ALTSCHOOL AFRICA

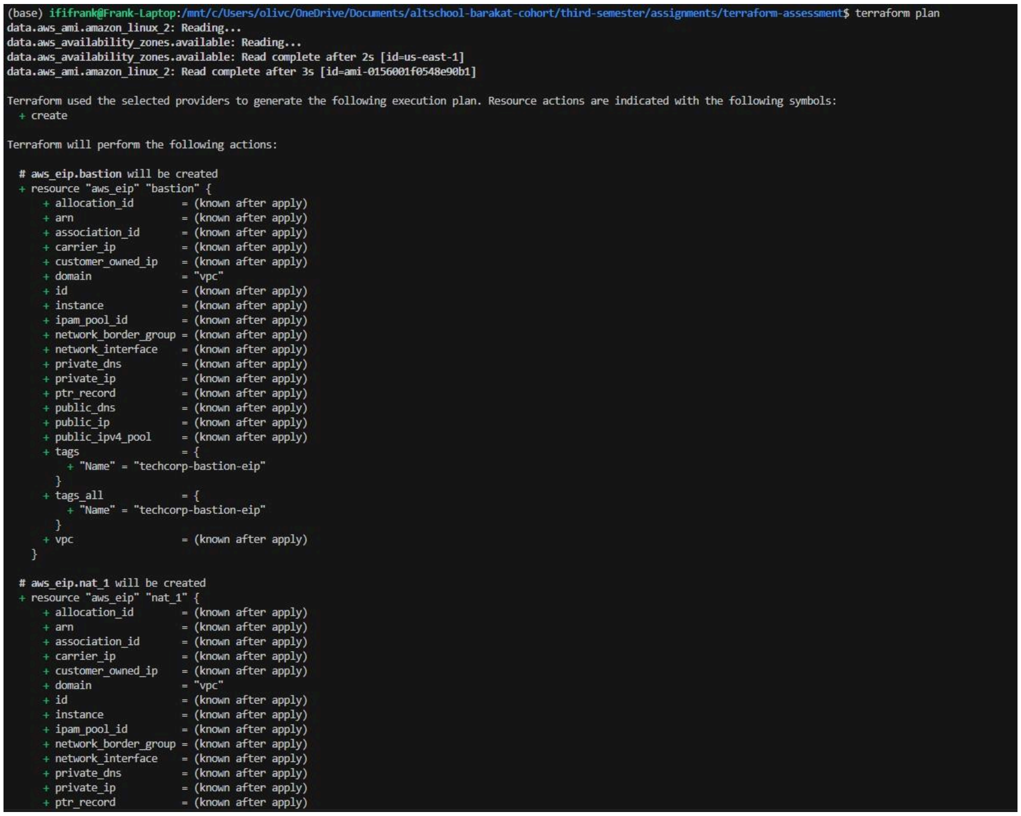
SCHOOL OF ENGINEERING (CLOUD ENGINEERING TRACK)

THIRD SEMESTER ASSESSMENT (MONTH 1 ASSESSMENT - SETTING UP AWS INSTRUCTURE WITH TERRAFORM**)**

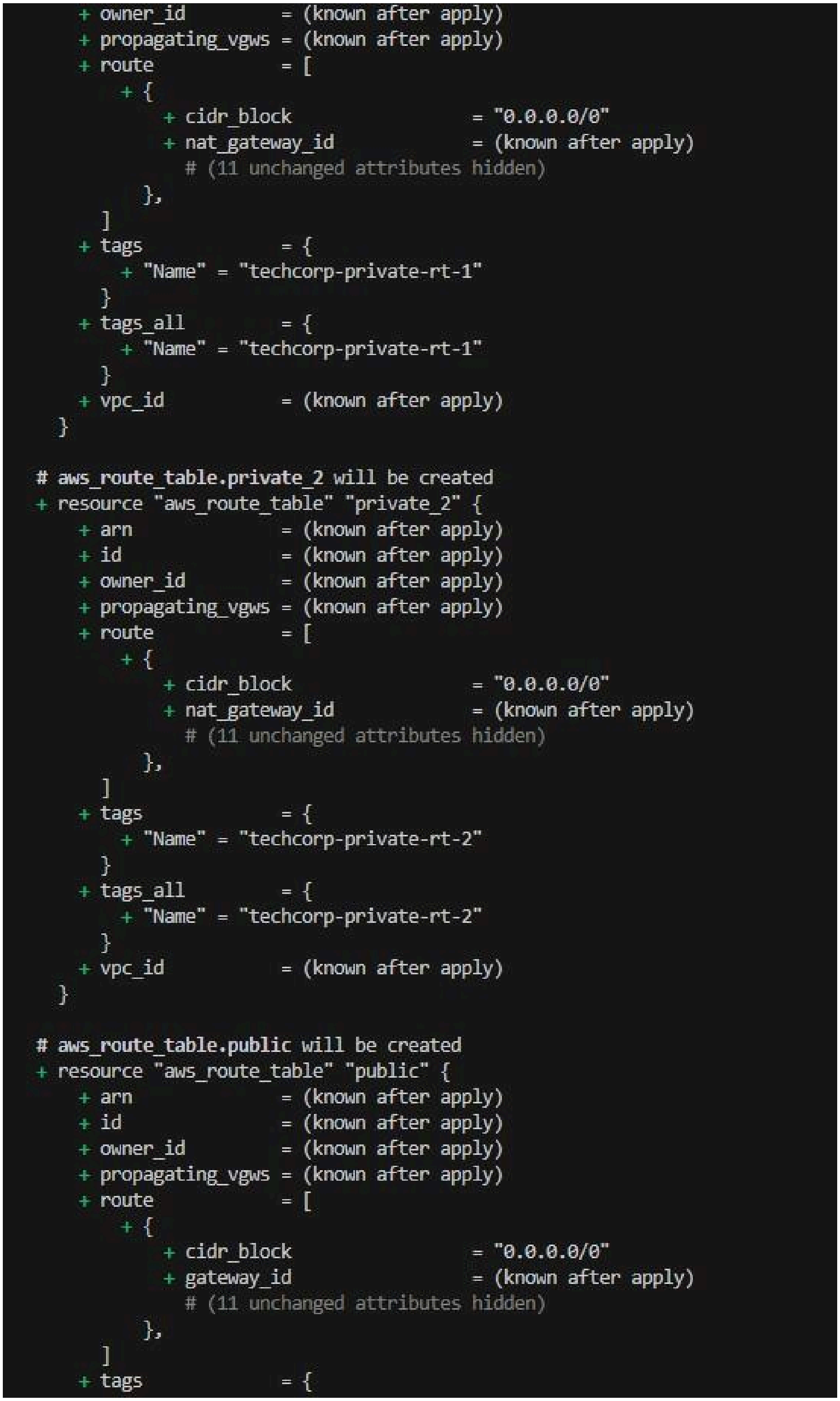
DEPLOYMENT EVIDENCE

Name: Ifeanyi F. Ike

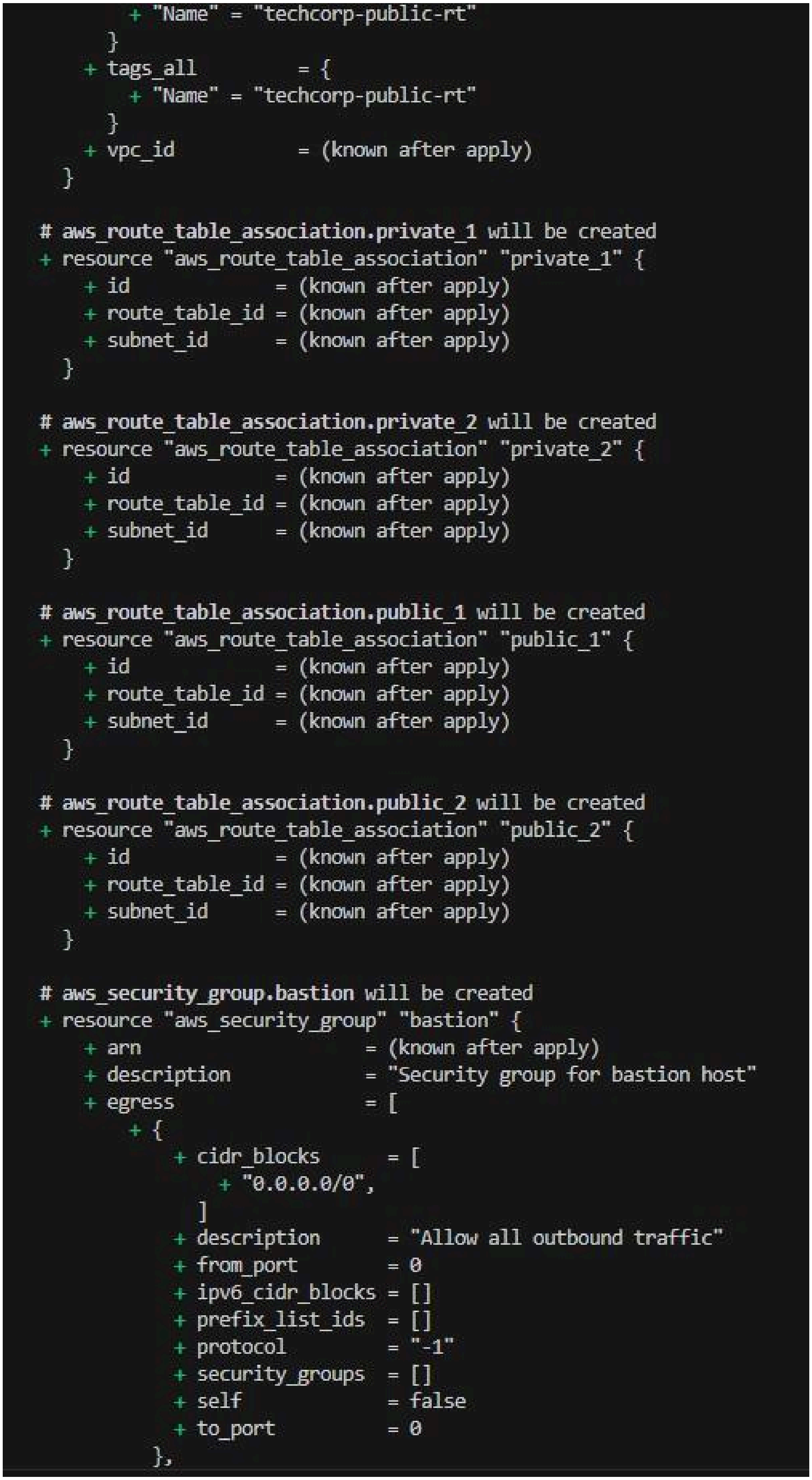
1. Terraform plan output



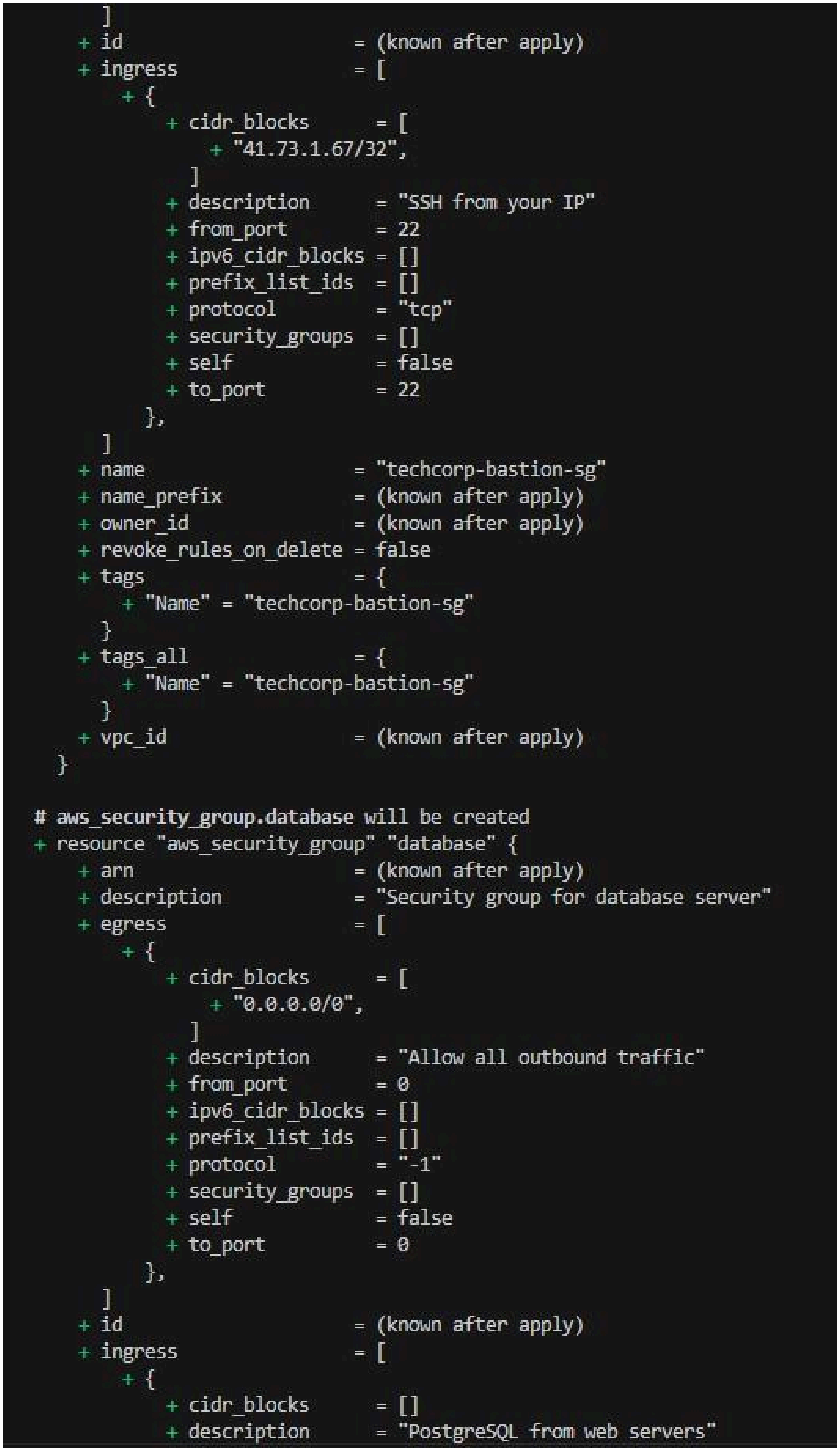
```
(base) ififrank@Frank-Laptop:/mnt/c/Users/olivc/OneDrive/Documents/altschool-barakat-cohort/third-semester/assignments/terraform-assessment$ terraform plan
data.aws_ami.amazon_linux_2: Reading...
data.aws_availability_zones.available: Reading...
data.aws_availability_zones.available: Read complete after 2s [id=us-east-1]
data.aws_ami.amazon_linux_2: Read complete after 3s [id=ami-0156001f0548e90b1]

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:

  # aws_eip.bastion will be created
  + resource "aws_eip" "bastion" {
      + 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                 = {
          + "Name" = "techcorp-bastion-eip"
        }
      + tags_all             = {
          + "Name" = "techcorp-bastion-eip"
        }
      + vpc                  = (known after apply)
    }

  # aws_eip.nat_1 will be created
  + resource "aws_eip" "nat_1" {
      + 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)
```



