 8:



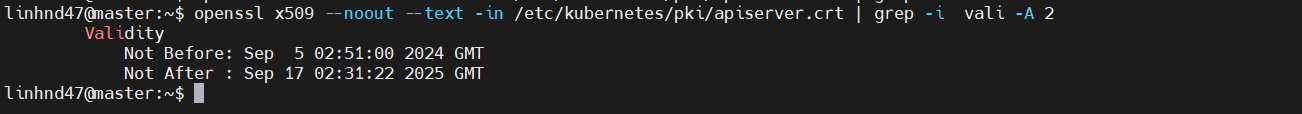
Lời giải:

1. Check file kubectl-apiserver.yaml trong manifest

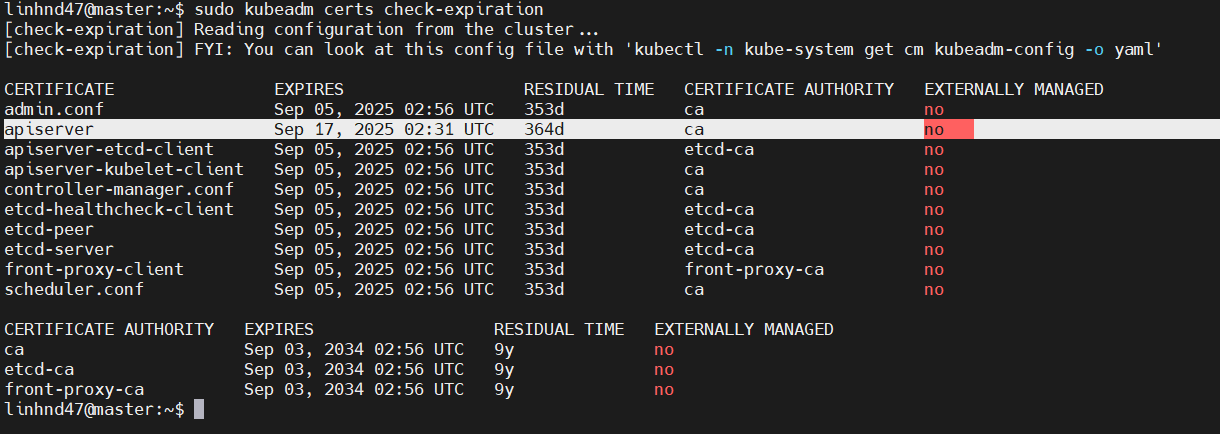
Vim /etc/kubenetes/manifest/kube-apiserver.yaml | grep tls



openssl x509 --noout --text -in /etc/kubernetes/pki/apiserver.crt | grep -i vali -A 2



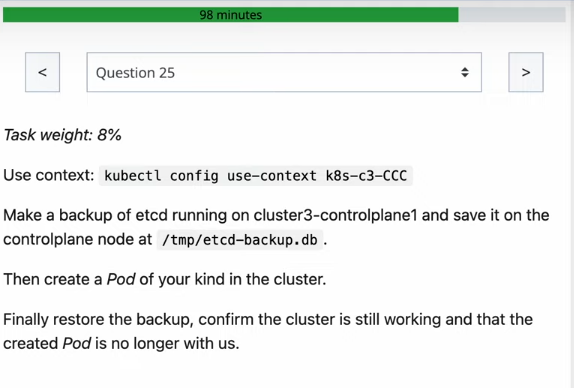
1. kubeadm certs check-expiration



* Ra hạn thì ta dùng lệnh sau rồi ghi lệnh vào file theo yêu cầu:

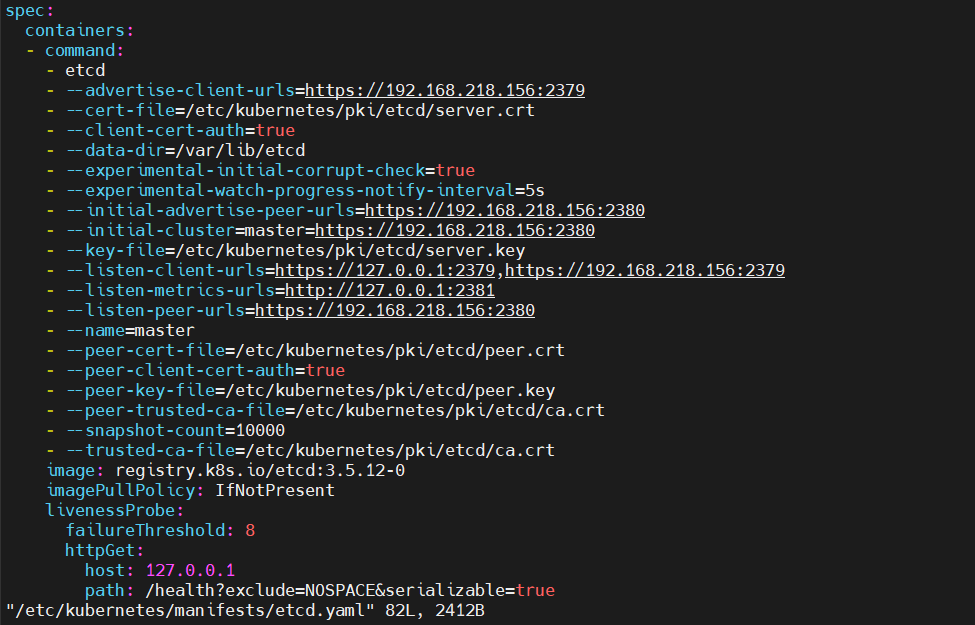
kubeadm certs renew apiserver

Câu 9: Backup and Restore etcd



1.Check thông số rồi điền vào lệnh

Vim /etc/kubernetes/manifests/etcd.yaml



* Lệnh chạy tạo snapshot etcd:

ETCDCTL\_API=3 etcdctl --endpoints=https://127.0.0.1:2379 **\**

--cacert=<trusted-ca-file> --cert=<cert-file> --key=<key-file> **\**

snapshot save <backup-file-location>

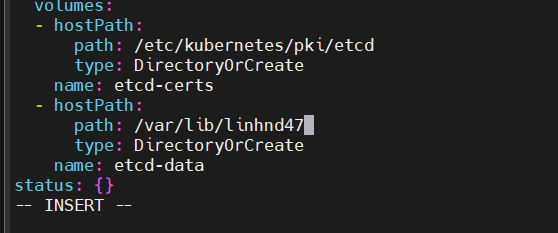
1. Khi tạo được file snapshot ta tạo một pod tên nginx-pod



1. Restore etcd cũ khi chưa có pod nginx-pod

ETCDCTL\_API=3 etcdctl --cacert=/etc/kubernetes/pki/etcd/ca.crt --cert=/etc/kubernetes/pki/etcd/server.crt --key=/etc/kubernetes/pki/etcd/server.key --data-dir /var/lib/linhnd47 snapshot restore /opt/snapshot-pre-boot1.db

1. Tiếp theo vào file /etc/kubernetes/manifests/etcd.yaml để sửa phần hostpath từ /var/lib/etcd -> /var/lib/linhnd47 để quay trở về etcd cũ.



1. Kiểm tra thì thấy đã mất pod nginx-pod

