

```
alvaroandrespinzoncortes@Alvaros-iMac stack1 % terraform apply
data.aws_caller_identity.current: Reading...
module.s3_bucket.data.aws_caller_identity.current: Reading...
module.s3_bucket.data.aws_partition.current: Reading...
module.s3_bucket.data.aws_canonical_user_id.this[0]: Reading...
module.s3_bucket.data.aws_region.current: Reading...
module.s3_bucket.data.aws_partition.current: Read complete after 0s [id=aws]
module.s3_bucket.data.aws_region.current: Read complete after 0s [id=us-west-2]
data.aws_caller_identity.current: Read complete after 0s [id=752227508740]
module.s3_bucket.data.aws_caller_identity.current: Read complete after 0s [id=752227508740]
module.s3_bucket.data.aws_canonical_user_id.this[0]: Read complete after 1s [id=0870d4792475274e4b67f60484303f85389e4954cded7f55777b4b61bde7047f]
```

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.aurora_serverless.aws_db_subnet_group.aurora will be created
+ resource "aws_db_subnet_group" "aurora" {
  + arn                = (known after apply)
  + description        = "Managed by Terraform"
  + id                 = (known after apply)
  + name               = "my-aurora-serverless-subnet-group"
  + name_prefix        = (known after apply)
  + subnet_ids         = (known after apply)
  + supported_network_types = (known after apply)
  + tags               = {
    + "Name" = "my-aurora-serverless-subnet-group"
  }
  + tags_all           = {
    + "Name" = "my-aurora-serverless-subnet-group"
  }
  + vpc_id              = (known after apply)
}

# module.aurora_serverless.aws_rds_cluster.aurora_serverless will be created
+ resource "aws_rds_cluster" "aurora_serverless" {
  + allocated_storage      = (known after apply)
  + allow_major_version_upgrade = true
  + apply_immediately      = true
  + arn                    = (known after apply)
  + availability_zones      = (known after apply)
  + backup_retention_period = (known after apply)
  + ca_certificate_identifier = (known after apply)
  + ca_certificate_valid_till = (known after apply)
  + cluster_identifier      = "my-aurora-serverless"
  + cluster_identifier_prefix = (known after apply)
  + cluster_members         = (known after apply)
  + cluster_resource_id     = (known after apply)
  + copy_tags_to_snapshot   = false
  + database_name           = "myapp"
  + db_cluster_parameter_group_name = (known after apply)
  + db_subnet_group_name    = "my-aurora-serverless-subnet-group"
  + db_system_id            = (known after apply)
  + delete_automated_backups = true
  + enable_global_write_forwarding = false
  + enable_http_endpoint     = true
  + enable_local_write_forwarding = false
  + endpoint                 = (known after apply)
  + engine                   = "aurora-postgresql"
  + engine_lifecycle_support = (known after apply)
  + engine_mode              = "provisioned"
  + engine_version           = "15.4"
  + engine_version_actual    = (known after apply)
  + hosted_zone_id           = (known after apply)
  + iam_roles                = (known after apply)
  + id                       = (known after apply)
  + kms_key_id               = (known after apply)
  + master_password          = (sensitive value)
  + master_user_secret       = (known after apply)
  + master_user_secret_kms_key_id = (known after apply)
  + master_username          = "dbadmin"
  + network_type             = (known after apply)
  + performance_insights_kms_key_id = (known after apply)
  + performance_insights_retention_period = (known after apply)
  + port                     = (known after apply)
  + preferred_backup_window  = (known after apply)
  + preferred_maintenance_window = (known after apply)
  + reader_endpoint          = (known after apply)
  + skip_final_snapshot      = true
  + storage_encrypted        = (known after apply)
  + storage_type             = (known after apply)
  + tags_all                 = (known after apply)
  + vpc_security_group_ids   = (known after apply)

  + serverlessv2_scaling_configuration {
    + max_capacity = 1
    + min_capacity = 0.5
    + seconds_until_auto_pause = (known after apply)
  }
}