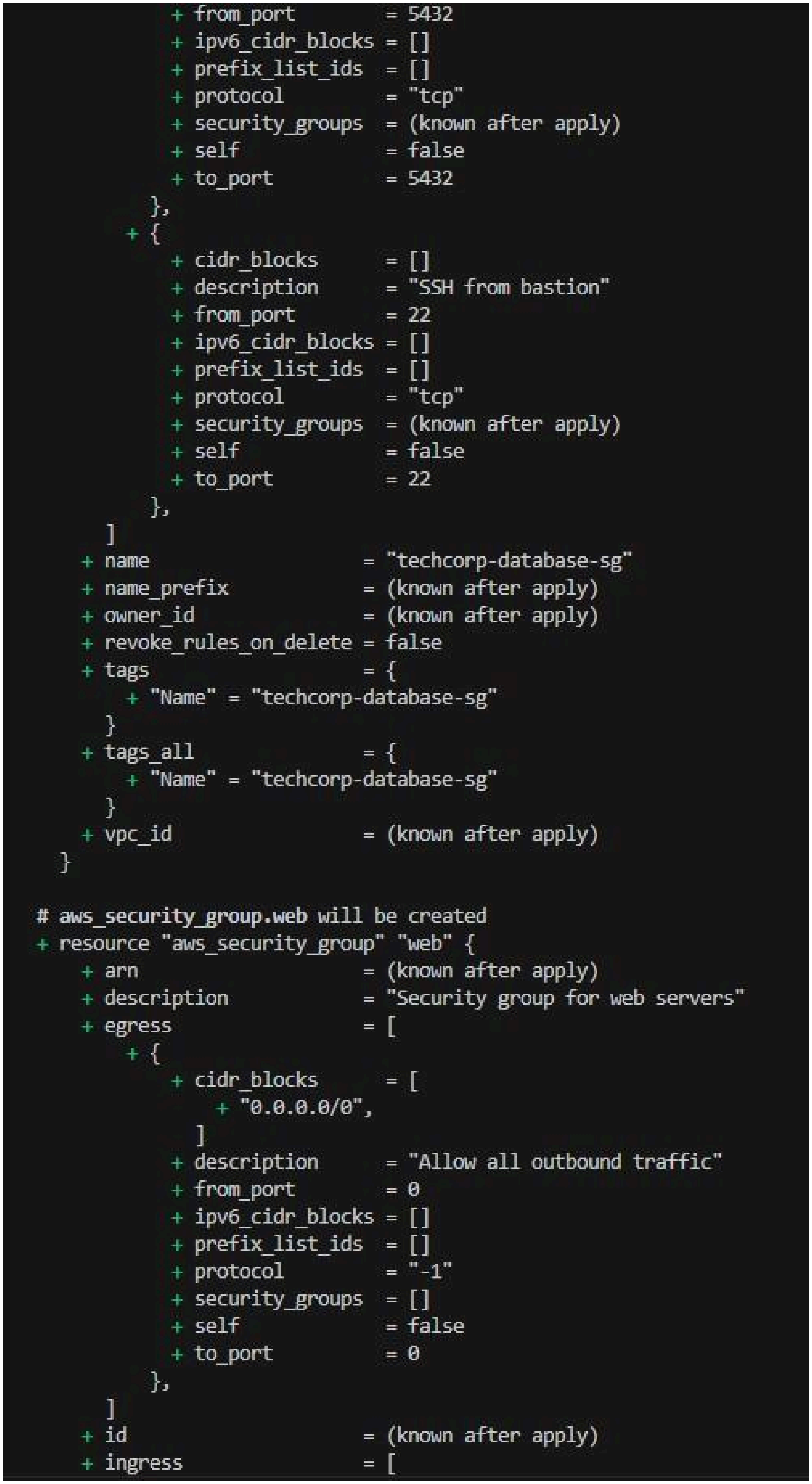
```
      + owner_id         = (known after apply)
      + propagating_vgws = (known after apply)
      + route            = [
          + {
              + cidr_block                 = "0.0.0.0/0"
              + nat_gateway_id             = (known after apply)
                # (11 unchanged attributes hidden)
            },
        ]
      + tags             = {
          + "Name" = "techcorp-private-rt-1"
        }
      + tags_all         = {
          + "Name" = "techcorp-private-rt-1"
        }
      + vpc_id           = (known after apply)
    }

  # aws_route_table.private_2 will be created
  + resource "aws_route_table" "private_2" {
      + arn              = (known after apply)
      + id               = (known after apply)
      + owner_id         = (known after apply)
      + propagating_vgws = (known after apply)
      + route            = [
          + {
              + cidr_block                 = "0.0.0.0/0"
              + nat_gateway_id             = (known after apply)
                # (11 unchanged attributes hidden)
            },
        ]
      + tags             = {
          + "Name" = "techcorp-private-rt-2"
        }
      + tags_all         = {
          + "Name" = "techcorp-private-rt-2"
        }
      + vpc_id           = (known after apply)
    }

  # aws_route_table.public 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            = [
          + {
              + cidr_block                 = "0.0.0.0/0"
              + gateway_id                 = (known after apply)
                # (11 unchanged attributes hidden)
            },
        ]
      + tags             = {
```



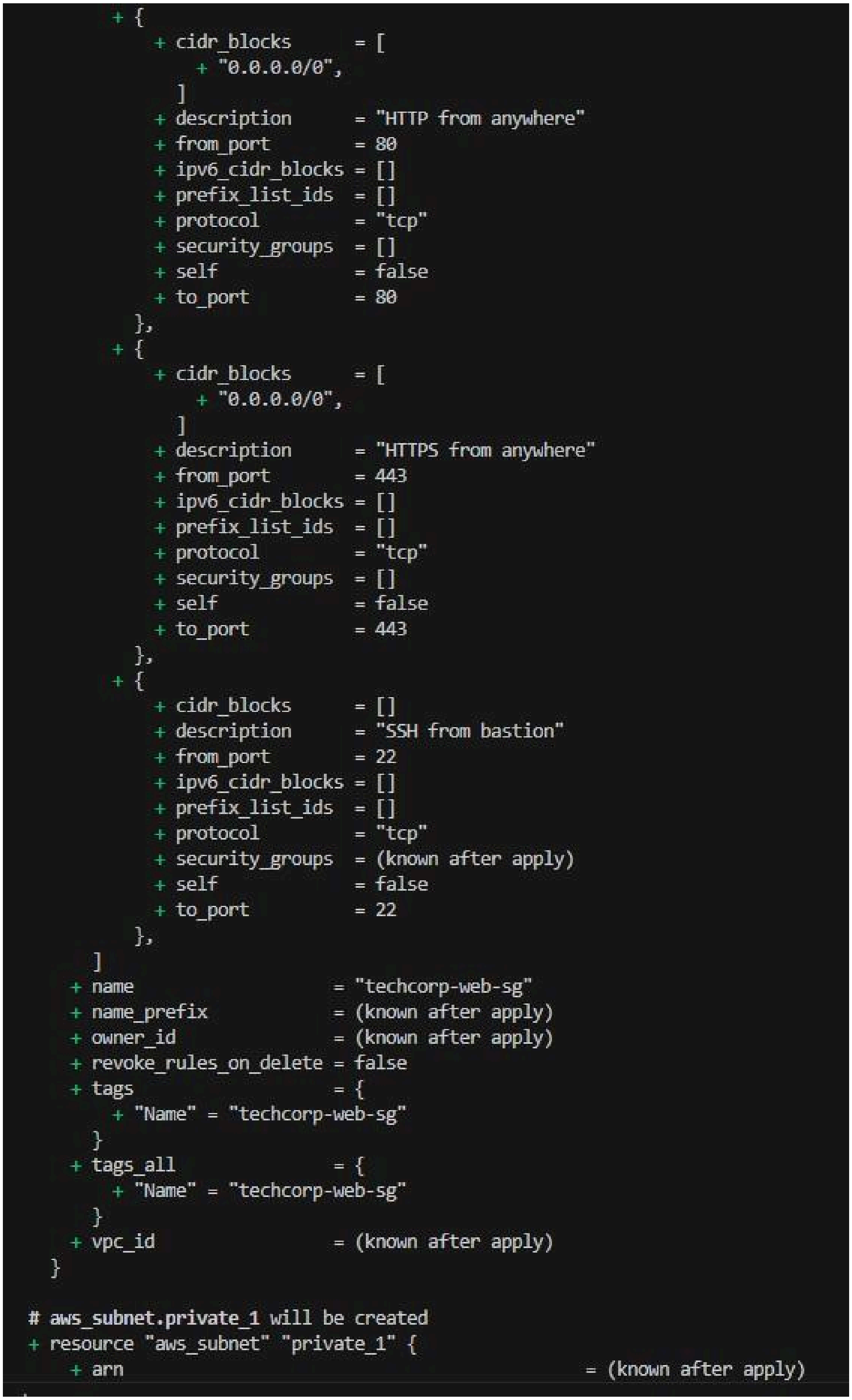
```
          + "Name" = "techcorp-public-rt"
        }
      + tags_all         = {
          + "Name" = "techcorp-public-rt"
        }
      + vpc_id           = (known after apply)
    }

  # aws_route_table_association.private_1 will be created
  + resource "aws_route_table_association" "private_1" {
      + id             = (known after apply)
      + route_table_id = (known after apply)
      + subnet_id      = (known after apply)
    }

  # aws_route_table_association.private_2 will be created
  + resource "aws_route_table_association" "private_2" {
      + id             = (known after apply)
      + route_table_id = (known after apply)
      + subnet_id      = (known after apply)
    }

  # aws_route_table_association.public_1 will be created
  + resource "aws_route_table_association" "public_1" {
      + id             = (known after apply)
      + route_table_id = (known after apply)
      + subnet_id      = (known after apply)
    }

  # aws_route_table_association.public_2 will be created
  + resource "aws_route_table_association" "public_2" {
      + id             = (known after apply)
      + route_table_id = (known after apply)
      + subnet_id      = (known after apply)
    }

  # aws_security_group.bastion will be created
  + resource "aws_security_group" "bastion" {
      + arn                    = (known after apply)
      + description            = "Security group for bastion host"
      + egress                 = [
          + {
              + cidr_blocks      = [
                  + "0.0.0.0/0",
                ]
              + description      = "Allow all outbound traffic"
              + from_port        = 0
              + ipv6_cidr_blocks = []
              + prefix_list_ids  = []
              + protocol         = "-1"
              + security_groups  = []
              + self             = false
              + to_port          = 0
            },
```



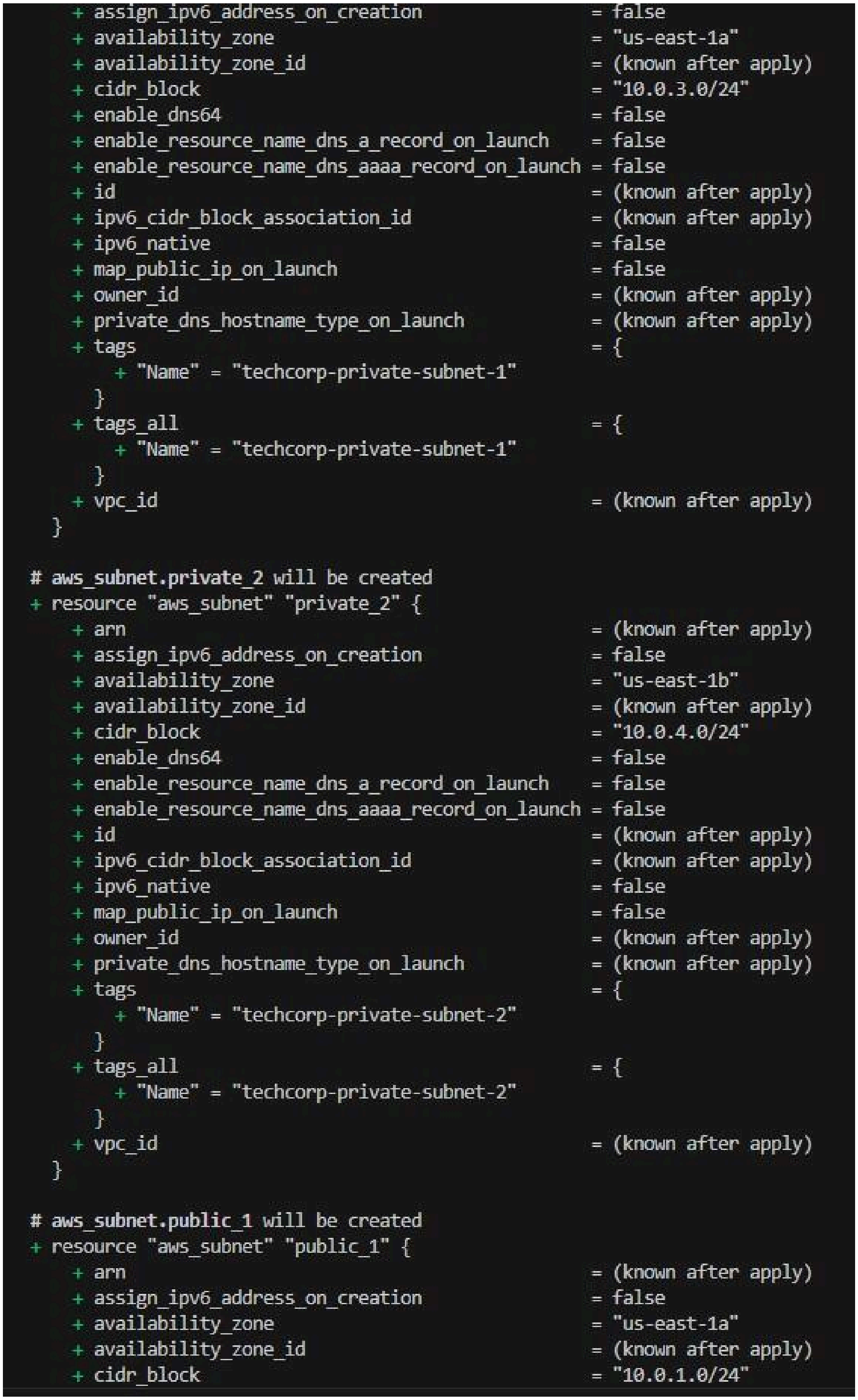
```
      ]
    + id                     = (known after apply)
    + ingress                = [
        + {
            + cidr_blocks      = [
                + "41.73.1.67/32",
              ]
            + description      = "SSH from your IP"
            + from_port        = 22
            + ipv6_cidr_blocks = []
            + prefix_list_ids  = []
            + protocol         = "tcp"
            + security_groups  = []
            + self             = false
            + to_port          = 22
          },
      ]
    + name                   = "techcorp-bastion-sg"
    + name_prefix            = (known after apply)
    + owner_id               = (known after apply)
    + revoke_rules_on_delete = false
    + tags                   = {
        + "Name" = "techcorp-bastion-sg"
      }
    + tags_all               = {
        + "Name" = "techcorp-bastion-sg"
      }
    + vpc_id                 = (known after apply)
  }

# aws_security_group.database will be created
+ resource "aws_security_group" "database" {
    + arn                    = (known after apply)
    + description            = "Security group for database server"
    + egress                 = [
        + {
            + cidr_blocks      = [
                + "0.0.0.0/0",
              ]
            + description      = "Allow all outbound traffic"
            + from_port        = 0
            + ipv6_cidr_blocks = []
            + prefix_list_ids  = []
            + protocol         = "-1"
            + security_groups  = []
            + self             = false
            + to_port          = 0
          },
      ]
    + id                     = (known after apply)
    + ingress                = [
        + {
            + cidr_blocks      = []
            + description      = "PostgreSQL from web servers"
```



