A HAIDAR HAFIZ
RAHADYAN DANANG SUSETYO PRAMONO
R.P.A LEXY MANGLU SAPUTRA
AQIL YOGA PRANAWA

File profider.tf

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

File vpc.tf

1. VPC

```
resource "aws_vpc" "latihan-vpc-awan_mendung" {
    cidr_block = "10.0.0.0/18"
    tags = {
        Name = "latihan vpc awan mendung"
    }
}
```

2. Access control list (ACL)

```
resource "aws_network_acl" "latihan-acl-awan_mendung" {
    vpc_id = aws_vpc.latihan-vpc-awan_mendung.id

    ingress{
        protocol = "tcp"
        rule_no = 100
        action = "allow"
        cidr_block = "0.0.0.0/0"
        from_port = 3306
        to_port = 3306
    }

    ingress {
        protocol = "tcp"
        rule_no = 101
        action = "allow"
        cidr_block = "0.0.0.0/0"
        from_port = 22
        to_port = 22
    }

    egress {
        protocol = -1
        rule_no = 100
        action = "allow"
        cidr_block = "0.0.0.0/0"
        from_port = 0
        to_port = 0
    }

    tags = {
        Name = "latihan acl awan mendung"
    }
}
```

3. Availability Zone a

```
resource "aws_subnet" "latihan-public-subnet-awan_mendung" {
    vpc_id = aws_vpc.latihan-vpc-awan_mendung.id
    cidr_block = "10.0.0.0/24"
    map_public_ip_on_launch = "true"
    availability_zone = "eu-west-2a"

    tags = {
        Name = "latihan public subnet awan mendung"
    }
}

resource "aws_subnet" "latihan-private-subnet-awan_mendung" {
    vpc_id = aws_vpc.latihan-vpc-awan_mendung.id
    cidr_block = "10.0.1.0/24"
    map_public_ip_on_launch = "false"
    availability_zone = "eu-west-2a"

    tags = {
        Name = "latihan private subnet awan mendung"
    }
}
```

4. Availability Zone b (update)

```
resource "aws_subnet" "latihan-public-subnet2-awan_mendung" {
    vpc_id = aws_vpc.latihan-vpc-awan_mendung.id
    cidr_block = "10.0.2.0/24"
    map_public_ip_on_launch = "true"
    availability_zone = "eu-west-2b"

    tags = {
        Name = "latihan public subnet 2 awan mendung"
    }
}

resource "aws_subnet" "latihan-private-subnet2-awan_mendung" {
    vpc_id = aws_vpc.latihan-vpc-awan_mendung.id
    cidr_block = "10.0.3.0/24"
    map_public_ip_on_launch = "false"
    availability_zone = "eu-west-2b"

    tags = {
        Name = "latihan private subnet 2 awan mendung"
    }
}
```

5. Asosiasi ACL dengan private subnet

```
resource "aws_network_acl_association" "latihan-acl-assoc-awan_mendung" {
    network_acl_id = aws_network_acl.latihan-acl-awan_mendung.id
    subnet_id = aws_subnet.latihan-private-subnet-awan_mendung.id
}

resource "aws_network_acl_association" "latihan-acl-assoc2-awan_mendung" {
    network_acl_id = aws_network_acl.latihan-acl-awan_mendung.id
    subnet_id = aws_subnet.latihan-private-subnet2-awan_mendung.id
}
```

6. Internet gateway

```
resource "aws_internet_gateway" "latihan-igw-awan_mendung" {
    vpc_id = aws_vpc.latihan-vpc-awan_mendung.id

    tags = {
        Name = "latihan igw awan mendung"
    }
}
```

7. Router table public

```
resource "aws_route_table" "latihan-public-rt-awan_mendung" {
    vpc_id = aws_vpc.latihan-vpc-awan_mendung.id

    route {
        cidr_block = "0.0.0.0/0"
        gateway_id = aws_internet_gateway.latihan-igw-awan_mendung.id
    }

    tags = {
        Name = "latihan public rt awan mendung"
    }
}
```

