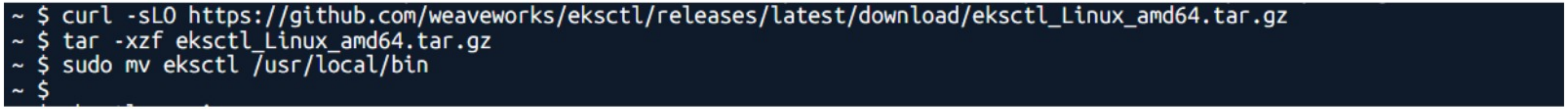
3. Deployment of Application using Kubernetes

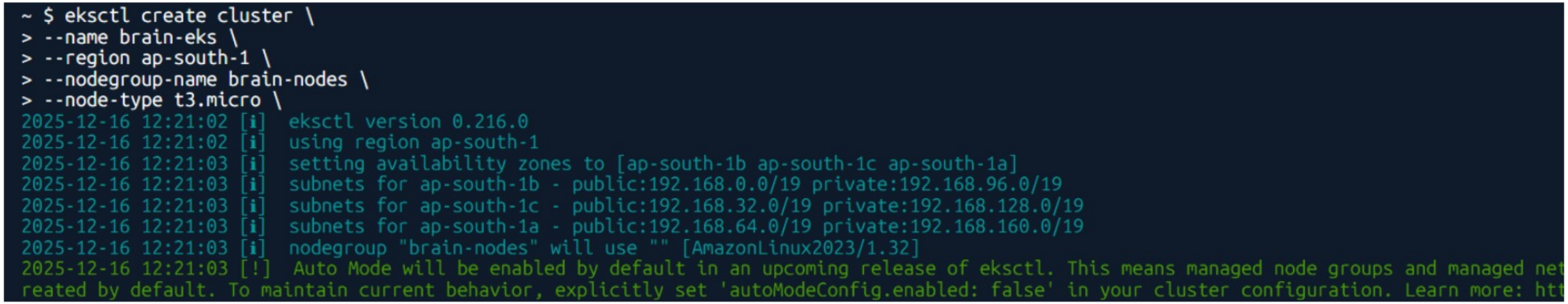
1. Install the eksctl



```
~ $ curl -sLO https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_Linux_amd64.tar.gz
~ $ tar -xzf eksctl_Linux_amd64.tar.gz
~ $ sudo mv eksctl /usr/local/bin
~ $
```

