VPC and Dynamic Subnets with availability Zone

```
chasrini@WKMZTA5DFE41 project % terraform plan
module.al-network.data.aws_availability_zones.available: Reading...
module.al-network.data.aws_availability_zones.available: Read
complete after 1s [id=us-east-1]
Terraform used the selected providers to generate the following
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

+ create

Terraform will perform the following actions:

```
# module.al-network.aws_subnet.vpc-al-subnets[0] will be created
 + resource "aws subnet" "vpc-a1-subnets" {
                                                        = (known
      + arn
after apply)
      + assign_ipv6_address_on_creation
                                                        = false
                                                        = "us-east-
      + availability_zone
1a"
      + availability_zone_id
                                                        = (known
after apply)
      + cidr block
"10.0.1.0/24"
      + enable dns64
                                                        = false
      + enable_resource_name_dns_a_record_on_launch
                                                        = false
      + enable_resource_name_dns_aaaa_record_on_launch = false
      + id
                                                        = (known
after apply)
      + ipv6_cidr_block_association_id
                                                        = (known
after apply)
      + ipv6_native
                                                        = false
                                                        = false
      + map_public_ip_on_launch
      + owner_id
                                                        = (known
after apply)
      + private_dns_hostname_type_on_launch
                                                        = (known
after apply)
      + tags all
                                                        = (known
after apply)
                                                        = (known
      + vpc id
after apply)
    }
 # module.al-network.aws_subnet.vpc-al-subnets[1] will be created
 + resource "aws_subnet" "vpc-a1-subnets" {
      + arn
                                                        = (known
after apply)
      + assign_ipv6_address_on_creation
                                                        = false
      + availability zone
                                                        = "us-east-
1b"
```

```
+ availability_zone_id
                                                        = (known
after apply)
      + cidr_block
"10.0.2.0/24"
      + enable dns64
                                                        = false
      + enable resource name dns a record on launch
                                                        = false
      + enable_resource_name_dns_aaaa_record_on_launch = false
                                                        = (known
after apply)
      + ipv6 cidr block association id
                                                        = (known
after apply)
      + ipv6_native
                                                        = false
      + map_public_ip_on_launch
                                                        = false
      + owner_id
                                                        = (known
after apply)
      + private_dns_hostname_type_on_launch
                                                        = (known
after apply)
      + tags_all
                                                        = (known
after apply)
                                                        = (known
      + vpc_id
after apply)
    }
 # module.al-network.aws_vpc.vpc-al will be created
  + resource "aws_vpc" "vpc-a1" {
                                              = (known after apply)
      + arn
      + cidr_block
                                              = "10.0.0.0/16"
      + default network acl id
                                             = (known after apply)
                                         = (known after apply)
= (known after apply)
      + default_route_table_id
      + default_security_group_id
      + dhcp options id
                                            = (known after apply)
                                              = (known after apply)
      + enable_dns_hostnames
      + enable dns support
                                              = true
      + enable_network_address_usage_metrics = (known after apply)
                                              = (known after apply)
                                              = "default"
      + instance_tenancy
      + ipv6_association_id
                                              = (known after apply)
      + ipv6_cidr_block
                                              = (known after apply)
      + ipv6_cidr_block_network_border_group = (known after apply)
      + main_route_table_id
                                              = (known after apply)
      + owner_id
                                              = (known after apply)
      + tags
                                              = {
          + "Environment" = "dev"
                          = "my-vpc"
          + "Name"
        }
      + tags_all
                                              = {
          + "Environment" = "dev"
          + "Name"
                          = "my-vpc"
        }
    }
```

Plan: 3 to add, 0 to change, 0 to destroy.

Terraform Workspaces

```
chasrini@WKMZTA5DFE41 project % terraform workspace list
  default
* dev
  prod
  test
```

Only one subnet for dev workspace (output below)

hasrini@WKMZTA5DFE41 project % terraform workspace select dev Switched to workspace "dev".