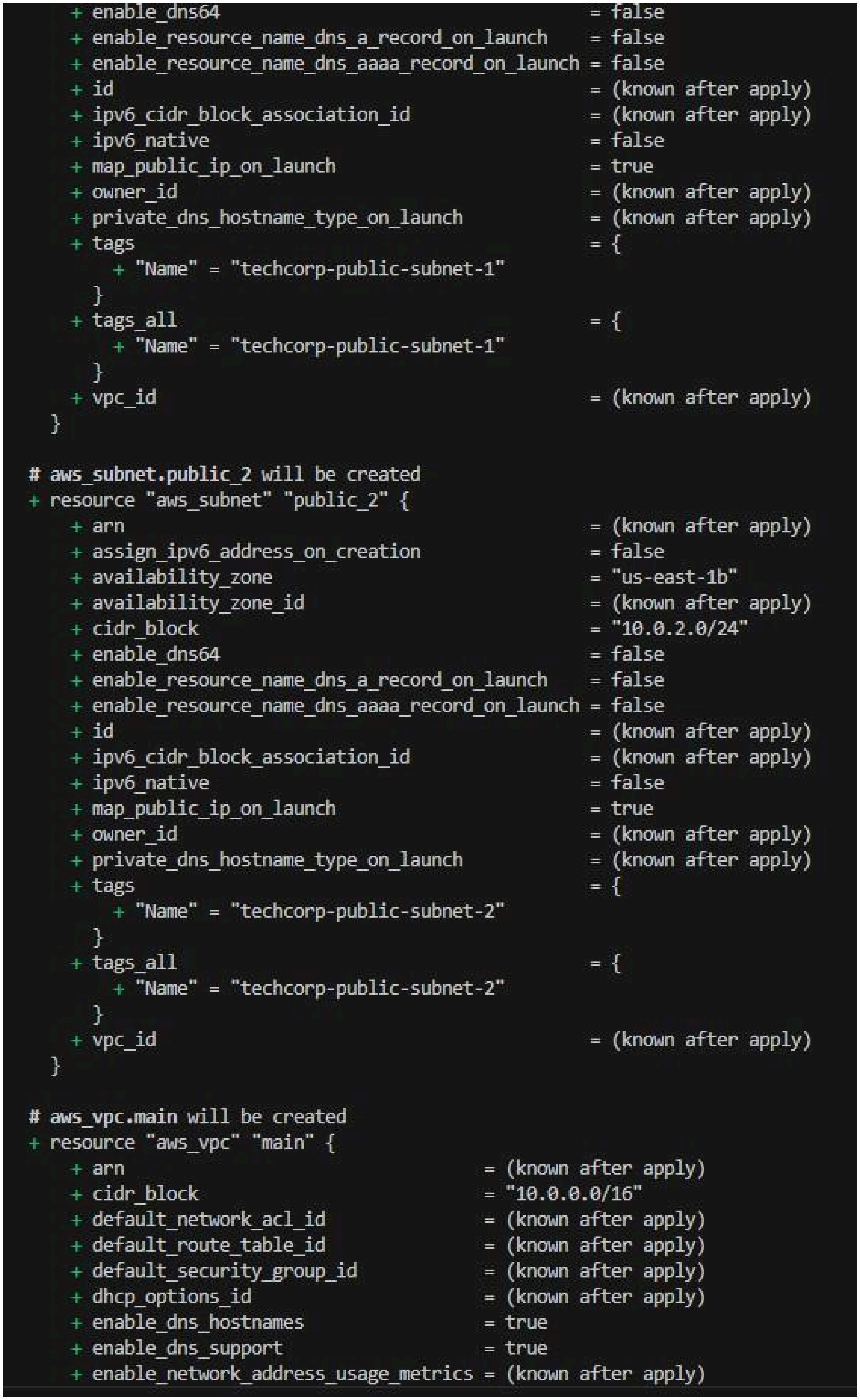
```
              + from_port        = 5432
              + ipv6_cidr_blocks = []
              + prefix_list_ids  = []
              + protocol         = "tcp"
              + security_groups  = (known after apply)
              + self             = false
              + to_port          = 5432
            },
          + {
              + cidr_blocks      = []
              + description      = "SSH from bastion"
              + from_port        = 22
              + ipv6_cidr_blocks = []
              + prefix_list_ids  = []
              + protocol         = "tcp"
              + security_groups  = (known after apply)
              + self             = false
              + to_port          = 22
            },
        ]
      + name                   = "techcorp-database-sg"
      + name_prefix            = (known after apply)
      + owner_id               = (known after apply)
      + revoke_rules_on_delete = false
      + tags                   = {
          + "Name" = "techcorp-database-sg"
        }
      + tags_all               = {
          + "Name" = "techcorp-database-sg"
        }
      + vpc_id                 = (known after apply)
    }

  # aws_security_group.web will be created
  + resource "aws_security_group" "web" {
      + arn                    = (known after apply)
      + description            = "Security group for web servers"
      + egress                 = [
          + {
              + cidr_blocks      = [
                  + "0.0.0.0/0",
                ]
              + description      = "Allow all outbound traffic"
              + from_port        = 0
              + ipv6_cidr_blocks = []
              + prefix_list_ids  = []
              + protocol         = "-1"
              + security_groups  = []
              + self             = false
              + to_port          = 0
            },
        ]
      + id                     = (known after apply)
      + ingress                = [
```



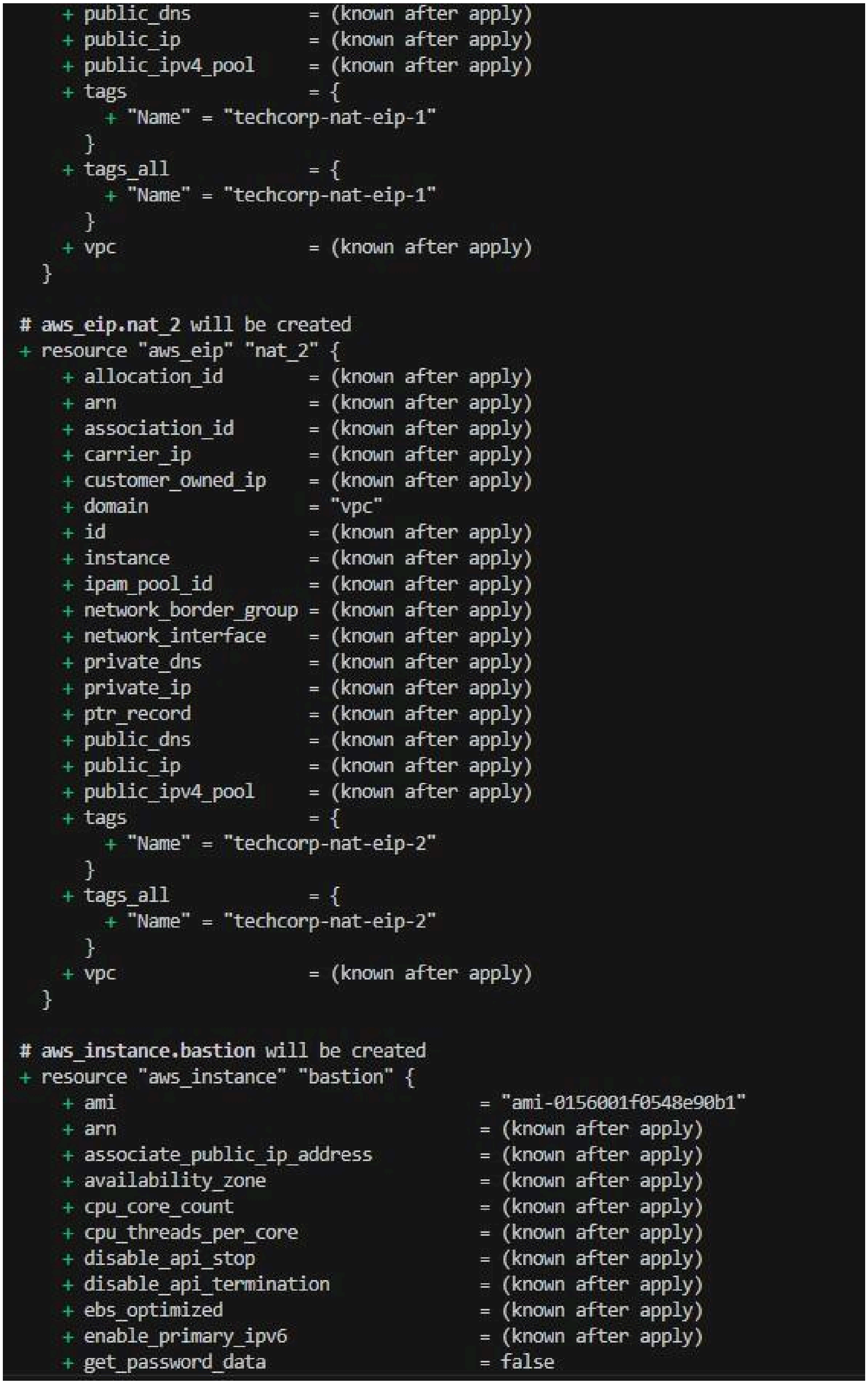
```
        + {
            + cidr_blocks      = [
                + "0.0.0.0/0",
              ]
            + description      = "HTTP from anywhere"
            + from_port        = 80
            + ipv6_cidr_blocks = []
            + prefix_list_ids  = []
            + protocol         = "tcp"
            + security_groups  = []
            + self             = false
            + to_port          = 80
          },
        + {
            + cidr_blocks      = [
                + "0.0.0.0/0",
              ]
            + description      = "HTTPS from anywhere"
            + from_port        = 443
            + ipv6_cidr_blocks = []
            + prefix_list_ids  = []
            + protocol         = "tcp"
            + security_groups  = []
            + self             = false
            + to_port          = 443
          },
        + {
            + cidr_blocks      = []
            + description      = "SSH from bastion"
            + from_port        = 22
            + ipv6_cidr_blocks = []
            + prefix_list_ids  = []
            + protocol         = "tcp"
            + security_groups  = (known after apply)
            + self             = false
            + to_port          = 22
          },
      ]
    + name                   = "techcorp-web-sg"
    + name_prefix            = (known after apply)
    + owner_id               = (known after apply)
    + revoke_rules_on_delete = false
    + tags                   = {
        + "Name" = "techcorp-web-sg"
      }
    + tags_all               = {
        + "Name" = "techcorp-web-sg"
      }
    + vpc_id                 = (known after apply)
  }

# aws_subnet.private_1 will be created
+ resource "aws_subnet" "private_1" {
    + arn                                            = (known after apply)
```



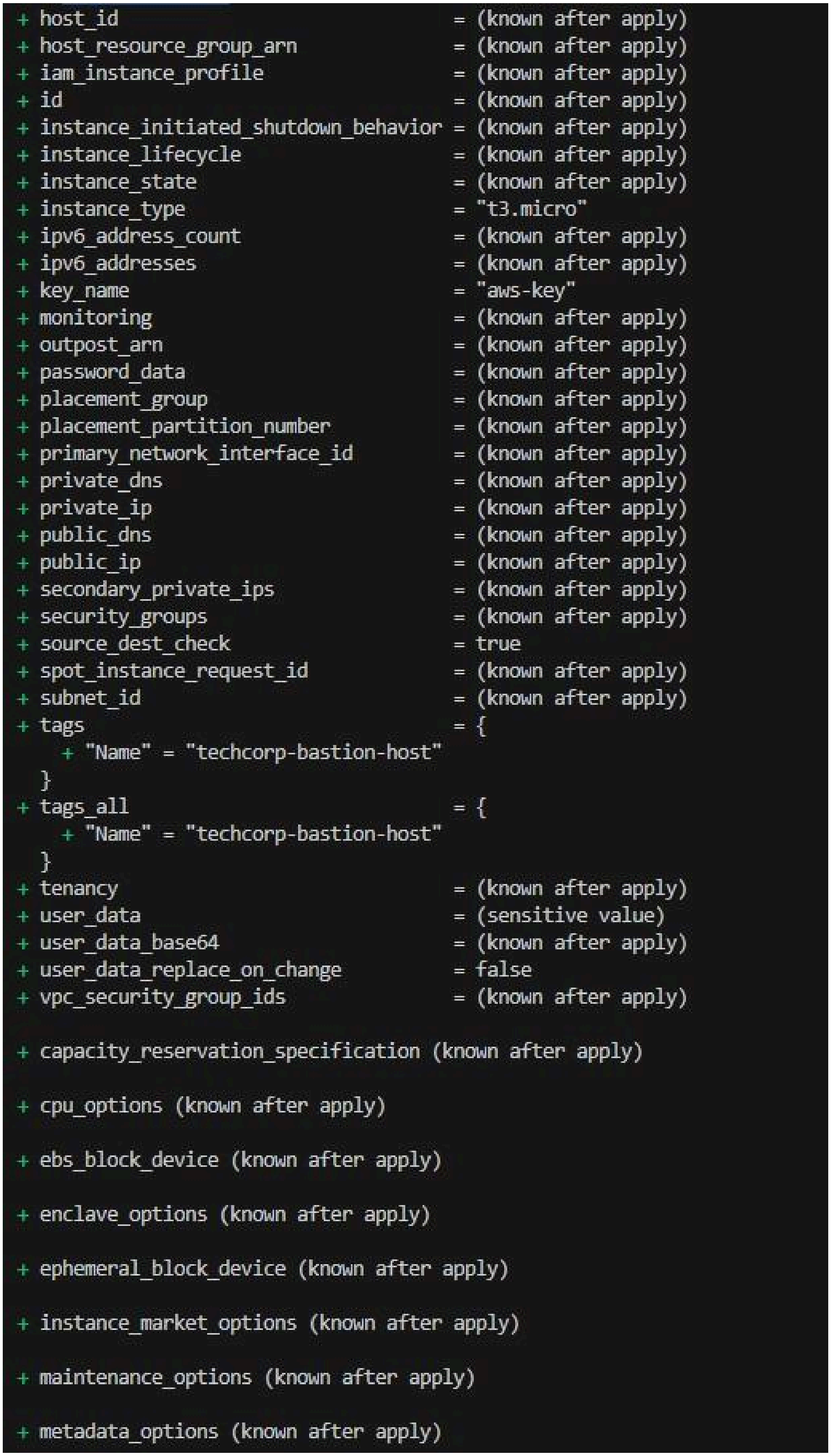
```
      + assign_ipv6_address_on_creation                = false
      + availability_zone                              = "us-east-1a"
      + availability_zone_id                           = (known after apply)
      + cidr_block                                     = "10.0.3.0/24"
      + enable_dns64                                   = 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                                           = {
          + "Name" = "techcorp-private-subnet-1"
        }
      + tags_all                                       = {
          + "Name" = "techcorp-private-subnet-1"
        }
      + vpc_id                                         = (known after apply)
    }

  # aws_subnet.private_2 will be created
  + resource "aws_subnet" "private_2" {
      + arn                                            = (known after apply)
      + assign_ipv6_address_on_creation                = false
      + availability_zone                              = "us-east-1b"
      + availability_zone_id                           = (known after apply)
      + cidr_block                                     = "10.0.4.0/24"
      + 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" = "techcorp-private-subnet-2"
        }
      + tags_all                                       = {
          + "Name" = "techcorp-private-subnet-2"
        }
      + vpc_id                                         = (known after apply)
    }

  # aws_subnet.public_1 will be created
  + resource "aws_subnet" "public_1" {
      + 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"
```



