Exercise 0.4 Terraform S3 Backend Configuration (mins)

For this exercise we will use the AWS web console to create and configure and Amazon S3 bucket to be used as a Terraform Backend.

# Create an S3 Bucket

* Open a browser in your vm and navigate to <https://console.aws.amazon.com>.
* Sign in using the credentials you have been using for the AWS CLI
* In the search box at the top type “S3” and choose the S3 service.

A screenshot of a computer

Description automatically generated

* A screenshot of a computer

  Description automatically generatedClick on “Create Bucket” and choose a bucket name (which must be unique) and a region.
* Scroll down and under “Bucket Versioning” select “Enable”
* A screenshot of a computer

  Description automatically generatedA screenshot of a computer

  Description automatically generatedScroll down to “Default Encryption” and click “Enable”. Choose “AWS Key Management Service key” as the encryption type and leave the other options in their default settings as shown. Click “Create Bucket” at the bottom

# Modify the Bucket Policy

* Click on your new bucket in the list of S3 buckets
* Look under the “Properties” tab of your bucket to find the ARN of your bucket and copy that to a text editor like Notepad.
* From a git bash command line run the aws command to get your user ARN and copy it to the same text editor:

$ aws sts get-caller-identity --output json

(Copy in a Git Bash shell can be achieved by selecting with the mouse and typing CTRL+INSERT to copy the text to the clipboard)

* A screenshot of a computer

  Description automatically generatedNow go to the “Permissions” tab of your S3 bucket, scroll down to the “Bucket Policy” box and click “Edit”
* Amend the policy to match the policy below, substituting your user ARN and S3 ARN, then click “Save Changes” at the bottom :

{

    "Version": "2012-10-17",

    "Statement": [

        {

            "Effect": "Allow",

            "Principal": {

                "AWS": "<your\_user\_arn>"

            },

            "Action": "s3:ListBucket",

            "Resource": "<your\_bucket\_arn>"

        },

        {

            "Effect": "Allow",

            "Principal": {

                "AWS": "<your\_user\_arn>"

            },

            "Action": [

                "s3:GetObject",

                "s3:PutObject"

            ],

            "Resource": "<your\_bucket\_arn>/\*"

        }

    ]

}

# Create a DynamoDB Table