

AWS CloudFormation

```
AWSTemplateFormatVersion: "2010-09-09"
Description: Creation of S3 Buckets
Parameters:
  env:
    Type: "String"
    Description: Environment Name
    AllowedValues:
      - dev
      - qa
      - prod
  CloudFormationBucket:
    Type: "String"
    Description: CloudFormation stacks Bucket Name
    Default: "stacks"
```

iam:

```
AWSTemplateFormatVersion: "2010-09-09"
Resources:
  RootRole:
    Type: "AWS::IAM::Role"
    Properties:
      AssumeRolePolicyDocument:
        Version: "2012-10-17"
        Statement:
          - Effect: "Allow"
            Principal:
              Service:
                - "ec2.amazonaws.com"
            Action:
              - "sts:AssumeRole"
      Path: "/"
  RolePolicies:
    Type: "AWS::IAM::Policy"
    Properties:
      PolicyName: "root"
      PolicyDocument:
        Version: "2012-10-17"
        Statement:
          - Effect: "Allow"
            Action: "*"
```

```

        Resource: "*"
    Roles:
        - Ref: "RootRole"
RootInstanceProfile:
    Type: "AWS::IAM::InstanceProfile"
    Properties:
        Path: "/"
    Roles:
        - Ref: "RootRole"

```

S3:

```

Resources:
    S3Bucket:
        Type: 'AWS::S3::Bucket'
        DeletionPolicy: Retain
        Properties:
            BucketName: !Sub ${var}

```

VPC:

```

myVPC:
    Type: AWS::EC2::VPC
    Properties:
        CidrBlock: 10.0.0.0/16
        EnableDnsSupport: 'true'
        EnableDnsHostnames: 'true'
        Tags:
            - Key: stack
              Value: production

```

Subnet:

```

mySubnet:
    Type: AWS::EC2::Subnet
    Properties:
        VpcId:
            Ref: myVPC
        CidrBlock: 10.0.0.0/24
        AvailabilityZone: "us-east-1a"
        Tags:
            - Key: stack

```

```
Value: production
```

security group

```
ServerSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: allow connections from specified CIDR ranges
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 80
        ToPort: 80
        CidrIp: 0.0.0.0/0
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        CidrIp: 192.168.1.1/32
```

EC2

```
EC2Instance:
  Type: AWS::EC2::Instance
  Properties:
    ImageId: !FindInMap [ AWSRegionArch2AMI, !Ref 'AWS::Region', PV64 ]
    KeyName: !Ref KeyName
    InstanceType: m1.small
    SecurityGroups:
      - !Ref Ec2SecurityGroup
    BlockDeviceMappings:
      -
        DeviceName: /dev/sdc
        VirtualName: ephemeral0
```

lambda

```
AWSTemplateFormatVersion: '2010-09-09'
Description: VPC function.
Resources:
  Function:
    Type: AWS::Lambda::Function
    Properties:
      Handler: index.handler
      Role: arn:aws:iam::123456789012:role/lambda-role
      Code:
        S3Bucket: my-bucket
        S3Key: function.zip
      Runtime: nodejs18.x
      Timeout: 5
      TracingConfig:
        Mode: Active
      VpcConfig:
        SecurityGroupIds:
```

```
    - sg-085912345678492fb
  SubnetIds:
    - subnet-071f712345678e7c8
    - subnet-07fd123456788a036
```

deploy

```
aws cloudformation deploy --template-file templates/s3.yaml --region
$REGION --stack-name s3-$ENVIRONMENT --parameter-overrides
env=$ENVIRONMENT
```