VPC and Dynamic Subnets with availability Zone

```
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 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.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)
      + 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
      + 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)
      + vpc_id                                        = (known
after apply)
    }

  # module.a1-network.aws_subnet.vpc-a1-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
        + id                                           = (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)
        + vpc_id                                       = (known
after apply)
      }

  # module.a1-network.aws_vpc.vpc-a1 will be created
  + resource "aws_vpc" "vpc-a1" {
      + arn                                  = (known after apply)
      + 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)
      + id                                   = (known after apply)
      + instance_tenancy                     = "default"
      + 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"
          + "Name"        = "my-vpc"
        }
      + 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 % <mark>terraform workspace select dev</mark>
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)
      + 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
      + 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)
      + vpc_id                                       = (known after apply)
    }

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

```
 + resource "aws_vpc" "vpc-a1" {
    + arn                        = (known after apply)
    + 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)
    + id                         = (known after apply)
    + instance_tenancy            = "default"
    + 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"
       + "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.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)
      + 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
      + 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)
      + vpc_id   = (known
after apply)
    }

  # module.a1-network.aws_subnet.vpc-a1-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
      + id   = (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)
        + vpc_id                                            = (known
after apply)
      }

  # module.a1-network.aws_vpc.vpc-a1 will be created
  + resource "aws_vpc" "vpc-a1" {
      + arn                                    = (known after apply)
      + 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)
      + id                                     = (known after apply)
      + instance_tenancy                       = "default"
      + 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" = "prod"
          + "Name"        = "my-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 %
```

Terraform output

```
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                                  = "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)
     + 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)
     + 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"
      + "Name"        = "appserver"
      + "Terraform"   = "true"
      }
    + tenancy                        = (known after apply)
    + 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)
    + 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)
```

```
      + 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"        = "dbserver"
        + "Terraform"   = "true"
        }
      + tags_all                       = {
        + "Environment" = "prod"
        + "Name"        = "dbserver"
        + "Terraform"   = "true"
        }
      + tenancy                        = (known after apply)
      + 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.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)
      + 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)
          + 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)
          + 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"        = "webserver"
            + "Terraform"   = "true"
            }
          + tags_all                      = {
            + "Environment" = "prod"
            + "Name"        = "webserver"
            + "Terraform"   = "true"
            }
          + tenancy                       = (known after apply)
          + 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_security_group.sg_mytraffic will be created
  + resource "aws_security_group" "sg_mytraffic" {
      + arn              = (known after apply)
      + description      = "Managed by Terraform"
      + egress           = (known after apply)
      + 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  = []
        + self             = false
        + to_port          = 80
      },
    ]
  + name                  = "sg_allow_webserver_traffic"
  + name_prefix           = (known after apply)
  + owner_id              = (known after apply)
  + revoke_rules_on_delete = false
  + tags_all              = (known after apply)
  + vpc_id                = (known after apply)
}


# 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)
  + name                    = "a2-vpc"
  + name_prefix             = (known after apply)
  + subnet_ids              = (known after apply)
  + supported_network_types = (known after apply)
  + tags                    = {
      + "Name" = "a2-vpc"
    }
  + tags_all                = {
      + "Name" = "a2-vpc"
    }
  + vpc_id                  = (known after apply)
}


# 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"
      }
    + vpc_id             = (known after apply)

    + egress {
      + action          = "allow"
      + from_port       = 0
      + ipv6_cidr_block = "::/0"
      + protocol        = "-1"
      + rule_no         = 101
      + to_port         = 0
      }
    + 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
      + to_port         = 0
      }
    + 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" {
    + arn                   = (known after apply)
    + default_route_table_id = (known after apply)
    + id                    = (known after apply)
    + owner_id              = (known after apply)
```

```
      + route              = (known after apply)
      + tags              = {
          + "Name" = "a2-vpc-default"
        }
      + tags_all            = {
          + "Name" = "a2-vpc-default"
        }
      + vpc_id             = (known after apply)


