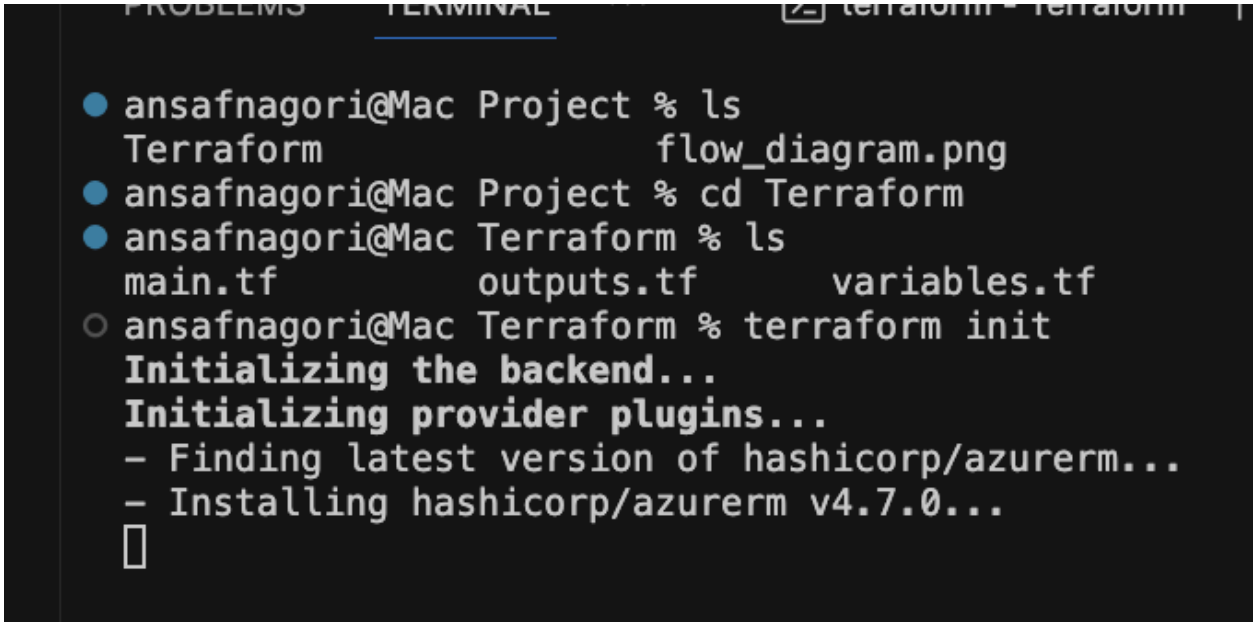


I installed az CLI: brew update && brew install azure-cli

I logged in from shell to az: az login

resource_group_name = "vpn_vm_rg"
location = "Central India"
vnet_cidr = "10.0.0.0/16"
subnet_cidr = "10.0.1.0/24"
admin_username = "vpn-vm-admin"
allowed_ip = "96.78.12.98" # Replace with your IP
ssh_public_key = "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDH0OsrR+TBUKORKgn5vopfzYEcQpsDK/JxIHh4e
T1kysdf6Qxx7rvNInS6BMBHDNlPGjRMiCM/4fjFaYTUI3//ZqMW2wcWuJhhv4tKM1INAl8XlXjzlN
HmzVZqnWals9N8Gu5goMimxvbAn/D6FL3qH9uaUVtcrY6RF7zKUkMgol2rrZWNj/L2zcrxDOnI+
MxNBTk+8Wax7kU5mjN+BMJx3X5wCYIeKHhvAeHJH68L7yJIGDbY/2GEEovqtPs09KX7TVOS
Au9ZFG9zpOOzbJylfAeHkCTzd68eSftrHiVcoe3+5NataWeEqeby+GkskIOra/bnxOCy3xjJAu9SV
kgZ ansafnagori@Mac.hsd1.co.comcast.net"

admin_username = "vpn-vm-admin"
location = "Central India"
vpn_public_ip = "13.71.47.182"
ansafnagori@Mac Terraform %

```
Generating public/private rsa key pair.
```

```
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in
/Users/ansafnagori/.ssh/vpn_ssh_key
Your public key has been saved in
/Users/ansafnagori/.ssh/vpn_ssh_key.pub
The key fingerprint is:
SHA256:XB5z8h+d6ABBv54iiBn9753HqYJCcQXgW358+LxzYjw
ansafnagori@Mac.hsd1.co.comcast.net
The key's randomart image is:
+---[RSA 2048]----+
|     ...oo       |
|    .    .o      |
|     . ...=..    |
|    ..+o.oo*. ...|
|   . oo.S+.+.....|
|    +.o . = +. . |
|   o.. o...=..o  |
|     . .o.oEo=   |
|      . .oo=B    |
+----[SHA256]-----+
ansafnagori@Mac Project %
```

ansafnagori@Mac Terraform % terraform plan 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: # azurerm_network_interface.vpn_nic will be created + resource "azurerm_network_interface" "vpn_nic" { + applied_dns_servers = (known after apply) + id = (known after apply) + internal_domain_name_suffix = (known after apply) + location = "centralindia" + mac_address = (known after apply) + name = "vpn-nic" + private_ip_address = (known after apply) + private_ip_addresses = (known after apply) + resource_group_name = "vpn_vm_rg" + virtual_machine_id = (known after apply) + ip_configuration { + gateway_load_balancer_frontend_ip_configuration_id = (known after apply) + name = "internal" + primary = (known after apply) + private_ip_address = (known after apply) + private_ip_address_allocation = "Dynamic" + private_ip_address_version = "IPv4" + public_ip_address_id = (known after apply) + subnet_id = (known after apply) } } # azurerm_network_security_group.subnet_nsg will be created + resource "azurerm_network_security_group" "subnet_nsg" { + id = (known after apply) + location = "centralindia" + name = "subnet-nsg" + resource_group_name = "vpn_vm_rg" + security_rule = [+ { + access = "Allow" + destination_address_prefix = "*" + destination_address_prefixes = [] +

destination_application_security_group_ids = [] + destination_port_range = "1194" + destination_port_ranges = [] + direction = "Inbound" + name = "allow-vpn-traffic" + priority = 100 + protocol = "Udp" + source_address_prefix = "96.78.12.98" + source_address_prefixes = [] + source_application_security_group_ids = [] + source_port_range = "*" + source_port_ranges = [] # (1 unchanged attribute hidden) }, + { + access = "Allow" + destination_address_prefix = "*" + destination_address_prefixes = [] + destination_application_security_group_ids = [] + destination_port_range = "22" + destination_port_ranges = [] + direction = "Inbound" + name = "allow-ssh" + priority = 200 + protocol = "Tcp" + source_address_prefix = "96.78.12.98" + source_address_prefixes = [] + source_application_security_group_ids = [] + source_port_range = "*" + source_port_ranges = [] # (1 unchanged attribute hidden) },] } # azurerm_public_ip.vpn_public_ip will be created + resource "azurerm_public_ip" "vpn_public_ip" { + allocation_method = "Static" + ddos_protection_mode = "VirtualNetworkInherited" + fqdn = (known after apply) + id = (known after apply) + idle_timeout_in_minutes = 4 + ip_address = (known after apply) + ip_version = "IPv4" + location = "centralindia" + name = "vpn-public-ip" + resource_group_name = "vpn_vm_rg" + sku = "Standard" + sku_tier = "Regional" } # azurerm_resource_group.vpn_rg will be created + resource "azurerm_resource_group" "vpn_rg" { + id = (known after apply) + location = "centralindia" + name = "vpn_vm_rg" } # azurerm_subnet.vpn_subnet will be created + resource "azurerm_subnet" "vpn_subnet" { + address_prefixes = [+ "10.0.1.0/24",] + default_outbound_access_enabled = true + id = (known after apply) + name = "vpn-subnet" + private_endpoint_network_policies = "Disabled" + private_link_service_network_policies_enabled = true + resource_group_name = "vpn_vm_rg" + virtual_network_name = "vpn-vnet" } # azurerm_subnet_network_security_group_association.subnet_nsg_assoc will be created + resource "azurerm_subnet_network_security_group_association" "subnet_nsg_assoc" { + id = (known after apply) + network_security_group_id = (known after apply) + subnet_id = (known after apply) } # azurerm_virtual_machine.vpn_vm will be created + resource "azurerm_virtual_machine" "vpn_vm" { + availability_set_id = (known after apply) + delete_data_disks_on_termination = false + delete_os_disk_on_termination = false + id = (known after apply) + license_type = (known after apply) + location = "centralindia" + name = "vpn-server" + network_interface_ids = (known after apply) + resource_group_name = "vpn_vm_rg" + vm_size = "Standard_B1ms" + os_profile { # At least one attribute in this block is (or was) sensitive, # so its contents will not be displayed. } + os_profile_linux_config { + disable_password_authentication = true + ssh_keys { + key_data = "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDH0OsrR+TBUKORKgn5vopfzYEcQpsDK/JxIHh4eT1kysdf6Qxx7rvNInS6BMBHDNIPGjRMiCM/4fjFaYTUI3//ZqMW2wcWuJhhv4tKM1INAI8XIXjzlNHmzVZqnWaIs9N8Gu5goMimxvbAn/D6FL3qH9uaUVtcrY6RF7zKUkMgol2rrZWNj/L2zcrxDOnI+MxNBTk+8Wax7kU5mjN+BMJx3X5wCYIeKHhvAeHJH68L7yJIGDbY/2GEEovqtPs09KX7TVOSAu9ZFG9zpOOzbJylfAeHkCTzd68eSftrHiVcoe3+5NataWeEqeby+GkskIOra/bnxOCy3xjJAu9SVkgZ ansafnagori@Mac.hsd1.co.comcast.net" + path = "/home/vpn-vm-admin/.ssh/authorized_keys" } } + storage_data_disk (known after apply) + storage_image_reference { id = null + offer = "UbuntuServer" + publisher = "Canonical" + sku = "18.04-LTS" + version = "latest" } + storage_os_disk { + caching = "ReadWrite" + create_option = "FromImage" + disk_size_gb = (known after apply) + managed_disk_id = (known after apply) + managed_disk_type = "Standard_LRS" + name = "vpn-os-disk" + os_type = (known after