8. Asosiasi Public Subnet dan Router Public (update)

```
resource "aws_route_table_association" "latihan-public-rta-awan_mendung" {
    subnet_id = aws_subnet.latihan-public-subnet-awan_mendung.id
    route_table_id = aws_route_table.latihan-public-rt-awan_mendung.id
}

resource "aws_route_table_association" "latihan-public-rta2-awan_mendung" {
    subnet_id = aws_subnet.latihan-public-subnet2-awan_mendung.id
    route_table_id = aws_route_table.latihan-public-rt-awan_mendung.id
}
```

File ec2.tf

1. Security Group untuk EC2

```
resource "aws_security_group" "latihan-security-group-awan_mendung" {
  description = "Allow limited inbound external traffic"
  vpc_id = "${aws_vpc.latihan-vpc-awan_mendung.id}"

  name = "latihan-sg-awan-mendung"

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

    ingress {
    protocol = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
    from_port = 3000
    to_port = 3000
  }

  ingress {
    protocol = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
    from_port = 3306
    to_port = 3306
  }

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

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

  tags = {
    Name = "latihan-sg-awan-mendung"
  }

}
```

2. Key Pair

```
resource  "tls_private_key" "rsa" {
  algorithm = "RSA"
  rsa_bits = 4096
}

resource "local_file" "LatihanPrivateKeyPairAwanMendung" {
  filename = "latihanKeyPairAwanMendung"
  content = tls_private_key.rsa.private_key_pem
}

resource "aws_key_pair" "latihanKeyPairAwanMendung" {
  key_name = "latihanKeyPairAwanMendung"
  public_key = tls_private_key.rsa.public_key_openssh
}
```

3. Data template file

```
data "template_file" "user_data" {
  template = "${file("scriptku.sh")}"
  vars = {
    rds_address = "${aws_db_instance.latihan_db_rds_awan_mendung.address}"
    rds_username = "${aws_db_instance.latihan_db_rds_awan_mendung.username}"
    rds_password = "${aws_db_instance.latihan_db_rds_awan_mendung.password}"
    rds_db_name = "${aws_db_instance.latihan_db_rds_awan_mendung.db_name}"
  }
}
```

4. scriptku.sh

```
#!/bin/bash
sudo apt-get update
sudo apt install default-mysql-client -y
#sudo mysql -u "${rds_username}" --password="${rds_password}" --host "${rds_address}" -e "CREATE DATABASE ${rds_db_name} /
*!40100 DEFAULT CHARACTER SET utf8 */;"
sudo apt install nodejs npm -y
cd /home/admin
sudo mkdir myapp
cd myapp
git clone https://github.com/jokoprsty/latihan_rds.git
cd latihan_rds
echo "DB_USER=${rds_username}" >> .env
echo "DB_PASS=${rds_password}" >> .env
echo "DB_NAME=${rds_db_name}" >> .env
echo "DB_HOST=${rds_address}" >> .env
sudo npm install
sudo npm install pm2 -g
sudo pm2 start /home/admin/myapp/latihan_rds/app.js -u admin --watch
sudo pm2 save
pm2 startup
```

5. EC2 instance (update)

```
locals {
  loc_ami = "ami-0efc5833b9d584374"
  loc_instance_type = "t2.micro"
}

resource "aws_instance" "latihan-ec2-awan_mendung" {
  depends_on = [ aws_db_instance.latihan_db_rds_awan_mendung ]
  ami = local.loc_ami
  instance_type = local.loc_instance_type
  key_name = aws_key_pair.latihanKeyPairAwanMendung.key_name
  vpc_security_group_ids = [ "${aws_security_group.latihan-security-group-awan_mendung.id}"]
  subnet_id = "${aws_subnet.latihan-public-subnet-awan_mendung.id}"
  user_data = data.template_file.user_data.rendered
  tags = {
    Name = "latihan-ec2-awan_mendung"
  }
}

resource "aws_instance" "latihan-ec2-2-awan_mendung" {
  depends_on = [ aws_db_instance.latihan_db_rds_awan_mendung ]
  ami = local.loc_ami
  instance_type = local.loc_instance_type
  key_name = aws_key_pair.latihanKeyPairAwanMendung.key_name
  vpc_security_group_ids = [ "${aws_security_group.latihan-security-group-awan_mendung.id}"]
  subnet_id = "${aws_subnet.latihan-public-subnet2-awan_mendung.id}"
  user_data = data.template_file.user_data.rendered
  tags = {
    Name = "latihan-ec2-2-awan_mendung"
  }
}
```