# module.aurora_serverless.aws_rds_cluster_instance.aurora_instance will be created
+ resource "aws_rds_cluster_instance" "aurora_instance" {
  + apply_immediately = (known after apply)
  + arn                = (known after apply)
  + auto_minor_version_upgrade = true
  + availability_zone    = (known after apply)
  + ca_cert_identifier   = (known after apply)
  + cluster_identifier    = (known after apply)
  + copy_tags_to_snapshot = false
  + db_parameter_group_name = (known after apply)
  + db_subnet_group_name  = (known after apply)
  + dbi_resource_id       = (known after apply)
  + endpoint              = (known after apply)
  + engine                 = "aurora-postgresql"
  + engine_version        = "15.4"
  + engine_version_actual = (known after apply)
  + force_destroy         = false
  + id                    = (known after apply)
  + identifier            = (known after apply)
  + identifier_prefix     = (known after apply)
  + instance_class        = "db.serverless"
  + kms_key_id            = (known after apply)
  + monitoring_interval   = 0
  + monitoring_role_arn   = (known after apply)
  + network_type          = (known after apply)
  + performance_insights_enabled = (known after apply)
  + performance_insights_kms_key_id = (known after apply)
  + performance_insights_retention_period = (known after apply)
  + port                  = (known after apply)
  + preferred_backup_window = (known after apply)
```

```

+ preferred_maintenance_window = (known after apply)
+ promotion_tier                 = 0
+ publicly_accessible            = (known after apply)
+ storage_encrypted              = (known after apply)
+ tags_all                      = (known after apply)
+ writer                        = (known after apply)
}

# module.aurora_serverless.aws_secretsmanager_secret.aurora_secret will be created
+ resource "aws_secretsmanager_secret" "aurora_secret" {
+   arn = (known after apply)
+   force_overwrite_replica_secret = false
+   id = (known after apply)
+   name = "my-aurora-serverless"
+   name_prefix = (known after apply)
+   policy = (known after apply)
+   recovery_window_in_days = 0
+   tags_all = (known after apply)

+   replica (known after apply)
}

# module.aurora_serverless.aws_secretsmanager_secret_version.aurora_secret_version will be created
+ resource "aws_secretsmanager_secret_version" "aurora_secret_version" {
+   arn = (known after apply)
+   id = (known after apply)
+   secret_id = (known after apply)
+   secret_string = (sensitive value)
+   version_id = (known after apply)
+   version_stages = (known after apply)
}

# module.aurora_serverless.aws_security_group.aurora_sg will be created
+ resource "aws_security_group" "aurora_sg" {
+   arn = (known after apply)
+   description = "Security group for Aurora Serverless"
+   egress = [
+     {
+       cidr_blocks = [
+         "0.0.0.0/0",
+       ]
+       from_port = 0
+       ipv6_cidr_blocks = []
+       prefix_list_ids = []
+       protocol = "-1"
+       security_groups = []
+       self = false
+       to_port = 0
+     },
+     # (1 unchanged attribute hidden)
+   ],
+   id = (known after apply)
+   ingress = [
+     {
+       cidr_blocks = [
+         "10.0.0.0/16",
+       ]
+       from_port = 5432
+       ipv6_cidr_blocks = []
+       prefix_list_ids = []
+       protocol = "tcp"
+       security_groups = []
+       self = false
+       to_port = 5432
+     },
+     # (1 unchanged attribute hidden)
+   ],
+   name = "my-aurora-serverless-sg"
+   name_prefix = (known after apply)
+   owner_id = (known after apply)
+   revoke_rules_on_delete = false
+   tags = {
+     "Name" = "my-aurora-serverless-sg"
+   }
+   tags_all = {
+     "Name" = "my-aurora-serverless-sg"
+   }
+   vpc_id = (known after apply)
}

# module.aurora_serverless.random_password.master_password will be created
+ resource "random_password" "master_password" {
+   bcrypt_hash = (sensitive value)
+   id = (known after apply)
+   length = 16
+   lower = true
+   min_lower = 0
+   min_numeric = 0
+   min_special = 0
+   min_upper = 0
+   number = true
+   numeric = true
+   result = (sensitive value)
+   special = true
+   upper = true
}

