Install:

- `django-storages`
- `botocore`

AWS Settings in settings.py

```python
# AWS S3 Bucket settings
if "USE_AWS" in os.environ:
    # Cache control
    AWS_S3_OBJECT_PARAMETERS = {
        "Expires": "Thu, 31 Dec 2099 20:00:00 GMT",
        "CacheControl": "max-age=94608000",
    }
    # Bucket config
    AWS_STORAGE_BUCKET_NAME = "bucket-name"
    AWS_S3_REGION_NAME = "region-name"
    AWS_ACCESS_KEY_ID = os.environ.get("AWS_ACCESS_KEY_ID")
    AWS_SECRET_ACCESS_KEY = os.environ.get("AWS_SECRET_ACCESS_KEY")
    AWS_S3_CUSTOM_DOMAIN = f"{AWS_STORAGE_BUCKET_NAME}.s3.amazonaws.com"

    # Static and media files storage
    STATICFILES_STORAGE = "custom_storages.StaticStorage"
    STATICFILES_LOCATION = "static"
    STATIC_URL = f"https://{AWS_S3_CUSTOM_DOMAIN}/{STATICFILES_LOCATION}/"
    DEFAULT_FILE_STORAGE = "custom_storages.MediaStorage"
    MEDIAFILES_LOCATION = "media"
    MEDIA_URL = f"https://{AWS_S3_CUSTOM_DOMAIN}/{MEDIAFILES_LOCATION}/"
```

Custom Storages in custom_storages.py in the root of the project

custom_storages.py > ...

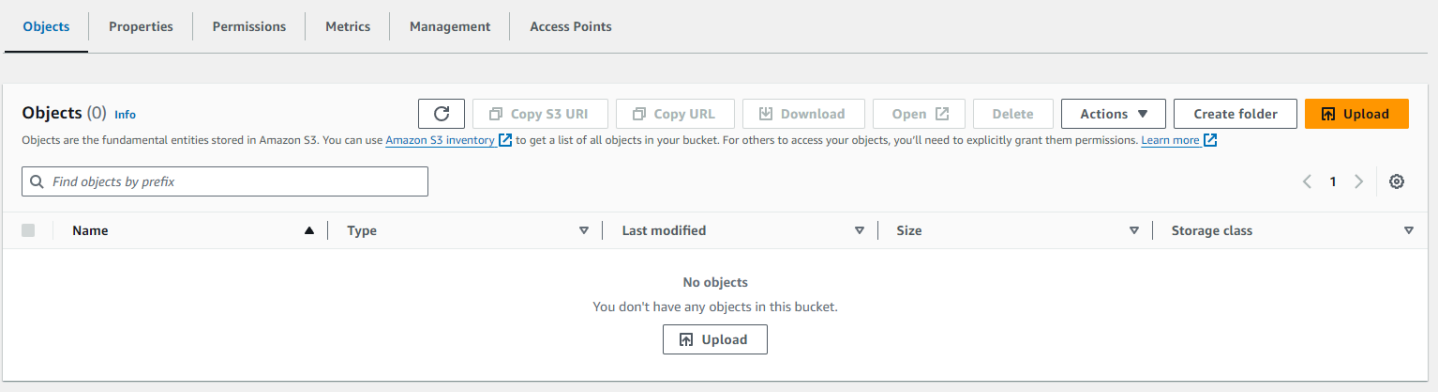
```python
...
from django.conf import settings
from storages.backends.s3boto3 import S3Boto3Storage


...
class StaticStorage(S3Boto3Storage):
    location = settings.STATICFILES_LOCATION


...
class MediaStorage(S3Boto3Storage):
    location = settings.MEDIAFILES_LOCATION
```

