

```

provider "aws" {
  region = "us-east-1"
  access_key = "AKIAZF2ONWCFVYQ7G6SX"
  secret_access_key = "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"
}

resource "aws_instance" "my-first-server" {
  ami = "ami-08962a4068733a2b6"
  instance_type = "t2.micro"

  tags = {
    name = "karthik"
  }
}

resource "aws_vpc" "first-vpc" {
  cidr_block = "10.0.0.0/16"
  tags = {
    name = "test"
  }
}

resource "aws_subnet" "subnet-1" {
  vpc_id = aws_vpc.first-vpc.vpc_id
  cidr_block = "10.0.0.1.0/24"
  tags = {
    name = "test1-subnet"
  }
}

resource "aws_security_group" "allow-web" {
  name = "allow-web-traffic"
  description = "Allow-web inbound traffic"
  vpc_id = aws_vpc.first-vpc.vpc_id
}

resource "aws_db_parameter_group" "default" {
  name = "mariadb"
  family = "mariadb10.2"

  parameter {
    name = "max_allowed_packet"
    value = "16777216"
  }
}

ingress{

```

```

    description = "https"
    from_port   = 443
    to_port     = 443
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
}
ingress{
    description = "http"
    from_port   = 80
    to_port     = 80
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
}
ingress{
    description = "http"
    from_port   = 22
    to_port     = 22
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
}
ingress{
    description = "mariabd"
    from_port   = 3306
    to_port     = 3306
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
}
egress{
    from_port = 0
    to_port   = 0
    protocol  = "-1"
    cidr_block = ["0.0.0.0/0"]
}
}

resource "aws_s3_bucket" "bucket" {
    bucket = "mybucket"
    acl    = "private"

    tags = {
        Name = "My bucket"
        environment= "task"
    }
}

locals {
    s3_origin_id = "myS3Origin"
}

```

```

resource "aws_cloudfront_distribution" "s3_distribution" {
  origin {
    domain_name = aws_s3_bucket.b.bucket_regional_domain_name
    origin_id   = local.s3_origin_id

    s3_origin_config {
      origin_access_identity = "origin-access-identity/cloudfront/myS3Origin"
    }
  }

  enabled          = true
  is_ipv6_enabled  = true
  comment          = "this is a interview test"
  default_root_object = "index.html"

  logging_config {
    include_cookies = false
    bucket          = "mybucket.s3.amazonaws.com"
    prefix          = "myprefix"
  }

  aliases = ["mysite.example.com", "yoursite.example.com"]

  default_cache_behavior {
    allowed_methods = ["DELETE", "GET", "HEAD", "OPTIONS", "PATCH", "POST", "PUT"]
    cached_methods = ["GET", "HEAD"]
    target_origin_id = local.s3_origin_id

    forwarded_values {
      query_string = false

      cookies {
        forward = "none"
      }
    }
  }

  viewer_protocol_policy = "allow-all"
  min_ttl                = 0
  default_ttl            = 3600
  max_ttl                = 86400
}

# Cache behavior with precedence 0
ordered_cache_behavior {
  path_pattern      = "/content/immutable/*"
  allowed_methods   = ["GET", "HEAD", "OPTIONS"]
  cached_methods    = ["GET", "HEAD", "OPTIONS"]
}

```

```

target_origin_id = local.s3_origin_id

forwarded_values {
  query_string = false
  headers      = ["Origin"]

  cookies {
    forward = "none"
  }
}

min_ttl          = 0
default_ttl      = 86400
max_ttl          = 31536000
compress         = true
viewer_protocol_policy = "redirect-to-https"
}

# Cache behavior with precedence 1
ordered_cache_behavior {
  path_pattern      = "/content/*"
  allowed_methods   = ["GET", "HEAD", "OPTIONS"]
  cached_methods    = ["GET", "HEAD"]
  target_origin_id = local.s3_origin_id

  forwarded_values {
    query_string = false

    cookies {
      forward = "none"
    }
  }

  min_ttl          = 0
  default_ttl      = 3600
  max_ttl          = 86400
  compress         = true
  viewer_protocol_policy = "redirect-to-https"
}

price_class = "PriceClass_200"

restrictions {
  geo_restriction {
    restriction_type = "whitelist"
    locations        = ["US", "CA", "GB", "DE"]
  }
}
}

```

```
tags = {  
    Environment = "production"  
}  
  
viewer_certificate {  
    cloudfront_default_certificate = true  
}  
}
```