apply) + write_accelerator_enabled = false } } # azurerm_virtual_machine_extension.vpn_install will be created + resource "azurerm_virtual_machine_extension" "vpn_install" { + failure_suppression_enabled = false + id = (known after apply) + name = "vpn-install" + publisher = "Microsoft.Azure.Extensions" + settings = <<-EOT { "commandToExecute": "sudo apt-get update && sudo apt-get install -y openvpn && wget https://git.io/vpn -O openvpn-install.sh && sudo bash openvpn-install.sh" } EOT + type = "CustomScript" + type_handler_version = "2.0" + virtual_machine_id = (known after apply) } # azurerm_virtual_network.vpn_vnet will be created + resource "azurerm_virtual_network" "vpn_vnet" { + address_space = [+ "10.0.0.0/16",] + dns_servers = (known after apply) + guid = (known after apply) + id = (known after apply) + location = "centralindia" + name = "vpn-vnet" + resource_group_name = "vpn_vm_rg" + subnet = (known after apply) } Plan: 9 to add, 0 to change, 0 to destroy. Changes to Outputs: + admin_username = "vpn-vm-admin" + location = "Central India" + vpn_public_ip = (known after apply)

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

ansafnagori@Mac Terraform % terraform apply

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:

```
  # azurerm_network_interface.vpn_nic will be created
  + resource "azurerm_network_interface" "vpn_nic" {
      + applied_dns_servers         = (known after apply)
      + id                          = (known after apply)
      + internal_domain_name_suffix = (known after apply)
      + location                    = "centralindia"
      + mac_address                 = (known after apply)
      + name                        = "vpn-nic"
      + private_ip_address          = (known after apply)
      + private_ip_addresses        = (known after apply)
      + resource_group_name         = "vpn_vm_rg"
      + virtual_machine_id          = (known after apply)

      + ip_configuration {
          + gateway_load_balancer_frontend_ip_configuration_id = (known after apply)
```

```
      + name                                          = "internal"
      + primary                                       = (known after apply)
      + private_ip_address                            = (known after apply)
      + private_ip_address_allocation                 = "Dynamic"
      + private_ip_address_version                    = "IPv4"
      + public_ip_address_id                          = (known after apply)
      + subnet_id                                     = (known after apply)
    }
  }

# azurerm_network_security_group.subnet_nsg will be created
+ resource "azurerm_network_security_group" "subnet_nsg" {
    + id                = (known after apply)
    + location          = "centralindia"
    + name              = "subnet-nsg"
    + resource_group_name = "vpn_vm_rg"
    + security_rule     = [
        + {
            + access                                   = "Allow"
            + destination_address_prefix               = "*"
            + destination_address_prefixes             = []
            + destination_application_security_group_ids = []
            + destination_port_range                   = "1194"
            + destination_port_ranges                  = []
            + direction                                = "Inbound"
            + name                                     = "allow-vpn-traffic"
            + priority                                 = 100
            + protocol                                 = "Udp"
            + source_address_prefix                    = "96.78.12.98"
            + source_address_prefixes                  = []
            + source_application_security_group_ids    = []
            + source_port_range                        = "*"
            + source_port_ranges                       = []
              # (1 unchanged attribute hidden)
          },
        + {
            + access                                   = "Allow"
            + destination_address_prefix               = "*"
            + destination_address_prefixes             = []
            + destination_application_security_group_ids = []
            + destination_port_range                   = "22"
            + destination_port_ranges                  = []
            + direction                                = "Inbound"
            + name                                     = "allow-ssh"
```

```
          + priority                                 = 200
          + protocol                                 = "Tcp"
          + source_address_prefix                    = "96.78.12.98"
          + source_address_prefixes                  = []
          + source_application_security_group_ids    = []
          + source_port_range                        = "*"
          + source_port_ranges                       = []
            # (1 unchanged attribute hidden)
        },
    ]
  }

# azurerm_public_ip.vpn_public_ip will be created
+ resource "azurerm_public_ip" "vpn_public_ip" {
    + allocation_method       = "Static"
    + ddos_protection_mode    = "VirtualNetworkInherited"
    + fqdn                    = (known after apply)
    + id                      = (known after apply)
    + idle_timeout_in_minutes = 4
    + ip_address              = (known after apply)
    + ip_version              = "IPv4"
    + location                = "centralindia"
    + name                    = "vpn-public-ip"
    + resource_group_name     = "vpn_vm_rg"
    + sku                     = "Standard"
    + sku_tier                = "Regional"
  }

# azurerm_resource_group.vpn_rg will be created
+ resource "azurerm_resource_group" "vpn_rg" {
    + id       = (known after apply)
    + location = "centralindia"
    + name     = "vpn_vm_rg"
  }

# azurerm_subnet.vpn_subnet will be created
+ resource "azurerm_subnet" "vpn_subnet" {
    + address_prefixes                          = [
        + "10.0.1.0/24",
      ]
    + default_outbound_access_enabled           = true
    + id                                        = (known after apply)
    + name                                      = "vpn-subnet"
    + private_endpoint_network_policies         = "Disabled"
```

```
    + private_link_service_network_policies_enabled = true
    + resource_group_name                          = "vpn_vm_rg"
    + virtual_network_name                         = "vpn-vnet"
  }

 # azurerm_subnet_network_security_group_association.subnet_nsg_assoc will be created
 + resource "azurerm_subnet_network_security_group_association" "subnet_nsg_assoc" {
    + id                        = (known after apply)
    + network_security_group_id = (known after apply)
    + subnet_id                 = (known after apply)
  }

 # azurerm_virtual_machine.vpn_vm will be created
 + resource "azurerm_virtual_machine" "vpn_vm" {
    + availability_set_id              = (known after apply)
    + delete_data_disks_on_termination = false
    + delete_os_disk_on_termination    = false
    + id                               = (known after apply)
    + license_type                     = (known after apply)
    + location                         = "centralindia"
    + name                             = "vpn-server"
    + network_interface_ids            = (known after apply)
    + resource_group_name              = "vpn_vm_rg"
    + vm_size                          = "Standard_B1ms"

    + os_profile {
        # At least one attribute in this block is (or was) sensitive,
