

1. Static website hosting using s3(what is index and error page).

Static website hosting

Endpoint : <http://ayush-public-bucket.s3-website-us-east-1.amazonaws.com>

☒ Use this bucket to host a website [Learn more](#)

Index document [i](#)

Error document [i](#)

Redirection rules (optional) [i](#)

☐ Redirect requests [Learn more](#)

☐ Disable website hosting

☐ Disabled

```
ayush@ayush:~/github/ttnbootcamp-tothenew$ vim index.html
ayush@ayush:~/github/ttnbootcamp-tothenew$ cat index.html
this is for static web hosting
ayush@ayush:~/github/ttnbootcamp-tothenew$ aws s3 cp index.html s3://ayush-public-bucket/
upload: ./index.html to s3://ayush-public-bucket/index.html
ayush@ayush:~/github/ttnbootcamp-tothenew$ aws s3 ls s3://ayush-public-bucket/
2020-03-01 20:24:22          31 index.html
ayush@ayush:~/github/ttnbootcamp-tothenew$
```

[Open](#)[Download](#)[Download as](#)[Make public](#)[Copy path](#)**Owner**

nitin.bhadauria

Last modified

Mar 1, 2020 8:24:22 PM GMT+0530

Etag

7f07553f59f731c38081a15222ed3d50

Storage class

Standard

Server-side encryption

None

Size

31.0 B

Key

index.html

Object URL

<https://ayush-public-bucket.s3.amazonaws.com/index.html>

← → ↻ 🔒 ayush-public-bucket.s3.amazonaws.com/index.html

🌐 Apps 🌐 Apple Final Cu... 🌐 Apple Final Cu... 📁 On Demand

this is for static web hosting

2. Create an assume role to access s3 using ec2.

Create a role that supports sts assume role

Attach policies

Policy name ▼

▼ ayush-s3-assume-role

Policy summary {} JSON Edit policy

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "s3:*",
8         "iam:ListRoles",
9         "sts:AssumeRole"
10      ],
11       "Resource": "*"
12     }
13   ]
14 }
```

Attach this role to first ec2 instance

Generate sts credentials for second instance

```
ubuntu@ip-172-31-71-45:~$ aws iam list-roles --query "Roles[?RoleName == 'ec2-s3-inatance'].[RoleName, Arn]"
[
  [
    "ec2-s3-inatance",
    "arn:aws:iam::187632318301:role/ec2-s3-inatance"
  ]
]
ubuntu@ip-172-31-71-45:~$ aws sts assume-role --role-arn "arn:aws:iam::123456789012:role/example-role" --role-session-name AWSCLI-Session

An error occurred (AccessDenied) when calling the AssumeRole operation: User: arn:aws:sts::187632318301:assumed-role/ec2-s3-inatance/i-0a5bb40810a8795d0 is not authorized to perform: sts:AssumeRole on resource: arn:aws:iam::123456789012:role/example-role
ubuntu@ip-172-31-71-45:~$ aws sts assume-role --role-arn "" --role-session-name AWS-assume-role
{
  "Credentials": {
    "AccessKeyId": "ASIASXL6B650V7WHOIE6",
    "SecretAccessKey": "N3cagQtTzr73TdiyTXylPPfW7elVcyX1QAY1nN36",
    "SessionToken": "FwoGZXIvYXZlEFaDLS/Z6lDiXqvnepnWSKzAZ5yEpcp6V4KG2aDY0aZ9Qv90bcp8RPC1+/K7a9qSnQTKJE4/mung4FzbU5hWSjyN0zL62hLLQXgn50ZNAZaj80ZW4jJKI2hql95Rmmn5zXw0Ig0g9hkzVkj1FJ32ikQ1eyRPZmGT/twa00lLMmkgjkE614p9kEYfp0kF58iF4oAdoY99IcgJMshVb3MUh5IKEHqjHwLI3y02xX0qqNJJaGD0JFkdjf2RhJGN2Le806F9jDmxKKG57/IFMi3e/ZgtI9xH5r7uCsZC1WWwqNgSxAijKPQYam+oP9l0bQiUmxqmZr5LLh/fHRY=",
    "Expiration": "2020-03-01T17:02:41Z"
  },
  "AssumedRoleUser": {
    "AssumedRoleId": "AROASXL6B650VCX7E5PAX:AWS-assume-role",
    "Arn": "arn:aws:sts::187632318301:assumed-role/ec2-s3-inatance/AWS-assume-role"
  }
}
ubuntu@ip-172-31-71-45:~$
```

Add those credentials to this instance and get access to s3