Create S3 Bucket

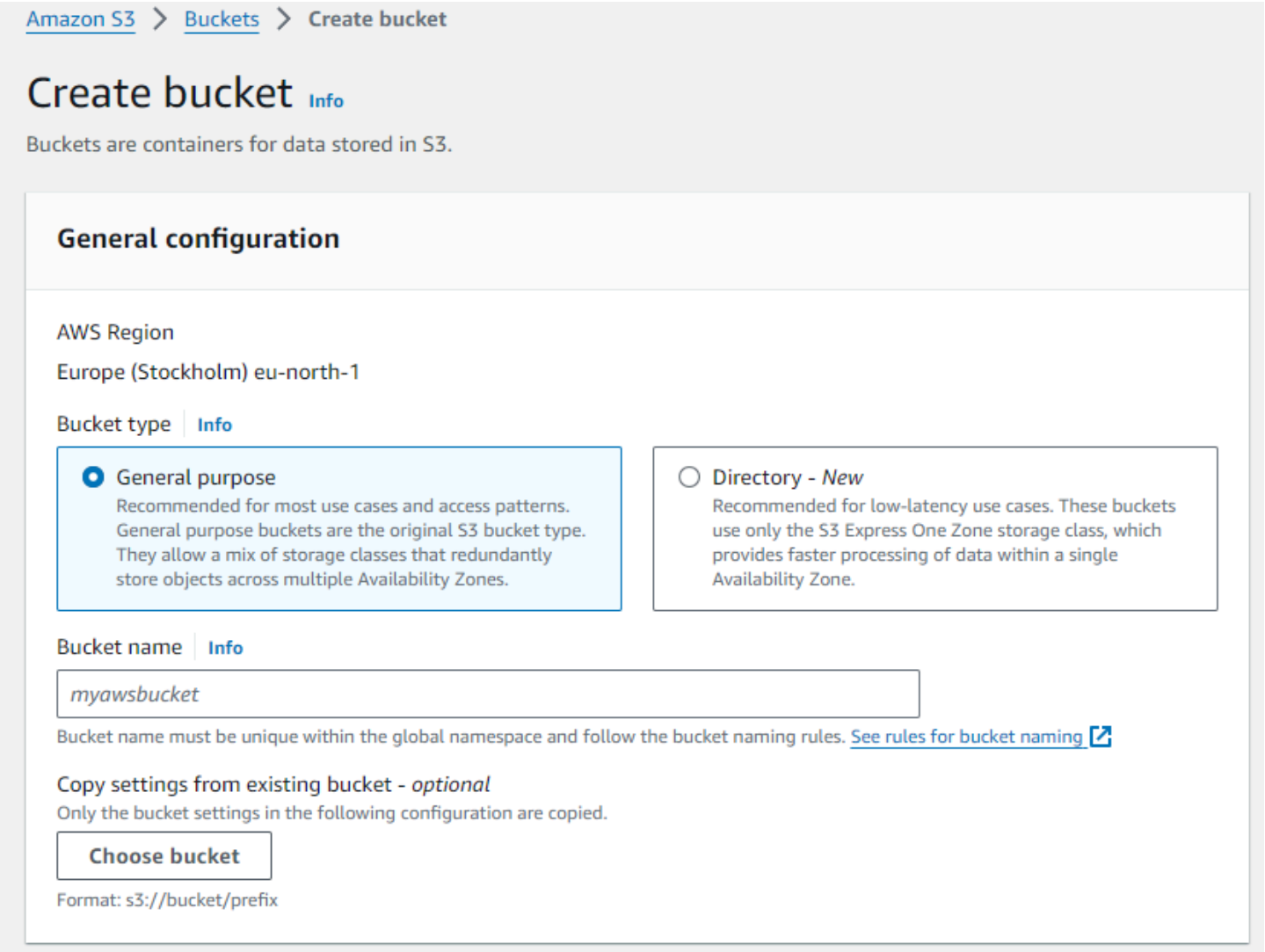
- Sign up for AWS
- Create bucket
- Set Static website hosting
- Create bucket policy and attach
- Set ACL to allow list access to everyone
- Edit CORS setting for bucket

Create Bucket

Sign in to AWS, go to S3 and select “Buckets”



Select “Create bucket”.



Enter bucket name and scroll down to Object Ownership.

Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

- ○ **ACLs disabled (recommended)**
  All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.
- ◉ **ACLs enabled**
  Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

⚠ We recommend disabling ACLs, unless you need to control access for each object individually or to have the object writer own the data they upload. Using a bucket policy instead of ACLs to share data with users outside of your account simplifies permissions management and auditing.

Object Ownership

- ◉ **Bucket owner preferred**
  If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.
- ○ **Object writer**
  The object writer remains the object owner.