        # so its contents will not be displayed.
      }

    + os_profile_linux_config {
        + disable_password_authentication = true

        + ssh_keys {
            + key_data = "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDH0OsrR+TBUKORKgn5vopfzYEcQpsDK/JxIHh4e
T1kysdf6Qxx7rvNInS6BMBHDNIPGjRMiCM/4fjFaYTUI3//ZqMW2wcWuJhhv4tKM1INAI8XIXjzlN
HmzVZqnWals9N8Gu5goMimxvbAn/D6FL3qH9uaUVtcrY6RF7zKUkMgol2rrZWNj/L2zcrxDOnI+
MxNBTk+8Wax7kU5mjN+BMJx3X5wCYIeKHhvAeHJH68L7yJIGDbY/2GEEovqtPs09KX7TVOS
Au9ZFG9zpOOzbJyIfAeHkCTzd68eSftrHiVcoe3+5NataWeEqeby+GkskIOra/bnxOCy3xjJAu9SV
kgZ ansafnagori@Mac.hsd1.co.comcast.net"
            + path     = "/home/vpn-vm-admin/.ssh/authorized_keys"
          }
      }
```

```
    + storage_data_disk (known after apply)

    + storage_image_reference {
        id        = null
      + offer     = "UbuntuServer"
      + publisher = "Canonical"
      + sku       = "18.04-LTS"
      + version   = "latest"
    }

    + storage_os_disk {
      + caching                 = "ReadWrite"
      + create_option           = "FromImage"
      + disk_size_gb            = (known after apply)
      + managed_disk_id         = (known after apply)
      + managed_disk_type       = "Standard_LRS"
      + name                    = "vpn-os-disk"
      + os_type                 = (known after apply)
      + write_accelerator_enabled = false
    }
  }

  # azurerm_virtual_machine_extension.vpn_install will be created
  + resource "azurerm_virtual_machine_extension" "vpn_install" {
    + failure_suppression_enabled = false
    + id                       = (known after apply)
    + name                     = "vpn-install"
    + publisher                = "Microsoft.Azure.Extensions"
    + settings                 = <<-EOT
        {
            "commandToExecute": "sudo apt-get update && sudo apt-get install -y openvpn &&
wget https://git.io/vpn -O openvpn-install.sh && sudo bash openvpn-install.sh"
        }
      EOT
    + type                     = "CustomScript"
    + type_handler_version     = "2.0"
    + virtual_machine_id       = (known after apply)
  }

  # azurerm_virtual_network.vpn_vnet will be created
  + resource "azurerm_virtual_network" "vpn_vnet" {
    + address_space       = [
      + "10.0.0.0/16",
```

```
      ]
    + dns_servers         = (known after apply)
    + guid                = (known after apply)
    + id                  = (known after apply)
    + location            = "centralindia"
    + name                = "vpn-vnet"
    + resource_group_name = "vpn_vm_rg"
    + subnet              = (known after apply)
  }

Plan: 9 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + admin_username = "vpn-vm-admin"
  + location       = "Central India"
  + vpn_public_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

azurerm_resource_group.vpn_rg: Creating...
azurerm_resource_group.vpn_rg: Still creating... [10s elapsed]
azurerm_resource_group.vpn_rg: Creation complete after 15s
[id=/subscriptions/bc86d3cb-dbde-4f24-be16-a3593c25ba9a/resourceGroups/vpn_vm_rg]
azurerm_virtual_network.vpn_vnet: Creating...
azurerm_public_ip.vpn_public_ip: Creating...
azurerm_network_security_group.subnet_nsg: Creating...
azurerm_network_security_group.subnet_nsg: Creation complete after 6s
[id=/subscriptions/bc86d3cb-dbde-4f24-be16-a3593c25ba9a/resourceGroups/vpn_vm_rg/provid
ers/Microsoft.Network/networkSecurityGroups/subnet-nsg]
azurerm_public_ip.vpn_public_ip: Creation complete after 7s
[id=/subscriptions/bc86d3cb-dbde-4f24-be16-a3593c25ba9a/resourceGroups/vpn_vm_rg/provid
ers/Microsoft.Network/publicIPAddresses/vpn-public-ip]
azurerm_virtual_network.vpn_vnet: Creation complete after 9s
[id=/subscriptions/bc86d3cb-dbde-4f24-be16-a3593c25ba9a/resourceGroups/vpn_vm_rg/provid
ers/Microsoft.Network/virtualNetworks/vpn-vnet]
azurerm_subnet.vpn_subnet: Creating...
azurerm_subnet.vpn_subnet: Creation complete after 9s
[id=/subscriptions/bc86d3cb-dbde-4f24-be16-a3593c25ba9a/resourceGroups/vpn_vm_rg/provid
ers/Microsoft.Network/virtualNetworks/vpn-vnet/subnets/vpn-subnet]
azurerm_subnet_network_security_group_association.subnet_nsg_assoc: Creating...
```

```
azurerm_network_interface.vpn_nic: Creating...
azurerm_subnet_network_security_group_association.subnet_nsg_assoc: Creation complete
after 9s
[id=/subscriptions/bc86d3cb-dbde-4f24-be16-a3593c25ba9a/resourceGroups/vpn_vm_rg/provid
ers/Microsoft.Network/virtualNetworks/vpn-vnet/subnets/vpn-subnet]
azurerm_network_interface.vpn_nic: Still creating... [10s elapsed]
azurerm_network_interface.vpn_nic: Still creating... [20s elapsed]
azurerm_network_interface.vpn_nic: Creation complete after 25s
[id=/subscriptions/bc86d3cb-dbde-4f24-be16-a3593c25ba9a/resourceGroups/vpn_vm_rg/provid
ers/Microsoft.Network/networkInterfaces/vpn-nic]
azurerm_virtual_machine.vpn_vm: Creating...
azurerm_virtual_machine.vpn_vm: Still creating... [10s elapsed]
azurerm_virtual_machine.vpn_vm: Still creating... [20s elapsed]
azurerm_virtual_machine.vpn_vm: Still creating... [30s elapsed]
azurerm_virtual_machine.vpn_vm: Still creating... [40s elapsed]
azurerm_virtual_machine.vpn_vm: Still creating... [50s elapsed]
azurerm_virtual_machine.vpn_vm: Creation complete after 58s
[id=/subscriptions/bc86d3cb-dbde-4f24-be16-a3593c25ba9a/resourceGroups/vpn_vm_rg/provid
ers/Microsoft.Compute/virtualMachines/vpn-server]
azurerm_virtual_machine_extension.vpn_install: Creating...
azurerm_virtual_machine_extension.vpn_install: Still creating... [10s elapsed]
azurerm_virtual_machine_extension.vpn_install: Still creating... [30s elapsed]
azurerm_virtual_machine_extension.vpn_install: Still creating... [40s elapsed]
azurerm_virtual_machine_extension.vpn_install: Creation complete after 48s
[id=/subscriptions/bc86d3cb-dbde-4f24-be16-a3593c25ba9a/resourceGroups/vpn_vm_rg/provid
ers/Microsoft.Compute/virtualMachines/vpn-server/extensions/vpn-install]

Apply complete! Resources: 9 added, 0 changed, 0 destroyed.

Outputs:

admin_username = "vpn-vm-admin"
location = "Central India"
vpn_public_ip = "13.71.47.182"
ansafnagori@Mac Terraform %


ansafnagori@Mac Terraform % ssh vpn-vm-admin@13.71.47.182
The authenticity of host '13.71.47.182 (13.71.47.182)' can't be established.
ED25519 key fingerprint is SHA256:HgRpKmWalt7rLoB2Jv4DXgnrxlCni5nOfW8MgwRqVnQ.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '13.71.47.182' (ED25519) to the list of known hosts.
vpn-vm-admin@13.71.47.182: Permission denied (publickey).
```

ansafnagori@Mac Terraform % cat ~/.ssh/vpn_ssh_key.pub

ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDH0OsrR+TBUKORKgn5vopfzYEcQpsDK/JxIHh4e
T1kysdf6Qxx7rvNInS6BMBHDNlPGjRMiCM/4fjFaYTUI3//ZqMW2wcWuJhhv4tKM1lNAl8XlXjzlN
HmzVZqnWals9N8Gu5goMimxvbAn/D6FL3qH9uaUVtcrY6RF7zKUkMgol2rrZWNj/L2zcrxDOnI+
MxNBTk+8Wax7kU5mjN+BMJx3X5wCYIeKHhvAeHJH68L7yJIGDbY/2GEEovqtPs09KX7TVOS
Au9ZFG9zpOOzbJylfAeHkCTzd68eSftrHiVcoe3+5NataWeEqeby+GkskIOra/bnxOCy3xjJAu9SV
kgZ ansafnagori@Mac.hsd1.co.comcast.net
ansafnagori@Mac Terraform % ssh -i ~/.ssh/vpn_ssh_key vpn-vm-admin@13.71.47.182

Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-1109-azure x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

  System information as of Fri Nov 1 06:08:26 UTC 2024

  System load: 0.08   Processes: 105
  Usage of /:   5.3% of 28.89GB   Users logged in: 0
  Memory usage: 10%   IP address for eth0: 10.0.1.4
  Swap usage: 0%

Expanded Security Maintenance for Infrastructure is not enabled.