```
chasrini@WKMZTA5DFE41 project % terraform plan module.a1-network.data.aws_availability_zones.available: Reading... module.a1-network.data.aws_availability_zones.available: Read complete after 2s [id=us-east-1]
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

+ create

Terraform will perform the following actions:

```
# module.a1-network.aws subnet.vpc-a1-subnets[0] will be created
+ resource "aws_subnet" "vpc-a1-subnets" {
  + arn
                              = (known after apply)
  + assign ipv6 address on creation
                                            = false
  + availability_zone
                                   = "us-east-1a"
  + availability zone id
                                     = (known after apply)
                                 = "10.0.1.0/24"
  + cidr block
  + enable dns64
                                   = false
  + enable_resource_name_dns_a_record_on_launch = false
  + enable resource name dns aaaa record on launch = false
  + id
                             = (known after apply)
  + ipv6 cidr block association id
                                          = (known after apply)
                                 = false
  + ipv6_native
  + map_public_ip_on_launch
                                          = false
                                 = (known after apply)
  + owner id
  + private_dns_hostname_type_on_launch
                                                = (known after apply)
  + tags all
                               = (known after apply)
  + vpc id
                               = (known after apply)
```

module.a1-network.aws_vpc.vpc-a1 will be created

```
+ resource "aws vpc" "vpc-a1" {
                         = (known after apply)
  + arn
  + cidr block
                            = "10.0.0.0/16"
  + default network acl id
                                  = (known after apply)
  + default route table id
                                 = (known after apply)
  + default security group id
                                   = (known after apply)
  + dhcp options id
                               = (known after apply)
  + enable dns hostnames
                                   = (known after apply)
  + enable dns support
                                 = true
  + enable network address usage metrics = (known after apply)
                        = (known after apply)
  + id
                               = "default"
  + instance tenancy
                                = (known after apply)
  + ipv6_association_id
  + ipv6 cidr block
                              = (known after apply)
  + ipv6 cidr block network border group = (known after apply)
  + main route table id
                                 = (known after apply)
                            = (known after apply)
  + owner id
  + tags
    + "Environment" = "dev"
    + "Name"
                 = "my-vpc"
   }
  + tags all
    + "Environment" = "dev"
    + "Name"
                 = "my-vpc"
   }
}
```

Plan: 2 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.

chasrini@WKMZTA5DFE41 project %

Two subnets for dev workspace (output below)

```
chasrini@WKMZTA5DFE41 project % terraform workspace select prod Switched to workspace "prod". chasrini@WKMZTA5DFE41 project % terraform plan module.al-network.data.aws_availability_zones.available: Reading... module.al-network.data.aws_availability_zones.available: Read complete after 2s [id=us-east-1]
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

+ create

Terraform will perform the following actions:

```
# module.al-network.aws_subnet.vpc-al-subnets[0] will be created
 + resource "aws subnet" "vpc-a1-subnets" {
                                                       = (known
after apply)
      + assign_ipv6_address_on_creation
                                                       = false
                                                       = "us-east-
      + availability_zone
                                                       = (known
      + availability_zone_id
after apply)
      + cidr block
"10.0.1.0/24"
      + enable dns64
                                                        = false
                                                      = false
      + enable_resource_name_dns_a_record_on_launch
      + enable resource name dns aaaa record on launch = false
                                                       = (known
after apply)
      + ipv6_cidr_block_association_id
                                                       = (known
after apply)
      + ipv6_native
                                                       = false
      + map_public_ip_on_launch
                                                       = false
      + owner id
                                                       = (known
after apply)
      + private_dns_hostname_type_on_launch
                                                       = (known
after apply)
      + tags_all
                                                       = (known
after apply)
                                                       = (known
      + vpc id
after apply)
    }
 # module.al-network.aws_subnet.vpc-al-subnets[1] will be created
 + resource "aws subnet" "vpc-a1-subnets" {
      + arn
                                                       = (known
after apply)
      + assign_ipv6_address_on_creation
                                                       = false
                                                       = "us-east-
      + availability_zone
1b"
      + availability zone id
                                                       = (known
after apply)
      + cidr_block
"10.0.2.0/24"
      + enable_dns64
                                                       = false
      + enable_resource_name_dns_a_record_on_launch
                                                       = false
      + enable_resource_name_dns_aaaa_record_on_launch = false
                                                       = (known
after apply)
```

