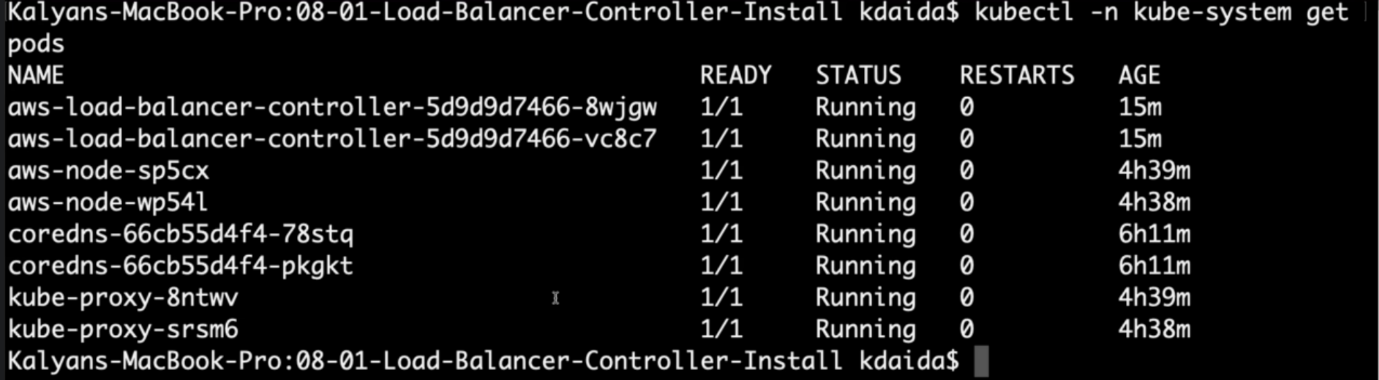
**6: LBC Service Account and TLS Cert Internals**

**Verify AWS Load Balancer Controller Logs**

**# List Pods**

--- **kubectl get pods -n kube-system**

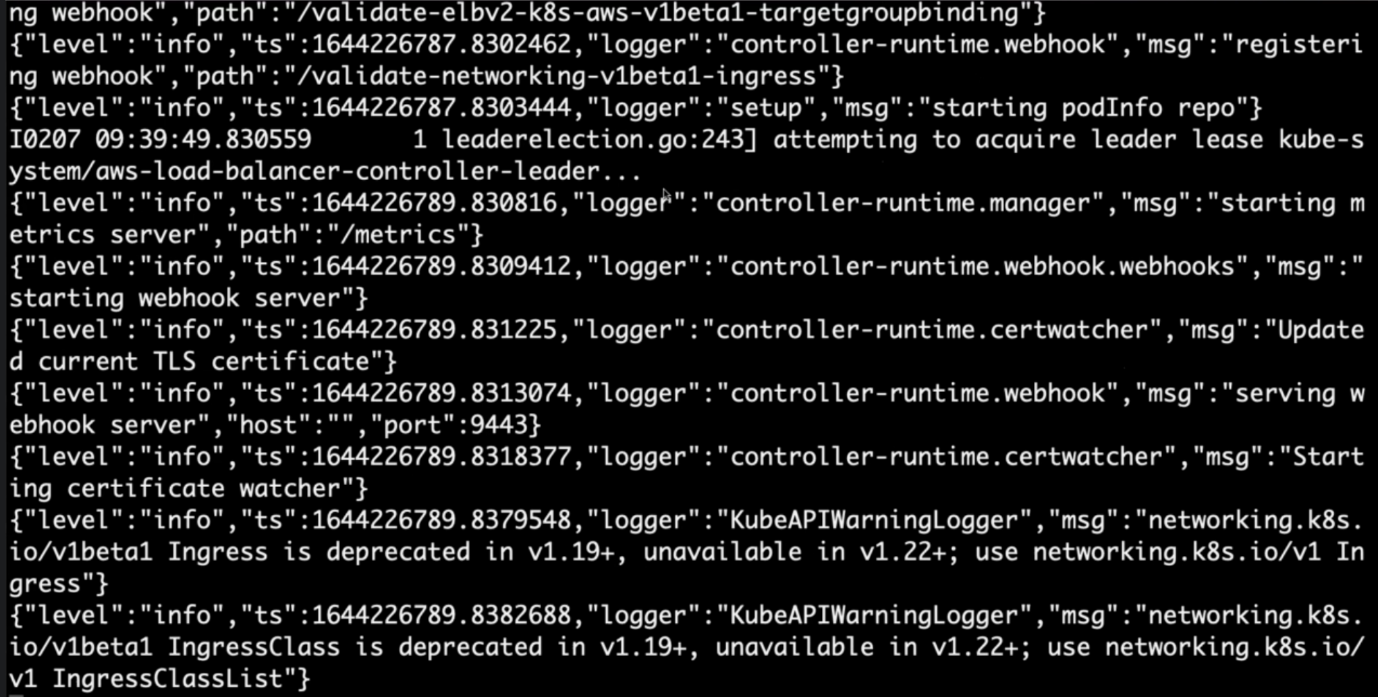


--- **note** – we have 2 pods which are related to aws-load-balancer-controller, those will be listed with other.

**# Review logs for AWS LB Controller POD-1**

--- **kubectl -n kube-system logs -f <POD-NAME>**

--- **kubectl -n kube-system logs -f aws-load-balancer-controller-86b598cbd6-5pjfk**



--- **note** – logs of that respective pod will be listed here.