# module.s3_bucket.aws_s3_bucket.this[0] will be created
+ resource "aws_s3_bucket" "this" {
+   acceleration_status = (known after apply)
+   acl = (known after apply)
+   arn = (known after apply)
+   bucket = "bedrock-kb-752227508740"
+   bucket_domain_name = (known after apply)
+   bucket_prefix = (known after apply)
+   bucket_regional_domain_name = (known after apply)
+   force_destroy = true
+   hosted_zone_id = (known after apply)
+   id = (known after apply)
+   object_lock_enabled = false
+   policy = (known after apply)
+   region = (known after apply)
+   request_payer = (known after apply)
+   tags = {
+     "Environment" = "dev"
+     "Terraform" = "true"
+   }
+   tags_all = {
+     "Environment" = "dev"
+     "Terraform" = "true"
+   }
+   website_domain = (known after apply)
}

```

```

+ website_endpoint           = (known after apply)
+ cors_rule (known after apply)
+ grant (known after apply)
+ lifecycle_rule (known after apply)
+ logging (known after apply)
+ object_lock_configuration (known after apply)
+ replication_configuration (known after apply)
+ server_side_encryption_configuration (known after apply)
+ versioning (known after apply)
+ website (known after apply)
}

# module.s3_bucket.aws_s3_bucket_acl.this[0] will be created
+ resource "aws_s3_bucket_acl" "this" {
+   acl = "private"
+   bucket = (known after apply)
+   id = (known after apply)

+   access_control_policy (known after apply)
}

# module.s3_bucket.aws_s3_bucket_ownership_controls.this[0] will be created
+ resource "aws_s3_bucket_ownership_controls" "this" {
+   bucket = (known after apply)
+   id = (known after apply)

+   rule {
+     object_ownership = "BucketOwnerPreferred"
+   }
}

# module.s3_bucket.aws_s3_bucket_public_access_block.this[0] will be created
+ resource "aws_s3_bucket_public_access_block" "this" {
+   block_public_acls = true
+   block_public_policy = true
+   bucket = (known after apply)
+   id = (known after apply)
+   ignore_public_acls = true
+   restrict_public_buckets = true
}

# module.s3_bucket.aws_s3_bucket_server_side_encryption_configuration.this[0] will be created
+ resource "aws_s3_bucket_server_side_encryption_configuration" "this" {
+   bucket = (known after apply)
+   id = (known after apply)

+   rule {
+     apply_server_side_encryption_by_default {
+       sse_algorithm = "AES256"
+       # (1 unchanged attribute hidden)
+     }
+   }
}

# module.s3_bucket.aws_s3_bucket_versioning.this[0] will be created
+ resource "aws_s3_bucket_versioning" "this" {
+   bucket = (known after apply)
+   id = (known after apply)

+   versioning_configuration {
+     mfa_delete = (known after apply)
+     status = "Enabled"
+   }
}

# 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 = {
+     "Environment" = "dev"
+     "Name" = "bedrock-poc-vpc-default"
+     "Terraform" = "true"
+   }
+   tags_all = {
+     "Environment" = "dev"
+     "Name" = "bedrock-poc-vpc-default"
+     "Terraform" = "true"
+   }
+   vpc_id = (known after apply)

+   egress {
+     action = "allow"
+     from_port = 0
+     ipv6_cidr_block = "::/0"
+     protocol = "-1"
+     rule_no = 101
+     to_port = 0
+     # (1 unchanged attribute hidden)
+   }
+   egress {
+     action = "allow"
+     cidr_block = "0.0.0.0/0"
+     from_port = 0
+     protocol = "-1"
+     rule_no = 100
+     to_port = 0
+     # (1 unchanged attribute hidden)
+   }
+   ingress {
+     action = "allow"
+     from_port = 0
+     ipv6_cidr_block = "::/0"
+     protocol = "-1"
+     rule_no = 101
+     to_port = 0
+     # (1 unchanged attribute hidden)
+   }
+   ingress {
+     action = "allow"

```