```
+ ipv6_cidr_block_association_id
                                                            = (known
after apply)
                                                            = false
      + ipv6_native
      + map_public_ip_on_launch
                                                            = false
                                                            = (known
      + owner_id
after apply)
      + private_dns_hostname_type_on_launch
                                                            = (known
after apply)
      + tags all
                                                            = (known
after apply)
      + vpc_id
                                                            = (known
after apply)
    }
  # module.al-network.aws_vpc.vpc-al will be created
  + resource "aws_vpc" "vpc-a1" {
      + arn
                                                 = (known after apply)
      + cidr_block
                                                 = "10.0.0.0/16"
                                       = 10.0.0/16"
= (known after apply)
= true
      + default_network_acl_id
+ default_route_table_id
      + default_security_group_id
      + dhcp options id
      + enable_dns_hostnames
      + enable_dns_support
      + enable_network_address_usage_metrics = (known after apply)
                                                = (known after apply)
                                                = "default"
      + instance_tenancy
      + ipv6_association_id
                                                 = (known after apply)
      + ipv6_cidr_block
                                                 = (known after apply)
      + ipv6_cidr_block_network_border_group = (known after apply)
                                               = (known after apply)
      + main route table id
      + owner_id
                                                 = (known after apply)
      + tags
                                                 = {
           + "Environment" = "prod"
           + "Name" = "mv-vpc"
        }
      + tags_all
                                                 = {
          + "Environment" = "prod"
          + "Name" = "my-vpc"
    }
```

Plan: 3 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.

chasrini@WKMZTA5DFE41 project %

chasrini@WKMZTA5DFE41 project % terraform output
vpc_id = "vpc-0cfd71f818f8f4066"
chasrini@WKMZTA5DFE41 project % terraform output vpc_id
"vpc-0cfd71f818f8f4066"

Assessment 2

```
terraform plan
```

```
Acquiring state lock. This may take a few moments... module.a2-network.data.aws_availability_zones.available: Reading... module.a2-network.data.aws_availability_zones.available: Read complete after 1s [id=us-east-1]
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