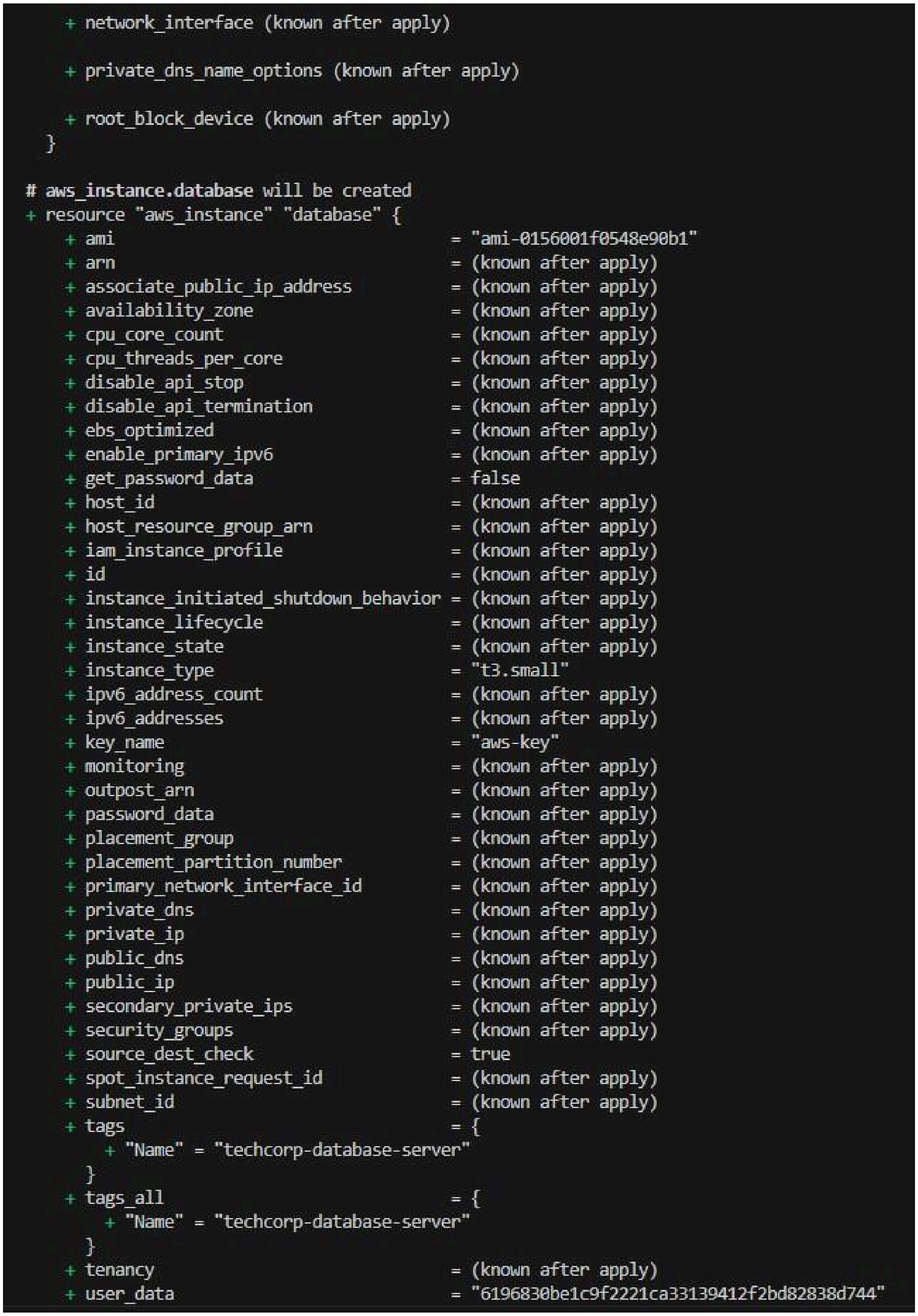
```
      + enable_dns64                                   = false
      + enable_resource_name_dns_a_record_on_launch    = false
      + enable_resource_name_dns_aaaa_record_on_launch = false
      + id                                             = (known after apply)
      + ipv6_cidr_block_association_id                 = (known after apply)
      + ipv6_native                                    = false
      + map_public_ip_on_launch                        = true
      + owner_id                                       = (known after apply)
      + private_dns_hostname_type_on_launch            = (known after apply)
      + tags                                           = {
          + "Name" = "techcorp-public-subnet-1"
        }
      + tags_all                                       = {
          + "Name" = "techcorp-public-subnet-1"
        }
      + vpc_id                                         = (known after apply)
    }

  # aws_subnet.public_2 will be created
  + resource "aws_subnet" "public_2" {
      + 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                        = true
      + owner_id                                       = (known after apply)
      + private_dns_hostname_type_on_launch            = (known after apply)
      + tags                                           = {
          + "Name" = "techcorp-public-subnet-2"
        }
      + tags_all                                       = {
          + "Name" = "techcorp-public-subnet-2"
        }
      + vpc_id                                         = (known after apply)
    }

  # aws_vpc.main will be created
  + resource "aws_vpc" "main" {
      + 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)
```



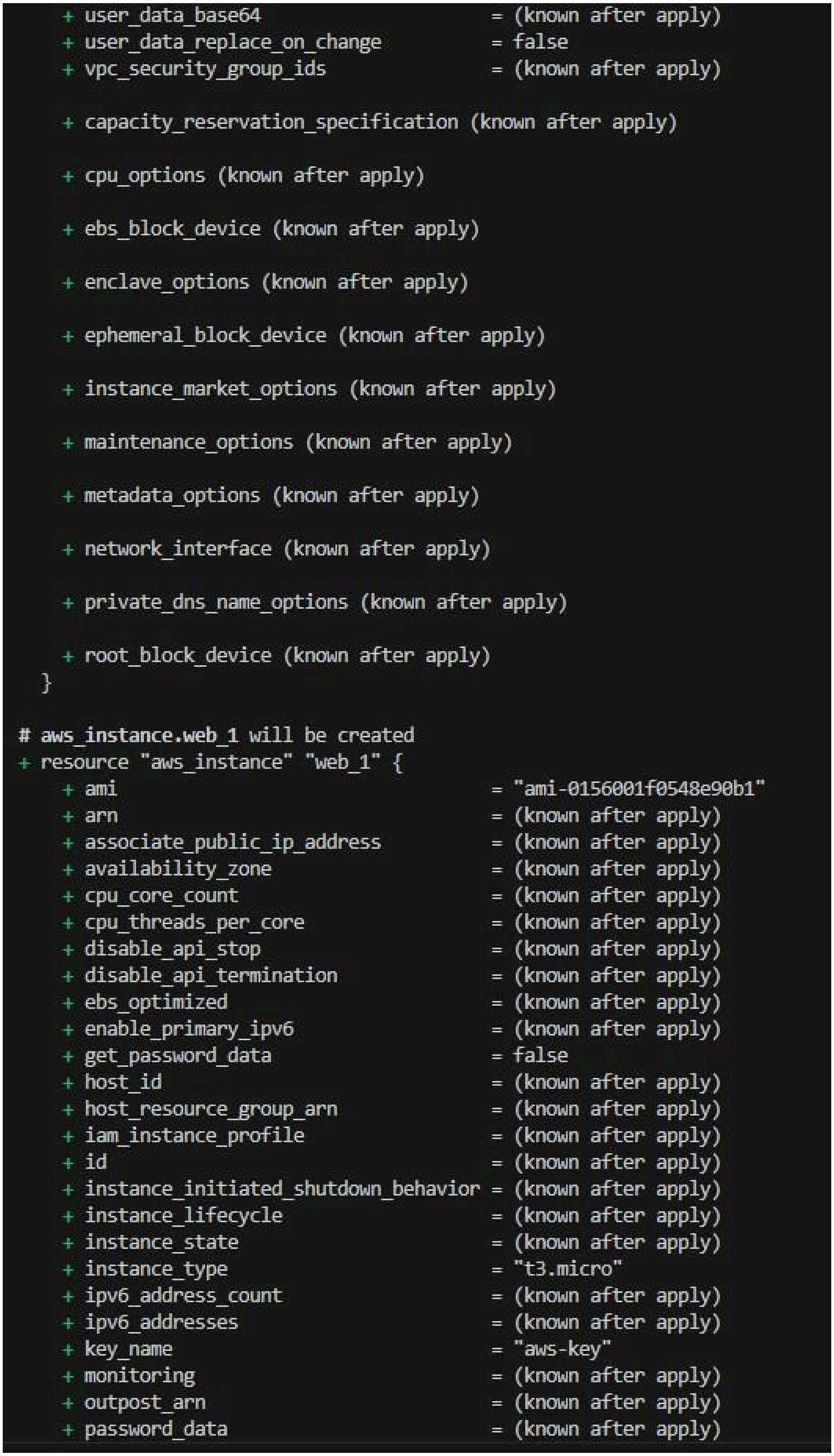
```
    + public_dns           = (known after apply)
    + public_ip            = (known after apply)
    + public_ipv4_pool     = (known after apply)
    + tags                 = {
        + "Name" = "techcorp-nat-eip-1"
      }
    + tags_all             = {
        + "Name" = "techcorp-nat-eip-1"
      }
    + vpc                  = (known after apply)
  }

# aws_eip.nat_2 will be created
+ resource "aws_eip" "nat_2" {
    + 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                 = {
        + "Name" = "techcorp-nat-eip-2"
      }
    + tags_all             = {
        + "Name" = "techcorp-nat-eip-2"
      }
    + vpc                  = (known after apply)
  }

# aws_instance.bastion will be created
+ resource "aws_instance" "bastion" {
    + ami                                  = "ami-0156001f0548e90b1"
    + 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)
    + enable_primary_ipv6                  = (known after apply)
    + get_password_data                    = false
```



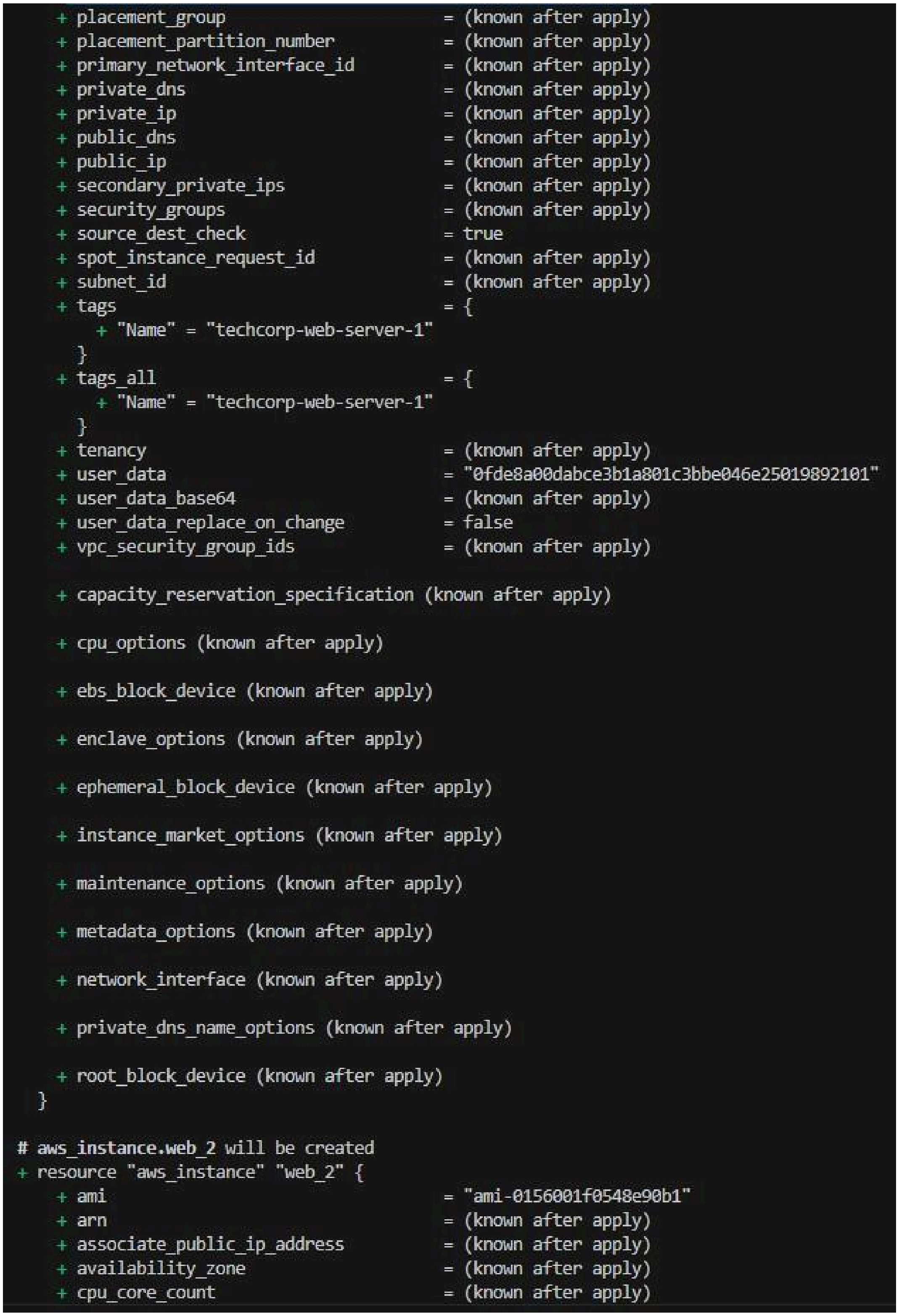
```
+ host_id                              = (known after apply)
+ host_resource_group_arn              = (known after apply)
+ iam_instance_profile                 = (known after apply)
+ id                                   = (known after apply)
+ instance_initiated_shutdown_behavior = (known after apply)
+ instance_lifecycle                   = (known after apply)
+ instance_state                       = (known after apply)
+ instance_type                        = "t3.micro"
+ ipv6_address_count                   = (known after apply)
+ ipv6_addresses                       = (known after apply)
+ key_name                             = "aws-key"
+ 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                                 = {
    + "Name" = "techcorp-bastion-host"
  }
+ tags_all                             = {
    + "Name" = "techcorp-bastion-host"
  }
+ tenancy                              = (known after apply)
+ user_data                            = (sensitive value)
+ user_data_base64                     = (known after apply)
+ user_data_replace_on_change          = false
+ vpc_security_group_ids               = (known after apply)

+ capacity_reservation_specification (known after apply)

+ cpu_options (known after apply)

+ ebs_block_device (known after apply)

+ enclave_options (known after apply)

+ ephemeral_block_device (known after apply)

+ instance_market_options (known after apply)

+ maintenance_options (known after apply)

+ metadata_options (known after apply)
```