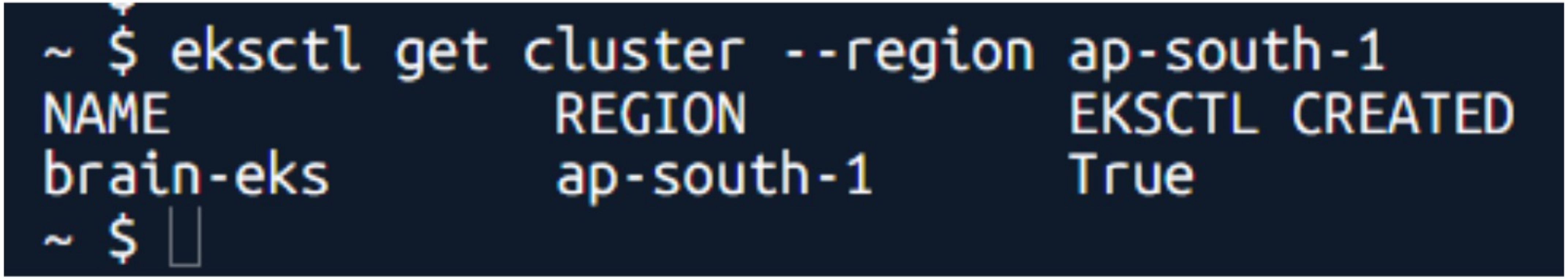
2. Check the status of eksctl

3. Create a kubernetes cluster



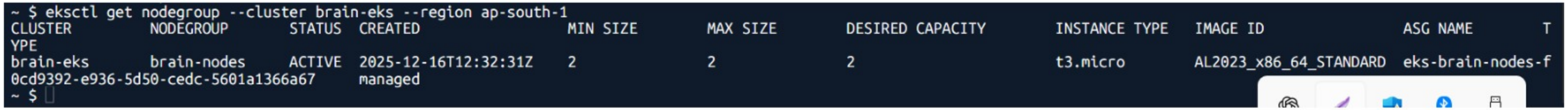
```
~ $ eksctl create cluster \
> --name brain-eks \
> --region ap-south-1 \
> --nodegroup-name brain-nodes \
> --node-type t3.micro \
2025-12-16 12:21:02 [ℹ]  eksctl version 0.216.0
2025-12-16 12:21:02 [ℹ]  using region ap-south-1
2025-12-16 12:21:03 [ℹ]  setting availability zones to [ap-south-1b ap-south-1c ap-south-1a]
2025-12-16 12:21:03 [ℹ]  subnets for ap-south-1b - public:192.168.0.0/19 private:192.168.96.0/19
2025-12-16 12:21:03 [ℹ]  subnets for ap-south-1c - public:192.168.32.0/19 private:192.168.128.0/19
2025-12-16 12:21:03 [ℹ]  subnets for ap-south-1a - public:192.168.64.0/19 private:192.168.160.0/19
2025-12-16 12:21:03 [ℹ]  nodegroup "brain-nodes" will use "" [AmazonLinux2023/1.32]
2025-12-16 12:21:03 [!]  Auto Mode will be enabled by default in an upcoming release of eksctl. This means managed node groups and managed net
reated by default. To maintain current behavior, explicitly set 'autoModeConfig.enabled: false' in your cluster configuration. Learn more: htt
```

4. Check cluster created or not



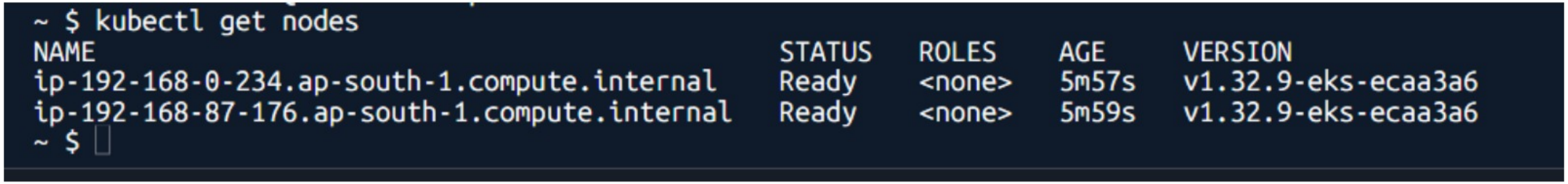
```
~ $ eksctl get cluster --region ap-south-1
NAME                REGION              EKSCTL CREATED
brain-eks           ap-south-1          True
~ $
```

5. Check the nodegroup status



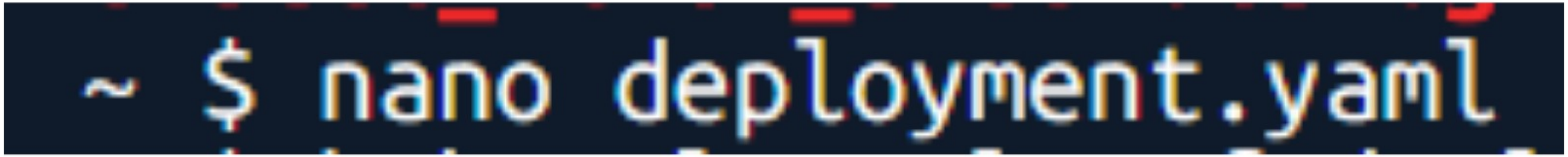
```
~ $ eksctl get nodegroup --cluster brain-eks --region ap-south-1
CLUSTER         NODEGROUP       STATUS  CREATED                 MIN SIZE        MAX SIZE        DESIRED CAPACITY        INSTANCE TYPE   IMAGE ID                ASG NAME        T
YPE
brain-eks       brain-nodes     ACTIVE  2025-12-16T12:32:31Z    2               2               2                       t3.micro        AL2023_x86_64_STANDARD  eks-brain-nodes-f
0cd9392-e936-5d50-cedc-5601a1366a67     managed
~ $ 
```

