Terraform AWS VPC Module Documentation

1. Overview

This Terraform configuration creates a **VPC with public and private subnets** across multiple AZs, including:

- Internet Gateway (IGW) for public subnets
- NAT Gateways for private subnet internet access
- Configurable subnet allocation and tagging
- DNS support and hostname enablement

2. Architecture

mermaid
Copy
graph TD
  VPC -->|CIDR| PublicSubnets
  VPC -->|CIDR| PrivateSubnets
  PublicSubnets --> IGW
  PrivateSubnets --> NAT
  NAT --> IGW

3. Module: VPC (`terraform-aws-modules/vpc/aws`)

Core Configuration

Parameter	Type	Default	Description
`name`	string	`"mdu-eks-vpc"`	VPC name
`cidr`	string	*Required*	Primary VPC CIDR (e.g., `10.0.0.0/16`)
`azs`	list(string)	First 3 AZs	Availability Zones for subnets
`private_subnets`	list(string)	3 subnets	Derived from VPC CIDR (slice 3-6)

`public_subnets`	list(string)	3 subnets	Derived from VPC CIDR (slice 0-3)

Network Services

Feature	Control Variable	Default	Notes
NAT Gateway	`enable_nat_gateway`	`true`	
Single NAT	`single_nat_gateway`	`false`	Conflicts with `one_nat_gateway_per_az`
AZ-distributed NAT	`one_nat_gateway_per_az`	`true`	
Internet Gateway	`create_igw`	`true`	Required for public subnets
DNS Support	`enable_dns_support`	`true`	
DNS Hostnames	`enable_dns_hostnames`	`true`	

CIDR Calculation Logic

hcl
Copy

```
locals {
  newbits = 8  # Creates /24 subnets from VPC CIDR
  netcount = 6 # Total subnets (3 public + 3 private)
  all_subnets = [for i in range(6) : cidrsubnet(var.vpc_cidr, 8, i)]
  public_subnets  = slice(local.all_subnets, 0, 3) # First 3 subnets
  private_subnets = slice(local.all_subnets, 3, 6) # Last 3 subnets
}
```

4. Variables Reference

Required Variables

Variable	Description	Example
`vpc_cidr`	Primary VPC CIDR block	`"10.0.0.0/16"`

Optional Variables

Variable	Type	Default	Description
`vpc_name`	string	`"mdu-eks-vpc"`	VPC name tag
`tags`	map(string)	`{}`	Base tags for all resources
`additional_public_subnet_tags`	map(string)	`{}`	Public subnet-specific tags
`additional_private_subnet_tags`	map(string)	`{}`	Private subnet-specific tags
`instance_tenancy`	string	`"default"`	`default` or `dedicated`

5. Outputs

Core Identifiers

Output	Description
`vpc_id`	Created VPC ID
`private_subnets`	List of private subnet IDs
`public_subnets`	List of public subnet IDs

Routing

Output	Description
`public_route_table_ids`	Public route table IDs
`private_route_table_ids`	Private route table IDs

NAT Components

Output	Description
`nat_ids`	Elastic IP allocation IDs for NAT

`nat_public_ips`	Public IPs of NAT Gateways
`natgw_ids`	NAT Gateway IDs

Internet Gateway

Output	Description
`igw_id`	Internet Gateway ID

6. Usage Example

hcl

```
module "network" {
  source = "./modules/vpc"

  vpc_cidr = "10.0.0.0/16"
  vpc_name = "prod-vpc"

  enable_nat_gateway     = true
  one_nat_gateway_per_az = true

  tags = {
    Environment = "production"
  }
}
```

7. Recommendations

1. **CIDR Planning**: Ensure `vpc_cidr` has enough space for all subnets (e.g., `/16` for 256 `/24` subnets)
2. **NAT Strategy**:
   - Use `one_nat_gateway_per_az` for high-availability
   - Use `single_nat_gateway` for cost savings
3. **Tagging**: Apply consistent tags for cost allocation and operations