Terraform EKS Module Documentation

Provider: AWS (`hashicorp/aws`)

1. Overview

This Terraform configuration deploys an **Amazon EKS cluster** with:

- Managed node groups (spot instances)
- IRSA (IAM Roles for Service Accounts) integration
- Core EKS add-ons (CoreDNS, kube-proxy, VPC-CNI)
- Amazon Managed Prometheus (AMP) for monitoring
- A service account (`ecr-puller`) with ECR read-only access

---

2. Module: EKS Cluster (`terraform-aws-modules/eks/aws`)

Configuration

Parameter	Description	Default/Value
`cluster_name`	Name of the EKS cluster	`mdu-aks-cluster` (from `var.eks_cluster_name`)
`cluster_version`	Kubernetes version	`1.27`
`vpc_id`	VPC ID for cluster security group	*Required* (`var.vpc_id`)
`control_plane_subnet_ids`	Subnets for EKS control plane	*Required* (`var.control_plane_subnet_ids`)
`subnet_ids`	Subnets for node groups	*Required* (`var.eks_node_groups_subnet_ids`)
`enable_irsa`	Enable IAM Roles for Service Accounts	`true`

`cluster_endpoint_*_access`	Public/private API endpoint access	Both enabled (`true`)

Managed Add-ons

- **CoreDNS** (preserved, most recent)
- **kube-proxy** (most recent)
- **VPC-CNI** (most recent)

Node Groups

Defined in `var.workers_config`:

- Instance type: `t3.large` (spot)
- Scaling: `min_size=1`, `max_size=2`, `desired_size=1`

---

3. IAM Integration (IRSA)

Resources

- **`kubernetes_service_account.ecr_puller`**
  - Name: `ecr-puller` (in `default` namespace)
  - IAM Role: `aws_iam_role.ecr_puller`
- **`aws_iam_role.ecr_puller`**
  - Trust policy: Allows EKS OIDC provider to assume role
  - Attached policy: `AmazonEC2ContainerRegistryReadOnly`

OIDC Provider

- ARN: `module.eks.oidc_provider_arn`
- Issuer URL: `module.eks.cluster_oidc_issuer_url`

---

4. Monitoring

- **Amazon Managed Prometheus (AMP)**
  - Workspace alias: `eks-monitoring`

---

5. Variables (`variables.tf`)

Variable	Type	Default	Description
`eks_cluster_name`	`string`	`"mdu-aks-cluster"`	EKS cluster name
`k8s_version`	`string`	`"1.27"`	Kubernetes version
`control_plane_subnet_ids`	`list(string)`	*Required*	Subnets for control plane
`eks_node_groups_subnet_ids`	`list(string)`	*Required*	Subnets for node groups
`vpc_id`	`string`	*Required*	VPC ID for security groups
`region`	`string`	`"us-east-1"`	AWS region
`workers_config`	`map(any)`	Spot `t3.large` x1	Node group configuration

---

6. Outputs (`outputs.tf`)

Output	Description
`cluster_arn`	ARN of the EKS cluster
`cluster_endpoint`	Kubernetes API endpoint
`cluster_certificate_authority_data`	Base64 CA cert for cluster auth
`cluster_oidc_issuer_url`	OIDC issuer URL for IRSA
`oidc_provider_arn`	ARN of the OIDC provider

---

7. Dependencies

- **AWS Provider**: Configured via `provider "aws"` (implied).
- **Kubernetes Provider**: Authenticates to EKS using `aws eks get-token`.

Usage Notes

1. Ensure `var.vpc_id`, `var.control_plane_subnet_ids`, and `var.eks_node_groups_subnet_ids` are provided.
2. The `ecr-puller` service account can pull images from ECR without hardcoded credentials.
3. AMP workspace (`eks-monitoring`) is created for Prometheus metrics.