6. Check nodes status



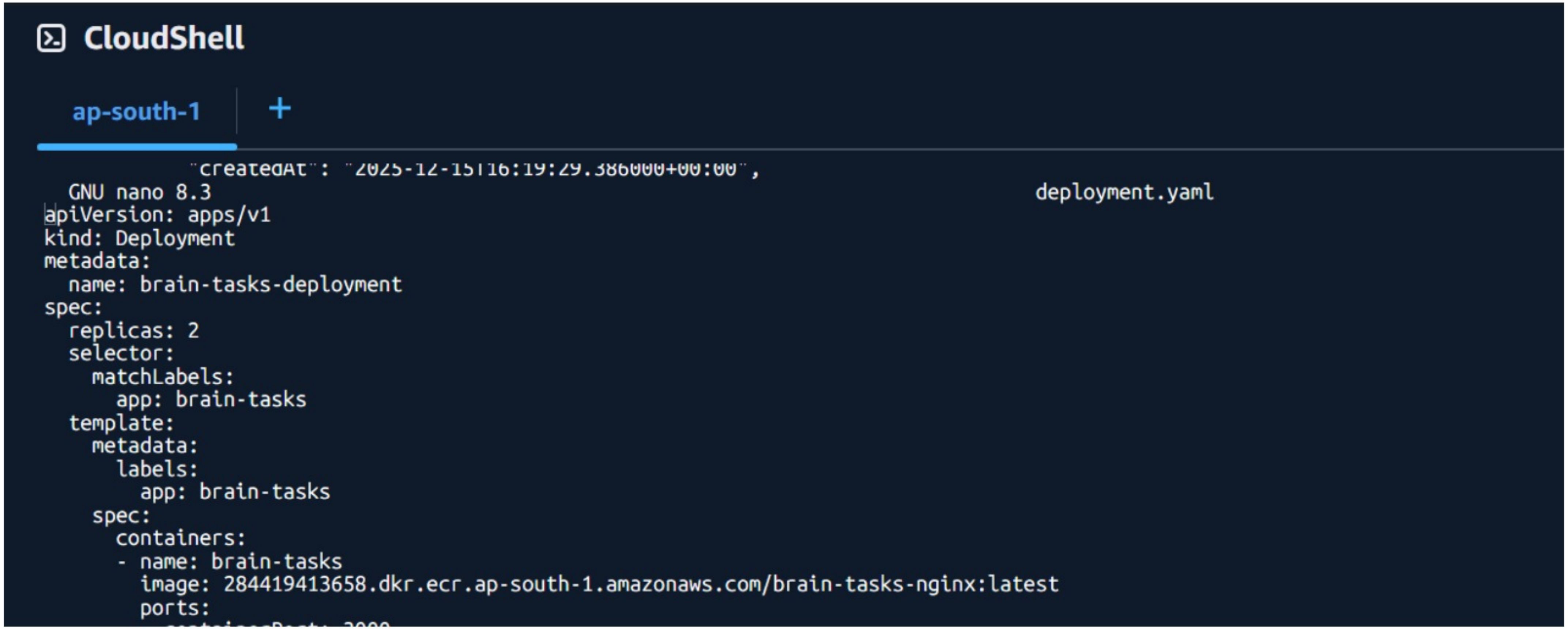
```
~ $ kubectl get nodes
NAME                                          STATUS   ROLES    AGE     VERSION
ip-192-168-0-234.ap-south-1.compute.internal   Ready    <none>   5m57s   v1.32.9-eks-ecaa3a6
ip-192-168-87-176.ap-south-1.compute.internal  Ready    <none>   5m59s   v1.32.9-eks-ecaa3a6
~ $ 
```