+ create

Terraform will perform the following actions:

```
# module.myec2s.aws instance.appserver will be created
+ resource "aws_instance" "appserver" {
                         = "ami-0230bd60aa48260c6"
  + ami
  + arn
                         = (known after apply)
                                     = (known after apply)
  + associate public ip address
  + availability zone
                              = (known after apply)
                                = (known after apply)
  + cpu core count
  + cpu_threads_per_core
                                   = (known after apply)
  + disable api stop
                               = (known after apply)
  + disable api termination
                                   = (known after apply)
                               = (known after apply)
  + ebs optimized
                                 = false
  + get password data
  + host id
                           = (known after apply)
  + host resource_group_arn
                                    = (known after apply)
                                 = (known after apply)
  + iam instance profile
  + id
                        = (known after apply)
  + instance_initiated_shutdown_behavior = (known after apply)
  + instance lifecycle
                               = (known after apply)
                              = (known after apply)
  + instance state
                              = "t2.micro"
  + instance type
                                 = (known after apply)
  + ipv6 address count
  + ipv6_addresses
                               = (known after apply)
  + key name
                             = (known after apply)
  + monitoring
                             = (known after apply)
  + outpost arn
                             = (known after apply)
  + password_data
                               = (known after apply)
  + placement group
                                 = (known after apply)
  + placement partition number
                                      = (known after apply)
  + primary network interface id
                                      = (known after apply)
  + private dns
                             = (known after apply)
  + private ip
                            = (known after apply)
  + public_dns
                             = (known after apply)
```

```
+ public ip
                            = (known after apply)
  + secondary private ips
                                  = (known after apply)
  + security_groups
                               = (known after apply)
  + source_dest_check
                                 = true
  + spot instance request id
                                    = (known after apply)
  + subnet id
                            = (known after apply)
                         = {
  + tags
    + "Environment" = "prod"
    + "Name"
                  = "appserver"
    + "Terraform" = "true"
   }
  + tags all
    + "Environment" = "prod"
                  = "appserver"
    + "Name"
    + "Terraform" = "true"
   }
                            = (known after apply)
  + tenancy
  + user_data
                            = (known after apply)
  + user_data_base64
                                 = (known after apply)
  + user data replace on change
                                       = false
  + vpc_security_group_ids
                                   = (known after apply)
 }
# module.myec2s.aws instance.dbserver will be created
+ resource "aws_instance" "dbserver" {
  + ami
                         = "ami-0230bd60aa48260c6"
  + arn
                         = (known after apply)
  + associate public ip address
                                     = (known after apply)
  + availability zone
                               = (known after apply)
  + cpu core count
                                = (known after apply)
  + cpu_threads_per_core
                                   = (known after apply)
  + disable api stop
                               = (known after apply)
  + disable_api_termination
                                   = (known after apply)
  + ebs optimized
                               = (known after apply)
  + get password data
                                  = false
  + host id
                           = (known after apply)
  + host resource group arn
                                    = (known after apply)
  + iam instance profile
                                 = (known after apply)
  + id
                        = (known after apply)
  + instance initiated shutdown behavior = (known after apply)
                               = (known after apply)
  + instance lifecycle
                              = (known after apply)
  + instance state
                              = "t2.micro"
  + instance type
                                 = (known after apply)
  + ipv6_address_count
  + ipv6_addresses
                               = (known after apply)
  + key name
                             = (known after apply)
  + monitoring
                             = (known after apply)
```

```
+ outpost arn
                             = (known after apply)
  + password data
                               = (known after apply)
  + placement_group
                                 = (known after apply)
  + placement_partition_number
                                      = (known after apply)
  + primary_network_interface_id
                                      = (known after apply)
                             = (known after apply)
  + private dns
                            = (known after apply)
  + private ip
                             = (known after apply)
  + public dns
                           = (known after apply)
  + public ip
  + secondary_private_ips
                                  = (known after apply)
  + security groups
                               = (known after apply)
  + source dest check
                                 = true
  + spot_instance_request_id
