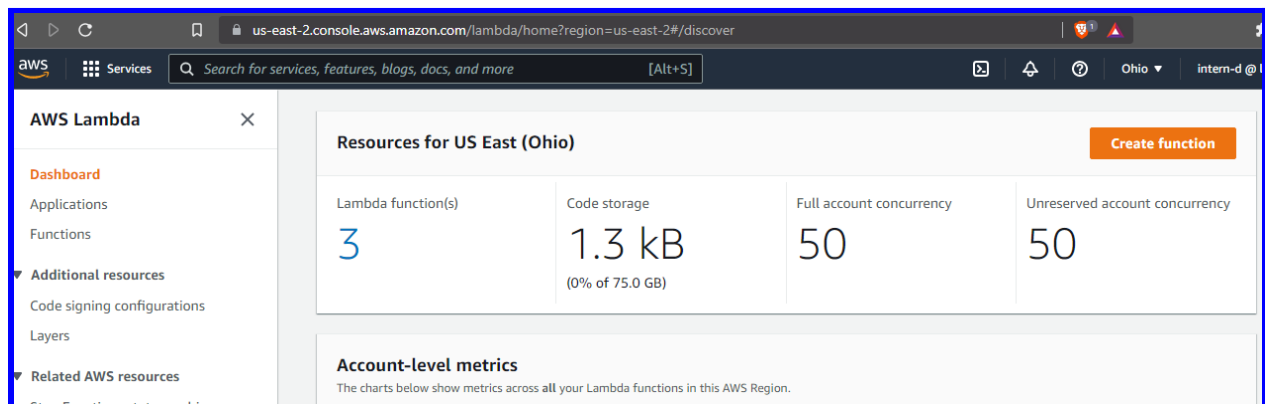


Create a Lambda function that is triggered by an object being uploaded to an S3 bucket.

If the object's name starts with make_public, ensure that the object is publicly accessible.

Creating Lambda Function for S3

Home Page of Lambda



Creating Lambda Function - bibek-LF-S3

The screenshot shows the 'Create function' wizard in the AWS Lambda console. The breadcrumb trail is 'Lambda > Functions > Create function'. The title is 'Create function' with an 'Info' link. Below the title, it says 'Choose one of the following options to create your function.' There are three options:

- Author from scratch** (selected): Start with a simple Hello World example.
- Use a blueprint**: Build a Lambda application from sample code and configuration presets for common use cases.
- Container image**: Select a container image to deploy for your function.

Below the options is the 'Basic information' section. It contains a 'Function name' field with the value 'bibek-LF-S3' and a note: 'Enter a name that describes the purpose of your function. Use only letters, numbers, hyphens, or underscores with no spaces.' Below this is a 'Runtime' dropdown menu with the value 'Python 3.9' and a note: 'Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.'

Using Default Options

Runtime [Info](#)
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.

Python 3.9 ▼

Architecture [Info](#)
Choose the instruction set architecture you want for your function code.

☒ x86_64

☐ arm64

Permissions [Info](#)
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

► **Change default execution role**

► **Advanced settings**

Successfully Created

aws

Services

Search for services, features, blogs, docs, and more


[Alt+S]


☑ Successfully created the function **bibek-LF-S3**. You can now change its code and configuration. To invoke your function with a t

Lambda > Functions > bibek-LF-S3

bibek-LF-S3

▼ **Function overview** [Info](#)