7. Create deployment file



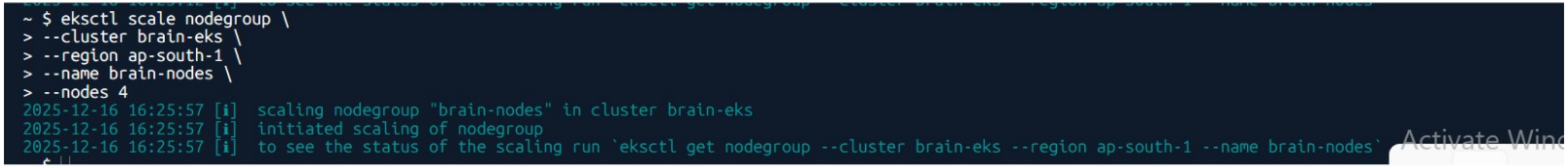
```
~ $ nano deployment.yaml
```

8. Write deployment code



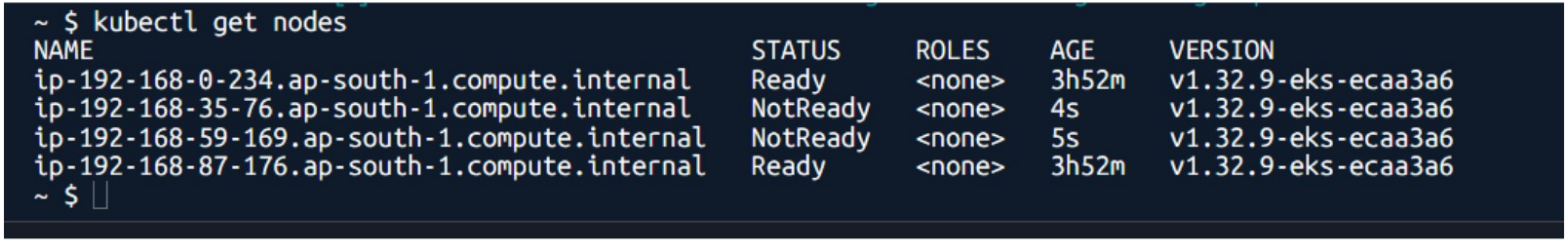
```
            "createdAt": "2025-12-15T16:19:29.386000+00:00",
  GNU nano 8.3                                                                   deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: brain-tasks-deployment
spec:
  replicas: 2
  selector:
    matchLabels:
      app: brain-tasks
  template:
    metadata:
      labels:
        app: brain-tasks
    spec:
      containers:
      - name: brain-tasks
        image: 284419413658.dkr.ecr.ap-south-1.amazonaws.com/brain-tasks-nginx:latest
        ports:
        - containerPort: 3000
```

9. Scale nodes in nodegroup



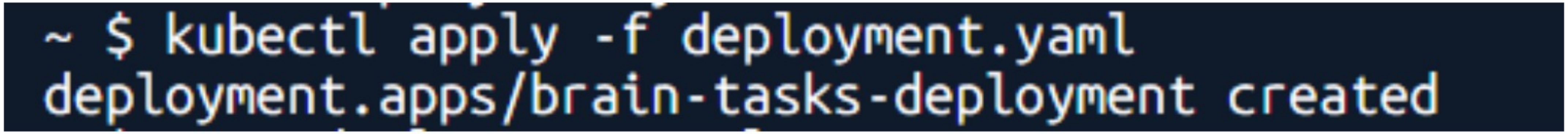
```
~ $ eksctl scale nodegroup \
> --cluster brain-eks \
> --region ap-south-1 \
> --name brain-nodes \
> --nodes 4
2025-12-16 16:25:57 [i]  scaling nodegroup "brain-nodes" in cluster brain-eks
2025-12-16 16:25:57 [i]  initiated scaling of nodegroup
2025-12-16 16:25:57 [i]  to see the status of the scaling run `eksctl get nodegroup --cluster brain-eks --region ap-south-1 --name brain-nodes`
```

10. Check the status of nodes



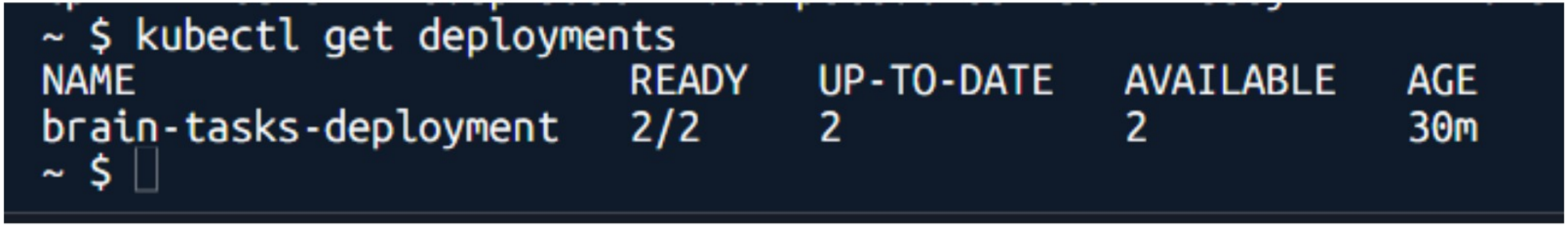
```
~ $ kubectl get nodes
NAME                                            STATUS     ROLES    AGE     VERSION
ip-192-168-0-234.ap-south-1.compute.internal    Ready      <none>   3h52m   v1.32.9-eks-ecaa3a6
ip-192-168-35-76.ap-south-1.compute.internal    NotReady   <none>   4s      v1.32.9-eks-ecaa3a6
ip-192-168-59-169.ap-south-1.compute.internal   NotReady   <none>   5s      v1.32.9-eks-ecaa3a6
ip-192-168-87-176.ap-south-1.compute.internal   Ready      <none>   3h52m   v1.32.9-eks-ecaa3a6
~ $ 
```