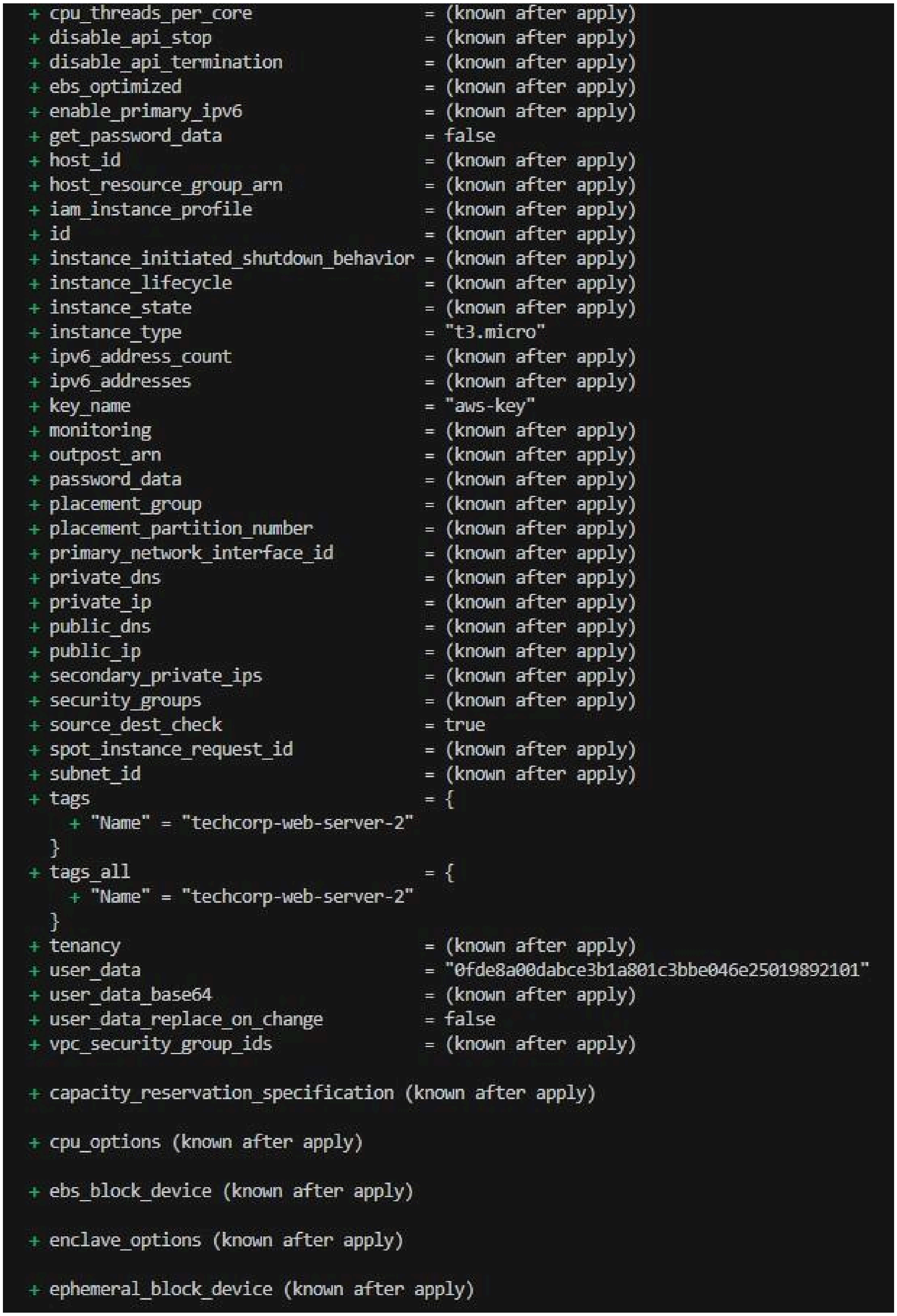
```
      + network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }

  # aws_instance.database will be created
  + resource "aws_instance" "database" {
      + ami                                  = "ami-0156001f0548e90b1"
      + 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)
      + enable_primary_ipv6                  = (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                        = "t3.small"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = "aws-key"
      + 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                                 = {
          + "Name" = "techcorp-database-server"
        }
      + tags_all                             = {
          + "Name" = "techcorp-database-server"
        }
      + tenancy                              = (known after apply)
      + user_data                            = "6196830be1c9f2221ca33139412f2bd82838d744"
```



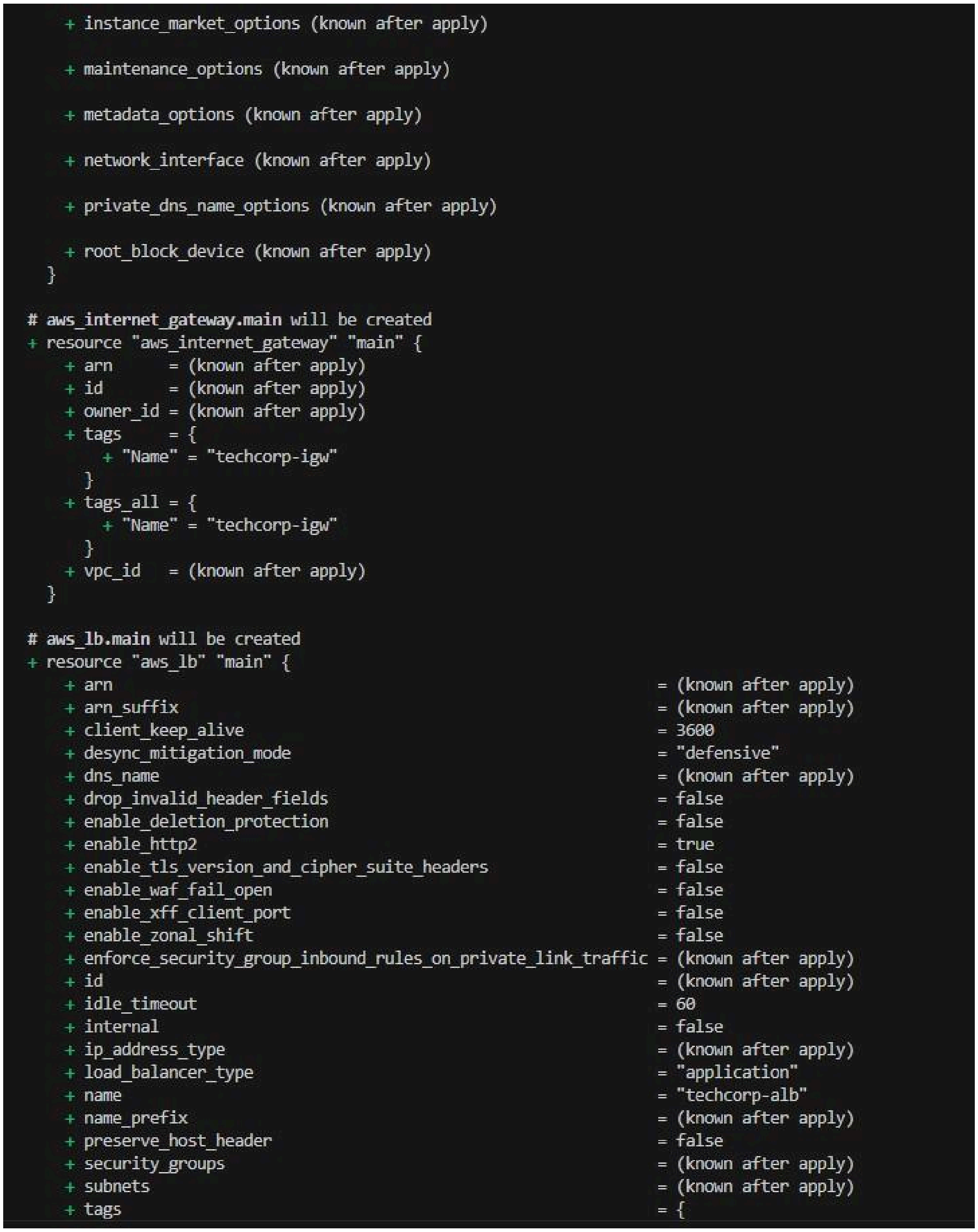
```
      + user_data_base64                     = (known after apply)
      + user_data_replace_on_change          = false
      + vpc_security_group_ids               = (known after apply)

      + capacity_reservation_specification (known after apply)

      + cpu_options (known after apply)

      + ebs_block_device (known after apply)

      + enclave_options (known after apply)

      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }

  # aws_instance.web_1 will be created
  + resource "aws_instance" "web_1" {
      + ami                                  = "ami-0156001f0548e90b1"
      + 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)
      + enable_primary_ipv6                  = (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                        = "t3.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = "aws-key"
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
```



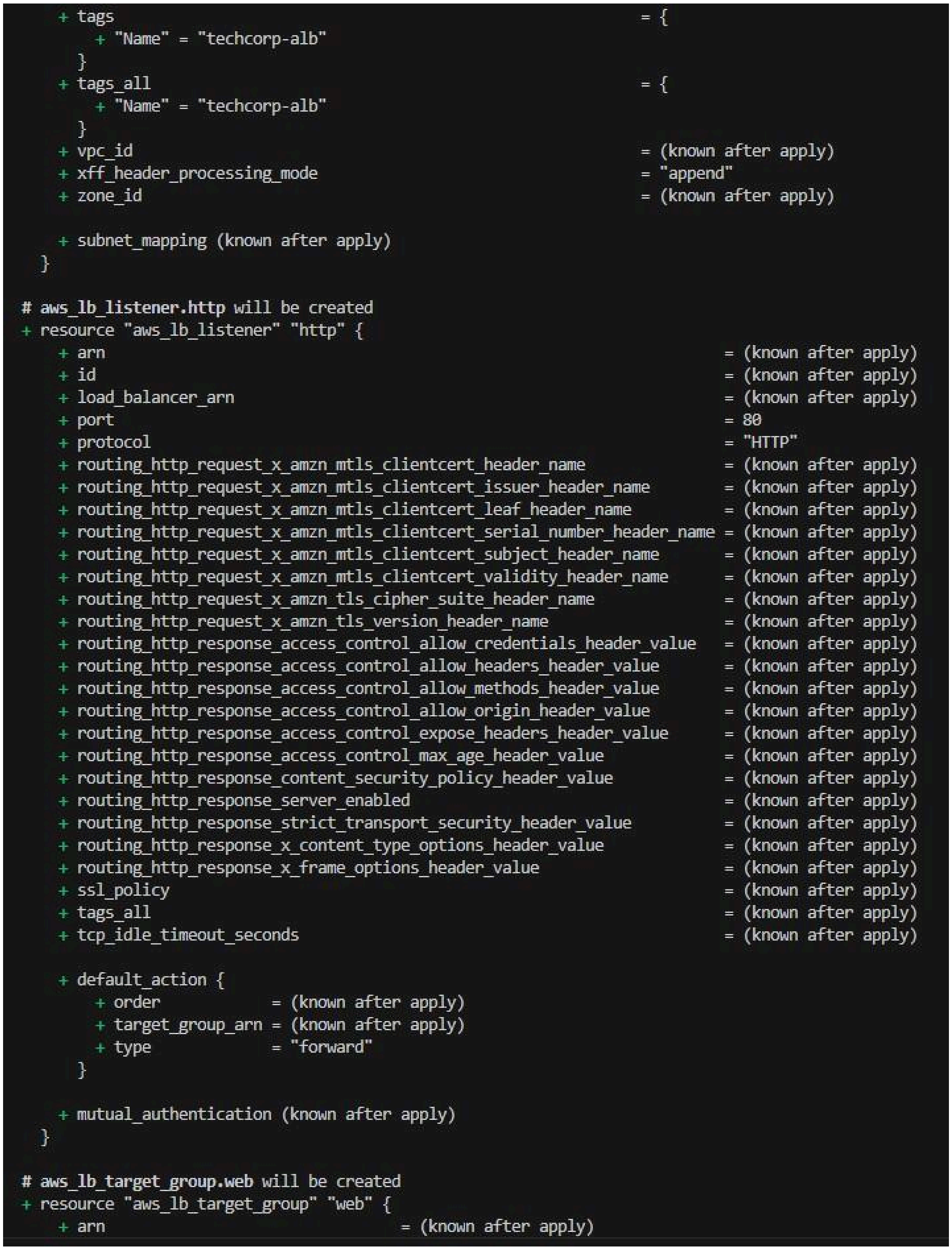
```
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (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                                 = {
          + "Name" = "techcorp-web-server-1"
        }
      + tags_all                             = {
          + "Name" = "techcorp-web-server-1"
        }
      + tenancy                              = (known after apply)
      + user_data                            = "0fde8a00dabce3b1a801c3bbe046e25019892101"
      + user_data_base64                     = (known after apply)
      + user_data_replace_on_change          = false
      + vpc_security_group_ids               = (known after apply)

      + capacity_reservation_specification (known after apply)

      + cpu_options (known after apply)

      + ebs_block_device (known after apply)

      + enclave_options (known after apply)

      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }

  # aws_instance.web_2 will be created
  + resource "aws_instance" "web_2" {
      + ami                                  = "ami-0156001f0548e90b1"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
```



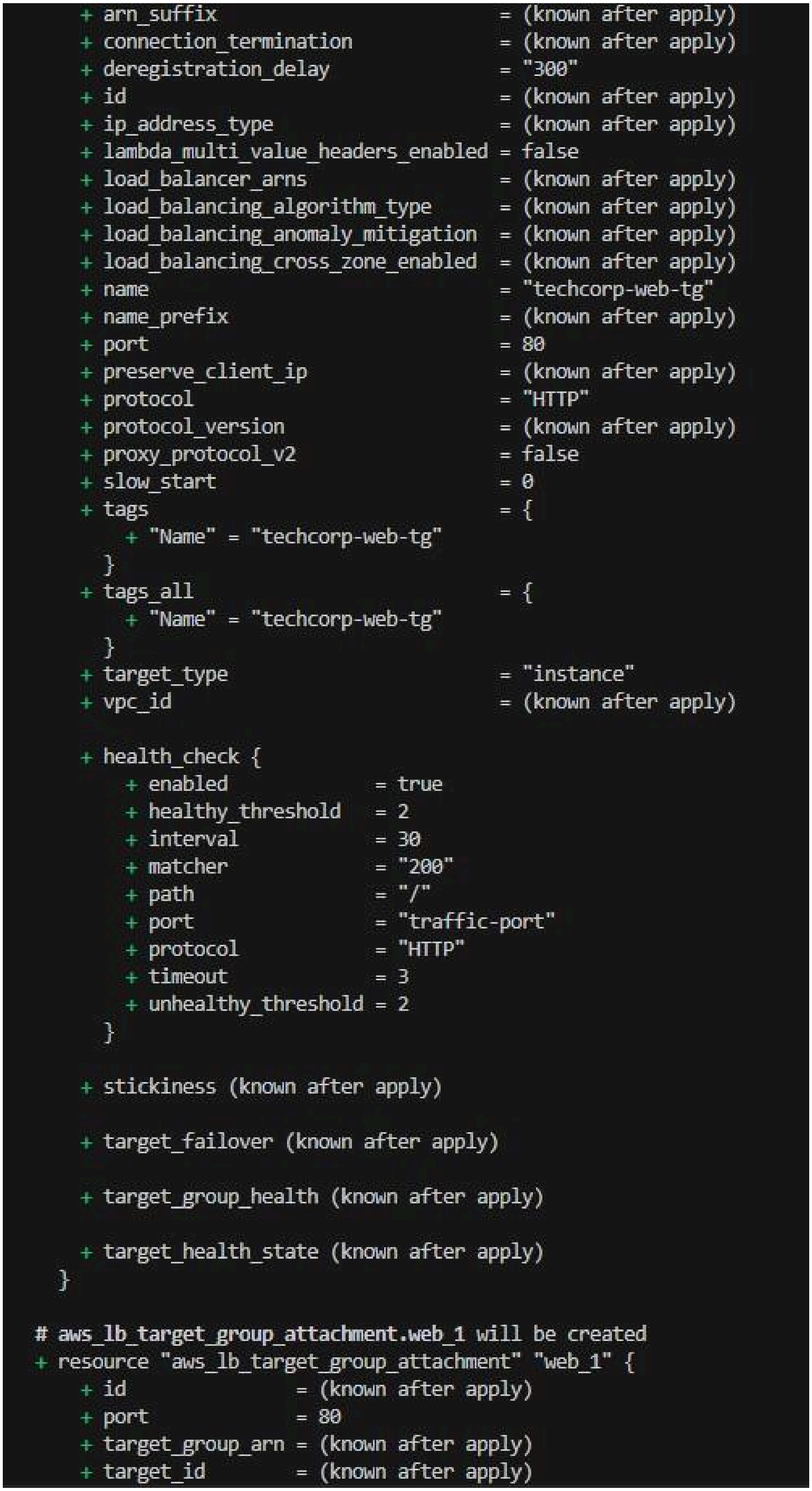
```
  + cpu_threads_per_core                 = (known after apply)
  + disable_api_stop                     = (known after apply)
  + disable_api_termination              = (known after apply)
  + ebs_optimized                        = (known after apply)
  + enable_primary_ipv6                  = (known 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                        = "t3.micro"
  + ipv6_address_count                   = (known after apply)
  + ipv6_addresses                       = (known after apply)
  + key_name                             = "aws-key"
  + 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                                 = {
      + "Name" = "techcorp-web-server-2"
    }
  + tags_all                             = {
      + "Name" = "techcorp-web-server-2"
    }
  + tenancy                              = (known after apply)
  + user_data                            = "0fde8a00dabce3b1a801c3bbe046e25019892101"
  + user_data_base64                     = (known after apply)
  + user_data_replace_on_change          = false
  + vpc_security_group_ids               = (known after apply)

  + capacity_reservation_specification (known after apply)

  + cpu_options (known after apply)

  + ebs_block_device (known after apply)

  + enclave_options (known after apply)

  + ephemeral_block_device (known after apply)
```