Select “ACLs enabled”.
Leave “Bucket owner preferred” selected.
Scroll down to “Block Public Access” settings.

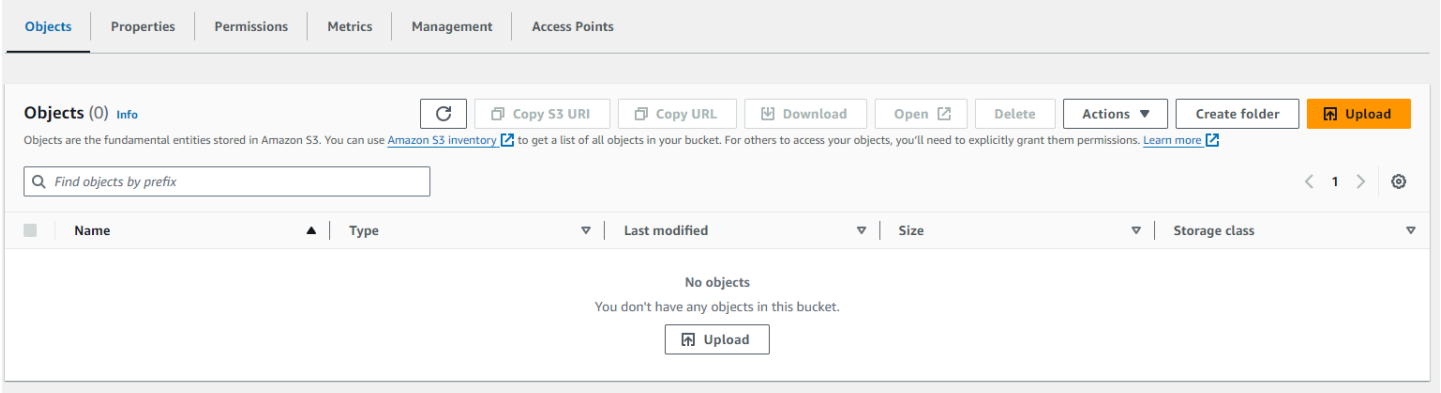
Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

- ☐ **Block *all* public access**
  Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.
  - ☐ **Block public access to buckets and objects granted through *new* access control lists (ACLs)**
    S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn’t change any existing permissions that allow public access to S3 resources using ACLs.
  - ☐ **Block public access to buckets and objects granted through *any* access control lists (ACLs)**
    S3 will ignore all ACLs that grant public access to buckets and objects.
  - ☐ **Block public access to buckets and objects granted through *new* public bucket or access point policies**
    S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
  - ☐ **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
    S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