```

+ cidr_block      = "0.0.0.0/0"
+ from_port      = 0
+ protocol        = "-1"
+ rule_no        = 100
+ to_port         = 0
# {1 unchanged attribute hidden}
}
}

# 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               = {
+     "Environment" = "dev"
+     "Name"        = "bedrock-poc-vpc-default"
+     "Terraform"   = "true"
+   }
+   tags_all           = {
+     "Environment" = "dev"
+     "Name"        = "bedrock-poc-vpc-default"
+     "Terraform"   = "true"
+   }
+   vpc_id              = (known after apply)
+   timeouts {
+     create = "5m"
+     update = "5m"
+   }
}

# 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               = {
+     "Environment" = "dev"
+     "Name"        = "bedrock-poc-vpc-default"
+     "Terraform"   = "true"
+   }
+   tags_all           = {
+     "Environment" = "dev"
+     "Name"        = "bedrock-poc-vpc-default"
+     "Terraform"   = "true"
+   }
+   vpc_id              = (known after apply)
}

# module.vpc.aws_eip.nat[0] will be created
+ resource "aws_eip" "nat" {
+   allocation_id      = (known after apply)
+   arn                 = (known after apply)
+   association_id      = (known after apply)
+   carrier_ip          = (known after apply)
+   customer_owned_ip   = (known after apply)
+   domain              = "vpc"
+   id                  = (known after apply)
+   instance             = (known after apply)
+   ipam_pool_id        = (known after apply)
+   network_border_group = (known after apply)
+   network_interface    = (known after apply)
+   private_dns          = (known after apply)
+   private_ip           = (known after apply)
+   ptr_record           = (known after apply)
+   public_dns           = (known after apply)
+   public_ip            = (known after apply)
+   public_ipv4_pool     = (known after apply)
+   tags                = {
+     "Environment" = "dev"
+     "Name"        = "bedrock-poc-vpc-us-west-2a"
+     "Terraform"   = "true"
+   }
+   tags_all           = {
+     "Environment" = "dev"
+     "Name"        = "bedrock-poc-vpc-us-west-2a"
+     "Terraform"   = "true"
+   }
+   vpc                 = (known after apply)
}

# 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               = {
+     "Environment" = "dev"
+     "Name"        = "bedrock-poc-vpc"
+     "Terraform"   = "true"
+   }
+   tags_all           = {
+     "Environment" = "dev"
+     "Name"        = "bedrock-poc-vpc"
+     "Terraform"   = "true"
+   }
+   vpc_id              = (known after apply)
}

# module.vpc.aws_nat_gateway.this[0] will be created
+ resource "aws_nat_gateway" "this" {
+   allocation_id      = (known after apply)
+   association_id      = (known after apply)
+   connectivity_type   = "public"
+   id                  = (known after apply)
+   network_interface_id = (known after apply)
+   private_ip          = (known after apply)
+   public_ip            = (known after apply)
+   secondary_private_ip_address_count = (known after apply)
+   secondary_private_ip_addresses     = (known after apply)
+   subnet_id            = (known after apply)
+   tags                 = {
+     "Environment" = "dev"

```

```

    + "Name"          = "bedrock-poc-vpc-us-west-2a"
    + "Terraform"     = "true"
  }
  + tags_all          = {
    + "Environment" = "dev"
    + "Name"        = "bedrock-poc-vpc-us-west-2a"
    + "Terraform"   = "true"
  }
}

# module.vpc.aws_route.private_nat_gateway[0] will be created
+ resource "aws_route" "private_nat_gateway" {
  + destination_cidr_block = "0.0.0.0/0"
  + id                    = (known after apply)
  + instance_id           = (known after apply)
  + instance_owner_id     = (known after apply)
  + nat_gateway_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.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.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         = {
    + "Environment" = "dev"
    + "Name"        = "bedrock-poc-vpc-private"
    + "Terraform"   = "true"
  }
  + tags_all     = {
    + "Environment" = "dev"
    + "Name"        = "bedrock-poc-vpc-private"
    + "Terraform"   = "true"
  }
  + vpc_id       = (known after apply)
}