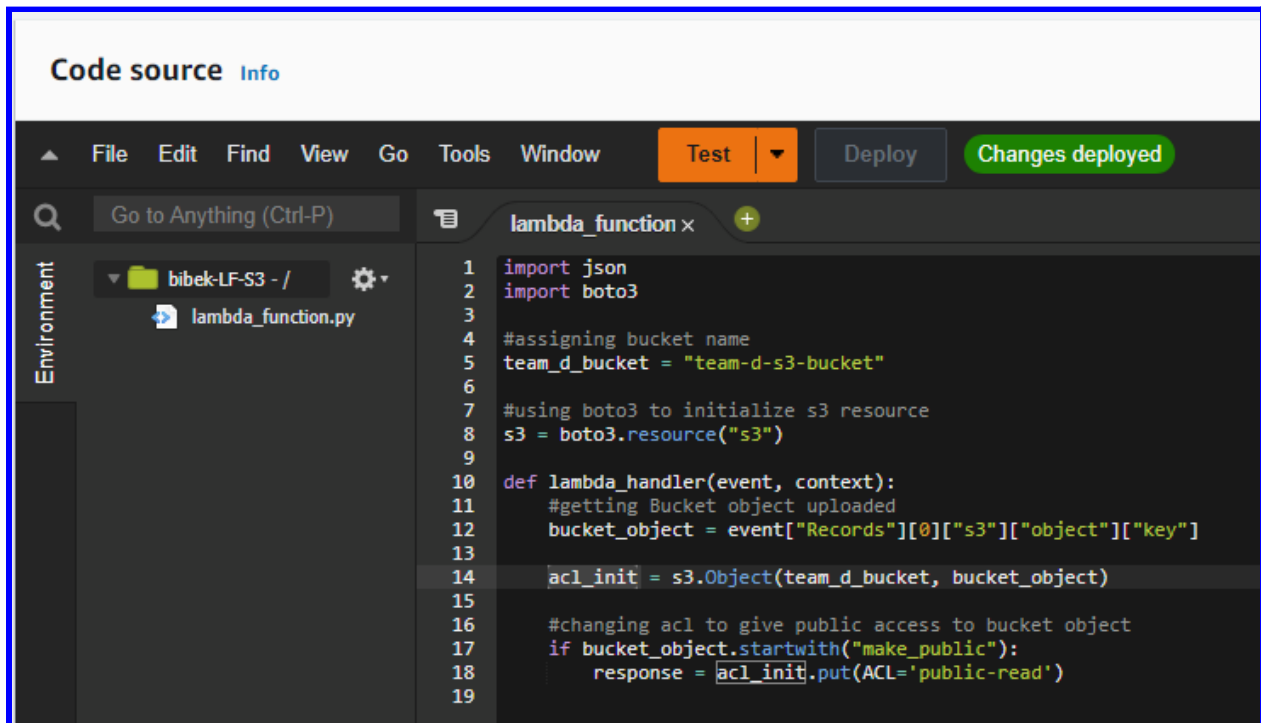
 bibek-LF-S3

 Layers (0)

+ Add trigger

+ Add destination

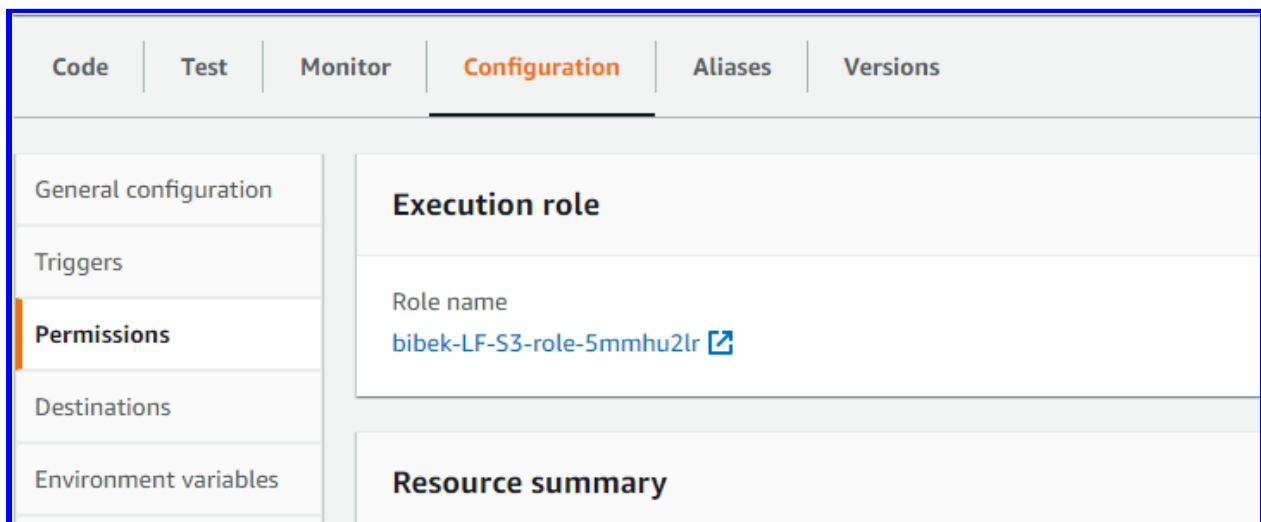
Writing the python code for using “S3” resource and changing ACL to “Public-read” if object name starts with “make_public”