File rds.tf

1. Subnet group RDS

```
resource "aws_db_subnet_group" "latihan_subnet_db_awan_mendung"{
    name = "latihan_subnet_db"
    subnet_ids = [aws_subnet.latihan-private-subnet-awan_mendung.id,aws_subnet.latihan-private-subnet2-awan_mendung.id]

    tags = {
        Name = "latihan_subnet_db_awan_mendung"
    }
}
```

2. Security group untuk RDS

```
resource "aws_security_group" "latihan-rds-sg-awan_mendung" {
    name = "latihan_rds_sg_awan_mendung"
    vpc_id = aws_vpc.latihan-vpc-awan_mendung.id

    ingress {
        from_port = 3306
        to_port = 3306
        protocol = "tcp"
        cidr_blocks = ["0.0.0.0/0"]
    }

    egress {
        from_port = 3306
        to_port = 3306
        protocol = "tcp"
        cidr_blocks = ["0.0.0.0/0"]
    }

    tags = {
        Name = "latihan_rds_sg_awan_mendung"
    }
}
```

3. RDS (update)

```
resource "aws_db_instance" "latihan_db_rds_awan_mendung" {
    identifier = "latihan-db-rds-awanmendung"
    instance_class = "db.t3.micro"
    allocated_storage = 20
    engine = "mariadb"
    engine_version = "10.6.14"
    username = "latihan"
    password = 12345678
    db_name = "my_project"
    db_subnet_group_name = aws_db_subnet_group.latihan_subnet_db_awan_mendung.name
    vpc_security_group_ids = [aws_security_group.latihan-rds-sg-awan_mendung.id]
    publicly_accessible = false
    skip_final_snapshot = true
    multi_az = true
}
```

File lb.tf

1. Jenis load balancer

```
resource "aws_lb" "latihan-lb-awan-mendung" {
  name = "latihan-lb-awan-mendung"
  internal = false
  load_balancer_type = "application"
  security_groups = [aws_security_group.latihan-security-group-awan_mendung.id]
  subnets = [ aws_subnet.latihan-public-subnet-awan_mendung.id,aws_subnet.latihan-public-subnet2-awan_mendung.id ]

}
```

2. Load balancer target group

```
resource "aws_lb_target_group" "latihan-lb-target-group-awan-mendung" {
  name = "latihan-lb-target-group-awan"
  port = 3000
  protocol = "HTTP"
  vpc_id = aws_vpc.latihan-vpc-awan_mendung.id
}
```

3. Mengasosiasi target group dengan instance

```
resource "aws_lb_target_group_attachment" "latihan-lb-tga1-awan-mendung" {
  target_group_arn = aws_lb_target_group.latihan-lb-target-group-awan-mendung.arn
  target_id = aws_instance.latihan-ec2-awan_mendung.id
  port = 3000
}


resource "aws_lb_target_group_attachment" "latihan-lb-tga2-awan-mendung" {
  target_group_arn = aws_lb_target_group.latihan-lb-target-group-awan-mendung.arn
  target_id = aws_instance.latihan-ec2-2-awan_mendung.id
  port = 3000
}
```

4. Load balancer listener

```
resource "aws_lb_listener" "latihan_lb_listener_awan_mendung" {
  load_balancer_arn = aws_lb.latihan-lb-awan-mendung.arn
  port = "3000"
  protocol = "HTTP"

  default_action {
    type = "forward"
    target_group_arn = aws_lb_target_group.latihan-lb-target-group-awan-mendung.arn
  }
}
```

Terraform init

```
PS C:\Users\acer\Documents\Tugas Ongoing\Pertemuan 12 Awan\AwanMendung> terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/local from the dependency lock file
- Reusing previous version of hashicorp/template from the dependency lock file
- Reusing previous version of hashicorp/aws from the dependency lock file
- Reusing previous version of hashicorp/tls from the dependency lock file
- Using previously-installed hashicorp/aws v5.94.1
- Using previously-installed hashicorp/tls v4.0.6
- Using previously-installed hashicorp/local v2.5.2
- Using previously-installed hashicorp/template v2.2.0

Terraform has been successfully initialized!
```