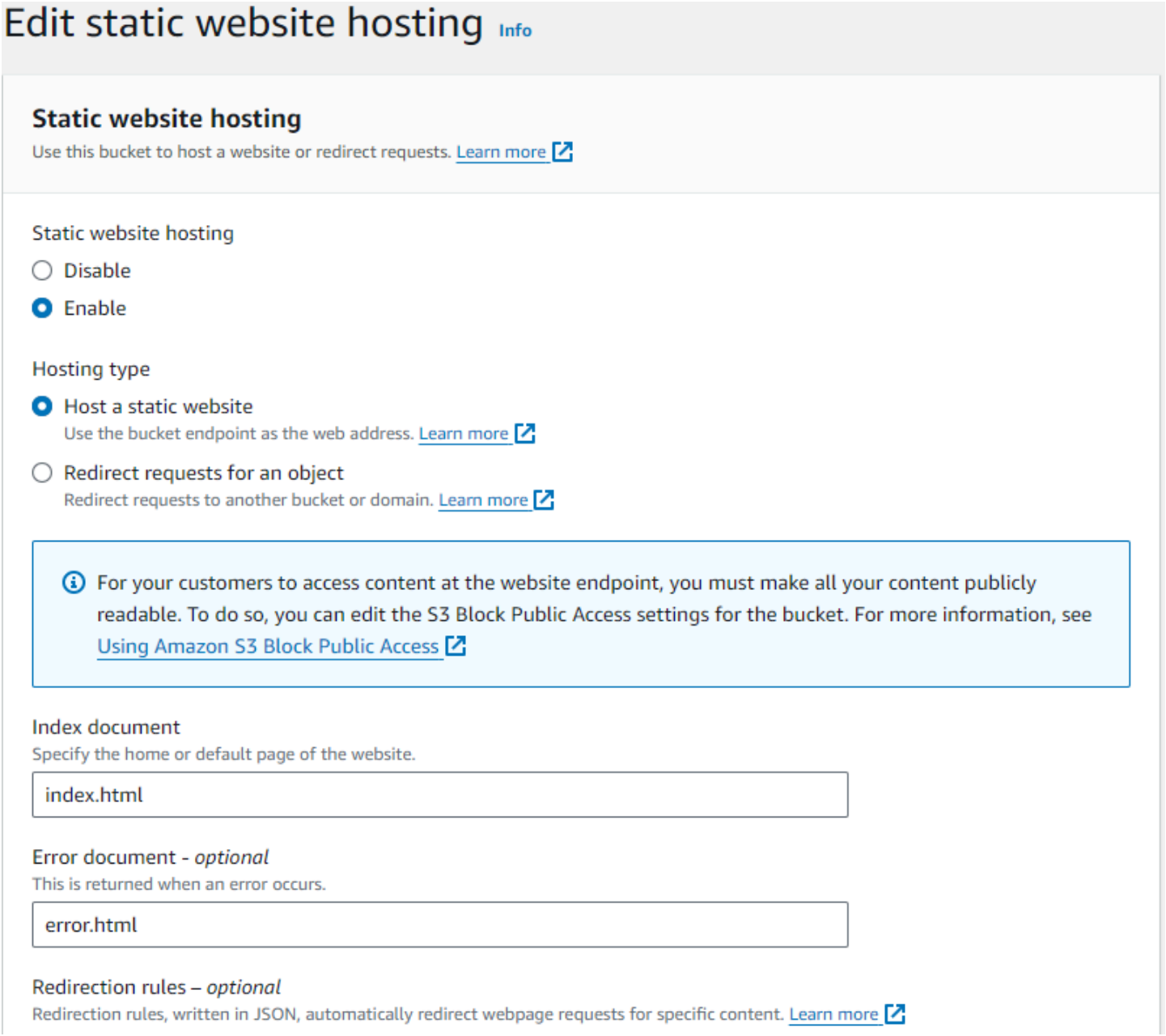
Deselect “Block all public access” and accept the warning.
Leave the remaining settings as they are.
Scroll to the end and select “Create bucket”.

Static website hosting



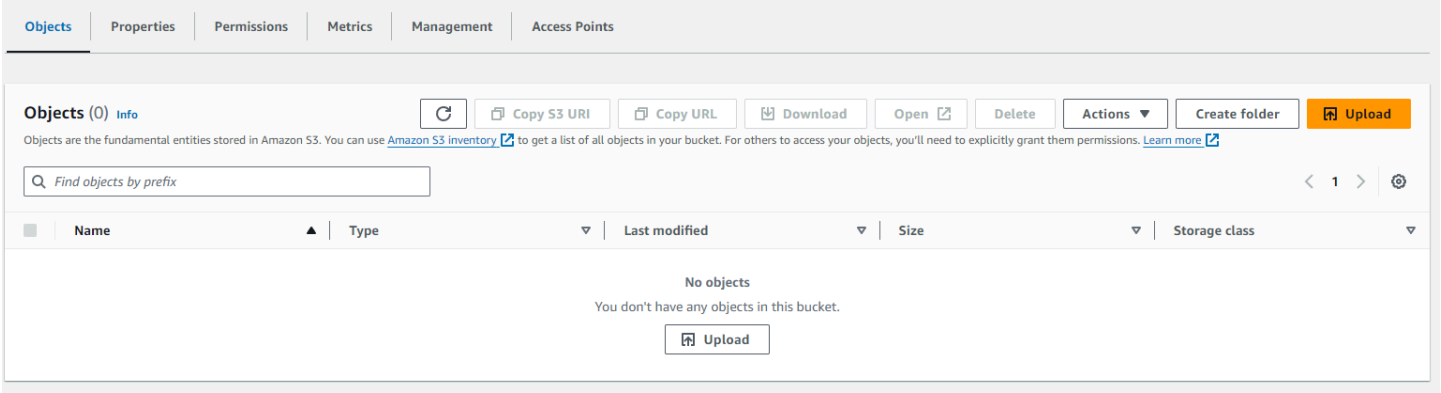
From the bucket's dashboard, select the "Properties" tab.