4 updates can be applied immediately.
1 of these updates is a standard security update.
To see these additional updates run: apt list --upgradable

131 additional security updates can be applied with ESM Infra.
Learn more about enabling ESM Infra service for Ubuntu 18.04 at
https://ubuntu.com/18-04

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

/usr/bin/xauth: file /home/vpn-vm-admin/.Xauthority does not exist

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

vpn-vm-admin@vpn-server:~$

vpn-vm-admin@vpn-server:~$ cat /etc/openvpn/server/server.conf cat:
/etc/openvpn/server/server.conf: No such file or directory vpn-vm-admin@vpn-server:~$

vpn-vm-admin@vpn-server:~$ sudo nano /etc/openvpn/server/server.conf

port 1194
proto udp
dev tun
ca /etc/openvpn/server/ca.crt
cert /etc/openvpn/server/server.crt
key /etc/openvpn/server/server.key
dh /etc/openvpn/server/dh.pem
server 10.8.0.0 255.255.255.0
ifconfig-pool-persist ipp.txt
keepalive 10 120
cipher AES-256-CBC
auth SHA256
compress lz4
persist-key
persist-tun
status openvpn-status.log
verb 3

vpn-vm-admin@vpn-server:~$ sudo nano /etc/openvpn/server/server.conf
vpn-vm-admin@vpn-server:~$ cat /etc/openvpn/server/server.conf
cat: /etc/openvpn/server/server.conf: No such file or directory
vpn-vm-admin@vpn-server:~$ sudo nano /etc/openvpn/server/server.conf
vpn-vm-admin@vpn-server:~$ ls /etc/openvpn/server/
server.conf
vpn-vm-admin@vpn-server:~$ sudo apt update
Hit:1 http://azure.archive.ubuntu.com/ubuntu bionic InRelease

```
Hit:2 http://azure.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://azure.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://azure.archive.ubuntu.com/ubuntu bionic-security InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
8 packages can be upgraded. Run 'apt list --upgradable' to see them.
vpn-vm-admin@vpn-server:~$ sudo apt install -y easy-rsa
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libccid libpcsclite1 opensc opensc-pkcs11 pcscd
Suggested packages:
  pcmciautils
The following NEW packages will be installed:
  easy-rsa libccid libpcsclite1 opensc opensc-pkcs11 pcscd
0 upgraded, 6 newly installed, 0 to remove and 8 not upgraded.
Need to get 1213 kB of archives.
After this operation, 4189 kB of additional disk space will be used.
Get:1 http://azure.archive.ubuntu.com/ubuntu bionic/main amd64 libpcsclite1 amd64 1.8.23-1
[21.3 kB]
Get:2 http://azure.archive.ubuntu.com/ubuntu bionic/universe amd64 opensc-pkcs11 amd64
0.17.0-3 [791 kB]
Get:3 http://azure.archive.ubuntu.com/ubuntu bionic/universe amd64 opensc amd64 0.17.0-3
[237 kB]
Get:4 http://azure.archive.ubuntu.com/ubuntu bionic/universe amd64 easy-rsa all 2.2.2-2 [17.4
kB]
Get:5 http://azure.archive.ubuntu.com/ubuntu bionic/universe amd64 libccid amd64 1.4.29-1
[88.4 kB]
Get:6 http://azure.archive.ubuntu.com/ubuntu bionic/universe amd64 pcscd amd64 1.8.23-1
[57.9 kB]
Fetched 1213 kB in 3s (406 kB/s)
Selecting previously unselected package libpcsclite1:amd64.
(Reading database ... 59286 files and directories currently installed.)
Preparing to unpack .../0-libpcsclite1_1.8.23-1_amd64.deb ...
Unpacking libpcsclite1:amd64 (1.8.23-1) ...
Selecting previously unselected package opensc-pkcs11:amd64.
Preparing to unpack .../1-opensc-pkcs11_0.17.0-3_amd64.deb ...
Unpacking opensc-pkcs11:amd64 (0.17.0-3) ...
Selecting previously unselected package opensc.
Preparing to unpack .../2-opensc_0.17.0-3_amd64.deb ...
Unpacking opensc (0.17.0-3) ...
Selecting previously unselected package easy-rsa.
```

```
Preparing to unpack .../3-easy-rsa_2.2.2-2_all.deb ...
Unpacking easy-rsa (2.2.2-2) ...
Selecting previously unselected package libccid.
Preparing to unpack .../4-libccid_1.4.29-1_amd64.deb ...
Unpacking libccid (1.4.29-1) ...
Selecting previously unselected package pcscd.
Preparing to unpack .../5-pcscd_1.8.23-1_amd64.deb ...
Unpacking pcscd (1.8.23-1) ...
Setting up libpcsclite1:amd64 (1.8.23-1) ...
Setting up opensc-pkcs11:amd64 (0.17.0-3) ...
Setting up easy-rsa (2.2.2-2) ...
Setting up libccid (1.4.29-1) ...
Setting up opensc (0.17.0-3) ...
Setting up pcscd (1.8.23-1) ...
Created symlink /etc/systemd/system/sockets.target.wants/pcscd.socket →
/lib/systemd/system/pcscd.socket.
Processing triggers for libc-bin (2.27-3ubuntu1.6) ...
Processing triggers for systemd (237-3ubuntu10.57) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ureadahead (0.100.0-21) ...
vpn-vm-admin@vpn-server:~$ make-cadir ~/easy-rsa
vpn-vm-admin@vpn-server:~$ cd ~/easy-rsa
vpn-vm-admin@vpn-server:~/easy-rsa$ ./easyrsa init-pki
-bash: ./easyrsa: No such file or directory
vpn-vm-admin@vpn-server:~/easy-rsa$ ls
build-ca        build-key-pkcs12  inherit-inter      pkitool
build-dh        build-key-server  list-crl           revoke-full
build-inter     build-req         openssl-0.9.6.cnf  sign-req
build-key       build-req-pass    openssl-0.9.8.cnf  vars
build-key-pass  clean-all         openssl-1.0.0.cnf  whichopensslcnf
vpn-vm-admin@vpn-server:~/easy-rsa$ nano vars
vpn-vm-admin@vpn-server:~/easy-rsa$ source ./vars
************************************************************
  No /home/vpn-vm-admin/easy-rsa/openssl.cnf file could be found
  Further invocations will fail
************************************************************
NOTE: If you run ./clean-all, I will be doing a rm -rf on /home/vpn-vm-admin/easy-rsa/keys
-bash: ./vars: line 80: unexpected EOF while looking for matching `"'
-bash: ./vars: line 81: syntax error: unexpected end of file
vpn-vm-admin@vpn-server:~/easy-rsa$ ./clean-all
erver server
./build-dh
vpn-vm-admin@vpn-server:~/easy-rsa$ ./build-ca
grep: /home/vpn-vm-admin/easy-rsa/openssl.cnf: No such file or directory
```

```
pkitool: KEY_CONFIG (set by the ./vars script) is pointing to the wrong
version of openssl.cnf: /home/vpn-vm-admin/easy-rsa/openssl.cnf
The correct version should have a comment that says: easy-rsa version 2.x
vpn-vm-admin@vpn-server:~/easy-rsa$ ./build-key-server server
grep: /home/vpn-vm-admin/easy-rsa/openssl.cnf: No such file or directory
pkitool: KEY_CONFIG (set by the ./vars script) is pointing to the wrong
version of openssl.cnf: /home/vpn-vm-admin/easy-rsa/openssl.cnf
The correct version should have a comment that says: easy-rsa version 2.x
vpn-vm-admin@vpn-server:~/easy-rsa$ ./build-dh
Generating DH parameters, 2048 bit long safe prime, generator 2
This is going to take a long time
.....+.....+.....+.....+.....
.....
.....+.....+.....
.....+.....
.....
.....+.....
.....++*++*++*++*
vpn-vm-admin@vpn-server:~/easy-rsa$
```

OpenVPN Config:

Enter New CA Key Passphrase: vpn-vm-project

Common Name (eg: your user, host, or server name) [Easy-RSA CA]:openvpn-ansaf

CA creation complete and you may now import and sign cert requests.
Your new CA certificate file for publishing is at:
/home/vpn-vm-admin/easy-rsa/pki/ca.crt

vpn-vm-admin@vpn-server:~/easy-rsa$

```
vpn-vm-admin@vpn-server:~/easy-rsa$ ./easyrsa gen-req server nopass
Using SSL: openssl OpenSSL 1.1.1  11 Sep 2018
Generating a RSA private key
.....+++++
.....+++++
writing new private key to
'/home/vpn-vm-admin/easy-rsa/pki/easy-rsa-5090.Xw5I9w/tmp.u9nZFS'
-----
You are about to be asked to enter information that will be incorporated
```

into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Common Name (eg: your user, host, or server name) [server]:openvpn-ansaf

Keypair and certificate request completed. Your files are:
req: /home/vpn-vm-admin/easy-rsa/pki/reqs/server.req
key: /home/vpn-vm-admin/easy-rsa/pki/private/server.key

vpn-vm-admin@vpn-server:~/easy-rsa$

vpn-vm-admin@vpn-server:~/easy-rsa$ ./easyrsa sign-req server server
Using SSL: openssl OpenSSL 1.1.1 11 Sep 2018

You are about to sign the following certificate.
Please check over the details shown below for accuracy. Note that this request
has not been cryptographically verified. Please be sure it came from a trusted
source or that you have verified the request checksum with the sender.