Terraform plan

```
PS C:\Users\acer\Documents\Tugas Ongoing\Pertemuan 12 Awan\AwanMendung> terraform plan

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create
 <= read (data resources)

Terraform will perform the following actions:

  # data.template_file.user_data will be read during apply
  # (config refers to values not yet known)
 <= data "template_file" "user_data" {
      + id       = (known after apply)
      + rendered = (known after apply)
      + template = <<-EOT
            #!/bin/bash
            sudo apt-get update
```

Terraform 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.
PS C:\Users\acer\Documents\Tugas Ongoing\Pertemuan 12 Awan\AwanMendung> terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create
 <= read (data resources)

Terraform will perform the following actions:

  # data.template_file.user_data will be read during apply
  # (config refers to values not yet known)
 <= data "template_file" "user_data" {
      + id       = (known after apply)
```

Jalankan aplikasi web

1. Buka browser, kemudian copy alamat dns pada menu load balancer di halaman ec2 ke browser, buka dengan port 3000(jika tidak muncul, tunggu beberapa saat, cek ec2 sampai selesai initializing): dns_load_balancer:3000/books

2. Insert data ke dalam mysql melalui ec2(gunakan ssh), insert data dengan menggunakan alamat mysql yang ada di /home/admin/myapp/latihan_rds/.env, kemudian buka di browser dns_load_balancer:3000/books/1

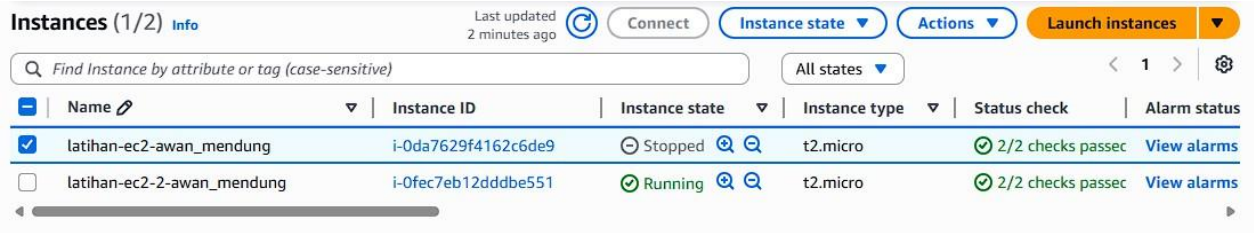
```
MariaDB [my_project]> desc books;
ERROR 1146 (42S02): Table 'my_project.books' doesn't exist
MariaDB [my_project]> desc Books;
+-----------+--------------+------+-----+---------+----------------+
| Field     | Type         | Null | Key | Default | Extra          |
+-----------+--------------+------+-----+---------+----------------+
| id        | int(11)      | NO   | PRI | NULL    | auto_increment |
| title     | varchar(255) | YES  |     | NULL    |                |
| author    | varchar(255) | YES  |     | NULL    |                |
| createdAt | datetime     | NO   |     | NULL    |                |
| updatedAt | datetime     | NO   |     | NULL    |                |
+-----------+--------------+------+-----+---------+----------------+
5 rows in set (0.002 sec)
```