                                    = (known after apply)
  + subnet id
                            = (known after apply)
  + tags
    + "Environment" = "prod"
                 = "dbserver"
    + "Name"
    + "Terraform" = "true"
   }
  + tags all
    + "Environment" = "prod"
    + "Name"
                 = "dbserver"
    + "Terraform" = "true"
   }
  + tenancy
                           = (known after apply)
  + user data
                            = (known after apply)
                                 = (known after apply)
  + user_data_base64
  + user data replace on change
                                       = false
  + vpc security group ids
                                   = (known after apply)
 }
# module.myec2s.aws instance.webserver will be created
+ resource "aws instance" "webserver" {
  + ami
                         = "ami-0230bd60aa48260c6"
  + arn
                         = (known after apply)
  + associate_public_ip_address
                                     = (known after apply)
  + availability zone
                               = (known after apply)
                                = (known after apply)
  + cpu core count
  + cpu_threads_per_core
                                   = (known after apply)
  + disable api stop
                               = (known after apply)
  + disable api termination
                                   = (known after apply)
  + ebs_optimized
                               = (known after apply)
                                 = false
  + get password data
                           = (known after apply)
  + host id
  + host_resource_group_arn
                                    = (known after apply)
  + iam instance profile
                                 = (known after apply)
  + id
                        = (known after apply)
```

```
+ instance_initiated_shutdown_behavior = (known after apply)
  + instance lifecycle
                               = (known after apply)
  + instance_state
                              = (known after apply)
  + instance_type
                              = "t2.micro"
  + ipv6_address_count
                                 = (known after apply)
  + ipv6 addresses
                               = (known after apply)
  + key_name
                             = (known after apply)
  + monitoring
                             = (known after apply)
                              = (known after apply)
  + outpost_arn
                               = (known after apply)
  + password_data
  + placement group
                                 = (known after apply)
  + placement_partition_number
                                      = (known after apply)
  + primary_network_interface_id
                                      = (known after apply)
  + private dns
                             = (known after apply)
  + private ip
                            = (known after apply)
  + public_dns
                             = (known after apply)
  + public ip
                            = (known after apply)
                                  = (known after apply)
  + secondary_private_ips
  + security_groups
                               = (known after apply)
                                 = true
  + source_dest_check
  + spot_instance_request_id
                                    = (known after apply)
  + subnet id
                             = (known after apply)
  + tags
                         = {
    + "Environment" = "prod"
    + "Name"
                 = "webserver"
    + "Terraform" = "true"
   }
  + tags all
                           = {
    + "Environment" = "prod"
    + "Name"
                 = "webserver"
    + "Terraform" = "true"
   }
                            = (known after apply)
  + tenancy
                            = (known after apply)
  + user data
  + user_data_base64
                                 = (known after apply)
  + user_data_replace_on_change
                                       = false
                                   = (known after apply)
  + vpc_security_group_ids
 }
# module.myec2s.aws_security_group.sg_mytraffic will be created
+ resource "aws_security_group" "sg_mytraffic" {
                  = (known after apply)
  + arn
                     = "Managed by Terraform"
  + description
                   = (known after apply)
  + egress
  + id
                 = (known after apply)
  + ingress
                   = [
    + {
```

```
+ cidr blocks
        + "0.0.0.0/0",
                      = ""
      + description
      + from port
                      = 80
      + ipv6 cidr blocks = []
      + prefix list ids = []
      + protocol
                     = "tcp"
      + security_groups = []
                = false
      + self
                    = 80
      + to port
     },
  1
  + name
                   = "sg allow webserver traffic"
  + name prefix
                      = (known after apply)
  + owner_id
                     = (known after apply)
  + revoke rules on delete = false
                  = (known after apply)
  + tags_all
                   = (known after apply)
  + vpc id
 }
# module.a2-network.module.vpc.aws_db_subnet_group.database[0] will be created
+ resource "aws_db_subnet_group" "database" {
  + arn
                  = (known after apply)
  + description
                     = "Database subnet group for a2-vpc"
  + id
                 = (known after apply)
                   = "a2-vpc"
  + name
                      = (known after apply)
  + name prefix
                     = (known after apply)
  + subnet ids
  + supported network types = (known after apply)
  + tags
    + "Name" = "a2-vpc"
   }
  + tags all
    + "Name" = "a2-vpc"
   }
                   = (known after apply)
  + vpc_id
 }
# module.a2-network.module.vpc.aws default network acl.this[0] will be created
+ resource "aws default network acl" "this" {
  + arn
                 = (known after apply)
  + default network acl id = (known after apply)
