

Terraform Task

Create a VPC with public subnet, private subnet, route table & IGW by Module in terraform

Name – Shashank Sharma

Note:

Before performing this practical you should configure your system

Use following commands to configure the **awscli**

Commands-

Use command as root user;

- `sudo apt install awscli`
- `aws configure --profile configs`
(put your access key & secret access key)
- `ls -a`
- `cd .aws`
- `pwd` (copy path & paste in DEV folder of main.tf -> module in module
=source “/home/username/.aws/credentials”)

1. Firstly create 1 main folder called as vpc-module-tf (your folder name).

📁 vpc module photo	07-03-2024 17:06	File folder
📁 vpc -module-tf	06-03-2024 10:54	File folder
📁 vpc-tf	04-03-2024 18:01	File folder

2. Create 2 sub-folder in main folder (module & resource).

Name	Date modified	Type	Size
📁 module	07-03-2024 12:55	File folder	
📁 resource	06-03-2024 10:56	File folder	

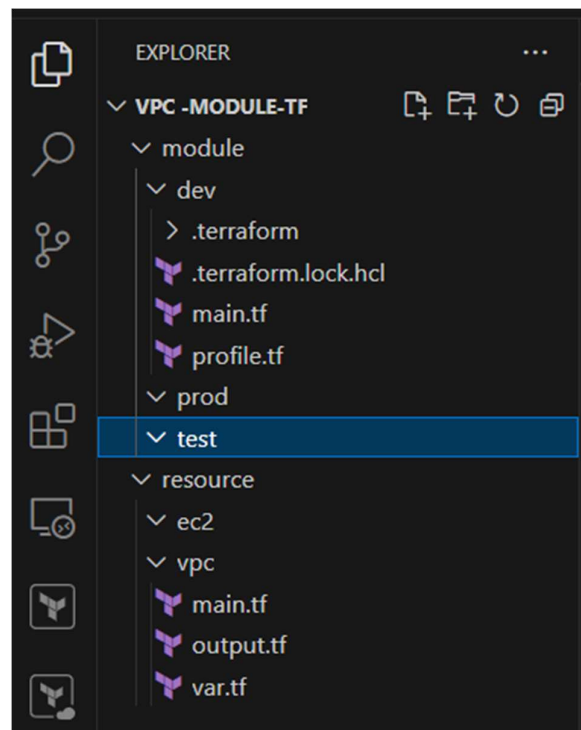
3. Create 3 sub-folder (dev, prod, test) in module folder.

Name	Date modified	Type	Size
dev	07-03-2024 13:09	File folder	
prod	06-03-2024 10:55	File folder	
test	06-03-2024 10:54	File folder	

4. Create 1 sub-folder (vpc) in resource folder.

Name	Date modified	Type	Size
ec2	06-03-2024 10:56	File folder	
vpc	06-03-2024 17:28	File folder	

5. Open VS code & make files in DEV folder as will as VPC folder.
6. In DEV folder
 - a. Main.tf
 - b. Profile.tf
7. In VPC folder
 - a. Main.tf
 - b. Output.tf
 - c. Var.tf



8. Write script in VPC main.tf

```

main.tf ...\vpc X  profile.tf  main.tf ...\dev  output.tf  var.tf
resource > vpc > main.tf > resource "aws_subnet" "private_sub" > vpc_id
1  #create vpc
2  resource "aws_vpc" "this_vpc" {
3      cidr_block = var.this_vpc_cidr_block //"26.11.0.0/16"
4
5      tags = {
6          Name = var.this_vpc_Name //"tf_vpc"
7      }
8
9  }
10
11 #create public subnet
12 resource "aws_subnet" "public_sub" {
13     vpc_id = aws_vpc.this_vpc.id
14     cidr_block = var.public_sub_cidr_block //"26.11.0.0/17"
15     tags = {
16         Name = var.public_sub_Name //"public_subnet"
17     }
18     map_public_ip_on_launch = var.public_sub_map_public_ip_on_launch //true    #to give internet access
19 }
20
21 #create private subnet
22 resource "aws_subnet" "private_sub" {
23     vpc_id = aws_vpc.this_vpc.id
24     cidr_block = var.private_sub_cidr_block //"26.11.128.0/18"
25     tags = {
26         Name = var.private_sub_Name //"private_subnet"
27     }
28     map_public_ip_on_launch = var.private_sub_map_public_ip_on_launch //false
29 }
30
31 #create private1 subnet
32 resource "aws_subnet" "private_one_sub" {
33     vpc_id = aws_vpc.this_vpc.id
34     cidr_block = var.private_one_sub_cidr_block //"26.11.192.0/20"
35     tags = {
36         Name = var.private_one_sub_Name //"private1_subnet"
37     }
38     map_public_ip_on_launch = var.private_one_sub_map_public_ip_on_launch //false
39 }
40