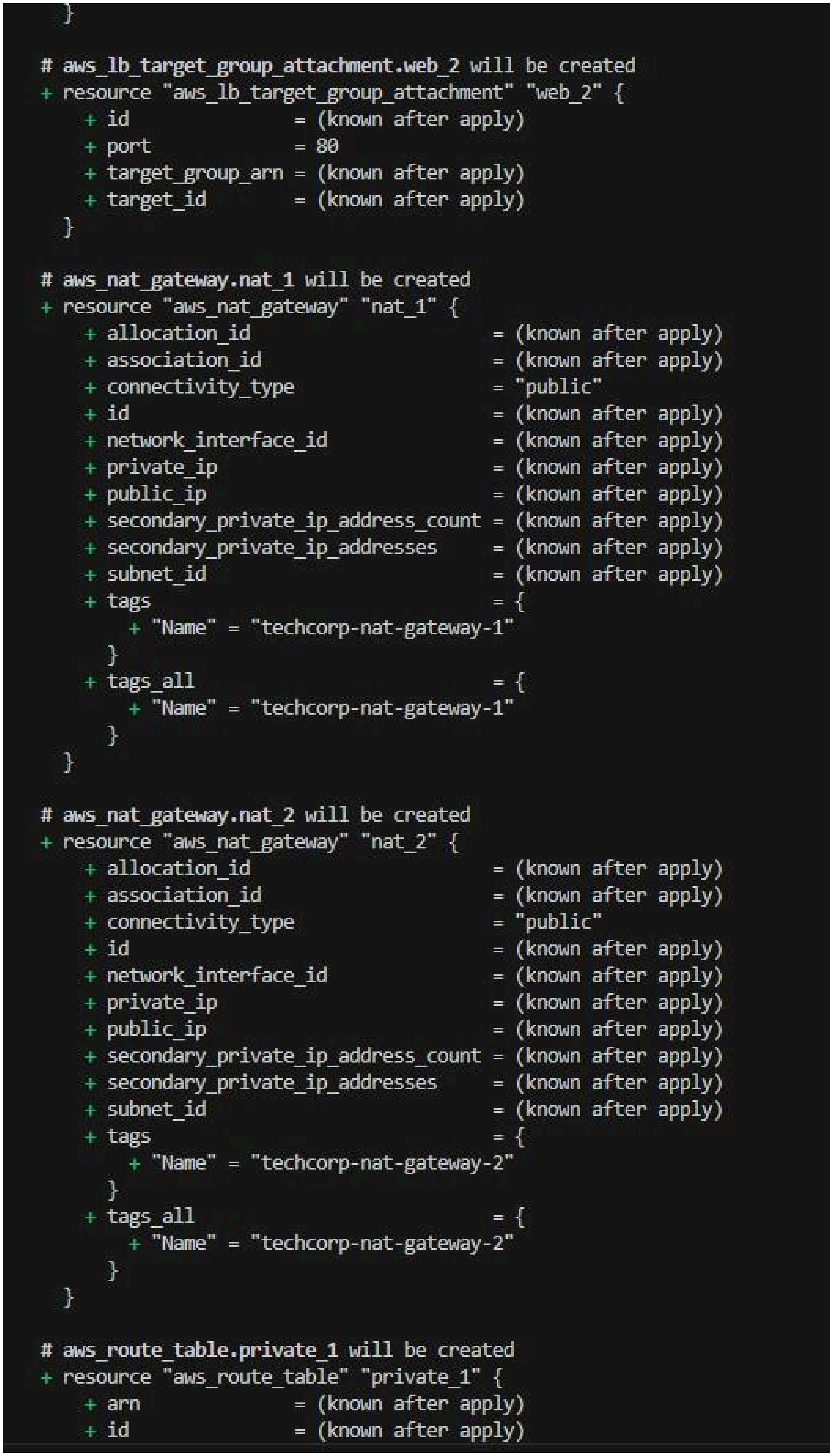
```
      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }

  # aws_internet_gateway.main will be created
  + resource "aws_internet_gateway" "main" {
      + arn      = (known after apply)
      + id       = (known after apply)
      + owner_id = (known after apply)
      + tags     = {
          + "Name" = "techcorp-igw"
        }
      + tags_all = {
          + "Name" = "techcorp-igw"
        }
      + vpc_id   = (known after apply)
    }

  # aws_lb.main will be created
  + resource "aws_lb" "main" {
      + arn                                                          = (known after apply)
      + arn_suffix                                                   = (known after apply)
      + client_keep_alive                                            = 3600
      + desync_mitigation_mode                                       = "defensive"
      + dns_name                                                     = (known after apply)
      + drop_invalid_header_fields                                   = false
      + enable_deletion_protection                                   = false
      + enable_http2                                                 = true
      + enable_tls_version_and_cipher_suite_headers                  = false
      + enable_waf_fail_open                                         = false
      + enable_xff_client_port                                       = false
      + enable_zonal_shift                                           = false
      + enforce_security_group_inbound_rules_on_private_link_traffic = (known after apply)
      + id                                                           = (known after apply)
      + idle_timeout                                                 = 60
      + internal                                                     = false
      + ip_address_type                                              = (known after apply)
      + load_balancer_type                                           = "application"
      + name                                                         = "techcorp-alb"
      + name_prefix                                                  = (known after apply)
      + preserve_host_header                                         = false
      + security_groups                                              = (known after apply)
      + subnets                                                      = (known after apply)
      + tags                                                         = {
```



```
      + tags                                                         = {
          + "Name" = "techcorp-alb"
        }
      + tags_all                                                     = {
          + "Name" = "techcorp-alb"
        }
      + vpc_id                                                       = (known after apply)
      + xff_header_processing_mode                                   = "append"
      + zone_id                                                      = (known after apply)

      + subnet_mapping (known after apply)
    }

  # aws_lb_listener.http will be created
  + resource "aws_lb_listener" "http" {
      + arn                                                                   = (known after apply)
      + id                                                                    = (known after apply)
      + load_balancer_arn                                                     = (known after apply)
      + port                                                                  = 80
      + protocol                                                              = "HTTP"
      + routing_http_request_x_amzn_mtls_clientcert_header_name               = (known after apply)
      + routing_http_request_x_amzn_mtls_clientcert_issuer_header_name        = (known after apply)
      + routing_http_request_x_amzn_mtls_clientcert_leaf_header_name          = (known after apply)
      + routing_http_request_x_amzn_mtls_clientcert_serial_number_header_name = (known after apply)
      + routing_http_request_x_amzn_mtls_clientcert_subject_header_name       = (known after apply)
      + routing_http_request_x_amzn_mtls_clientcert_validity_header_name      = (known after apply)
      + routing_http_request_x_amzn_tls_cipher_suite_header_name              = (known after apply)
      + routing_http_request_x_amzn_tls_version_header_name                   = (known after apply)
      + routing_http_response_access_control_allow_credentials_header_value   = (known after apply)
      + routing_http_response_access_control_allow_headers_header_value       = (known after apply)
      + routing_http_response_access_control_allow_methods_header_value       = (known after apply)
      + routing_http_response_access_control_allow_origin_header_value        = (known after apply)
      + routing_http_response_access_control_expose_headers_header_value      = (known after apply)
      + routing_http_response_access_control_max_age_header_value             = (known after apply)
      + routing_http_response_content_security_policy_header_value            = (known after apply)
      + routing_http_response_server_enabled                                  = (known after apply)
      + routing_http_response_strict_transport_security_header_value          = (known after apply)
      + routing_http_response_x_content_type_options_header_value             = (known after apply)
      + routing_http_response_x_frame_options_header_value                    = (known after apply)
      + ssl_policy                                                            = (known after apply)
      + tags_all                                                              = (known after apply)
      + tcp_idle_timeout_seconds                                              = (known after apply)

      + default_action {
          + order            = (known after apply)
          + target_group_arn = (known after apply)
          + type             = "forward"
        }

      + mutual_authentication (known after apply)
    }

  # aws_lb_target_group.web will be created
  + resource "aws_lb_target_group" "web" {
      + arn                                = (known after apply)
```

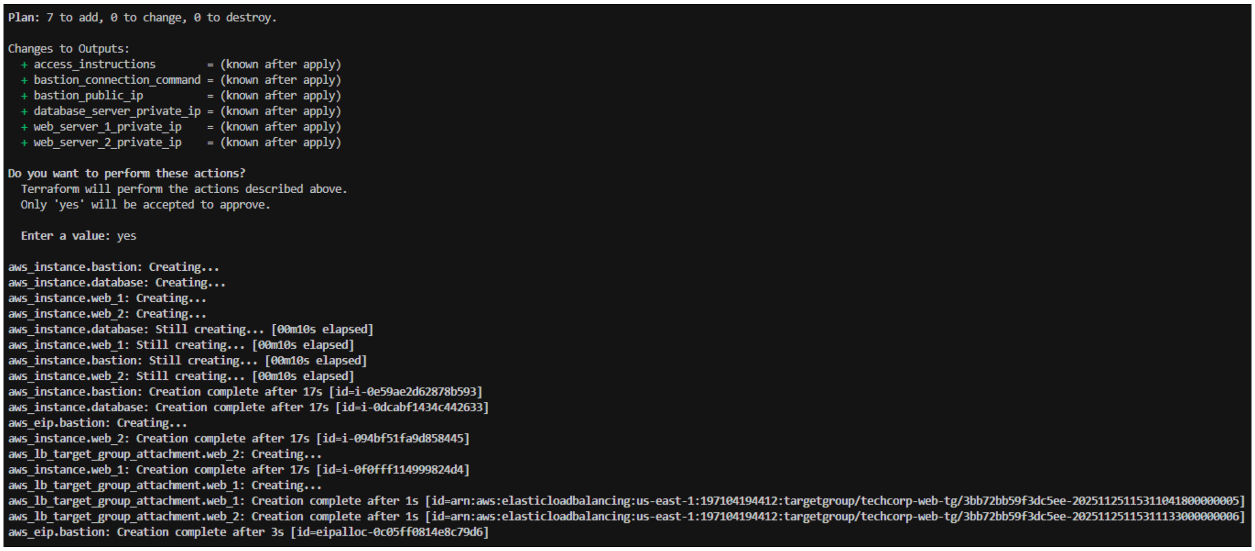


```
      + arn_suffix                         = (known after apply)
      + connection_termination             = (known after apply)
      + deregistration_delay               = "300"
      + id                                 = (known after apply)
      + ip_address_type                    = (known after apply)
      + lambda_multi_value_headers_enabled = false
      + load_balancer_arns                 = (known after apply)
      + load_balancing_algorithm_type      = (known after apply)
      + load_balancing_anomaly_mitigation  = (known after apply)
      + load_balancing_cross_zone_enabled  = (known after apply)
      + name                               = "techcorp-web-tg"
      + name_prefix                        = (known after apply)
      + port                               = 80
      + preserve_client_ip                 = (known after apply)
      + protocol                           = "HTTP"
      + protocol_version                   = (known after apply)
      + proxy_protocol_v2                  = false
      + slow_start                         = 0
      + tags                               = {
          + "Name" = "techcorp-web-tg"
        }
      + tags_all                           = {
          + "Name" = "techcorp-web-tg"
        }
      + target_type                        = "instance"
      + vpc_id                             = (known after apply)

      + health_check {
          + enabled             = true
          + healthy_threshold   = 2
          + interval            = 30
          + matcher             = "200"
          + path                = "/"
          + port                = "traffic-port"
          + protocol            = "HTTP"
          + timeout             = 3
          + unhealthy_threshold = 2
        }

      + stickiness (known after apply)

      + target_failover (known after apply)

      + target_group_health (known after apply)

      + target_health_state (known after apply)
    }

  # aws_lb_target_group_attachment.web_1 will be created
  + resource "aws_lb_target_group_attachment" "web_1" {
      + id               = (known after apply)
      + port             = 80
      + target_group_arn = (known after apply)
      + target_id        = (known after apply)
```