# 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         = {
    + "Environment" = "dev"
    + "Name"        = "bedrock-poc-vpc-public"
    + "Terraform"   = "true"
  }
  + tags_all     = {
    + "Environment" = "dev"
    + "Name"        = "bedrock-poc-vpc-public"
    + "Terraform"   = "true"
  }
  + vpc_id       = (known after apply)
}

# 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.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.vpc.aws_route_table_association.private[2] 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.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.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.vpc.aws_route_table_association.public[2] 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.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-west-2a"
+   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                               = {
+     "Environment" = "dev"
+     "Name"         = "bedrock-poc-vpc-private-us-west-2a"
+     "Terraform"    = "true"
+   }
+   tags_all                           = {
+     "Environment" = "dev"
+     "Name"         = "bedrock-poc-vpc-private-us-west-2a"
+     "Terraform"    = "true"
+   }
+   vpc_id                             = (known after apply)
}

# 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-west-2b"
+   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                               = {
+     "Environment" = "dev"
+     "Name"         = "bedrock-poc-vpc-private-us-west-2b"
+     "Terraform"    = "true"
+   }
+   tags_all                           = {
+     "Environment" = "dev"
+     "Name"         = "bedrock-poc-vpc-private-us-west-2b"
+     "Terraform"    = "true"
+   }
+   vpc_id                             = (known after apply)
}

# module.vpc.aws_subnet.private[2] will be created
+ resource "aws_subnet" "private" {
+   arn                                = (known after apply)
+   assign_ipv6_address_on_creation    = false
+   availability_zone                  = "us-west-2c"
+   availability_zone_id               = (known after apply)
+   cidr_block                         = "10.0.3.0/24"
+   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                               = {
+     "Environment" = "dev"
+     "Name"         = "bedrock-poc-vpc-private-us-west-2c"
+     "Terraform"    = "true"
+   }
+   tags_all                           = {
+     "Environment" = "dev"
+     "Name"         = "bedrock-poc-vpc-private-us-west-2c"
+     "Terraform"    = "true"
+   }
+   vpc_id                             = (known after apply)
}

# 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-west-2a"
+   availability_zone_id               = (known after apply)
+   cidr_block                         = "10.0.101.0/24"
+   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                               = {
+     "Environment" = "dev"
+     "Name"         = "bedrock-poc-vpc-public-us-west-2a"
+     "Terraform"    = "true"
+   }
+   tags_all                           = {
+     "Environment" = "dev"
+     "Name"         = "bedrock-poc-vpc-public-us-west-2a"
+     "Terraform"    = "true"
+   }
+   vpc_id                             = (known after apply)
}

# 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-west-2b"
+ availability_zone_id = (known after apply)
+ cidr_block = "10.0.102.0/24"
+ 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 = {
+   "Environment" = "dev"
+   "Name" = "bedrock-poc-vpc-public-us-west-2b"
+   "Terraform" = "true"
+ }
+ tags_all = {
+   "Environment" = "dev"
+   "Name" = "bedrock-poc-vpc-public-us-west-2b"
+   "Terraform" = "true"
+ }
+ vpc_id = (known after apply)
}

# module.vpc.aws_subnet.public[2] will be created
+ resource "aws_subnet" "public" {
+   arn = (known after apply)
+   assign_ipv6_address_on_creation = false
+   availability_zone = "us-west-2c"
+   availability_zone_id = (known after apply)
+   cidr_block = "10.0.103.0/24"
+   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 = {
+     "Environment" = "dev"
+     "Name" = "bedrock-poc-vpc-public-us-west-2c"
+     "Terraform" = "true"
+   }
+   tags_all = {
+     "Environment" = "dev"
+     "Name" = "bedrock-poc-vpc-public-us-west-2c"
+     "Terraform" = "true"
+   }
+   vpc_id = (known after apply)
}