Request subject, to be signed as a server certificate for 825 days:

subject=
    commonName   = openvpn-ansaf

Type the word 'yes' to continue, or any other input to abort.
  Confirm request details: yes
Using configuration from /home/vpn-vm-admin/easy-rsa/pki/easy-rsa-5138.hrSxNb/tmp.z1Bxv6
Enter pass phrase for /home/vpn-vm-admin/easy-rsa/pki/private/ca.key:
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName   :ASN.1 12:'openvpn-ansaf'
Certificate is to be certified until Feb 4 06:32:43 2027 GMT (825 days)

Write out database with 1 new entries
Data Base Updated

Certificate created at: /home/vpn-vm-admin/easy-rsa/pki/issued/server.crt

vpn-vm-admin@vpn-server:~/easy-rsa$

vpn-vm-admin@vpn-server:~/easy-rsa$ ./easyrsa gen-dh
Using SSL: openssl OpenSSL 1.1.1 11 Sep 2018
Generating DH parameters, 2048 bit long safe prime, generator 2
This is going to take a long time
.....+.....+.....
.....+.....+.....+.....+.....+.....
.....+....+.....+.....+.....
.....
.....+.....+.....+....
.....+.....+....+.....+.....
.....++*++*++*++*

DH parameters of size 2048 created at /home/vpn-vm-admin/easy-rsa/pki/dh.pem

vpn-vm-admin@vpn-server:~/easy-rsa$

pn-vm-admin@vpn-server:~/easy-rsa$ ls -l /etc/openvpn/server/server.conf
-rw-r--r-- 1 root root 338 Nov 1 06:14 /etc/openvpn/server/server.conf
vpn-vm-admin@vpn-server:~/easy-rsa$ sudo chmod 644 /etc/openvpn/server/server.conf
vpn-vm-admin@vpn-server:~/easy-rsa$ sudo chmod 755 /etc/openvpn/server
vpn-vm-admin@vpn-server:~/easy-rsa$ cat /lib/systemd/system/openvpn@.service
[Unit]
Description=OpenVPN connection to %i
PartOf=openvpn.service
Before=systemd-user-sessions.service
After=network-online.target
Wants=network-online.target
Documentation=man:openvpn(8)
Documentation=https://community.openvpn.net/openvpn/wiki/Openvpn24ManPage
Documentation=https://community.openvpn.net/openvpn/wiki/HOWTO

[Service]
Type=notify
PrivateTmp=true
WorkingDirectory=/etc/openvpn
ExecStart=/usr/sbin/openvpn --daemon ovpn-%i --status /run/openvpn/%i.status 10 --cd
/etc/openvpn --script-security 2 --config /etc/openvpn/%i.conf --writepid /run/openvpn/%i.pid

```
PIDFile=/run/openvpn/%i.pid
KillMode=process
CapabilityBoundingSet=CAP_IPC_LOCK CAP_NET_ADMIN CAP_NET_BIND_SERVICE
CAP_NET_RAW CAP_SETGID CAP_SETUID CAP_SYS_CHROOT CAP_DAC_OVERRIDE
CAP_AUDIT_WRITE
LimitNPROC=10
DeviceAllow=/dev/null rw
DeviceAllow=/dev/net/tun rw
ProtectSystem=true
ProtectHome=true
RestartSec=5s
Restart=on-failure

[Install]
WantedBy=multi-user.target
vpn-vm-admin@vpn-server:~/easy-rsa$ sudo mv /etc/openvpn/server/server.conf
/etc/openvpn/server.conf
vpn-vm-admin@vpn-server:~/easy-rsa$ sudo ln -s /etc/openvpn/server/server.conf
/etc/openvpn/server.conf
ln: failed to create symbolic link '/etc/openvpn/server.conf': File exists
vpn-vm-admin@vpn-server:~/easy-rsa$ sudo systemctl daemon-reload
vpn-vm-admin@vpn-server:~/easy-rsa$ sudo systemctl restart openvpn@server
vpn-vm-admin@vpn-server:~/easy-rsa$ sudo systemctl status openvpn@server
● openvpn@server.service - OpenVPN connection to server
   Loaded: loaded (/lib/systemd/system/openvpn@.service; indirect; vendor preset: enabled)
   Active: active (running) since Fri 2024-11-01 06:41:40 UTC; 5s ago
     Docs: man:openvpn(8)
           https://community.openvpn.net/openvpn/wiki/Openvpn24ManPage
           https://community.openvpn.net/openvpn/wiki/HOWTO
 Main PID: 5633 (openvpn)
   Status: "Initialization Sequence Completed"
    Tasks: 1 (limit: 2259)
   CGroup: /system.slice/system-openvpn.slice/openvpn@server.service
           └─5633 /usr/sbin/openvpn --daemon ovpn-server --status /run/openvpn/server.status 10
--cd

Nov 01 06:41:40 vpn-server ovpn-server[5633]: /sbin/ip addr add dev tun0 local 10.8.0.1 peer
10.8.0.2
Nov 01 06:41:40 vpn-server ovpn-server[5633]: /sbin/ip route add 10.8.0.0/24 via 10.8.0.2
Nov 01 06:41:40 vpn-server ovpn-server[5633]: Could not determine IPv4/IPv6 protocol. Using
AF_INET
Nov 01 06:41:40 vpn-server ovpn-server[5633]: Socket Buffers: R=[212992->212992]
S=[212992->212992]
```

Nov 01 06:41:40 vpn-server ovpn-server[5633]: UDPv4 link local (bound): [AF_INET][undef]:1194
Nov 01 06:41:40 vpn-server ovpn-server[5633]: UDPv4 link remote: [AF_UNSPEC]
Nov 01 06:41:40 vpn-server ovpn-server[5633]: MULTI: multi_init called, r=256 v=256
Nov 01 06:41:40 vpn-server ovpn-server[5633]: IFCONFIG POOL: base=10.8.0.4 size=62, ipv6=0
Nov 01 06:41:40 vpn-server ovpn-server[5633]: IFCONFIG POOL LIST
Nov 01 06:41:40 vpn-server ovpn-server[5633]: Initialization Sequence Completed
lines 1-22/22 (END)

Step 1: Generate a Client Certificate and Key

From your server, navigate to the Easy-RSA directory and generate a certificate for the client. Replace `client1` with any unique name for your client:

bash

```
cd ~/easy-rsa

./easyrsa gen-req ansaf nopass  # Generate the client request

./easyrsa sign-req client ansaf  # Sign the request with the CA
```

Client Config File:

```
client
dev tun
proto udp
remote 13.71.47.182 1194
resolv-retry infinite
nobind
persist-key
persist-tun
remote-cert-tls server
cipher AES-256-CBC
auth SHA256
compress lz4
verb 3

<ca>
-----BEGIN CERTIFICATE-----
MIIDSzCCAjOgAwIBAgIULw9CW5kRI859HxeqMDURYM/szFwwDQYJKoZIhvcNAQEL
BQAwFjEUMBIGA1UEAwwLRWFzeS1SU0EgQ0EwHhcNMjQxMTAxMDY1MjAyWhcNMzQx
MDMwMDY1MjAyWjAWMRQwEgYDVQQDDAtFYXN5LVJTQSBDQTCCASIwDQYJKoZIhvcN
AQEBBQADggEPADCCAQoCggEBAMAkLPq/+kcn+PQskyiB/GGwT3b0Z/P+9CCajy3a
Gtsu38pLHBGDQAFd8L2ha6QahA0ms2rnjRFiNtf5Psl7Y+E+zEJA2OIEBzAQNVYR
Am4VoJLGa5XSTd/9gHCVYNodWPoGhrPviWUAf7869/y1srZGqaKX5q4NaLXmRbsS
TWAj3x8fFzHTyrrLN4U7Dy7GPkBIUCS/c4kFXtvuK872FZwKNcEAOH2R39wZ0Qze
es+mgIKKzvhJSFuBZ1H0naGP3BFOxBJsaWb8e0kQfbRzfjKto8V+YZL+veqEXIZp
LLmpoeqeKukbg61SY5/vW5ecZsNV4SwSAe2i1ynE1iCb3vUCAwEAAaOBkDCBjTAd
BgNVHQ4EFgQU+GXrsn6uS1BQMeP26ezT9lk4gpEwUQYDVR0jBEowSIAU+GXrsn6u
S1BQMeP26ezT9lk4gpGhGqQYMBYxFDASBgNVBAMMC0Vhc3ktUlNBIENBghQvD0Jb
mREjzn0fF6owNRFgz+zMXDAMBgNVHRMEBTADAQH/MAsGA1UdDwQEAwIBBjANBgkq
hkiG9w0BAQsFAAOCAQEAWs7N+HRsrmNW8qYqHEKhiObYqN/7OnYOwpqTEhQp+LEF
mrvEbqbvcAL2NoT8wDVuPTVhdCf1ZrbS+xlnNDY5C16vRU6oywjJxjs5pC6zlh34
jv6MMjK3M/73s5aRYHDxvvz/OMPeqVMbWY9madjAGSIvULKd1GBke3TsN8ym5qHp
xBROF6Dc7JcQN7ZrjV+KbPa0d8t4br6+4RO0+lO20ox35T+Fe62CindxKCPJeptB
7k/Ch5pUw+znFXtzGs7Bc+2pfPvHwM5wtxAM+A+sl6+YNLJgke3K+Id5Mr9hZx07
3Y/fimnv8fySdNDDC++1k67H5XJHD014nCnZ0HIX8Q==
-----END CERTIFICATE-----
</ca>
<cert>
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            da:02:51:80:16:b9:42:31:85:08:fd:aa:ae:4f:c4:13
```