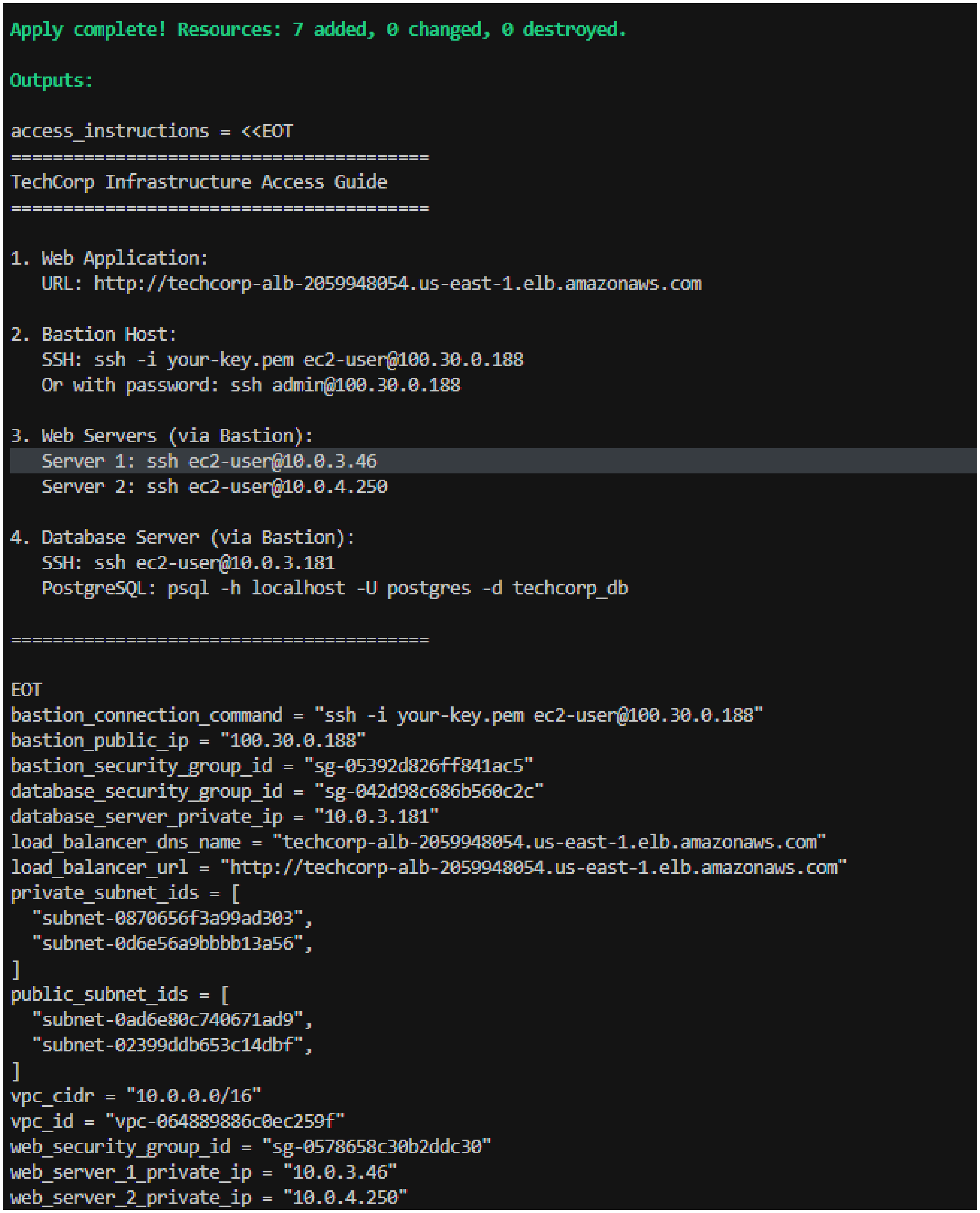


```
    }

  # aws_lb_target_group_attachment.web_2 will be created
  + resource "aws_lb_target_group_attachment" "web_2" {
      + id               = (known after apply)
      + port             = 80
      + target_group_arn = (known after apply)
      + target_id        = (known after apply)
    }

  # aws_nat_gateway.nat_1 will be created
  + resource "aws_nat_gateway" "nat_1" {
      + 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                               = {
          + "Name" = "techcorp-nat-gateway-1"
        }
      + tags_all                           = {
          + "Name" = "techcorp-nat-gateway-1"
        }
    }

  # aws_nat_gateway.nat_2 will be created
  + resource "aws_nat_gateway" "nat_2" {
      + 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                               = {
          + "Name" = "techcorp-nat-gateway-2"
        }
      + tags_all                           = {
          + "Name" = "techcorp-nat-gateway-2"
        }
    }

  # aws_route_table.private_1 will be created
  + resource "aws_route_table" "private_1" {
      + arn              = (known after apply)
      + id               = (known after apply)
```

2. Terraform apply completion

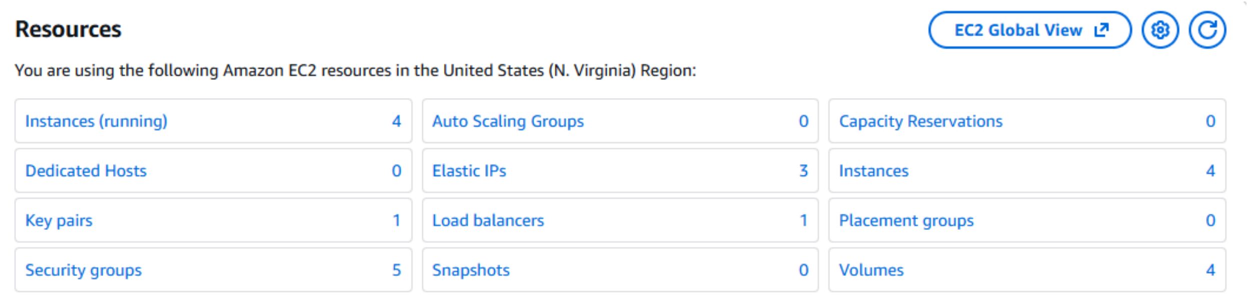


```
Plan: 7 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + access_instructions        = (known after apply)
  + bastion_connection_command = (known after apply)
  + bastion_public_ip          = (known after apply)
  + database_server_private_ip = (known after apply)
  + web_server_1_private_ip    = (known after apply)
  + web_server_2_private_ip    = (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

aws_instance.bastion: Creating...
aws_instance.database: Creating...
aws_instance.web_1: Creating...
aws_instance.web_2: Creating...
aws_instance.database: Still creating... [00m10s elapsed]
aws_instance.web_1: Still creating... [00m10s elapsed]
aws_instance.bastion: Still creating... [00m10s elapsed]
aws_instance.web_2: Still creating... [00m10s elapsed]
aws_instance.bastion: Creation complete after 17s [id=i-0e59ae2d62878b593]
aws_instance.database: Creation complete after 17s [id=i-0dcabf1434c442633]
aws_eip.bastion: Creating...
aws_instance.web_2: Creation complete after 17s [id=i-094bf51fa9d858445]
aws_lb_target_group_attachment.web_2: Creating...
aws_instance.web_1: Creation complete after 17s [id=i-0f0fff114999824d4]
aws_lb_target_group_attachment.web_1: Creating...
aws_lb_target_group_attachment.web_1: Creation complete after 1s [id=arn:aws:elasticloadbalancing:us-east-1:197104194412:targetgroup/techcorp-web-tg/3bb72bb59f3dc5ee-20251125115311041800000005]
aws_lb_target_group_attachment.web_2: Creation complete after 1s [id=arn:aws:elasticloadbalancing:us-east-1:197104194412:targetgroup/techcorp-web-tg/3bb72bb59f3dc5ee-20251125115311133000000006]
aws_eip.bastion: Creation complete after 3s [id=eipalloc-0c05ff0814e8c79d6]
```



```
Apply complete! Resources: 7 added, 0 changed, 0 destroyed.

Outputs:

access_instructions = <<EOT
========================================
TechCorp Infrastructure Access Guide
========================================

1. Web Application:
   URL: http://techcorp-alb-2059948054.us-east-1.elb.amazonaws.com

2. Bastion Host:
   SSH: ssh -i your-key.pem ec2-user@100.30.0.188
   Or with password: ssh admin@100.30.0.188

3. Web Servers (via Bastion):
   Server 1: ssh ec2-user@10.0.3.46
   Server 2: ssh ec2-user@10.0.4.250

4. Database Server (via Bastion):
   SSH: ssh ec2-user@10.0.3.181
   PostgreSQL: psql -h localhost -U postgres -d techcorp_db

========================================

EOT
bastion_connection_command = "ssh -i your-key.pem ec2-user@100.30.0.188"
bastion_public_ip = "100.30.0.188"
bastion_security_group_id = "sg-05392d826ff841ac5"
database_security_group_id = "sg-042d98c686b560c2c"
database_server_private_ip = "10.0.3.181"
load_balancer_dns_name = "techcorp-alb-2059948054.us-east-1.elb.amazonaws.com"
load_balancer_url = "http://techcorp-alb-2059948054.us-east-1.elb.amazonaws.com"
private_subnet_ids = [
  "subnet-0870656f3a99ad303",
  "subnet-0d6e56a9bbbb13a56",
]
public_subnet_ids = [
  "subnet-0ad6e80c740671ad9",
  "subnet-02399ddb653c14dbf",
]
vpc_cidr = "10.0.0.0/16"
vpc_id = "vpc-064889886c0ec259f"
web_security_group_id = "sg-0578658c30b2ddc30"
web_server_1_private_ip = "10.0.3.46"
web_server_2_private_ip = "10.0.4.250"
```

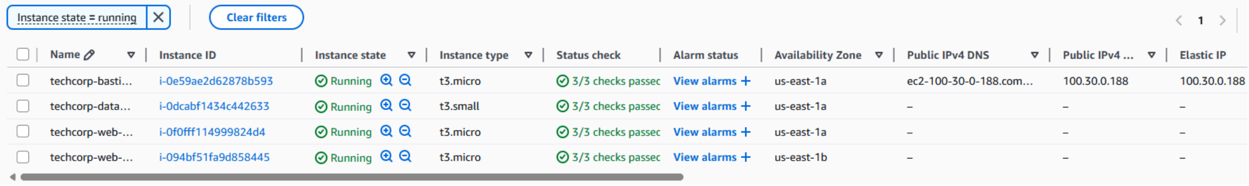
3. AWS Console showing created resources



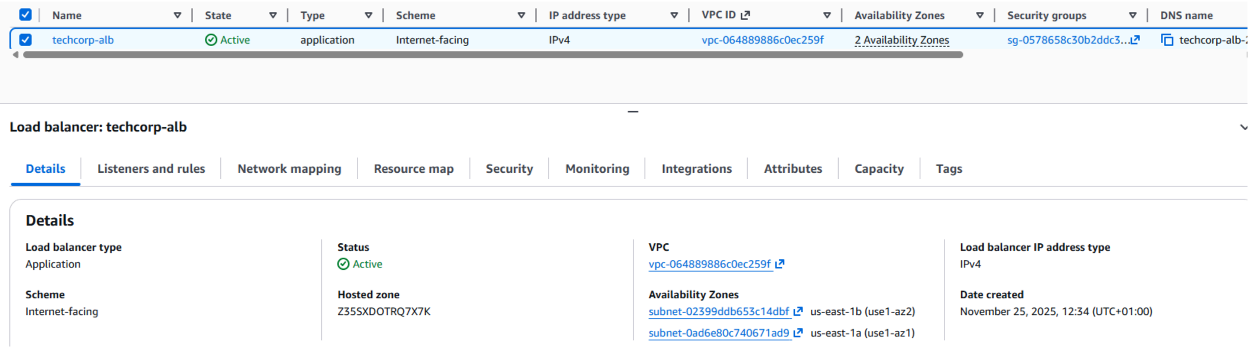
Resources

You are using the following Amazon EC2 resources in the United States (N. Virginia) Region:

Instances (running)	4	Auto Scaling Groups	0	Capacity Reservations	0
Dedicated Hosts	0	Elastic IPs	3	Instances	4
Key pairs	1	Load balancers	1	Placement groups	0
Security groups	5	Snapshots	0	Volumes	4



Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
techcorp-basti...	i-0e59ae2d62878b593	Running	t3.micro	3/3 checks passec	View alarms +	us-east-1a	ec2-100-30-0-188.com...	100.30.0.188	100.30.0.188
techcorp-data...	i-0dcabf1434c442633	Running	t3.small	3/3 checks passec	View alarms +	us-east-1a	–	–	–
techcorp-web-...	i-0f0fff114999824d4	Running	t3.micro	3/3 checks passec	View alarms +	us-east-1a	–	–	–
techcorp-web-...	i-094bf51fa9d858445	Running	t3.micro	3/3 checks passec	View alarms +	us-east-1b	–	–	–



☑	Name	State	Type	Scheme	IP address type	VPC ID	Availability Zones	Security groups	DNS name
☑	techcorp-alb	Active	application	Internet-facing	IPv4	vpc-064889886c0ec259f	2 Availability Zones	sg-0578658c30b2ddc3...	techcorp-alb-

Load balancer: techcorp-alb

Details

Load balancer type
Application

Scheme
Internet-facing

Status
Active

Hosted zone
Z35SXDOTRQ7X7K

VPC
vpc-064889886c0ec259f

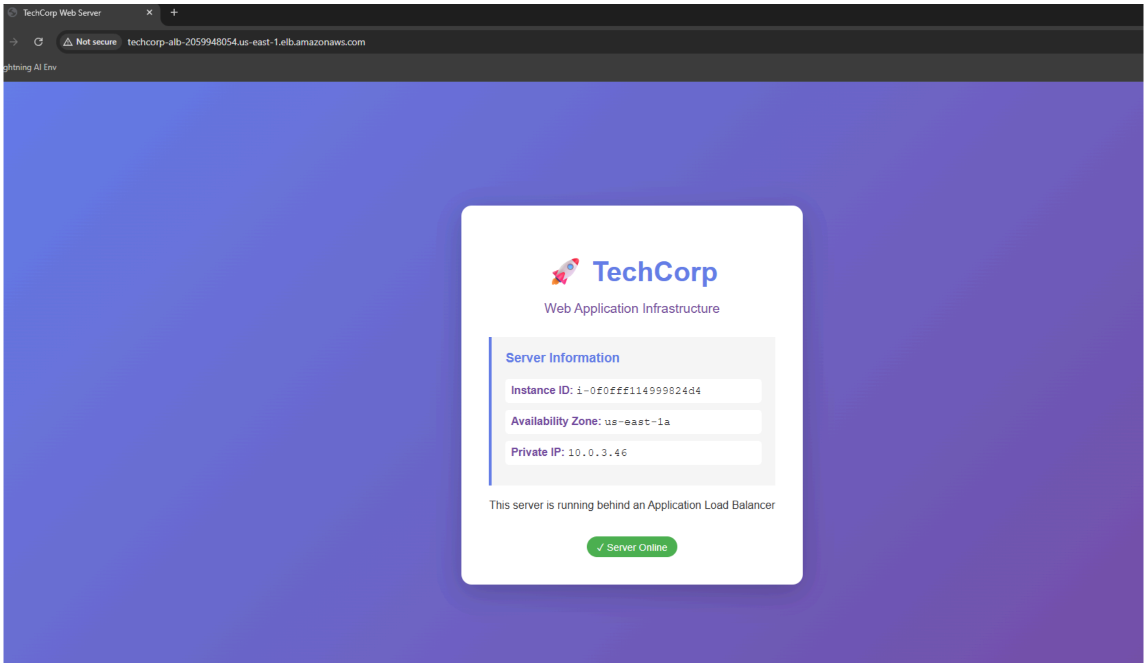
Availability Zones
subnet-02399ddb653c14dbf us-east-1b (use1-az2)
subnet-0ad6e80c740671ad9 us-east-1a (use1-az1)

Load balancer IP address type
IPv4

Date created
November 25, 2025, 12:34 (UTC+01:00)

4. Load balancer serving web pages from both instances

Web page 1:



