**Step 1:**

**Create role for access eks service**

IAM role attached to the EC2 instance with permissions to manage EKS (AmazonEKSClusterPolicy, AmazonEKSWorkerNodePolicy, AmazonEC2FullAccess, AmazonS3ReadOnlyAccess, etc.).

Login ec2 instance

**Prerequisites**

# Update package manager

sudo yum update -y

**# Install Ansible**

sudo amazon-linux-extras install ansible2 -y

**# Install AWS CLI**

curl "https://awscli.amazonaws.com/awscli-exe-linux-x86\_64.zip" -o "awscliv2.zip"

unzip awscliv2.zip

sudo ./aws/install

**# Install eksctl**

curl --silent --location "https://github.com/weaveworks/eksctl/releases/download/latest\_release/eksctl\_$(uname -s)\_amd64.tar.gz" | tar xz -C /tmp

sudo mv /tmp/eksctl /usr/local/bin

**# Install kubectl**

curl -LO "https://dl.k8s.io/release/$(curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt)/bin/linux/amd64/kubectl"

chmod +x ./kubectl

sudo mv ./kubectl /usr/local/bin

**ANSIBlE:**

**Create an inventory file**

[local]

localhost ansible\_connection=local

**Create the Ansible playbook to update the EKS cluster control plane and node groups:**

**Playbook.yml**

---

- hosts: localhost

  gather\_facts: no

  vars:

    cluster\_name: "my-eks-cluster"  # Change this to your EKS cluster name

    eksctl\_version: "0.190.0"       # Specify the desired eksctl version

    desired\_version: "1.30"         # The target Kubernetes version for the EKS cluster

    region: "us-east-1"             # The region where your EKS cluster is located

  tasks:

    # Upgrade EKS cluster control plane

    - name: Upgrade EKS control plane

      ansible.builtin.command: >

        eksctl upgrade cluster

        --name {{ cluster\_name }}

        --region {{ region }}

        --version {{ desired\_version }}

      register: control\_plane\_upgrade

      ignore\_errors: yes

    - name: Output control plane upgrade result

      debug:

        msg: "{{ control\_plane\_upgrade.stdout }}"

    # Get node groups for the cluster

    - name: Get nodegroups for the cluster

      ansible.builtin.command: eksctl get nodegroup --cluster {{ cluster\_name }} --region {{ region }}

      register: nodegroups

      ignore\_errors: yes

    - name: Output nodegroup information

      debug:

        msg: "{{ nodegroups.stdout }}"

    # Upgrade each nodegroup to the desired Kubernetes version

    - name: Upgrade nodegroups

      ansible.builtin.command: >

        eksctl upgrade nodegroup

        --cluster {{ cluster\_name }}

        --name {{ item }}

        --kubernetes-version {{ desired\_version }}

      loop: "{{ nodegroups.stdout\_lines | select('search', 'ng-') | list }}"

      ignore\_errors: yes

    - name: Verify cluster upgrade

      ansible.builtin.command: eksctl get cluster --name {{ cluster\_name }} --region {{ region }}

      register: cluster\_info

    - name: Output final cluster information

      debug:

        msg: "{{ cluster\_info.stdout }}"

**check the syntax is correct**

ansible-playbook -i dev playbook.yml –syntax-check

**Run playbook**

ansible-playbook -i dev playbook.yml

**upgrade eksctl**

eksctl upgrade cluster --name [my-eks-cluster](https://us-east-1.console.aws.amazon.com/eks/home?region=us-east-1#/clusters/my-eks-cluster) --region [**us-east-1**](https://us-east-1.console.aws.amazon.com/eks/home?region=us-east-1#/clusters)—approve

it will upgrade we can check eks in aws