11. Apply deployment



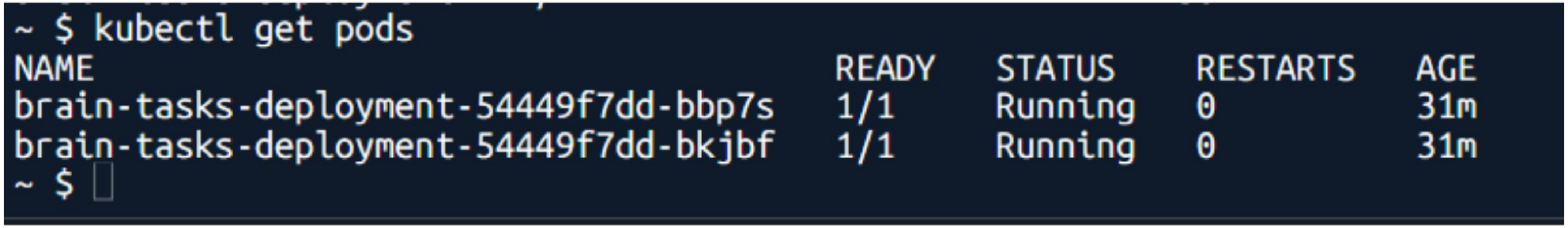
```
~ $ kubectl apply -f deployment.yaml
deployment.apps/brain-tasks-deployment created
```

12. Check deployment status



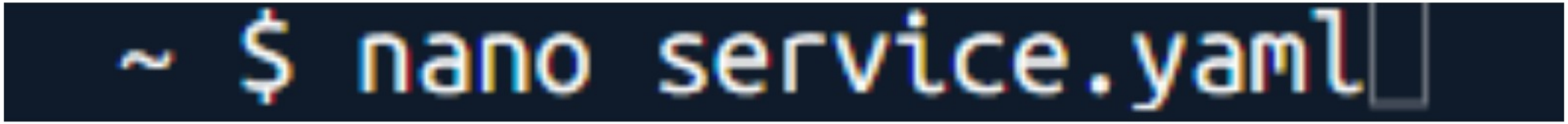
```
~ $ kubectl get deployments
NAME                     READY   UP-TO-DATE   AVAILABLE   AGE
brain-tasks-deployment   2/2     2            2           30m
~ $ 
```