**# Review logs for AWS LB Controller POD-2**

--- **kubectl -n kube-system logs -f <POD-NAME>**

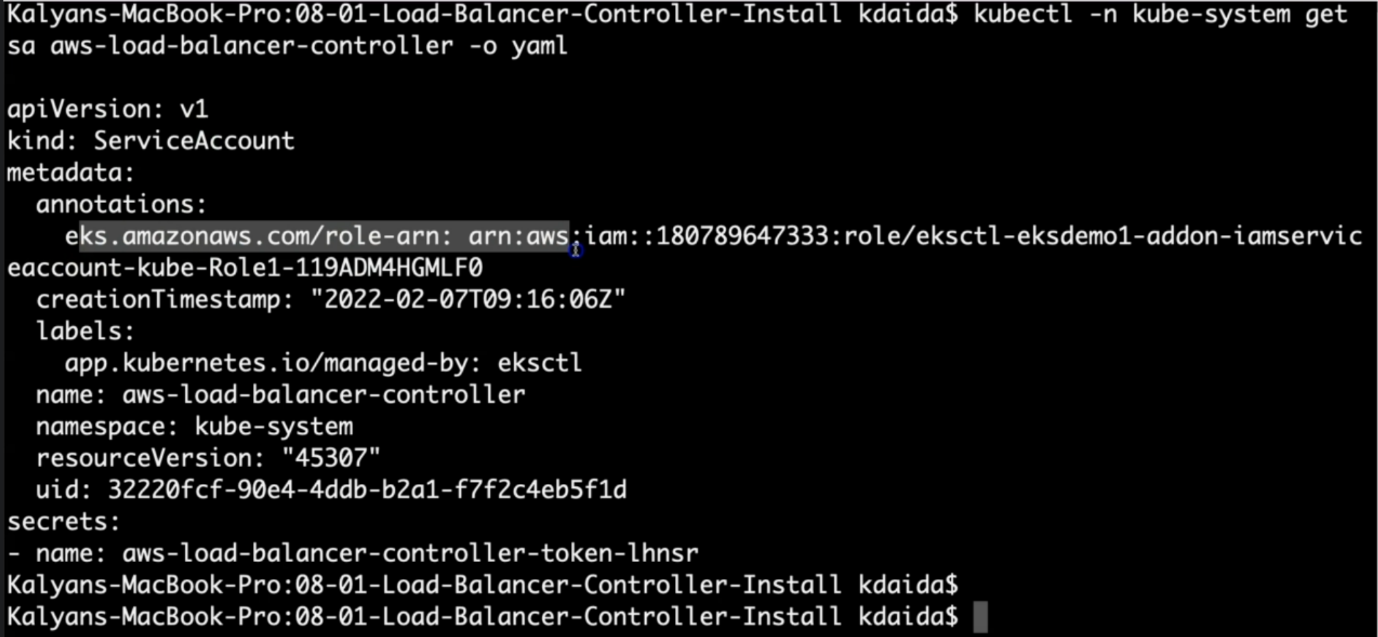
--- **kubectl -n kube-system logs -f aws-load-balancer-controller-86b598cbd6-vqqsk**

**Verify AWS Load Balancer Controller k8s Service Account – Internals**

**# List Service Account and its secret**

--- **kubectl -n kube-system get sa aws-load-balancer-controller**

--- **kubectl -n kube-system get sa aws-load-balancer-controller -o yaml** – the service account in yml format.



--- **labels**: this is managed by eksctl command.

--- **note** – we have secrets here, inside of secrets we have aws-load-balancer-controller-token-lhnsr. Note the secret name.

--- **kubectl -n kube-system get secret <GET\_FROM\_PREVIOUS\_COMMAND - secrets.name> -o yaml**





--- **note** – you can see that the **ca.crt** and **token**

--- **kubectl -n kube-system get secret aws-load-balancer-controller-token-5w8th**

--- **kubectl -n kube-system get secret aws-load-balancer-controller-token-5w8th -o yaml**

**## Decoce ca.crt using below two websites**

--- [**https://www.base64decode.org/**](https://www.base64decode.org/)

--- [**https://www.sslchecker.com/certdecoder**](https://www.sslchecker.com/certdecoder)

**## Decode token using below two websites**

--- <https://www.base64decode.org/>

--- <https://jwt.io/>

**Observation:**

1. Review decoded JWT Token

**# List Deployment in YAML format**

--- kubectl -n kube-system get deploy aws-load-balancer-controller -o yaml

**Observation:**

1. Verify "spec.template.spec.serviceAccount" and "spec.template.spec.serviceAccountName" in "aws-load-balancer-controller" Deployment

2. We should find the Service Account Name as "aws-load-balancer-controller"

**# List Pods in YAML format**

--- kubectl -n kube-system get pods

--- kubectl -n kube-system get pod <AWS-Load-Balancer-Controller-POD-NAME> -o yaml

--- kubectl -n kube-system get pod aws-load-balancer-controller-65b4f64d6c-h2vh4 -o yaml

**Observation:**

1. Verify "spec.serviceAccount" and "spec.serviceAccountName"

2. We should find the Service Account Name as "aws-load-balancer-controller"

3. Verify "spec.volumes". You should find something as below, which is a temporary credentials to access AWS Services

--- **CHECK-1:** Verify "spec.volumes.name = aws-iam-token"

- name: aws-iam-token

projected:

defaultMode: 420

sources:

- serviceAccountToken:

audience: sts.amazonaws.com

expirationSeconds: 86400

path: token

--- **CHECK-2**: Verify Volume Mounts

volumeMounts:

- mountPath: /var/run/secrets/eks.amazonaws.com/serviceaccount

name: aws-iam-token

readOnly: true

--- **CHECK-3**: Verify ENVs whose path name is "token"

- name: AWS\_WEB\_IDENTITY\_TOKEN\_FILE

value: /var/run/secrets/eks.amazonaws.com/serviceaccount/token

---