# module.vpc.aws_vpc.this[0] will be created
+ resource "aws_vpc" "this" {
+   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 = 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 = {
+     "Environment" = "dev"
+     "Name" = "bedrock-poc-vpc"
+     "Terraform" = "true"
+   }
+   tags_all = {
+     "Environment" = "dev"
+     "Name" = "bedrock-poc-vpc"
+     "Terraform" = "true"
+   }
}

```

Plan: 36 to add, 0 to change, 0 to destroy.

Changes to Outputs:

```

+ aurora_arn = (known after apply)
+ aurora_endpoint = (known after apply)
+ db_endpoint = (known after apply)
+ db_reader_endpoint = (known after apply)
+ private_subnet_ids = [
+   (known after apply),
+   (known after apply),
+   (known after apply),
+ ]
+ public_subnet_ids = [
+   (known after apply),
+   (known after apply),
+   (known after apply),
+ ]
+ rds_secret_arn = (known after apply)
+ s3_bucket_name = (known after apply)
+ vpc_id = (known after apply)

```

Do you want to perform these actions?

Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

```

module.aurora_serverless.random_password.master_password: Creating...
module.aurora_serverless.random_password.master_password: Creation complete after 0s [id=none]
module.aurora_serverless.aws_secretsmanager_secret.aurora_secret: Creating...
module.vpc.aws_vpc.this[0]: Creating...
module.s3_bucket.aws_s3_bucket.this[0]: Creating...
module.aurora_serverless.aws_secretsmanager_secret.aurora_secret: Creation complete after 2s [id=arn:aws:secretsmanager:us-west-2:752227508740:secret:my-aurora-serverless-BQ6kIZ]
module.s3_bucket.aws_s3_bucket.this[0]: Creation complete after 5s [id=bedrock-kb-752227508740]
module.s3_bucket.aws_s3_bucket_public_access_block.this[0]: Creating...
module.s3_bucket.aws_s3_bucket_server_side_encryption_configuration.this[0]: Creating...

```