```
MariaDB [my_project]> insert into Books (title, author, createdAt, updatedAt) va
lues("pelangi", "giring", NOW(), NOW());
Query OK, 1 row affected (0.003 sec)
```

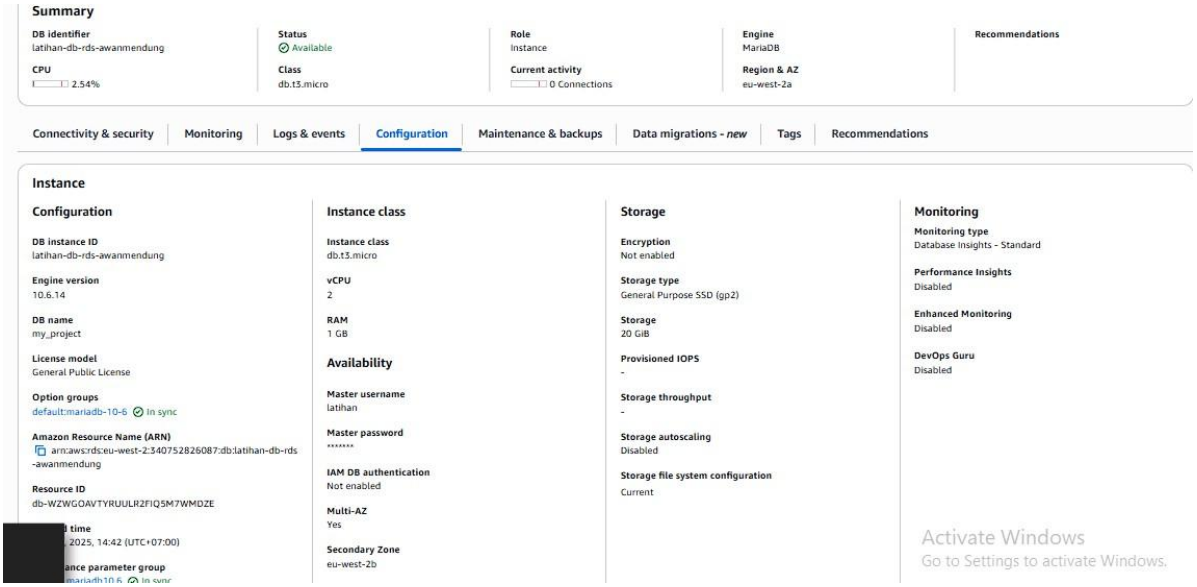
```
PuTTY (inactive)
gi", "giring", NOW());
ERROR 1364 (HY000): Field 'updatedAt' doesn't have a default value
MariaDB [my_project]> insert into Books (title, author, createdAt, updatedAt) va
lues("pelangi", "giring", NOW(), NOW());
Query OK, 1 row affected (0.003 sec)

MariaDB [my_project]> delete from Books where title="pelangi";
Query OK, 1 row affected (0.003 sec)

MariaDB [my_project]> insert into Books (title, author, createdAt, updatedAt) va
lues("pelangi", "giring", NOW(), NOW());
Query OK, 1 row affected (0.003 sec)

MariaDB [my_project]>
Broadcast message from root@ip-10-0-0-233 (Fri 2025-05-23 08:25:03 UTC):

The system will power off now!


Broadcast message from root@ip-10-0-0-233 (Fri 2025-05-23 08:25:03 UTC):