  + id
                 = (known after apply)
  + owner_id
                     = (known after apply)
                 = {
  + tags
    + "Name" = "a2-vpc-default"
```

```
}
  + tags all = {
    + "Name" = "a2-vpc-default"
   }
                   = (known after apply)
  + vpc id
  + egress {
                 = "allow"
    + action
    + from_port
                   = 0
    + ipv6_cidr_block = "::/0"
    + protocol = "-1"
                  = 101
    + rule no
                 = 0
    + to_port
   }
  + egress {
    + action = "allow"
    + cidr_block = "0.0.0.0/0"
    + from_port = 0
    + protocol = "-1"
    + rule_no = 100
    + to_port = 0
   }
  + ingress {
    + action
                 = "allow"
    + from_port = 0
    + ipv6_cidr_block = "::/0"
    + protocol
               = "-1"
    + rule no
                 = 101
                 = 0
    + to port
   }
  + ingress {
    + action = "allow"
    + cidr block = "0.0.0.0/0"
    + from port = 0
    + protocol = "-1"
    + rule no = 100
    + to port = 0
   }
}
# module.a2-network.module.vpc.aws_default_route_table.default[0] will be created
+ resource "aws default route table" "default" {
                 = (known after apply)
  + default_route_table_id = (known after apply)
  + id
                = (known after apply)
  + owner id
                    = (known after apply)
```

```
= (known after apply)
  + route
  + tags
                 = {
    + "Name" = "a2-vpc-default"
  + tags all = {
    + "Name" = "a2-vpc-default"
                  = (known after apply)
  + vpc id
  + timeouts {
    + create = "5m"
    + update = "5m"
   }
}
# module.a2-network.module.vpc.aws_default_security_group.this[0] will be created
+ resource "aws_default_security_group" "this" {
  + arn
               = (known after apply)
                    = (known after apply)
  + description
                  = (known after apply)
  + egress
  + id
                = (known after apply)
  + ingress
                  = (known after apply)
  + name
                 = (known after apply)
                    = (known after apply)
  + name prefix
  + owner_id
                    = (known after apply)
  + revoke rules on delete = false
  + tags
                = {
    + "Name" = "a2-vpc-default"
   }
  + tags all
    + "Name" = "a2-vpc-default"
                  = (known after apply)
  + vpc_id
 }
# module.a2-network.module.vpc.aws_internet_gateway.this[0] will be created
+ resource "aws internet gateway" "this" {
         = (known after apply)
  + id
         = (known after apply)
  + owner id = (known after apply)
  + tags = {
    + "Name" = "a2-vpc"
   }
  + tags all = {
    + "Name" = "a2-vpc"
   }
  + vpc id = (known after apply)
```

```
}
# module.a2-network.module.vpc.aws route.public internet gateway[0] will be created
+ resource "aws_route" "public_internet_gateway" {
  + destination cidr block = "0.0.0.0/0"
  + gateway id
                      = (known after apply)
  + id
                 = (known after apply)
                     = (known after apply)
  + instance id
  + instance_owner_id
                         = (known after apply)
  + network interface id = (known after apply)
                  = (known after apply)
  + origin
  + route table id
                       = (known after apply)
                  = (known after apply)
  + state
  + timeouts {
    + create = "5m"
   }
 }
# module.a2-network.module.vpc.aws route table.private[0] will be created
+ resource "aws route table" "private" {
  + arn
              = (known after apply)
  + id
             = (known after apply)
                 = (known after apply)
  + owner id
  + propagating_vgws = (known after apply)
  + route
               = (known after apply)
  + tags
               = {
    + "Name" = "a2-vpc-private"
   }
  + tags all
    + "Name" = "a2-vpc-private"
  + vpc_id
                = (known after apply)
 }
# module.a2-network.module.vpc.aws_route_table.public[0] will be created
+ resource "aws_route_table" "public" {
              = (known after apply)
  + arn
  + id
             = (known after apply)
  + owner id
                 = (known after apply)
  + propagating vgws = (known after apply)
  + route
               = (known after apply)
              = {
  + tags
    + "Name" = "a2-vpc-public"
  + tags all
                = {
    + "Name" = "a2-vpc-public"
```

```
}
  + vpc id
               = (known after apply)
# module.a2-network.module.vpc.aws route table association.database[0] will be created
+ resource "aws route table association" "database" {
  + id
            = (known after apply)
  + route table id = (known after apply)
  + subnet id = (known after apply)
}
# module.a2-network.module.vpc.aws route table association.database[1] will be created
+ resource "aws_route_table_association" "database" {
  + id
            = (known after apply)
  + route table id = (known after apply)
  + subnet_id = (known after apply)
}
# module.a2-network.module.vpc.aws route table association.private[0] will be created
+ resource "aws route table association" "private" {
            = (known after apply)
  + route table id = (known after apply)
  + subnet id = (known after apply)
}
# module.a2-network.module.vpc.aws route table association.private[1] will be created
+ resource "aws_route_table_association" "private" {
  + id
            = (known after apply)
  + route table id = (known after apply)
  + subnet id = (known after apply)
}
# module.a2-network.module.vpc.aws_route_table_association.public[0] will be created
+ resource "aws route table association" "public" {
            = (known after apply)
  + route table id = (known after apply)
  + subnet id = (known after apply)
}
# module.a2-network.module.vpc.aws route table association.public[1] will be created
+ resource "aws route table association" "public" {
  + id
            = (known after apply)
  + route table id = (known after apply)
  + subnet id = (known after apply)
 }
```

