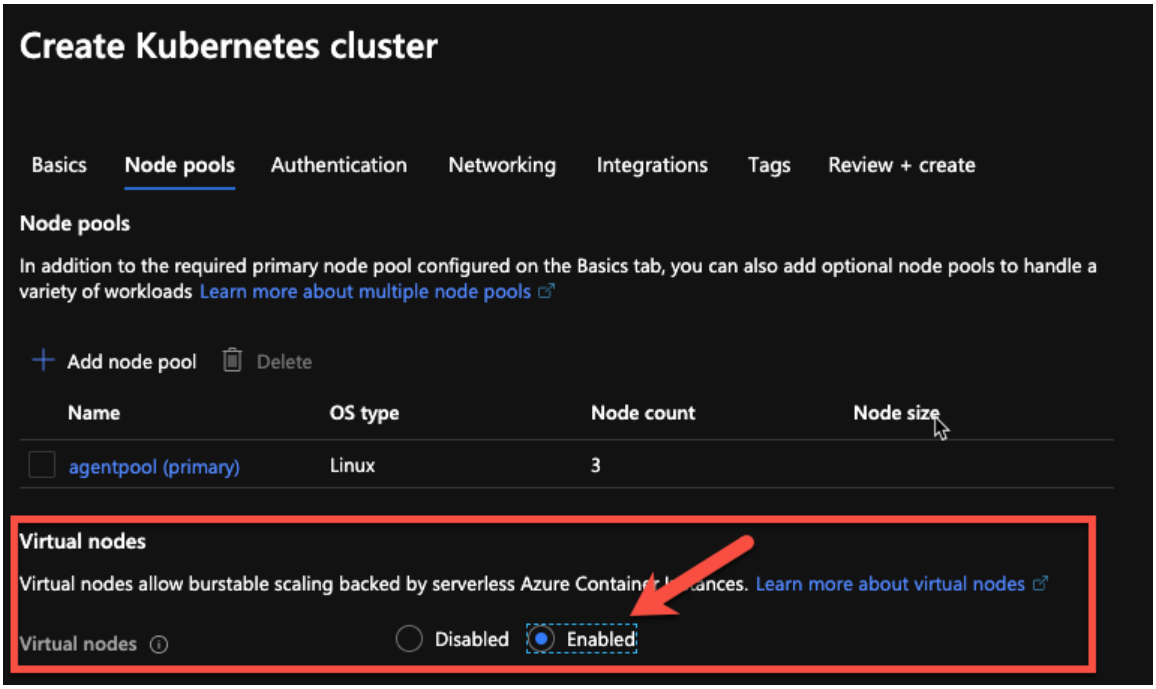
Bursting to ACI from AKS

Use the AKS virtual node to provision pods inside ACI that start in seconds. This enables AKS to run with just enough capacity for your average workload. As you run out of capacity in your AKS cluster, scale out additional pods in ACI without any additional servers to manage.

1. Ensure that you create an AKS cluster with Bursting enabled.
    1. If you use the web portal, ensure this option is checked off.



    2. If you are using the Azure CLI to enable bursting, run the following code:

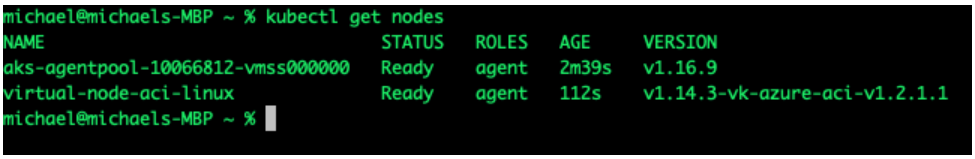
```
az aks install-connector --resource-group <myResourceGroup> --name <myK8sCluster> --connector-name myaciconnector
```

2. Pull down the Kubernetes configuration by running the following command:

```
az aks get-credentials --resource-group <myResourceGroup> --name <myK8sCluster> --admin
```

3. Run the following command to list the nodes. You will see that the ACI burst node is available.

```
kubectl get nodes
```



4. Next, you will need an application to run on AKS and burst into ACI. The application can be anything, so we'll choose Nginx-ingress. To install Nginx-ingress with Helm, use the following command:

```
helm install ingress stable/nginx-ingress --namespace kube-system
```

5. To see the Kubernetes service running, run the following command:

```
kubectl get services --namespace kube-system
```

6. To scale the nginx-ingress application up, run the following command:

```
kubectl scale deploy ingress-nginx-ingress-controller --replicas 10 --namespace kube-system
```

Even though we are using kubectl, the ACI Connector is dispatching pods to Azure Container Instances transparently, via the ACI connector node. This virtual node has unlimited capacity and a per-second billing model, making it perfect for burst compute scenarios like this one. If we wait a minute or so for the ACI containers to warm up, we should see image recognizer throughput increase dramatically.

7. Head back over to the Azure portal and go to the AKS cluster.

8. Once you are in the AKS cluster, click on Insights and go to containers. You will be able to see the newly/scaled containers running.