      + 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)
      + description        = (known after apply)
      + egress            = (known after apply)
      + id              = (known after apply)
      + ingress           = (known after apply)
      + name             = (known after apply)
      + name_prefix        = (known after apply)
      + owner_id          = (known after apply)
      + revoke_rules_on_delete = false
      + tags             = {
          + "Name" = "a2-vpc-default"
        }
      + tags_all            = {
          + "Name" = "a2-vpc-default"
        }
      + vpc_id             = (known after apply)
  }

# module.a2-network.module.vpc.aws_internet_gateway.this[0] will be created
+ resource "aws_internet_gateway" "this" {
      + arn      = (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)
    + instance_id          = (known after apply)
    + instance_owner_id    = (known after apply)
    + network_interface_id = (known after apply)
    + origin               = (known after apply)
    + route_table_id       = (known after apply)
    + state                = (known after apply)

    + 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)
    + owner_id        = (known after apply)
    + 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" {
    + arn             = (known after apply)
    + 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" {
    + 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[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" {
    + 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[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" {
    + arn                                          = (known after apply)
    + assign_ipv6_address_on_creation              = false
    + availability_zone                            = "us-east-1a"
    + availability_zone_id                         = (known after apply)
    + 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)
    + ipv6_native                                  = false
    + map_public_ip_on_launch                      = false
    + owner_id                                     = (known after apply)
    + private_dns_hostname_type_on_launch          = (known after apply)
    + tags                                         = {
        + "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
    + availability_zone                            = "us-east-1b"
    + availability_zone_id                         = (known after apply)
    + cidr_block                                   = "172.20.22.0/24"
    + 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
    + map_public_ip_on_launch                      = false
    + owner_id                                     = (known after apply)
    + private_dns_hostname_type_on_launch          = (known after apply)
    + tags                                         = {
        + "Name" = "a2-vpc-db-us-east-1b"
      }
    + tags_all                                     = {
        + "Name" = "a2-vpc-db-us-east-1b"
      }
    + vpc_id                                       = (known after apply)
```

```
  }

# 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
    + availability_zone                              = "us-east-1a"
    + availability_zone_id                           = (known after apply)
    + cidr_block                                     = "172.20.11.0/24"
    + 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
    + map_public_ip_on_launch                        = false
    + owner_id                                       = (known after apply)
    + 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
    + id                                             = (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                                           = {
        + "Name" = "a2-vpc-private-us-east-1b"
      }
    + tags_all                                       = {
```

```
      + "Name" = "a2-vpc-private-us-east-1b"
    }
  + vpc_id                          = (known after apply)
}


# module.a2-network.module.vpc.aws_subnet.public[0] will be created
+ resource "aws_subnet" "public" {
    + arn                             = (known after apply)
    + assign_ipv6_address_on_creation       = false
    + availability_zone               = "us-east-1a"
    + availability_zone_id                = (known after apply)
    + cidr_block                   = "172.20.1.0/24"
    + 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
    + map_public_ip_on_launch             = false
    + owner_id                      = (known after apply)
    + 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
    + availability_zone               = "us-east-1b"
    + availability_zone_id                = (known after apply)
    + cidr_block                   = "172.20.2.0/24"
    + 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
    + map_public_ip_on_launch             = false
    + owner_id                      = (known after apply)
    + 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"
      }
    + vpc_id                        = (known after apply)
  }

  # module.a2-network.module.vpc.aws_vpc.this[0] will be created
  + resource "aws_vpc" "this" {
    + arn                             = (known after apply)
    + cidr_block                      = "172.20.0.0/16"
    + 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            = true
    + enable_dns_support              = true
    + enable_network_address_usage_metrics = (known after apply)
    + id                              = (known after apply)
    + instance_tenancy                = "default"
    + 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                            = {
        + "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 %
```