The system will power off now!
```

```
Not secure  latihan-lb-awan-mendung-398068383.eu-west-2.elb.amazonaws.com:3000/books
Pretty-print
[{"id":2,"title":"pelangi","author":"giring","createdAt":"2025-05-23T08:17:36.000Z","updatedAt":"2025-05-23T08:17:36.000Z"}]
```

3. stop ec2 pada availability zone a, tunggu sebentar sampai status stopped, kemudian cek lagi di browser ke-2 langkah di atas

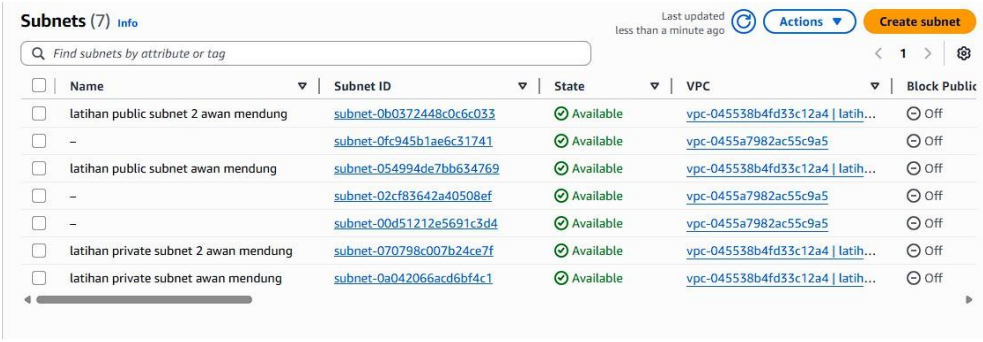


Cek failover pada RDS

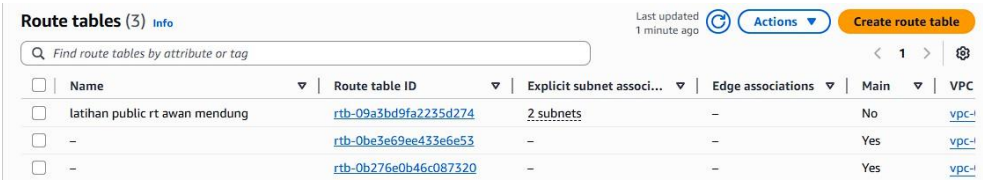


TUGAS:
Screenshot pada AWS

- VPC
  - Subnet

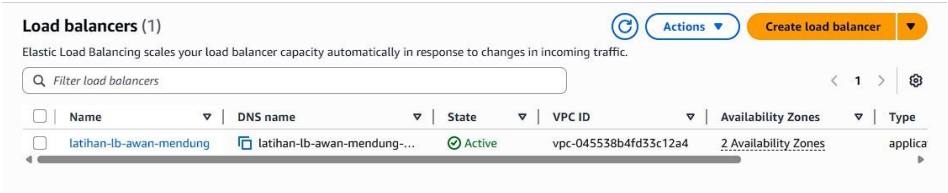


  - Route

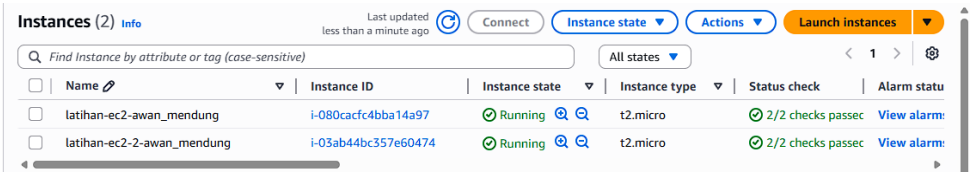


  - Router Association

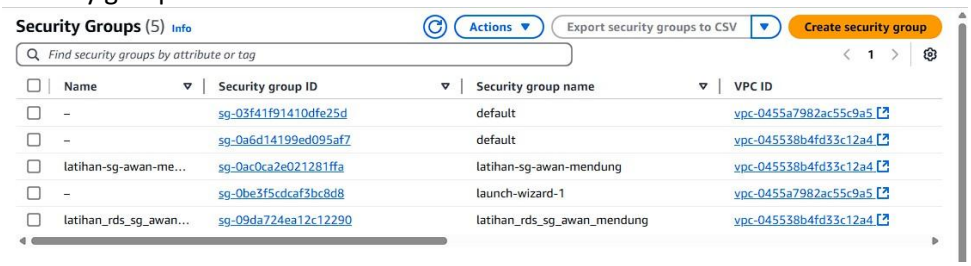
- Load balancers



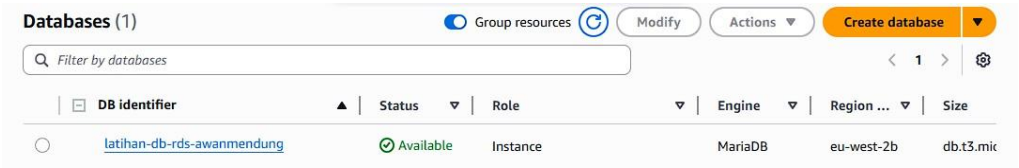
- EC2
  - Instance



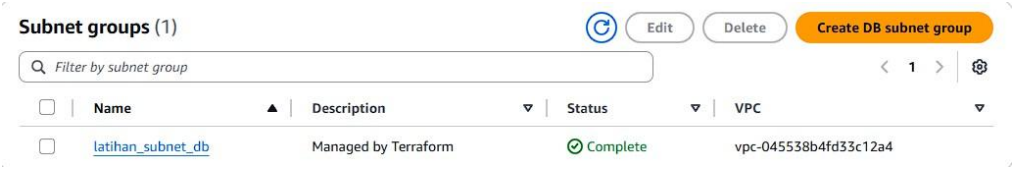
  - Security group



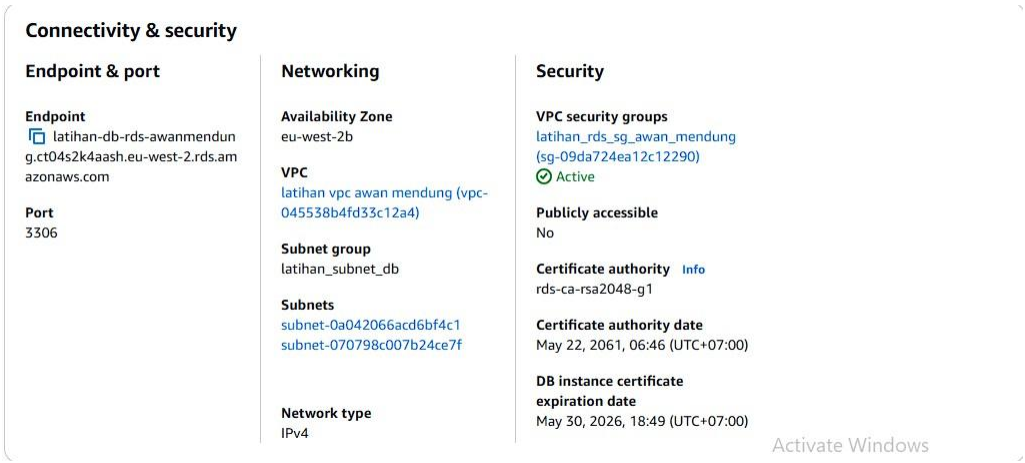
- RDS
  - Database



  - Subnet group



  - Security group



- Setelah semua praktikum selesai, lakukan:
    - terraform destroy

