# Setting Up an EKS Cluster and SSH Gateway on Kubernetes: Step-by-Step Guide

This document provides a comprehensive guide for setting up an Amazon Elastic Kubernetes Service (EKS) cluster and configuring a Kubernetes-based SSH gateway for secure customer access. Each step is explained in detail to ensure successful implementation.

## Step 1: Create an EKS Cluster

Purpose: To set up a Kubernetes cluster on AWS to manage and deploy your SSH gateway.

### Steps:

* Create the cluster using `eksctl`:  
   eksctl create cluster --name ssh-customer-cluster --region us-east-1 --nodes 3  
   - Nodes: Number of EC2 instances that will run your Kubernetes pods.  
   - Region: AWS region where the cluster will reside.
* Verify the cluster status:  
   aws eks describe-cluster --name ssh-customer-cluster --query 'cluster.status'  
   - Ensure the cluster is active.
* Configure `kubectl` to connect to the cluster:  
   aws eks update-kubeconfig --region us-east-1 --name ssh-customer-cluster
* Verify connectivity:  
   kubectl get nodes

## Step 2: Create a Storage Class

Purpose: To define how Kubernetes dynamically provisions storage for customer data.

### Steps:

* Create the StorageClass YAML file as shown below:  
   apiVersion: storage.k8s.io/v1  
   kind: StorageClass  
   metadata:  
   name: standard  
   provisioner: ebs.csi.aws.com  
   parameters:  
   type: gp2  
   reclaimPolicy: Delete  
   volumeBindingMode: Immediate
* Apply the StorageClass:  
   kubectl apply -f storageclass.yaml
* Verify the StorageClass:  
   kubectl get storageclass