

# Terraform lab1

aws | CentOS | Blog | Document... | Forums

https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#vpcDetails/vpcid=vpc-0ccdd611efbbb031

United States (N. Virginia) | mostafa\_practice @ mostafa-aws-practice

VPC > Your VPCs > vpc-0ccdd611efbbb031

### VPC dashboard

EC2 Global View

Filter by VPC

- Virtual private cloud
  - Your VPCs
  - Subnets
  - Route tables
  - Internet gateways
  - Egress-only internet gateways
  - Carrier gateways
  - DHCP option sets
  - Elastic IPs
  - Managed prefix lists
  - NAT gateways
  - Peering connections
  - Route servers
- Security
  - Network ACLs
  - Security groups
- PrivateLink and Lattice
  - Getting started
  - Endpoints
  - Endpoint services
  - Service networks
  - Lattice services
  - Resource configurations

## vpc-0ccdd611efbbb031 / terraform-lab1-vpc

Details

VPC ID: vpc-0ccdd611efbbb031

State: Available

Block Public Access: Off

DNS hostnames: Enabled

DNS resolution: Enabled

Tenancy: default

DHCP option set: dhcp1-04264dbc97b1197da

Main network ACL: acl-0c947fbc87bc7f50d

Default VPC: No

IPv4 CIDR: 10.0.0.0/16

Route 53 Resolver DNS Firewall rule groups: Disabled

Main route table: rtb-02ba64a4de45bd2ca

IPv6 pool: -

Owner ID: 774305593526

Resource map

Subnets (4): us-east-1a (public\_subnet1, private\_subnet1), us-east-1b (public\_subnet2, private\_subnet2)

Route tables (3): public-rt, rtb-02ba64a4de45bd2ca, private-rt

Network connections (1): terraform-lab1-gw

aws | CentOS | Blog | Document... | Forums

https://us-east-1.console.aws.amazon.com/ec2console/home?region=us-east-1#Instances

United States (N. Virginia) | mostafa\_practice @ mostafa-aws-practice

EC2 > Instances

### Instances (2)

Find Instance by attribute or tag (case-sensitive)

All states

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public
Application	i-00f4be14d4f9f0200	Running	t2.micro	Initializing	View alarms +	us-east-1a	-	-
Bastion	i-05a169216362e8059	Running	t2.micro	Initializing	View alarms +	us-east-1a	ec2-3-81-31-206.comp...	3.81.3

Select an instance

## Provider & backend

```
provider.tf x
provider.tf > provider "aws"
1 terraform {
2   required_providers {
3     aws = {
4       source = "hashicorp/aws"
5       version = "~> 5.0"
6     }
7   }
8 }
9
10 provider "aws" {
11   region = "us-east-1"
12 }

backend.tf x
backend.tf > _
1 terraform {
2   backend "s3" {
3     bucket = "terraform-state-bucket-mostafa"
4     key    = "terraform.tfstate"
5     region = "us-east-1"
6   }
7 }
8
```

## VPC & Subnet

```
vpc.tf x
vpc.tf > resource "aws_vpc" "lab1-vpc"
1 resource "aws_vpc" "lab1-vpc" {
2   cidr_block      = "10.0.0.0/16"
3   instance_tenancy = "default"
4   enable_dns_support = true
5   enable_dns_hostnames = true
6 }
7
8 tags = {
9   Name = "terraform-lab1-vpc"
10 }

subnet.tf x
subnet.tf > resource "aws_subnet" "private-sub2"
1 resource "aws_subnet" "pub-sub1" {
2   vpc_id      = aws_vpc.lab1-vpc.id
3   cidr_block  = "10.0.1.0/24"
4 }
5
6 tags = {
7   Name = "public_subnet1"
8 }
9
10 availability_zone = "us-east-1a"
11 map_public_ip_on_launch = true
12
13 resource "aws_subnet" "pub-sub2" {
14   vpc_id      = aws_vpc.lab1-vpc.id
15   cidr_block  = "10.0.2.0/24"
16   availability_zone = "us-east-1b"
17   map_public_ip_on_launch = true
18 }
19
20 tags = {
21   Name = "public_subnet2"
22 }
23
24 resource "aws_subnet" "private-sub1" {
25   vpc_id      = aws_vpc.lab1-vpc.id
26   cidr_block  = "10.0.3.0/24"
27   availability_zone = "us-east-1a"
28 }
29
30 tags = {
31   Name = "private_subnet1"
32 }
33
34 resource "aws_subnet" "private-sub2" {
35   vpc_id      = aws_vpc.lab1-vpc.id
36   cidr_block  = "10.0.4.0/24"
37   availability_zone = "us-east-1b"
38 }
39
40 tags = {
41   Name = "private_subnet2"
42 }
```

## Route tables & IGW

```
routeables.tf x
resource "aws_route_table" "public-rt" {
  vpc_id = aws_vpc.lab1-vpc.id

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.lab1-gw.id
  }

  tags = {
    Name = "public-rt"
  }
}

resource "aws_route_table_association" "public1-association" {
  subnet_id = aws_subnet.pub-sub1.id
  route_table_id = aws_route_table.public-rt.id
}

resource "aws_route_table_association" "public2-association" {
  subnet_id = aws_subnet.pub-sub2.id
  route_table_id = aws_route_table.public-rt.id
}

resource "aws_route_table" "private-rt" {
  vpc_id = aws_vpc.lab1-vpc.id

  tags = {
    Name = "private-rt"
  }
}

resource "aws_route_table_association" "private1-association" {
  subnet_id = aws_subnet.private-sub1.id
  route_table_id = aws_route_table.private-rt.id
}

resource "aws_route_table_association" "private2-association" {
  subnet_id = aws_subnet.private-sub2.id
  route_table_id = aws_route_table.private-rt.id
}

igw.tf x
resource "aws_internet_gateway" "lab1-gw" {
  vpc_id = aws_vpc.lab1-vpc.id

  tags = {
    Name = "terraform-lab1-gw"
  }
}
```

## Instances & SG s

```
sg.tf x
resource "aws_security_group" "bastion_sg" {
  name = "bastion-sg"
  vpc_id = aws_vpc.lab1-vpc.id

  ingress {
    from_port = 22
    to_port = 22
    protocol = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  egress {
    from_port = 0
    to_port = 0
    protocol = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }
}

resource "aws_security_group" "app_sg" {
  name = "app-sg"
  vpc_id = aws_vpc.lab1-vpc.id

  ingress {
    from_port = 22
    to_port = 22
    protocol = "tcp"
    cidr_blocks = [aws_vpc.lab1-vpc.cidr_block]
  }

  ingress {
    from_port = 3000
    to_port = 3000
    protocol = "tcp"
    cidr_blocks = [aws_vpc.lab1-vpc.cidr_block]
  }

  egress {
    from_port = 0
    to_port = 0
    protocol = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }
}

ec2.tf x
resource "aws_instance" "Bastion" {
  ami = "ami-084568db4383264d4"
  instance_type = "t2.micro"
  subnet_id = aws_subnet.pub-sub1.id
  vpc_security_group_ids = [aws_security_group.bastion_sg.id]
  associate_public_ip_address = true
  key_name = "TF-key"
  tags = {
    Name = "Bastion"
  }
}

resource "aws_instance" "Private" {
  ami = "ami-084568db4383264d4"
  instance_type = "t2.micro"
  subnet_id = aws_subnet.private-sub1.id
  vpc_security_group_ids = [aws_security_group.app_sg.id]
  associate_public_ip_address = false
  key_name = "TF-key"
  tags = {
    Name = "Application"
  }
}
```