module.a2-network.module.vpc.aws subnet.database[0] will be created

```
+ resource "aws subnet" "database" {
                              = (known after apply)
  + arn
  + assign_ipv6_address_on_creation
                                            = false
  + availability zone
                                   = "us-east-1a"
                                    = (known after apply)
  + availability zone id
  + cidr block
                                 = "172.20.21.0/24"
  + enable dns64
                                   = false
  + enable resource name dns a record on launch = false
  + enable_resource_name_dns_aaaa_record_on_launch = false
  + id
                             = (known after apply)
  + ipv6 cidr block association id
                                          = (known after apply)
                                 = false
  + ipv6 native
  + map_public_ip_on_launch
                                         = false
  + owner id
                                 = (known after apply)
                                            = (known after apply)
  + private dns hostname type on launch
                              = {
    + "Name" = "a2-vpc-db-us-east-1a"
   }
  + tags all
                               = {
    + "Name" = "a2-vpc-db-us-east-1a"
   }
  + vpc_id
                               = (known after apply)
 }
# module.a2-network.module.vpc.aws subnet.database[1] will be created
+ resource "aws subnet" "database" {
  + arn
                              = (known after apply)
  + assign ipv6 address on creation
                                            = false
                                   = "us-east-1b"
  + availability zone
  + availability zone id
                                    = (known after apply)
                                 = "172.20.22.0/24"
  + cidr_block
  + enable dns64
                                   = false
  + enable_resource_name_dns_a_record_on_launch = false
  + enable resource name dns aaaa record on launch = false
  + id
                             = (known after apply)
  + ipv6_cidr_block_association_id
                                          = (known after apply)
  + ipv6 native
                                 = false
                                         = false
  + map public ip on launch
  + owner id
                                 = (known after apply)
  + private dns hostname type on launch = (known after apply)
    + "Name" = "a2-vpc-db-us-east-1b"
   }
  + tags all
    + "Name" = "a2-vpc-db-us-east-1b"
   }
                               = (known after apply)
  + vpc id
```

```
}
# module.a2-network.module.vpc.aws subnet.private[0] will be created
+ resource "aws_subnet" "private" {
  + arn
                              = (known after apply)
  + assign ipv6 address on creation
                                            = false
                                   = "us-east-1a"
  + availability zone
                                     = (known after apply)
  + availability zone id
                                 = "172.20.11.0/24"
  + cidr_block
                                    = false
  + enable dns64
  + enable resource name dns a record on launch = false
  + enable resource name dns aaaa record on launch = false
  + id
                             = (known after apply)
  + ipv6 cidr block association id
                                          = (known after apply)
  + ipv6 native
                                  = false
  + map_public_ip_on_launch
                                          = false
                                 = (known after apply)
  + owner id
  + private_dns_hostname_type_on_launch
                                              = (known after apply)
  + tags
                              = {
    + "Name" = "a2-vpc-private-us-east-1a"
   }
  + tags all
                                = {
    + "Name" = "a2-vpc-private-us-east-1a"
   }
  + vpc_id
                                = (known after apply)
 }
# module.a2-network.module.vpc.aws subnet.private[1] will be created
+ resource "aws subnet" "private" {
  + arn
                              = (known after apply)
  + assign_ipv6_address_on_creation
                                            = false
  + availability_zone
                                   = "us-east-1b"
  + availability_zone_id
                                     = (known after apply)
  + cidr block
                                 = "172.20.12.0/24"
  + enable dns64
                                    = false
  + enable_resource_name_dns_a_record_on_launch = false
  + enable resource name dns aaaa record on launch = false
                             = (known after apply)
  + ipv6_cidr_block_association_id
                                          = (known after apply)