13. Check pods running or not



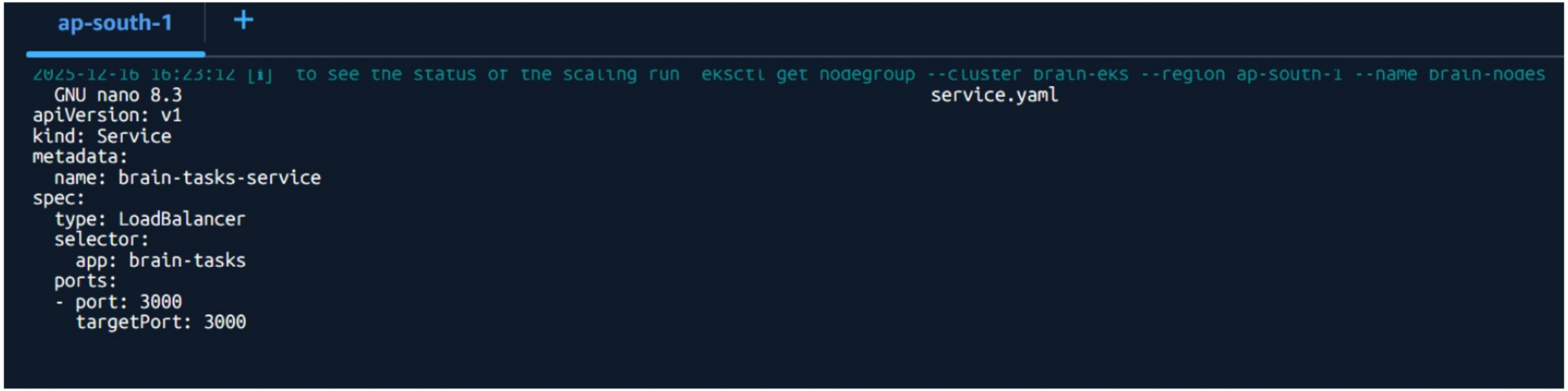
```
~ $ kubectl get pods
NAME                                     READY   STATUS    RESTARTS   AGE
brain-tasks-deployment-54449f7dd-bbp7s   1/1     Running   0          31m
brain-tasks-deployment-54449f7dd-bkjbf   1/1     Running   0          31m
~ $
```

14. Create service file



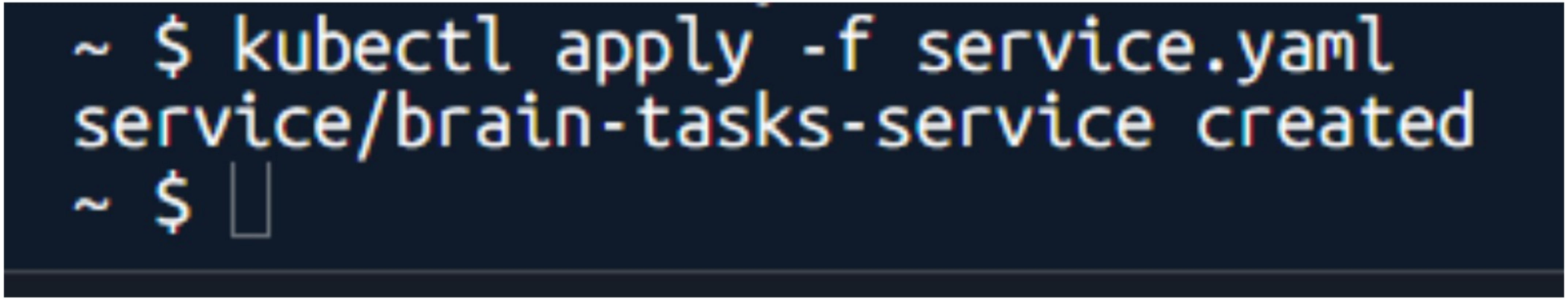
```
~ $ nano service.yaml
```

15. Write service configs



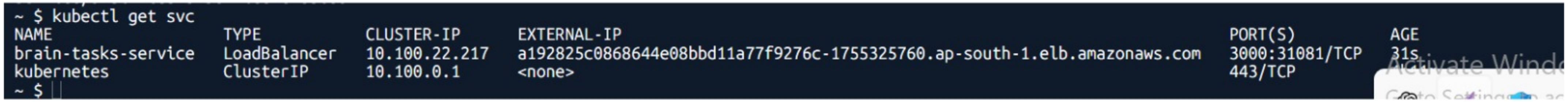
```
2025-12-16 16:23:12 [ℹ]  to see the status of the scaling run  eksctl get nodegroup --cluster brain-eks --region ap-south-1 --name brain-nodes
  GNU nano 8.3                                                                                service.yaml
apiVersion: v1
kind: Service
metadata:
  name: brain-tasks-service
spec:
  type: LoadBalancer
  selector:
    app: brain-tasks
  ports:
  - port: 3000
    targetPort: 3000
```