Scroll to the bottom of the page to "Static website hosting".

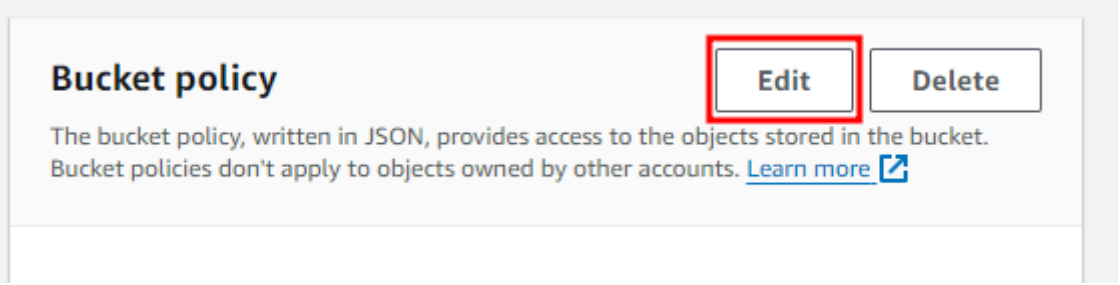


Select "Enable".
Enter the suggested file names in the "Index document" and "Error document" inputs and save.

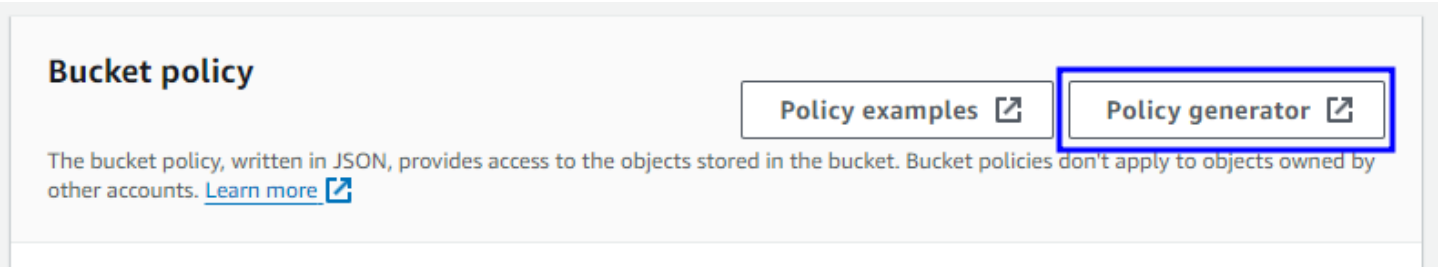
Bucket Policy



From the bucket's dashboard, select the "Permissions" tab.
Scroll down to "Bucket policy".



Select "Edit".



The "Policy generator" option will then be available. Select this to open the Amazon policy generator.

Step 1: Select Policy Type

A Policy is a container for permissions. The different types of policies you can create are an IAM Policy, an S3 Bucket Policy, an SNS Queue Policy.

Select Type of Policy S3 Bucket Policy

Step 2: Add Statement(s)

A statement is the formal description of a single permission. See a description of elements that you can use in statements.

Effect Allow Deny

Principal *

Use a comma to separate multiple values.

AWS Service Amazon S3 All Services ('*')

Use multiple statements to add permissions for more than one service.

Actions 1 Action(s) Selected All Actions ('*')

Amazon Resource Name (ARN) arn:aws:s3:::bucketname

ARN should follow the following format: arn:aws:s3:::${BucketName}/${KeyName}.
Use a comma to separate multiple values.