The screenshot shows the AWS Lambda console's code editor for a function named `lambda_function`. The code is written in Python and uses the `boto3` library to interact with Amazon S3. It defines a `lambda_handler` function that takes an event and context as input. The handler extracts the object key from the event and creates an `Object` instance. It then checks if the object key starts with `"make_public"`. If it does, it calls `put` on the `Object` instance to set the ACL to `"public-read"`.

```
1 import json
2 import boto3
3
4 #assigning bucket name
5 team_d_bucket = "team-d-s3-bucket"
6
7 #using boto3 to initialize s3 resource
8 s3 = boto3.resource("s3")
9
10 def lambda_handler(event, context):
11     #getting Bucket object uploaded
12     bucket_object = event["Records"][0]["s3"]["object"]["key"]
13
14     acl_init = s3.Object(team_d_bucket, bucket_object)
15
16     #changing acl to give public access to bucket object
17     if bucket_object.startswith("make_public"):
18         response = acl_init.put(ACL='public-read')
19
```

Granting permission of S3 to execution role of Lambda Function





Clicking on the Execution role and Attaching S3 Access rules

Permissions Trust relationships Tags Access Advisor Revoke sessions

▼ Permissions policies (3 policies applied)

Attach policies


Policy name ▼	Policy type ▼
▶ AWSLambdaBasicExecutionRole-6941f7ab-fc80-4a62-83b7-6334422a431f	Managed policy
▶  AmazonS3FullAccess	AWS managed policy
▶  AmazonS3ReadOnlyAccess	AWS managed policy


Adding trigger for S3 object upload

Lambda > Functions > bibek-LF-S3

bibek-LF-S3

▼ Function overview Info

 bibek-LF-S3

 Layers (0)

+ Add trigger

+ Add destination

Adding Trigger for PUT action in team-d-s3-bucket

Add trigger

Trigger configuration

 S3
aws storage

Bucket

Please select the S3 bucket that serves as the event source. The bucket must be in the same region as the function.

team-d-s3-bucket

Event type

Select the events that you want to have trigger the Lambda function. You can optionally set up a prefix or suffix for an event. However, for each bucket, individual events cannot have multiple configurations with overlapping prefixes or suffixes that could match the same object key.

PUT

Prefix - optional

Enter a single optional prefix to limit the notifications to objects with keys that start with matching characters.

make_public

Lambda > Functions > bibek-LF-S3

bibek-LF-S3

✓ The trigger team-d-s3-bucket was successfully added to function bibek-LF-S3. The function is now receiving events

▼ Function overview [Info](#)



bibek-LF-S3



Layers

(0)