```

```
main.tf ...vpc X  profile.tf  main.tf ...dev  output.tf  var.tf
resource > vpc > main.tf > resource "aws_subnet" "private_sub" > vpc_id

40
41 #route table
42 resource "aws_default_route_table" "this_route_table" {
43     default_route_table_id = aws_vpc.this_vpc.default_route_table_id //aws_vpc.this_d_rt.default_route_table.id
44
45     route {
46
47         cidr_block = var.this_route_table_cidr_block //"0.0.0.0/0"
48         gateway_id = aws_internet_gateway.IGW.id //aws_internet_gateway.this_d_rt.id
49     }
50     tags = {
51         Name = var.this_route_table_Name //"tf_route"
52     }
53 }
54
55 #create internet gateway
56 resource "aws_internet_gateway" "IGW" {
57     vpc_id = aws_vpc.this_vpc.id
58     tags = {
59         Name = var.IGW_Name //"tf_IGW"
60     }
61 }
62 }
63
64 #create subnet associate route table
65 resource "aws_route_table_association" "this_rta" {
66     subnet_id = aws_subnet.public_sub.id
67     route_table_id = aws_default_route_table.this_route_table.id
68 }
```

9. Write script in VPC var.tf.



```
resource > vpc > var.tf > ...
1  #vpc
2  variable "this_vpc_cidr_block" {
3      type = string
4      //default = "26.11.0.0/16"
5  }
6
7  variable "this_vpc_Name" {
8      type = string
9      //default = "tf_vpc"
10 }
11
12 #public subnet
13 variable "public_sub_cidr_block" {
14     type = string
15     //default = "26.11.0.0/17"
16 }
17
18 variable "public_sub_Name" {
19     type = string
20     // default = "public_subnet"
21 }
22
23 #private subnet
24 variable "private_sub_cidr_block" {
25     type = string
26     // default = "26.11.128.0/18"
27 }
28
29 variable "private_sub_Name" {
30     type = string
31     // default = "private_subnet"
32 }
33 }
34
35 #private1 subnet
36 variable "private_one_sub_cidr_block" {
37     type = string
38     //default = "26.11.192.0/20"
39 }
40
```

```
main.tf ...\vpc  profile.tf  main.tf ...\dev  output.tf  var.tf  X
resource > vpc > var.tf > variable "private_sub_Name"

34  #private1 subnet
35  variable "private_one_sub_cidr_block" {
36      type = string
37      //default = "26.11.192.0/20"
38  }
39
40  variable "private_one_sub_Name" {
41      type = string
42      //default = "private1_subnet"
43  }
44
45  #route table
46  variable "this_route_table_cidr_block" {
47      type = string
48      //default = "0.0.0.0/0"
49  }
50
51  variable "this_route_table_Name" {
52      type = string
53      //default = "tf_route"
54  }
55
56  #internet gateway
57  variable "IGW_Name" {
58      type = string
59      // default = "tf_IGW"
60  }
61
62  #public subnet map
63  variable "public_sub_map_public_ip_on_launch" {
64      type = bool
65      //default = true
66  }
67
68  #private subnet map
69  variable "private_sub_map_public_ip_on_launch" {
70      type = bool
71      // default = false
72  }
73
```

```
main.tf ...\vpc  profile.tf  main.tf ...\dev  output.tf  var.tf X
resource > vpc > var.tf > variable "private_sub_Name"

56 #internet gateway
57 variable "IGW_Name" {
58     type = string
59     // default = "tf_IGW"
60 }
61
62 #public subnet map
63 variable "public_sub_map_public_ip_on_launch" {
64     type = bool
65     //default = true
66 }
67
68 #private subnet map
69 variable "private_sub_map_public_ip_on_launch" {
70     type = bool
71     // default = false
72 }
73
74 #private1 subnet map
75 variable "private_one_sub_map_public_ip_on_launch" {
76     type = bool
77     //default = false
78 }
79
```