  + ipv6 native
                                  = false
                                          = false
  + map public ip on launch
  + owner id
                                 = (known after apply)
  + private dns hostname type on launch
                                             = (known after apply)
    + "Name" = "a2-vpc-private-us-east-1b"
   }
  + tags_all
                                = {
```

```
+ "Name" = "a2-vpc-private-us-east-1b"
   }
                                = (known after apply)
  + vpc_id
 }
# module.a2-network.module.vpc.aws subnet.public[0] will be created
+ resource "aws_subnet" "public" {
                              = (known after apply)
  + arn
  + assign ipv6 address on creation
                                            = false
                                   = "us-east-1a"
  + availability zone
  + availability zone id
                                     = (known after apply)
  + cidr block
                                 = "172.20.1.0/24"
                                    = false
  + enable_dns64
  + enable resource name dns a record on launch = false
  + enable resource name dns aaaa record on launch = false
  + id
                             = (known after apply)
  + ipv6 cidr block association id
                                          = (known after apply)
  + ipv6_native
                                  = false
  + map public ip on launch
                                          = false
                                 = (known after apply)
  + owner id
  + private_dns_hostname_type_on_launch
                                               = (known after apply)
                              = {
  + tags
    + "Name" = "a2-vpc-public-us-east-1a"
   }
  + tags all
    + "Name" = "a2-vpc-public-us-east-1a"
  + vpc id
                               = (known after apply)
 }
# module.a2-network.module.vpc.aws_subnet.public[1] will be created
+ resource "aws subnet" "public" {
  + arn
                              = (known after apply)
  + assign ipv6 address on creation
                                            = false
                                   = "us-east-1b"
  + availability zone
  + availability zone id
                                    = (known after apply)
  + cidr block
                                 = "172.20.2.0/24"
                                    = false
  + enable dns64
  + enable resource name dns a record on launch = false
  + enable resource name dns aaaa record on launch = false
                             = (known after apply)
                                          = (known after apply)
  + ipv6_cidr_block_association_id
                                  = false
  + ipv6 native
                                          = false
  + map_public_ip_on_launch
                                 = (known after apply)
  + owner id
  + private dns hostname type on launch
                                             = (known after apply)
                              = {
  + tags
```

```
+ "Name" = "a2-vpc-public-us-east-1b"
    }
   + tags all
     + "Name" = "a2-vpc-public-us-east-1b"
    }
                                 = (known after apply)
   + vpc id
 }
 # module.a2-network.module.vpc.aws_vpc.this[0] will be created
 + resource "aws_vpc" "this" {
                          = (known after apply)
   + arn
   + cidr block
                             = "172.20.0.0/16"
                                   = (known after apply)
   + default_network_acl_id
   + default route table id
                                   = (known after apply)
   + default_security_group_id
                                    = (known after apply)
   + dhcp options id
                                = (known after apply)
   + enable dns hostnames
                                     = true
   + enable_dns_support
                                   = true
   + enable_network_address_usage_metrics = (known after apply)
                         = (known after apply)
   + id
                                 = "default"
   + instance tenancy
   + ipv6 association id
                                 = (known after apply)
                               = (known after apply)
   + ipv6 cidr block
   + ipv6 cidr block network border group = (known after apply)
   + main_route_table_id
                                   = (known after apply)
   + owner id
                             = (known after apply)
   + tags
     + "Name" = "a2-vpc"
    }
   + tags all
     + "Name" = "a2-vpc"
  }
Plan: 25 to add, 0 to change, 0 to destroy.
Changes to Outputs:
 + vpc id = (known after apply)
```

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.

Releasing state lock. This may take a few moments... chasrini@WKMZTA5DFE41 project %