S3

+ Add destination

+ Add trigger

Making some changes in script to give `make_public` suffix

vi `pg-dump-S3.sh`

```
#!/bin/bash

#assigning a name to sql file with date (Month,day,Hour,Minute and seconds)

read -p "Do you want to grant Public access in S3 to Dump File ? y/n: " value

if [ $value = "y" ]; then
    dump=make_public_${ date +%m%d_%H%M%S" )
else
    dump=pgdump_EC2_${ date +%m%d_%H%M%S" )
fi

#taking dump file of lf technology
pg_dump -U postgres -d lf_technology >> /home/ec2-user/pg-backups/dumpfiles/$dump.sql

#uploading to aws s3 team-d-s3-bucket
aws s3 cp /home/ec2-user/pg-backups/dumpfiles/$dump.sql s3://team-d-s3-bucket/$dump.sql --profile lft-training
```

Save & exit

In this script if we want to make the sql file public it will start with suffix `make_public`

Else it will start with `pgdump_EC2_` suffix

Changing Block All Public Access

So that objects can be public too through ACL rules

Block public access (bucket settings)

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 all your S3 buckets and 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 your buckets 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.

Cancel

Save changes

Editing Object Ownership - ACLs Enabled

So that lambda function can change the ACL rules for the object uploaded in the S3

Amazon S3 > team-d-s3-bucket > Edit Object Ownership


Edit Object Ownership [Info](#)

Object Ownership

Control ownership of objects written to this bucket from other AWS accounts and granted using 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.

**Enabling ACLs turns off the bucket owner enforced setting for Object Ownership**

Once the bucket owner enforced setting is turned off, access control lists (ACLs) and their associated permissions are restored. Access to objects that you do not own will be based on ACLs and not the bucket policy.

☒ I acknowledge that ACLs will be restored.

**Enabling ACLs turns off the bucket owner enforced setting for Object Ownership**



Once the bucket owner enforced setting is turned off, access control lists (ACLs) and their associated permissions are restored. Access to objects that you do not own will be based on ACLs and not the bucket policy.

☒ I acknowledge that ACLs will be restored.

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.

 If you want to enforce object ownership for new objects only, your bucket policy must specify that the bucket-owner-full-control canned ACL is required for object uploads. [Learn more](#) 

Cancel

Save changes

Now running the Script

Accepting to make the dump file public - "y"

```
[ec2-user@ip-10-15-32-111 pg-backups]$ ./pg-dump-S3.sh
Do you want to grant Public access in S3 to Dump File ? y/n: y
upload: dumpfiles/make_public_1210_1559_13.sql to s3://team-d-s3-bucket/make_public_1210_1559_13.sql
[ec2-user@ip-10-15-32-111 pg-backups]$ cd dumpfiles/
[ec2-user@ip-10-15-32-111 dumpfiles]$ ls
make_public_1210_1559_13.sql  pgdump_EC2_1210_1017_55.sql  pgdump_EC2_1210_1130_01.sql  pgdump_EC2_1210_1136_01.sql
[ec2-user@ip-10-15-32-111 dumpfiles]$ cd
```

*We can see we have dump files in the name of **make_public** now*

The make_public files are uploaded to the S3 through script

team-d-s3-bucket [Info](#)

Objects | Properties | Permissions | Metrics | Management | Access Points

Objects (5)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 Inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

[Refresh](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#) [Delete](#) [Actions](#) [Create folder](#) [Upload](#)

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	Dockerfile	-	December 7, 2021, 17:45:40 (UTC+05:45)	151.0 B	Standard
<input type="checkbox"/>	make_public_1210_1550_59.sql	sql	December 10, 2021, 21:47:19 (UTC+05:45)	0 B	Standard
<input type="checkbox"/>	make_public_1210_1559_13.sql	sql	December 10, 2021, 21:47:19 (UTC+05:45)	0 B	Standard
<input type="checkbox"/>	pgdump_EC2_1210_1017_55.sql	sql	December 10, 2021, 16:02:56 (UTC+05:45)	2.3 KB	Standard
<input type="checkbox"/>	pgdump_EC2_1210_1136_01.sql	sql	December 10, 2021, 17:21:02 (UTC+05:45)	2.3 KB	Standard