10. Write script in VPC output.tf

```
main.tf ...\vpc  profile.tf  main.tf ...\dev  output.tf X  var.tf
resource > vpc > output.tf > output "private_subnet_id" > value

1  #vpc
2  output "vpc_id" {
3      description = "vpc id"
4      value = aws_vpc.this_vpc.id
5  }
6
7  #public subnet
8  output "public_subnet_id" {
9      description = "show public subnet id"
10     value = aws_subnet.private_sub.id
11 }
12
13 #private subnet
14 output "private_subnet_id" {
15     description = "show private subnet id"
16     value = aws_subnet.private_sub.id
17 }
18
```


11. Write script in DEV profile.tf

```
main.tf ...\vpc  profile.tf X  main.tf ...\dev  output.tf  var.tf
module > dev > profile.tf > ...
1  provider "aws" {
2      region                = "us-west-1"
3      profile                = "configs"
4      shared_credentials_files = ["/home/shashank/.aws/credentials"]
5
6  }
7
8  terraform {
9      backend "s3" {
10         bucket                = "tf-demo-new"
11         key                   = "terraform.tfstate"
12         dynamodb_table       = "tf-new-table"
13         region                = "us-west-1"
14         profile                = "configs"
15         shared_credentials_files = ["/home/shashank/.aws/credentials"]
16     }
17 }
18
19
```

12. Write script in DEV main.tf .

```
main.tf ...\vpc  profile.tf  main.tf ...\dev X  output.tf  var.tf
module > dev > main.tf > module "vpc" > private_one_sub_map_public_ip_on_launch
1  module "vpc" {
2      source                = "/mnt/c/shashank/shashank-module-tf/resource/vpc" #go vpc folder & copy path from terminal
3      this_vpc_cidr_block    = "26.11.0.0/16" #variable file->label & default values
4      this_vpc_Name          = "tf_vpc"
5      public_sub_cidr_block  = "26.11.0.0/17"
6      public_sub_Name        = "public_subnet"
7      private_sub_cidr_block = "26.11.128.0/18"
8      private_sub_Name       = "private_subnet"
9      private_one_sub_cidr_block = "26.11.192.0/20"
10     private_one_sub_Name    = "private1_subnet"
11     this_route_table_cidr_block = "0.0.0.0/0"
12     this_route_table_Name    = "tf_route"
13     IGW_Name                 = "tf_IGW"
14     public_sub_map_public_ip_on_launch = true
15     private_sub_map_public_ip_on_launch = false
16     private_one_sub_map_public_ip_on_launch = false
17
18
19
20
21
22
23
24
```

13. The final output is;

The screenshot displays the AWS Management Console interface for a specific VPC. The left-hand navigation pane lists various AWS services, with 'Virtual private cloud' expanded to show options like Subnets, Route tables, and Internet gateways. The main content area is titled 'vpc-0a2b2a6d634d7d9e7 / tf_vpc' and includes a 'Details' section with a table of VPC attributes and a 'Resource map' section showing a visual diagram of the network resources.

VPC Details:

Attribute	Value
VPC ID	vpc-0a2b2a6d634d7d9e7
State	Available
Tenancy	Default
Default VPC	No
Network Address Usage metrics	Disabled
DHCP option set	dopt-01de2ecfb57aad34
IPv4 CIDR	26.11.0.0/16
Route 53 Resolver DNS Firewall rule groups	-
DNS hostnames	Disabled
Main route table	rtb-0af6752b661a99983 / tf_route
IPv6 pool	-
Owner ID	381492218806
DNS resolution	Enabled
Main network ACL	acl-0ed56bac3cedbf33f
IPv6 CIDR	-

Resource map:

The resource map visualizes the VPC's components and their interconnections:

- VPC:** tf_vpc
- Subnets (3):** private_subnet, public_subnet, private1_subnet
- Route tables (1):** tf_route
- Network connections (1):** tf_IGW

Connections shown in the diagram include links from the subnets to the route table and from the route table to the internet gateway.