16. Apply service file



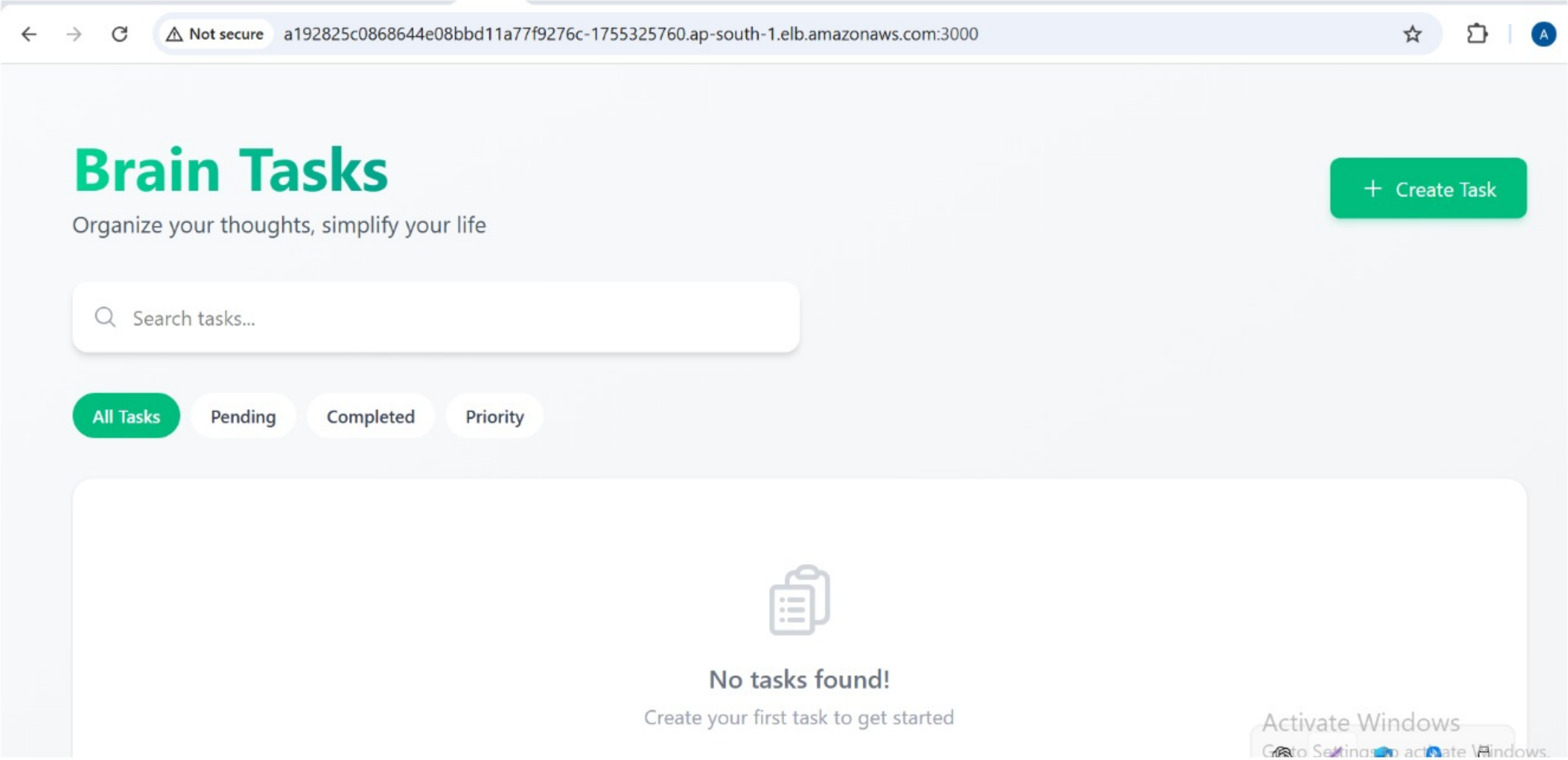
```
~ $ kubectl apply -f service.yaml
service/brain-tasks-service created
~ $ 
```

17. Check service running or not



```
~ $ kubectl get svc
NAME                  TYPE           CLUSTER-IP      EXTERNAL-IP                                                                PORT(S)          AGE
brain-tasks-service   LoadBalancer   10.100.22.217   a192825c0868644e08bbd11a77f9276c-1755325760.ap-south-1.elb.amazonaws.com   3000:31081/TCP   31s
kubernetes            ClusterIP      10.100.0.1      <none>                                                                     443/TCP
~ $
```

18. Here website successfully runing on port 3000



Not secure a192825c0868644e08bbd11a77f9276c-1755325760.ap-south-1.elb.amazonaws.com:3000

Brain Tasks

Organize your thoughts, simplify your life

No tasks found!

Create your first task to get started