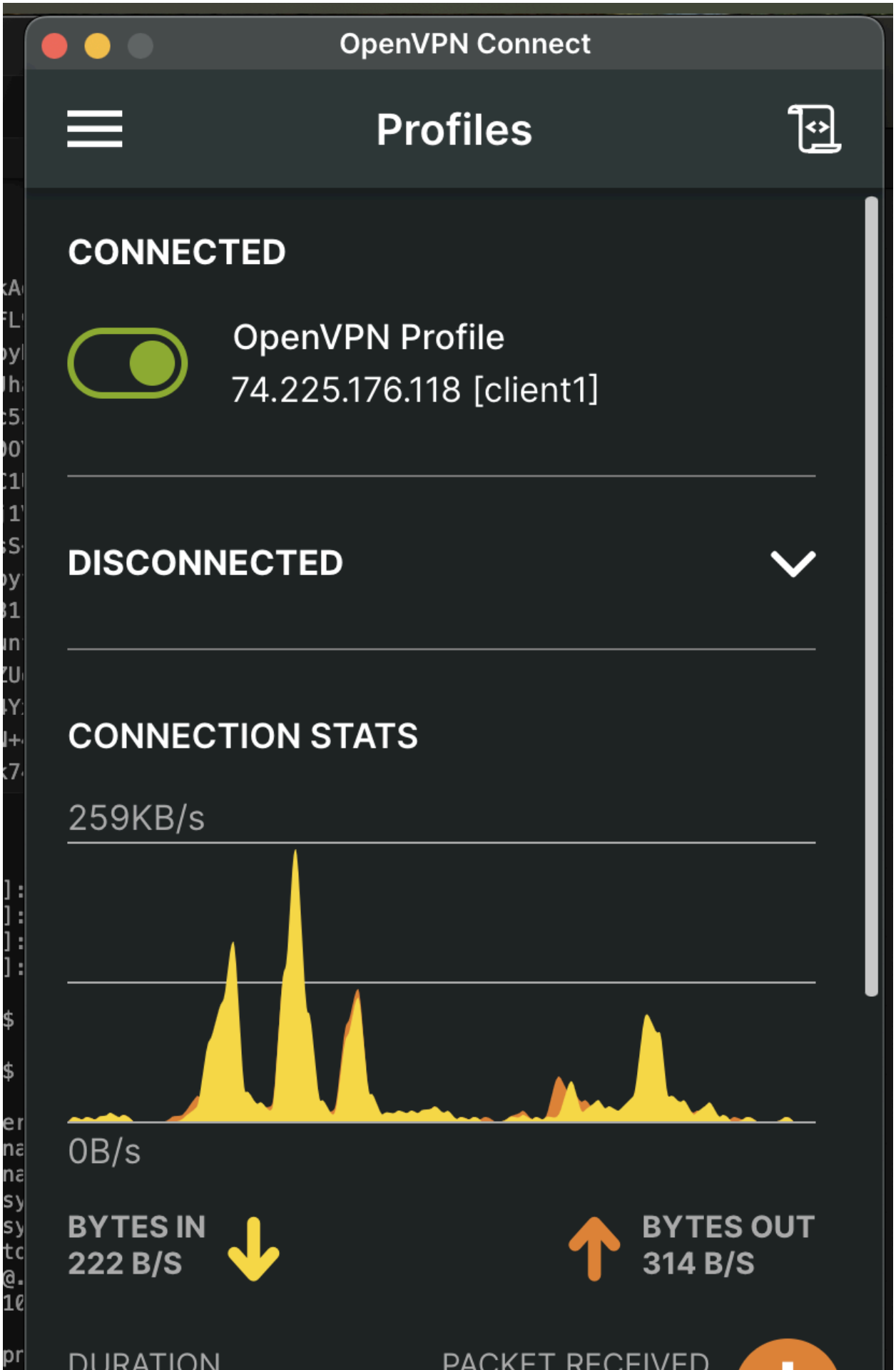
```
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: CN=Easy-RSA CA
    Validity
        Not Before: Nov  1 06:53:31 2024 GMT
        Not After : Feb  4 06:53:31 2027 GMT
    Subject: CN=ansaf
    Subject Public Key Info:
        Public Key Algorithm: rsaEncryption
            RSA Public-Key: (2048 bit)
            Modulus:
                00:d3:e3:63:98:9e:e1:5e:a4:73:9c:1f:99:07:f2:
                c8:af:cc:f7:67:fd:e7:e1:2c:94:1d:91:75:b0:56:
                d7:07:60:ba:bd:1c:ad:e0:20:c3:69:26:04:b1:9f:
                07:80:49:2e:51:8f:d6:5f:e4:95:5d:4f:7a:8d:81:
                ed:b9:ae:8c:2c:7a:f4:d9:e5:4b:60:67:98:c8:f4:
                73:e2:50:43:49:3d:5e:ca:1e:25:fb:5c:08:e4:75:
                42:70:e0:c1:92:96:00:ad:95:aa:4e:df:88:81:6a:
                95:1d:ae:0b:cd:aa:09:13:2c:b0:de:59:50:dc:f1:
                a8:77:66:ba:b2:3a:1f:8a:ce:be:8a:c7:29:79:30:
                93:a3:c6:dd:32:7b:2b:96:cd:2a:4e:1a:66:8e:db:
                69:af:6a:5f:ac:e1:07:e1:39:a6:38:92:72:bf:b6:
                8e:bc:db:7e:11:dc:38:21:b4:06:75:36:e6:ef:16:
                df:9c:09:7e:ac:1a:eb:a1:dd:2f:31:1b:7c:58:b8:
                d0:85:6c:9b:d3:c4:17:f5:cd:05:a9:d2:39:0b:3d:
                58:d9:72:1e:4c:34:69:81:18:60:a3:5f:a4:f2:09:
                d7:48:59:7c:ea:b0:ed:2c:c5:1d:aa:33:50:5f:42:
                6d:cd:79:6a:b8:2d:2f:47:d5:8a:fe:90:35:2c:eb:
                5e:27
            Exponent: 65537 (0x10001)
    X509v3 extensions:
        X509v3 Basic Constraints:
            CA:FALSE
        X509v3 Subject Key Identifier:
            10:AA:FD:13:6B:5F:27:7E:FF:0B:4E:32:88:0B:4E:55:47:03:6D:DB
        X509v3 Authority Key Identifier:
            keyid:F8:65:EB:B2:7E:AE:4B:50:50:31:E3:F6:E9:EC:D3:F6:59:38:82:91
            DirName:/CN=Easy-RSA CA
            serial:2F:0F:42:5B:99:11:23:CE:7D:1F:17:AA:30:35:11:60:CF:EC:CC:5C

        X509v3 Extended Key Usage:
            TLS Web Client Authentication
        X509v3 Key Usage:
            Digital Signature
Signature Algorithm: sha256WithRSAEncryption
```