🚀 TechCorp

Web Application Infrastructure

Server Information

Instance ID: i-0f0fff114999824d4

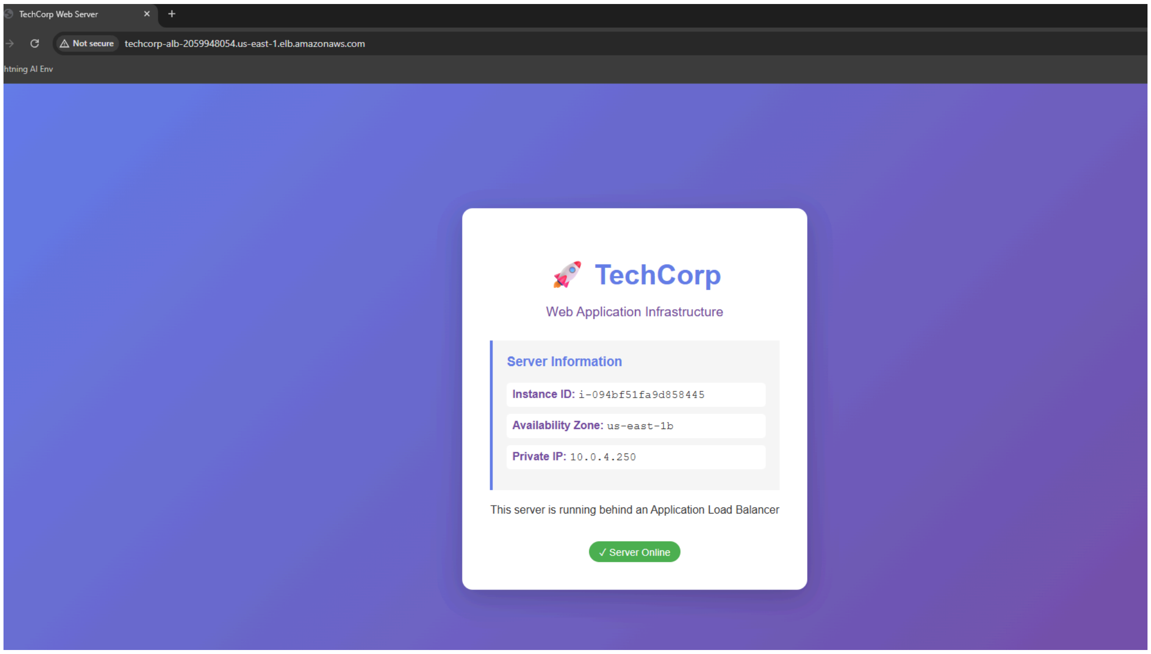
Availability Zone: us-east-1a

Private IP: 10.0.3.46

This server is running behind an Application Load Balancer

✓ Server Online

Web page 2:



🚀 TechCorp

Web Application Infrastructure

Server Information

Instance ID: i-094bf51fa9d858445

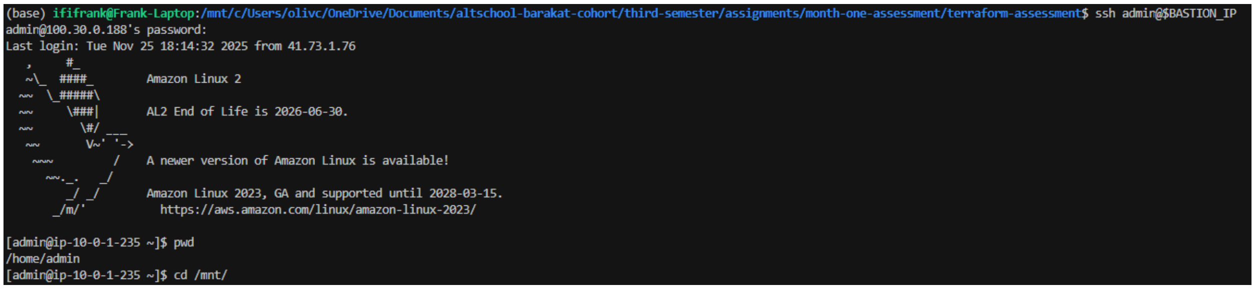
Availability Zone: us-east-1b

Private IP: 10.0.4.250

This server is running behind an Application Load Balancer

✓ Server Online

5. SSH access through bastion host

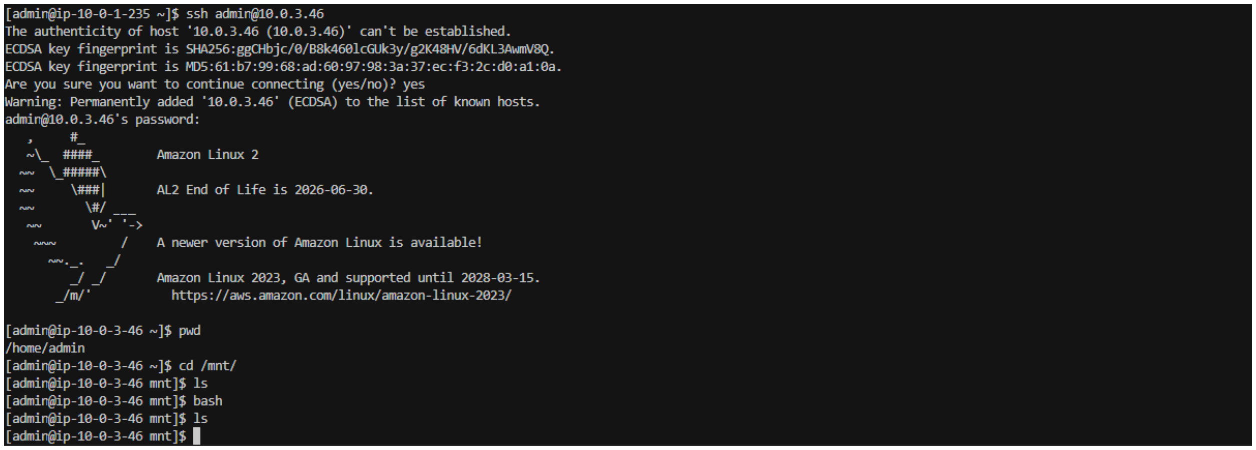


```
(base) ififrank@Frank-Laptop:/mnt/c/Users/olivc/OneDrive/Documents/altschool-barakat-cohort/third-semester/assignments/month-one-assessment/terraform-assessment$ ssh admin@$BASTION_IP
admin@100.30.0.188's password:
Last login: Tue Nov 25 18:14:32 2025 from 41.73.1.76
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[admin@ip-10-0-1-235 ~]$ pwd
/home/admin
[admin@ip-10-0-1-235 ~]$ cd /mnt/
```

6. SSH access to the Web and DB servers

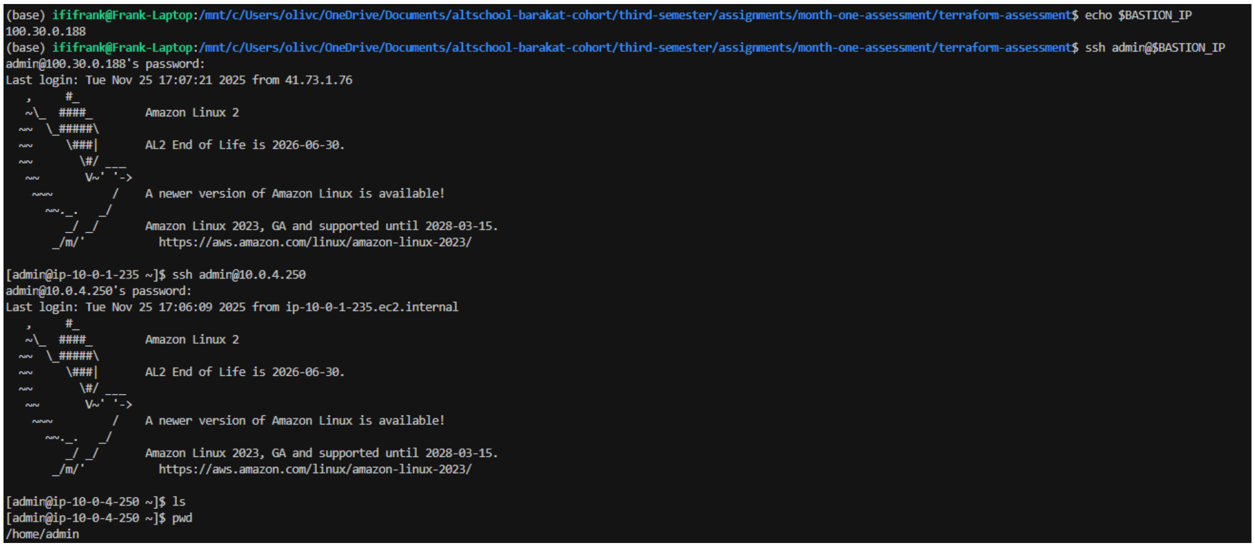
<u>6.a SSH to web server:</u>

Ssh into private IP 1a from Bastion:



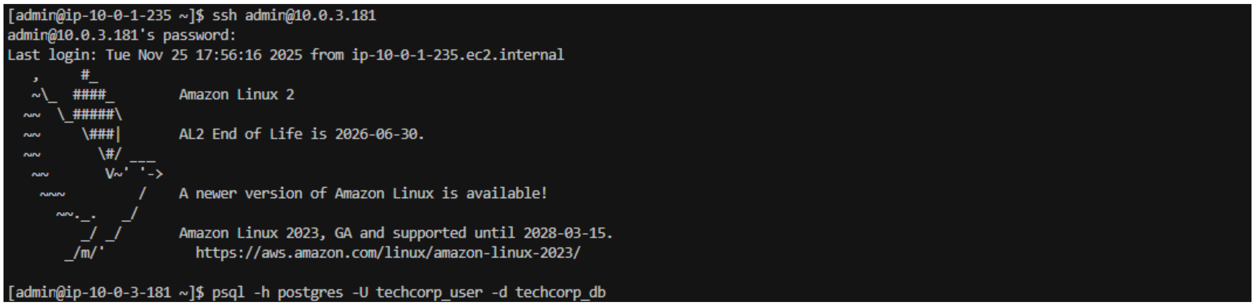
```
[admin@ip-10-0-1-235 ~]$ ssh admin@10.0.3.46
The authenticity of host '10.0.3.46 (10.0.3.46)' can't be established.
ECDSA key fingerprint is SHA256:ggCHbjc/0/B8k460lcGUk3y/g2K48HV/6dKL3AwmV8Q.
ECDSA key fingerprint is MD5:61:b7:99:68:ad:60:97:98:3a:37:ec:f3:2c:d0:a1:0a.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.3.46' (ECDSA) to the list of known hosts.
admin@10.0.3.46's password:
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[admin@ip-10-0-3-46 ~]$ pwd
/home/admin
[admin@ip-10-0-3-46 ~]$ cd /mnt/
[admin@ip-10-0-3-46 mnt]$ ls
[admin@ip-10-0-3-46 mnt]$ bash
[admin@ip-10-0-3-46 mnt]$ ls
[admin@ip-10-0-3-46 mnt]$ 
```

Ssh into private IP 1b from Bastion:



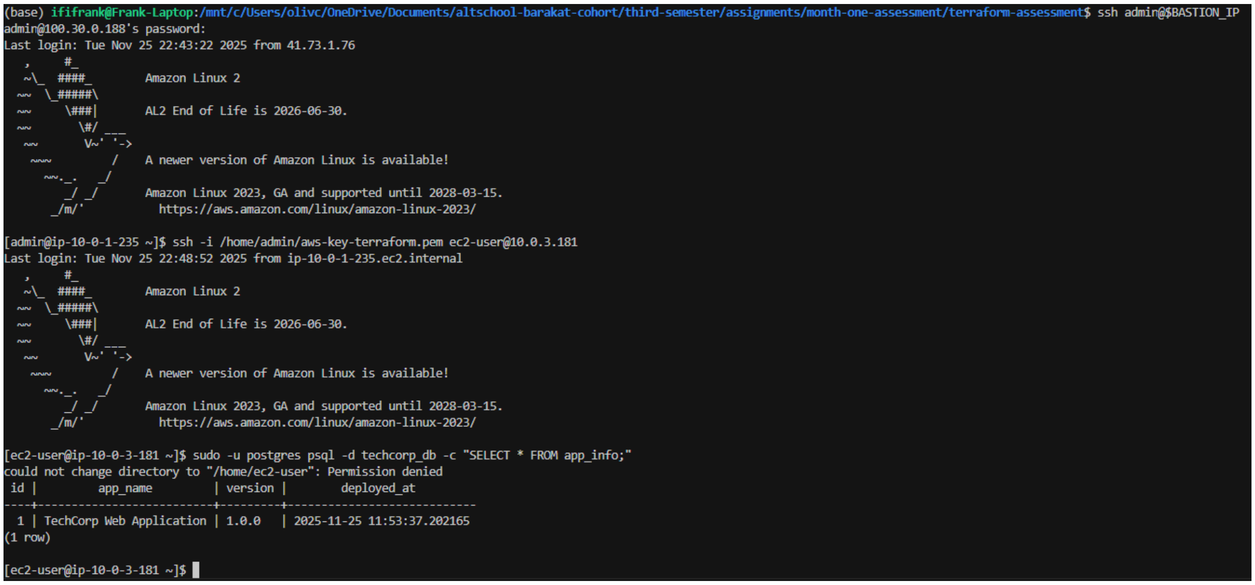
```
(base) ififrank@Frank-Laptop:/mnt/c/Users/olivc/OneDrive/Documents/altschool-barakat-cohort/third-semester/assignments/month-one-assessment/terraform-assessment$ echo $BASTION_IP
100.30.0.188
(base) ififrank@Frank-Laptop:/mnt/c/Users/olivc/OneDrive/Documents/altschool-barakat-cohort/third-semester/assignments/month-one-assessment/terraform-assessment$ ssh admin@$BASTION_IP
admin@100.30.0.188's password:
Last login: Tue Nov 25 17:07:21 2025 from 41.73.1.76
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[admin@ip-10-0-1-235 ~]$ ssh admin@10.0.4.250
admin@10.0.4.250's password:
Last login: Tue Nov 25 17:06:09 2025 from ip-10-0-1-235.ec2.internal
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[admin@ip-10-0-4-250 ~]$ ls
[admin@ip-10-0-4-250 ~]$ pwd
/home/admin
```

6.b SSH access to the Web and DB servers



```
[admin@ip-10-0-1-235 ~]$ ssh admin@10.0.3.181
admin@10.0.3.181's password:
Last login: Tue Nov 25 17:56:16 2025 from ip-10-0-1-235.ec2.internal
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[admin@ip-10-0-3-181 ~]$ psql -h postgres -U techcorp_user -d techcorp_db
```

7. Connect to the postgres instance on the DB server



```
(base) ififrank@Frank-Laptop:/mnt/c/Users/olivc/OneDrive/Documents/altschool-barakat-cohort/third-semester/assignments/month-one-assessment/terraform-assessment$ ssh admin@$BASTION_IP
admin@100.30.0.188's password:
Last login: Tue Nov 25 22:43:22 2025 from 41.73.1.76
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[admin@ip-10-0-1-235 ~]$ ssh -i /home/admin/aws-key-terraform.pem ec2-user@10.0.3.181
Last login: Tue Nov 25 22:48:52 2025 from ip-10-0-1-235.ec2.internal
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2026-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-10-0-3-181 ~]$ sudo -u postgres psql -d techcorp_db -c "SELECT * FROM app_info;"
could not change directory to "/home/ec2-user": Permission denied
 id |         app_name         | version |        deployed_at
----+--------------------------+---------+----------------------------
  1 | TechCorp Web Application | 1.0.0   | 2025-11-25 11:53:37.202165
(1 row)

[ec2-user@ip-10-0-3-181 ~]$
```