```
module.s3_bucket.aws_s3_bucket_versioning.this[0]: Creating...
module.s3_bucket.aws_s3_bucket_public_access_block.this[0]: Creation complete after 1s [id=bedrock-kb-752227508740]
module.s3_bucket.aws_s3_bucket_ownership_controls.this[0]: Creating...
module.s3_bucket.aws_s3_bucket_server_side_encryption_configuration.this[0]: Creation complete after 1s [id=bedrock-kb-752227508740]
module.s3_bucket.aws_s3_bucket_ownership_controls.this[0]: Creation complete after 1s [id=bedrock-kb-752227508740]
module.s3_bucket.aws_s3_bucket_acl.this[0]: Creating...
module.s3_bucket.aws_s3_bucket_versioning.this[0]: Creation complete after 3s [id=bedrock-kb-752227508740]
module.s3_bucket.aws_s3_bucket_acl.this[0]: Creation complete after 1s [id=bedrock-kb-752227508740,private]
module.vpc.aws_vpc.this[0]: Still creating... [10s elapsed]
module.vpc.aws_vpc.this[0]: Creation complete after 14s [id=vpc-00b9ee8ba1aa45b5d]
module.vpc.aws_default_route_table.default[0]: Creating...
module.vpc.aws_default_security_group.this[0]: Creating...
module.vpc.aws_internet_gateway.this[0]: Creating...
module.vpc.aws_route_table.private[0]: Creating...
module.vpc.aws_subnet.public[0]: Creating...
module.vpc.aws_default_network_acl.this[0]: Creating...
module.vpc.aws_route_table.public[0]: Creating...
module.aurora_serverless.aws_security_group.aurora_sg: Creating...
module.vpc.aws_subnet.private[0]: Creating...
module.vpc.aws_subnet.private[2]: Creating...
module.vpc.aws_default_route_table.default[0]: Creation complete after 1s [id=rtb-07a6dc8a775678b2a]
module.vpc.aws_subnet.private[1]: Creating...
module.vpc.aws_internet_gateway.this[0]: Creation complete after 2s [id=igw-01087cc4fbca95132]
module.vpc.aws_subnet.public[1]: Creating...
module.vpc.aws_route_table.private[0]: Creation complete after 2s [id=rtb-0bccda4c3af8a631f]
module.vpc.aws_subnet.private[0]: Creation complete after 2s [id=subnet-08085bb0fd03a4d06]
module.vpc.aws_subnet.public[2]: Creating...
module.vpc.aws_subnet.public[0]: Creation complete after 2s [id=subnet-067e40d7cf76547bc]
module.vpc.aws_eip_nat[0]: Creating...
module.vpc.aws_route_table.public[0]: Creation complete after 2s [id=rtb-0e57b34a3c00881c8]
module.vpc.aws_route.public_internet_gateway[0]: Creating...
module.vpc.aws_subnet.private[2]: Creation complete after 2s [id=subnet-0e62a5e1b55f5737d]
module.vpc.aws_subnet.private[1]: Creation complete after 1s [id=subnet-081f0ccde13ea4158f]
module.vpc.aws_route_table_association.private[1]: Creating...
module.vpc.aws_route_table_association.private[2]: Creating...
module.vpc.aws_route_table_association.private[0]: Creating...
module.vpc.aws_subnet.public[1]: Creation complete after 1s [id=subnet-0943448b12785054c]
module.vpc.aws_subnet.public[2]: Creation complete after 1s [id=subnet-0644e7abb17c65d66]
module.vpc.aws_route_table_association.public[2]: Creating...
module.vpc.aws_route_table_association.public[1]: Creating...
module.vpc.aws_eip_nat[0]: Creation complete after 1s [id=eipalloc-0ef48915c21a16860]
module.vpc.aws_route_table_association.public[0]: Creating...
module.vpc.aws_route.public_internet_gateway[0]: Creation complete after 1s [id=r-rtb-0e57b34a3c00881c81080289494]
module.aurora_serverless.aws_db_subnet_group.aurora: Creating...
module.vpc.aws_default_network_acl.this[0]: Creation complete after 3s [id=acl-0fcb78e656072956]
module.vpc.aws_nat_gateway.this[0]: Creating...
module.vpc.aws_default_security_group.this[0]: Creation complete after 3s [id=sg-019861d1c2dbac866]
module.vpc.aws_route_table_association.private[2]: Creation complete after 1s [id=rtbassoc-0f93c3cbd4bf8e004]
module.vpc.aws_route_table_association.private[0]: Creation complete after 1s [id=rtbassoc-01fb3aab722b96277]
module.vpc.aws_route_table_association.private[1]: Creation complete after 1s [id=rtbassoc-04c03f43df664bb0]
module.vpc.aws_route_table_association.public[1]: Creation complete after 1s [id=rtbassoc-0c557133d5d12071]
module.vpc.aws_route_table_association.public[2]: Creation complete after 1s [id=rtbassoc-03e020964eb2ab8cc]
module.vpc.aws_route_table_association.public[0]: Creation complete after 1s [id=rtbassoc-09f15e8e609298d61]
module.aurora_serverless.aws_security_group.aurora_sg: Creation complete after 5s [id=sg-07fd3cc95ed706c31]
module.aurora_serverless.aws_db_subnet_group.aurora: Creation complete after 2s [id=my-aurora-serverless-subnet-group]
module.aurora_serverless.aws_rds_cluster.aurora_serverless: Creating...
module.vpc.aws_nat_gateway.this[0]: Still creating... [10s elapsed]
module.aurora_serverless.aws_rds_cluster.aurora_serverless: Still creating... [10s elapsed]
module.vpc.aws_nat_gateway.this[0]: Still creating... [20s elapsed]
module.aurora_serverless.aws_rds_cluster.aurora_serverless: Still creating... [20s elapsed]
module.vpc.aws_nat_gateway.this[0]: Still creating... [30s elapsed]
module.aurora_serverless.aws_rds_cluster.aurora_serverless: Still creating... [30s elapsed]
module.vpc.aws_nat_gateway.this[0]: Still creating... [40s elapsed]
module.aurora_serverless.aws_rds_cluster.aurora_serverless: Still creating... [40s elapsed]
module.aurora_serverless.aws_rds_cluster.aurora_serverless: Creation complete after 42s [id=my-aurora-serverless]
module.aurora_serverless.aws_secretsmanager_secret_version.aurora_secret_version: Creating...
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Creating...
module.aurora_serverless.aws_secretsmanager_secret_version.aurora_secret_version: Creation complete after 0s [id=arn:aws:secretsmanager:us-west-2:752227508740:secret:my-aurora-serverless-BQ6kIZ]
module.vpc.aws_nat_gateway.this[0]: Still creating... [50s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [10s elapsed]
module.vpc.aws_nat_gateway.this[0]: Still creating... [1m0s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [20s elapsed]
module.vpc.aws_nat_gateway.this[0]: Still creating... [1m10s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [30s elapsed]
module.vpc.aws_nat_gateway.this[0]: Still creating... [1m20s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [40s elapsed]
module.vpc.aws_nat_gateway.this[0]: Still creating... [1m30s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [50s elapsed]
module.vpc.aws_nat_gateway.this[0]: Still creating... [1m40s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [1m0s elapsed]
module.vpc.aws_nat_gateway.this[0]: Still creating... [1m50s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [1m10s elapsed]
module.vpc.aws_nat_gateway.this[0]: Creation complete after 1m57s [id=nat-09650cbbc6282461a]
module.vpc.aws_route.private_nat_gateway[0]: Creating...
module.vpc.aws_route.private_nat_gateway[0]: Creation complete after 2s [id=r-rtb-0bccda4c3af8a631f1080289494]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [1m20s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [1m30s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [1m40s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [1m50s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [2m0s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [2m10s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [2m20s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [2m30s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [2m40s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [2m50s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [3m0s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [3m10s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [3m20s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [3m30s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [3m40s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [3m50s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [4m0s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [4m10s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [4m20s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [4m30s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [4m40s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [4m50s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [5m0s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [5m10s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [5m20s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [5m30s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [5m40s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [5m50s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [6m0s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [6m10s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [6m20s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [6m30s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [6m40s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [6m50s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [7m0s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [7m10s elapsed]
```