89:9e:e2:ad:ef:70:26:54:08:1b:99:e3:d2:8d:06:d1:bc:47:
df:f5:a4:5e:00:ae:07:fa:15:e4:15:9b:92:e1:5c:11:51:fc:
94:4c:24:e9:89:82:fc:fd:00:ba:1d:7e:ec:32:0e:d3:49:c3:
58:56:f1:1c:a8:86:35:f9:da:32:9e:0f:13:58:b9:15:c0:e1:
b7:09:8e:6d:94:31:c6:70:6d:d9:b3:83:d7:3f:63:35:51:0d:
35:0d:54:5d:8b:98:ad:02:7c:39:56:4a:27:9b:17:15:98:d8:
92:c8:9b:83:00:68:6a:3d:76:28:93:73:84:e1:fd:2a:27:02:
68:a4:36:10:0f:9b:47:64:f2:8e:2f:77:0c:ea:ca:9a:97:15:
51:04:47:00:26:2b:e2:e1:8d:ef:19:a4:44:f0:1f:bb:62:54:
7d:4e:a2:37:5a:e0:e6:03:a7:e8:65:84:5b:5f:04:de:b6:71:
98:d1:32:9e:a4:46:0b:92:06:5e:a1:8c:07:d1:03:3a:a3:4f:
11:15:07:28:a0:85:85:a8:1f:d8:da:ae:68:d5:a7:65:c2:8d:
15:a1:d4:3f:f4:7b:03:22:8b:18:f9:2e:ed:6a:da:c3:c0:ed:
de:e6:4b:74:d2:45:a4:64:19:ba:32:8f:d4:a2:ba:bf:c5:f1:
7c:f3:2a:1f
-----BEGIN CERTIFICATE-----
MIIDVDCCAjygAwIBAgIRANoCUYAWuUIxhQj9qq5PxBMwDQYJKoZIhvcNAQELBQAw
FjEUMBIGA1UEAwwLRWFzeS1SU0EgQ0EwHhcNMjQxMTAxMDY1MzMxWhcNMjcwMjA0
MDY1MzMxWjAQMQ4wDAYDVQQDDAVhbnNhZjCCASIwDQYJKoZIhvcNAQEBBQADggEP
ADCCAQoCggEBANPjY5ie4V6kc5wfmQfyyK/M92f95+EsIB2RdbBW1wdgur0creAg
w2kmBLGfB4BJLIGP1l/klV1Peo2B7bmujCx69NnlS2BnmMj0c+JQQ0k9XsoeJftc
COR1QnDgwZKWAK2Vqk7fiIFqlR2uC82qCRMssN5ZUNzxqHdmurl6H4rOvorHKXkw
k6PG3TJ7K5bNKk4aZo7baa9qX6zhB+E5pjiScr+2jrzbfhHcOCG0BnU25u8W35wJ
fqwa66HdLzEbfFi40IVsm9PEF/XNBanSOQs9WNlyHkw0aYEYYKNfpPIJ10hZfOqw
7SzFHaozUF9Cbc15argtL0fViv6QNSzrXicCAwEAAaOBojCBnzAJBgNVHRMEAjAA
MB0GA1UdDgQWBBQQqv0Ta18nfv8LTjKIC05VRwNt2zBRBgNVHSMESjBIgBT4Zeuy
fq5LUFAx4/bp7NP2WTiCkaEapBgwFjEUMBIGA1UEAwwLRWFzeS1SU0EgQ0GCFC8P
QluZESPOfR8XqjA1EWDP7MxcMBMGA1UdJQQMMAoGCCsGAQUFBwMCMAsGA1UdDwQE
AwIHgDANBgkqhkiG9w0BAQsFAAOCAQEAiZ7ire9wJlQIG5nj0o0G0bxH3/WkXgCu
B/oV5BWbkuFcEVH8lEwk6YmC/P0Auh1+7DIO00nDWFbxHKiGNfnaMp4PE1i5FcDh
twmObZQxxnBt2bOD1z9jNVENNQ1UXYuYrQJ8OVZKJ5sXFZjYksibgwBoaj12KJNz
hOH9KicCaKQ2EA+bR2Tyji93DOrKmpcVUQRHACYr4uGN7xmkRPAfu2JUfU6iN1rg
5gOn6GWEW18E3rZxmNEynqRGC5IGXqGMB9EDOqNPERUHKKCFhagf2NquaNWnZcKN
FaHUP/R7AyKLGPku7Wraw8Dt3uZLdNJFpGQZujKP1KK6v8XxfPMqHw==
-----END CERTIFICATE-----
</cert>
<key>
-----BEGIN PRIVATE KEY-----
MIIEvAIBADANBgkqhkiG9w0BAQEFAASCBKYwggSiAgEAAoIBAQDT42OYnuFepHOc
H5kH8sivzPdn/efhLJQdkXWwVtcHYLq9HK3gIMNpJgSxnweASS5Rj9Zf5JVdT3qN
ge25rowsevTZ5UtgZ5jI9HPiUENJPV7KHiX7XAjkdUJw4MGSlgCtlapO34iBapUd
rgvNqgkTLLDeWVDc8ah3ZrqyOh+Kzr6Kxyl5MJOjxt0yeyuWzSpOGmaO22mval+s
4QfhOaY4knK/to68234R3DghtAZ1NubvFt+cCX6sGuuh3S8xG3xYuNCFbJvTxBf1
zQWp0jkLPVjZch5MNGmBGGCjX6TyCddIWXzqsO0sxR2qM1BfQm3NeWq4LS9H1Yr+

kDUs614nAgMBAAECggEAFow2BwGxPd8GJnD+WeguDvcixMAyMrYJtPPLsE4tZ62V
cZZhsl4KLWBMU5J2u+INucQyrdWnR1yzz+cpov6+D+WttJo/4SxlB81rn4OnBV4w
fhWjORJcJ+OKhHSGZNDtlYs9qlMDNbJf196zhtk3SgEReTfL8RcONFaMO3cP8DdK
OtfgIvRT7Gh/MIqnsdxbVBqb1O64GjAUFkLmgDz+EV/BNDlVhbPhlvQfMfwz/iHX
16wflc8lBQmui1H+VlZDxEeAWNd182/uQ7XrjpsIy0Y8IyST4zWiNlyOaPPrM+G8
92aZZp0WYkelL7kaqaTGafMknAGIkwJ9L02tiOcyoQKBgQD77GRjmMpRVU2eeZst
Ajy4mEoLadLxX+Lsvh9Q9uU5tvOCE/a0qigBHZe3PiCC772r00uB5cM6zbqQOg8/
Y0nspy2rlSXSfS6LFuqKedAcaOaAl+uP1RZZ08b2koYGaYuy4bNHwhMc+GvWxQd5
QfJfipCcbtdKRxMo3j3GHxr26QKBgQDXUSYYgxI9SYoSahsV1Yj8TUW/9gbj55Do
01Yh/KD8KsbbYZR4+sogReqQYC86m4EWWF0MvLVP+bwRgz37PqLdfnkibgtdPNHV
294tM27TWZVXzTWKlQPfhEAiIvi945EgLKq64JPwYjt9ZIlbh1hsR7Kax62toLFY
AKXzLjyijwKBgGKMXdUMPxDEAuIchhR8TI6VnFmMoyUjMl0UHqIlgzEpnWZkMt0H
tzCAiAn5trQh3pupw17kJ5QISJQE8lQEnjWCTH5TkQEfUSa95zAWaM/ERW9GfrK8
U8r96lYoiV9WyHxIiF5o0BtEHsNcPgk0QAEZ7moTfgqjdR5Gmfm3z0+JAoGAPgxu
lAXBjdPRbDkTkRk2Hq8N+KGnnqmuwsmUOrsYkVKqd7IJHUv4T5CWZgCx24vQiWXx
eqgu/9sR7WIKzIpiWL4Hnnpj3/yU47I2toPp3hIkzWaflU4YDaOI781wLRiVS9ZT
ZQMu3skQJ7R/ONcqDhojSCNe03hJYvjc2dVrN9UCgYB9NVvX/SWDxzDEUQ2zR+5o
wYKKlrGgkFDAgfkpziha/VvWtB0Cquw1WvOqN/eqLQUkdjCGkcdHlHnZGH/SU2It
B60FpZbzMOWvZcZ9l9WvF620yyOc4sdL8Z3+/XBsqm6WOM+xHO62jJySL2B2nsJS
QDI2WKuZQz+nZ7wnZLm+Yw==
-----END PRIVATE KEY-----
</key>