And the permission of make_public files are public read accessible

make_public_1210_1559_13.sql [Info](#) [Copy S3 URI](#) [Download](#) [Open](#) [Object actions](#)

Properties | **Permissions** | Versions

Access control list (ACL) [Learn more](#) [Edit](#)

Grant basic read/write permissions to AWS accounts. [Learn more](#)

Grantee	Object	Object ACL
Object owner (your AWS account) Canonical ID: 8de29480ffa4ee98322f982d8529714aef0ac3dbbaed902de72083a1b4e2837b	Read	Read, Write
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	Read	-
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	-	-

And if we select “n”, i.e. not to make public while running script

```
[ec2-user@ip-10-15-32-111 pg-backups]$ ./pg-dump-S3.sh
Do you want to grant Public access in S3 to Dump File ? y/n: n
upload: pgdump_EC2_1210_1604_55.sql to s3://team-d-s3-bucket/pgdump_EC2_1210_1604_55.sql
[ec2-user@ip-10-15-32-111 pg-backups]$ cd dumpfiles/
[ec2-user@ip-10-15-32-111 dumpfiles]$ ls
-bash: ls: command not found
[ec2-user@ip-10-15-32-111 dumpfiles]$ ls
make_public_1210_1559_13.sql  pgdump_EC2_1210_1017_55.sql  pgdump_EC2_1210_1130_01.sql  pgdump_EC2_1210_1136_01.sql  pgdump_EC2_1210_1604_55.sql
[ec2-user@ip-10-15-32-111 dumpfiles]$ cd
```

It had created dump files with suffix pgdump_EC2

In the S3 bucket, pgdump_EC2 file is stored

team-d-s3-bucket Info							
Objects Properties Permissions Metrics Management Access Points							
Objects (6) Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more							
<div>Refresh Copy S3 URI Copy URL Download Open Delete Actions Create folder Upload</div> <div><input type="text" value="Find objects by prefix"/></div>							
<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class		
<input type="checkbox"/>	Dockerfile	-	December 7, 2021, 17:45:40 (UTC+05:45)	151.0 B	Standard		
<input type="checkbox"/>	make_public_1210_1550_59.sql	sql	December 10, 2021, 21:51:54 (UTC+05:45)	0 B	Standard		
<input type="checkbox"/>	make_public_1210_1559_13.sql	sql	December 10, 2021, 21:51:54 (UTC+05:45)	0 B	Standard		
<input type="checkbox"/>	pgdump_EC2_1210_1017_55.sql	sql	December 10, 2021, 16:02:56 (UTC+05:45)	2.3 KB	Standard		
<input type="checkbox"/>	pgdump_EC2_1210_1136_01.sql	sql	December 10, 2021, 17:21:02 (UTC+05:45)	2.3 KB	Standard		
<input type="checkbox"/>	pgdump_EC2_1210_1604_55.sql	sql	December 10, 2021, 21:49:56 (UTC+05:45)	2.3 KB	Standard		

And the file has not Public access read ACL

Amazon S3

>

team-d-s3-bucket

>

pgdump_EC2_1210_1604_55.sql

pgdump_EC2_1210_1604_55.sql

Info

Copy S3 URI

Download

Open

Object actions

Properties

Permissions

Versions

Access control list (ACL)

Edit

Grant basic read/write permissions to AWS accounts. Learn more

Grantee	Object	Object ACL
<div>Object owner (your AWS account)</div> <div>Canonical ID: 8de29480ffa4ee98322f982d8529714aef0ac3dbbaed902de72083a1b4e2837b</div>	Read	Read, Write
<div>Everyone (public access)</div> <div>Group: http://acs.amazonaws.com/groups/global/AllUsers</div>	-	-
<div>Authenticated users group (anyone with an AWS account)</div> <div>Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers</div>	-	-

In this way we can make specific file public if we want

Thank you !!