```
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [7m20s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [7m30s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [7m40s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [7m50s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [8m0s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [8m10s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [8m20s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [8m30s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [8m40s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [8m50s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [9m0s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [9m10s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [9m20s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Still creating... [9m30s elapsed]
module.aurora_serverless.aws_rds_cluster_instance.aurora_instance: Creation complete after 9m36s [id=tf-20241228202445921100000006]
```

Apply complete! Resources: 36 added, 0 changed, 0 destroyed.

Outputs:

```
aurora_arn = "arn:aws:rds:us-west-2:752227508740:cluster:my-aurora-serverless"
aurora_endpoint = "my-aurora-serverless.cluster-c3asosq08wp4.us-west-2.rds.amazonaws.com"
db_endpoint = "my-aurora-serverless.cluster-c3asosq08wp4.us-west-2.rds.amazonaws.com"
db_reader_endpoint = "my-aurora-serverless.cluster-ro-c3asosq08wp4.us-west-2.rds.amazonaws.com"
private_subnet_ids = [
    "subnet-08085bb0fd03a4d06",
    "subnet-081f0ccde13e4156f",
    "subnet-0e62a5e1b55f5737d",
]
public_subnet_ids = [
    "subnet-067e40d7cf76547bc",
    "subnet-0943448b12785054c",
    "subnet-0644e7abb17c65d66",
]
rds_secret_arn = "arn:aws:secretsmanager:us-west-2:752227508740:secret:my-aurora-serverless-BQ6kIZ"
s3_bucket_name = "arn:aws:s3:::bedrock-kb-752227508740"
vpc_id = "vpc-00b9ee8ba1aa45b5d"
```