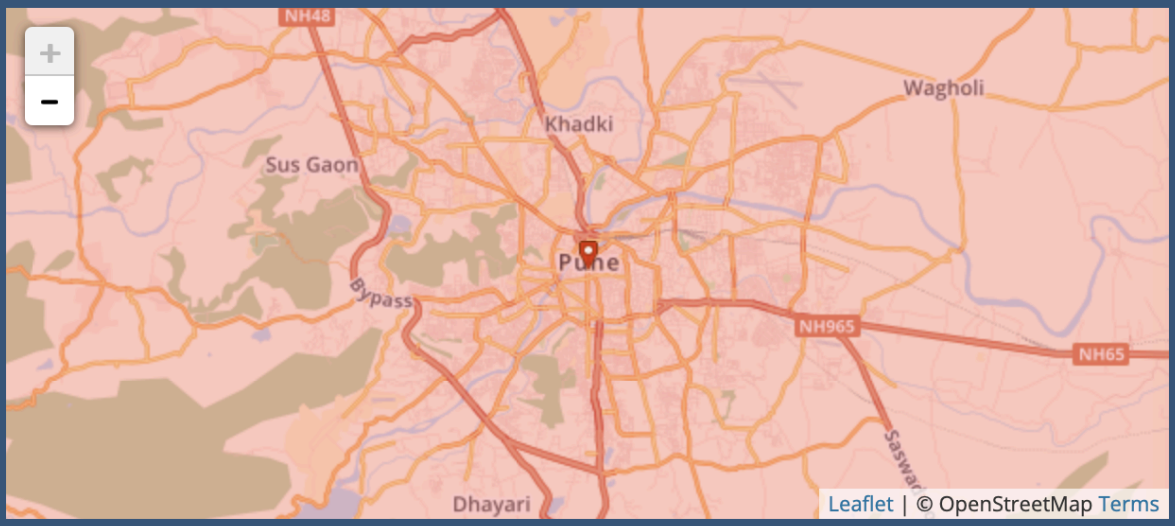
Redeploy vm CLI:
1. First start the machine:
az vm start --resource-group vpn_vm_rg --name vpn-server

2. Redeploy the machine:
az vm redeploy --resource-group vpn_vm_rg --name vpn-server


OpenVPN Connect
Profiles
CONNECTED
OpenVPN Profile
74.225.176.118 [client1]
DISCONNECTED
CONNECTION STATS
259KB/s
0B/s
BYTES IN
222 B/S
BYTES OUT
314 B/S
DURATION
PACKET RECEIVED

IP Details For: 74.225.176.118

Decimal:	1256304758
Hostname:	74.225.176.118
ASN:	8075
ISP:	Bellsouth.net Inc.
Services:	None detected
Country:	India
State/Region:	Maharashtra
City:	Pune
Latitude:	18.5197 (18° 31′ 10.79″ N)
Longitude:	73.8545 (73° 51′ 16.23″ E)



OpenVPN Setup with Terraform and Detailed Configuration Guide

This README provides step-by-step instructions on how to set up an OpenVPN server on a cloud virtual machine using Terraform, configure the VPN server and client, and troubleshoot

connection issues. It covers every step, from infrastructure provisioning to client connection verification.

Table of Contents

---

1. Introduction

In this guide, we set up an OpenVPN server on a cloud virtual machine, configured via Terraform. OpenVPN provides a secure VPN tunnel that allows clients to connect securely to a network, with all traffic routed through the server. This document includes all necessary commands, configurations, and troubleshooting steps to achieve a successful VPN connection.

---

2. Requirements

- **Terraform** installed for infrastructure provisioning
- **OpenVPN** installed on the server and client
- **Root or sudo access** on the server
- **Basic knowledge of SSH and server configuration**

---

3. Infrastructure Setup with Terraform

Step 1: Create a Terraform Configuration File

Write a Terraform configuration file (e.g., `main.tf`) to provision the virtual machine.

hcl
Copy code

```
provider "aws" {
  region = "us-west-2"
}

resource "aws_instance" "vpn_server" {
  ami           = "ami-0a313d6098716f372" # Update with a relevant
Linux AMI
  instance_type = "t2.micro"
  tags = {
    Name = "OpenVPN-Server"
  }
}
```

Step 2: Initialize and Apply Terraform

Initialize Terraform:
bash
Copy code

```
terraform init
```

1.

Apply the configuration to create the instance:
bash
Copy code

```
terraform apply
```

2.

Step 3: Connect to the Server via SSH

Once the instance is up, connect to it:

bash
Copy code

```
ssh -i path_to_key.pem ubuntu@<your_instance_public_ip>
```

---

4. OpenVPN Server Setup

Step 4: Update and Install OpenVPN

Update the system:
bash
Copy code

```
sudo apt update && sudo apt upgrade -y
```

1.

Install OpenVPN and Easy-RSA:
bash
Copy code

```
sudo apt install openvpn easy-rsa -y
```

2.

---

5. Certificate Authority and Key Generation

Step 5: Configure Easy-RSA and Build the Certificate Authority (CA)

Set up the Easy-RSA directory:
bash
Copy code

```
make-cadir ~/EasyRSA-3.0.8
cd ~/EasyRSA-3.0.8
```

1.

Initialize the PKI (Public Key Infrastructure):
bash
Copy code

```
./easyrsa init-pki
```

2.

Build the CA:
bash
Copy code
```
./easyrsa build-ca
```

3.

Generate the server certificate and key:
bash
Copy code
```
./easyrsa gen-req server nopass
./easyrsa sign-req server server
```

4.

Generate Diffie-Hellman parameters for encryption:
bash
Copy code
```
./easyrsa gen-dh
```

5.

Generate the client certificate and key:
bash
Copy code
```
./easyrsa gen-req client1 nopass
./easyrsa sign-req client client1
```

6.

Step 6: Transfer Certificates to the OpenVPN Directory

Copy certificates and keys to the `/etc/openvpn/server` directory:

bash
Copy code
```
sudo cp pki/ca.crt pki/private/server.key pki/issued/server.crt
pki/dh.pem /etc/openvpn/server
```

---

6. OpenVPN Server Configuration

Step 7: Create the OpenVPN Server Configuration File

Open the configuration file:
bash
Copy code

```
sudo nano /etc/openvpn/server/server.conf
```

1.

Paste the following configuration:
bash
Copy code

```
port 1194
proto udp
dev tun
ca ca.crt
cert server.crt
key server.key
dh dh.pem
topology subnet
server 10.8.0.0 255.255.255.0
push "redirect-gateway def1 bypass-dhcp"
push "dhcp-option DNS 8.8.8.8"
push "dhcp-option DNS 8.8.4.4"
keepalive 10 120
cipher AES-256-CBC
auth SHA256
compress lz4
persist-key
persist-tun
status /var/log/openvpn-status.log
verb 3
```

2.
3. Save and exit the file.

Step 8: Start and Enable OpenVPN

Start and enable the OpenVPN service:

bash
Copy code

```
sudo systemctl start openvpn@server
sudo systemctl enable openvpn@server
```

---

7. Firewall Rules Configuration

Step 9: Configure UFW or iptables

Enable firewall rules to allow OpenVPN traffic on port 1194:
bash
Copy code

```
sudo ufw allow 1194/udp
```

1.

Verify the rules:
bash
Copy code

```
sudo ufw status
```

2.

---

8. Client Configuration

Step 10: Generate Client Configuration File

Create a client configuration file on the server:
bash
Copy code

```
sudo nano /etc/openvpn/client/client1.ovpn
```

1.

Paste the following configuration, replacing `<SERVER_IP>` with your server's public IP:
bash
Copy code

```
client
dev tun
```

```
proto udp
remote <SERVER_IP> 1194
resolv-retry infinite
nobind
persist-key
persist-tun
cipher AES-256-CBC
auth SHA256
compress lz4
verb 3

<ca>
-----BEGIN CERTIFICATE-----
# Paste CA certificate here
-----END CERTIFICATE-----
</ca>
<cert>
-----BEGIN CERTIFICATE-----
# Paste Client certificate here
-----END CERTIFICATE-----
</cert>
<key>
-----BEGIN PRIVATE KEY-----
# Paste Client key here
-----END PRIVATE KEY-----
</key>
```

2.
3. Transfer `client1.ovpn` to the client machine.

---

9. Troubleshooting

Common Issues and Fixes

- **Peer Certificate Verification Failure**: Ensure the client certificate and key match the CA and server certificates on the server.
- **TLS Error in Logs**: Verify time sync between client and server.

- **IP Address Not Changing**: Check if `redirect-gateway` is set in the server configuration.

Example Commands for Debugging

Check OpenVPN service status:
bash
Copy code
```
sudo systemctl status openvpn@server
```

-

View OpenVPN logs:
bash
Copy code
```
sudo journalctl -u openvpn@server -e
```

-

---

10. Testing the VPN Connection

1. Import the `.ovpn` file on the client machine.

Connect using OpenVPN and verify the IP change:
bash
Copy code
```
curl ifconfig.me
```

2.

---

11. Final Notes

This completes the setup of an OpenVPN server with a secure client connection. Review the configurations and logs regularly for any issues.