```
aws_db_instance.latihan_db_rds_awan_mendung: Still destroying... [id=db-7LTMN3RCDHYR63LHEQYUHDOYEQ, 5m40s elapsed]
aws_db_instance.latihan_db_rds_awan_mendung: Still destroying... [id=db-7LTMN3RCDHYR63LHEQYUHDOYEQ, 5m50s elapsed]
aws_db_instance.latihan_db_rds_awan_mendung: Still destroying... [id=db-7LTMN3RCDHYR63LHEQYUHDOYEQ, 6m0s elapsed]
aws_db_instance.latihan_db_rds_awan_mendung: Still destroying... [id=db-7LTMN3RCDHYR63LHEQYUHDOYEQ, 6m10s elapsed]
aws_db_instance.latihan_db_rds_awan_mendung: Still destroying... [id=db-7LTMN3RCDHYR63LHEQYUHDOYEQ, 6m20s elapsed]
aws_db_instance.latihan_db_rds_awan_mendung: Still destroying... [id=db-7LTMN3RCDHYR63LHEQYUHDOYEQ, 6m30s elapsed]
aws_db_instance.latihan_db_rds_awan_mendung: Destruction complete after 6m33s
aws_db_subnet_group.latihan_subnet_db_awan_mendung: Destroying... [id=latihan_subnet_db]
aws_security_group.latihan-rds-sg-awan_mendung: Destroying... [id=sg-09da724ea12c12290]
aws_db_subnet_group.latihan_subnet_db_awan_mendung: Destruction complete after 0s
aws_subnet.latihan-private-subnet2-awan_mendung: Destroying... [id=subnet-070798c007b24ce7f]
aws_subnet.latihan-private-subnet-awan_mendung: Destroying... [id=subnet-0a042066acd6bf4c1]
aws_subnet.latihan-private-subnet2-awan_mendung: Destruction complete after 2s
aws_subnet.latihan-private-subnet-awan_mendung: Destruction complete after 2s
aws_security_group.latihan-rds-sg-awan_mendung: Destruction complete after 2s
aws_vpc.latihan-vpc-awan_mendung: Destroying... [id=vpc-045538b4fd33c12a4]
aws_vpc.latihan-vpc-awan_mendung: Destruction complete after 1s

Destroy complete! Resources: 26 destroyed.
PS C:\Users\acer\Documents\Tugas Ongoing\Pertemuan 12 Awan\AwanMendung>
```