- From the "Select Type of Policy" dropdown (red) select "S3 Bucket Policy".
- In the "Principal" input (blue), enter an asterisk "*" to allow all.
- From the "Actions" dropdown menu (purple) select "GetObject".
- In the "Amazon Resource Name" input (green), copy the ARN for the bucket (Can be found in the "Bucket overview" under the "Properties" tab of the bucket)
- Select "Generate policy"

Policy JSON Document

Click below to edit. To save the policy, copy the text below to a text editor.
Changes made below will **not be reflected in the policy generator tool.**

```
{
  "Id": "Policy1714743345352",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1714743341352",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::bucketname",
      "Principal": "*"
    }
  ]
}
```

This AWS Policy Generator is provided for informational purposes only, you are still responsible for your use of Amazon Web Services technologies and ensuring

Close

Copy the generated policy and return to the "Bucket policy" in the "Permissions" tab of the bucket.

Policy

```
1  {
2    "Id": "Policy1714743345352",
3    "Version": "2012-10-17",
4    "Statement": [
5      {
6        "Sid": "Stmt1714743341352",
7        "Action": [
8          "s3:GetObject"
9        ],
10       "Effect": "Allow",
11       "Resource": "arn:aws:s3:::bucketname/*",
12       "Principal": "*"
13     }
14   ]
15 }
```

Edit statement

Select a statement

Select an existing statement in the policy or add a new statement.

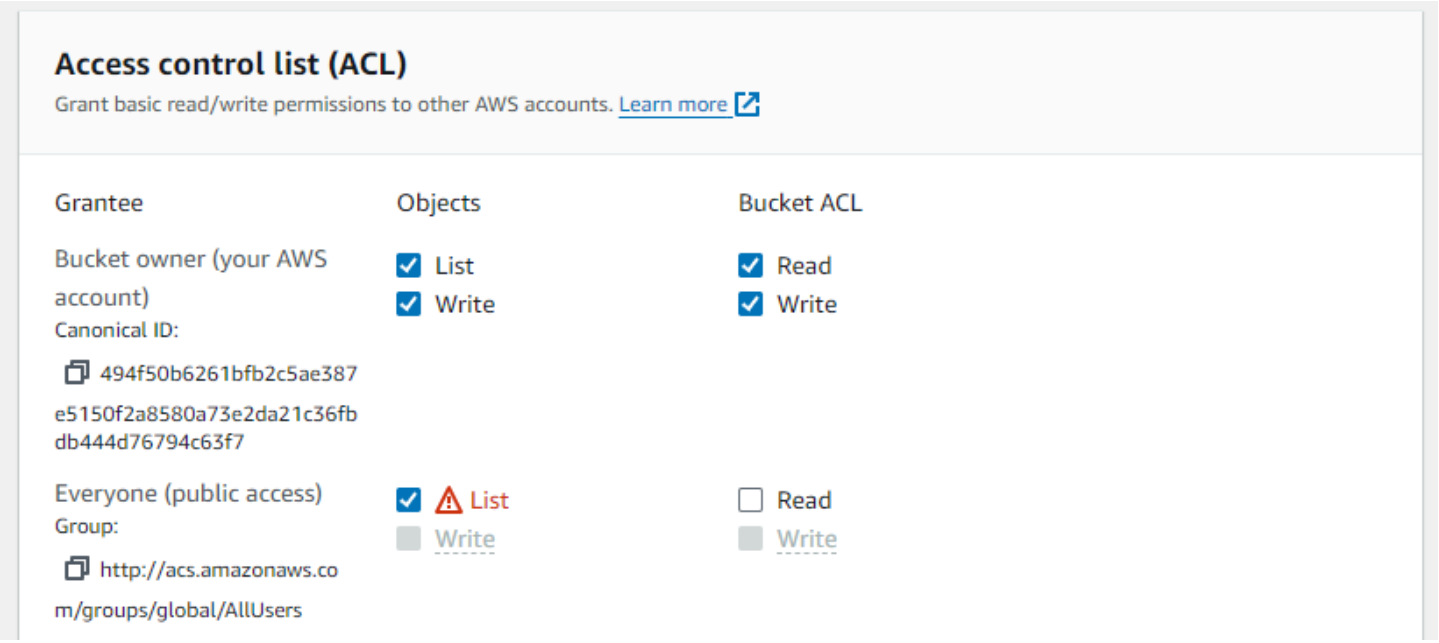
+ Add new statement

Paste the policy into the field and add “/*” to the end of the “Resource” to allow access to all resources.

Access Control List

Still within the “Properties” tab of the bucket.

Scroll down to “Access control list” and select “Edit”.



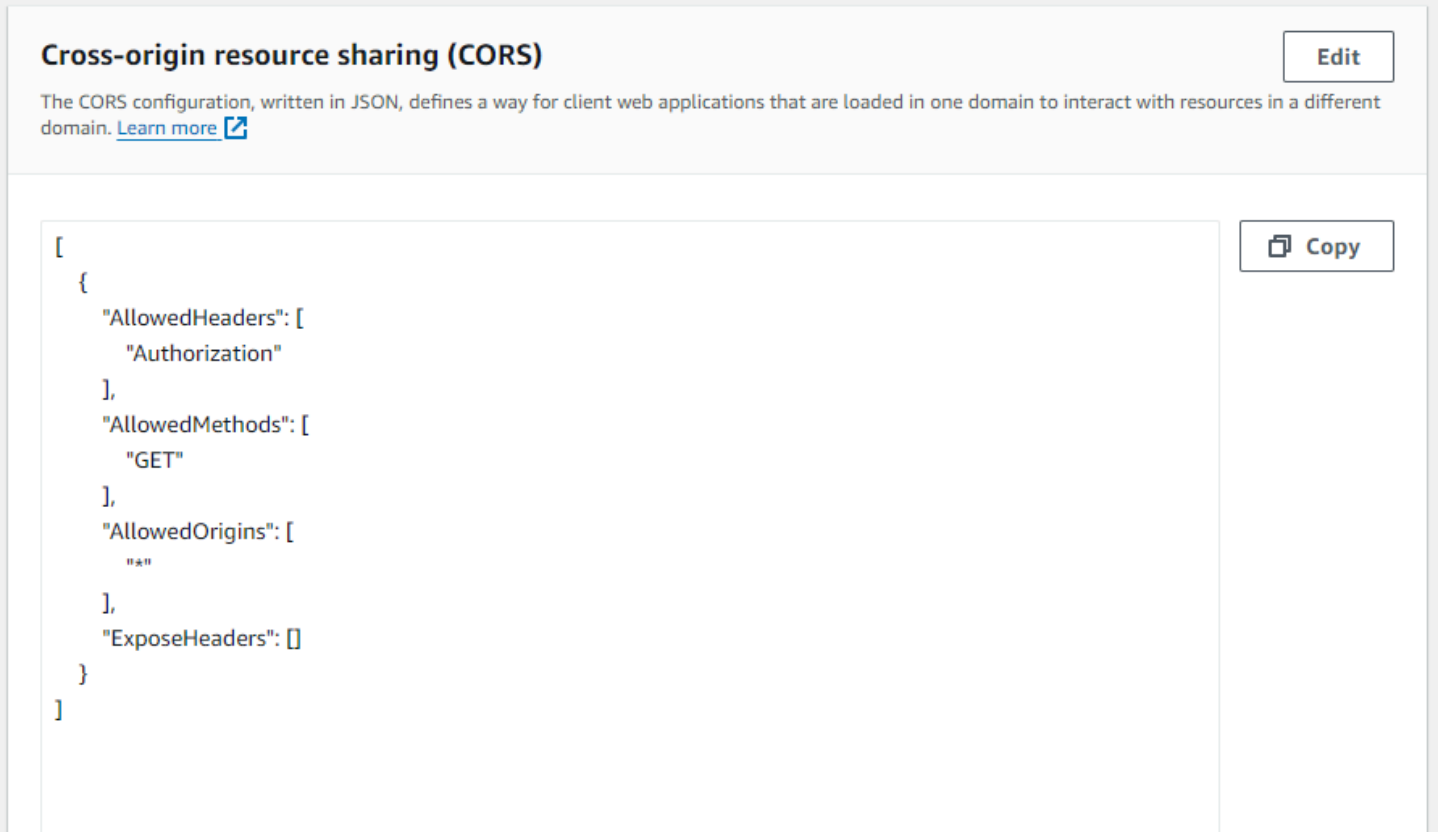
Select “List” for “Everyone” to allow public access.

Accept the warning and save.

Bucket CORS settings

Still within the “Properties” tab of the bucket.
Scroll to the bottom of the bucket properties.

“Edit” the CORS settings to match below:



Save the changes and the bucket is set up.

Next, create a group and user who will have access to the bucket.