```
ubuntu@ip-10-0-2-177:~$ aws s3 ls s3://ayush-s3/

An error occurred (AccessDenied) when calling the ListObjects operation: Access Denied
ubuntu@ip-10-0-2-177:~$ export AWS_ACCESS_KEY_ID=ASIASXL6B650V7WHOIE6
ubuntu@ip-10-0-2-177:~$ export AWS_SECRET_ACCESS_KEY=N3cagQtTzr73TdiyTXylPPfW7elVcyX1QAY1nN36
ubuntu@ip-10-0-2-177:~$ export AWS_SESSION_TOKEN=FwoGZXIvYXZlEFaDLS/Z6lDiXqvnepnWSKzAZ5yEpcp6V4KG2aDY0aZ9Qv90bcp8RPC1+/K7a9qSnQTKJE4/mung4FzbU5hWSjyN0zL62hLLQXgn50ZNAZaj80ZW4jJKI2hql95Rmmn5zXw0Ig0g9hkzVkj1FJ32ikQ1eyRPZmGT/twa00lLMmkgjkE614p9kEYfp0kF58iF4oAdoY99IcgJMshVb3MUh5IKEHqjHwLI3y02xX0qqNJJaGD0JFkdjf2RhJGN2Le806F9jDmxKKG57/IFMi3e/ZgtI9xH5r7uCsZC1WWwqNgSxAijKPQYam+oP9l0bQiUmxqmZr5LLh/fHRY=
ubuntu@ip-10-0-2-177:~$ aws sts get-caller-identity
{
  "UserId": "AROASXL6B650VCX7E5PAX:AWS-assume-role",
  "Account": "187632318301",
  "Arn": "arn:aws:sts::187632318301:assumed-role/ec2-s3-inatance/AWS-assume-role"
}
ubuntu@ip-10-0-2-177:~$ aws s3 ls s3://ayush-s3/
2020-03-01 12:47:58      1089 Mongo-DB.txt
2020-03-01 12:47:22      1675 codecommit
2020-03-01 12:47:37      4606 sample.war
ubuntu@ip-10-0-2-177:~$
```

3. Block s3 access on the basis of

i. IP

Block public access

Access Control List

Bucket Policy

CORS configuration

Bucket policy editor ARN: arn:aws:s3:::ayush-public-bucket

Type to add a new policy or edit an existing policy in the text area below.

```
1 {
2   "Version": "2012-10-17",
3   "Id": "S3PolicyId1",
4   "Statement": [
5     {
6       "Sid": "IPAllow",
7       "Effect": "Allow",
8       "Principal": "*",
9       "Action": "s3:*",
10      "Resource": "arn:aws:s3:::ayush-public-bucket/*",
11      "Condition": {
12        "IpAddress": {
13          "aws:SourceIp": "10.0.1.0/24"
14        }
15      }
16    }
17  ]
18 }
```

ii. Domain

Block public access

Access Control List

Bucket Policy

CORS configuration

CORS configuration editor ARN: arn:aws:s3:::ayush-public-bucket

Add a new cors configuration or edit an existing one in the text area below.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
3   <CORSRule>
4     <AllowedOrigin>https://example.com</AllowedOrigin>
5     <AllowedOrigin>https://www.example.com</AllowedOrigin>
6     <AllowedMethod>GET</AllowedMethod>
7     <AllowedMethod>POST</AllowedMethod>
8     <AllowedMethod>PUT</AllowedMethod>
9     <MaxAgeSeconds>3000</MaxAgeSeconds>
10    <AllowedHeader>Authorization</AllowedHeader>
11  </CORSRule>
12 </CORSConfiguration>
```


iii. Pre-signed URL(Time based)

Query string authentication and URL-based access are hidden gems in Amazon S3. These methods allow you to grant permissions based on a specific URL. There are two common patterns for using this type of authentication

- Allowing someone or something to upload a key to your bucket
- Providing temporary access to a specific key

This is a great method of securing providing one-time access to your Amazon S3 buckets.

Block public access

Access Control List

Bucket Policy

Bucket policy editor ARN: arn:aws:s3:::ayush-public-bucket



Type to add a new policy or edit an existing policy in the text area below.

```
1  {
2    "Id": "Policy1583075785735",
3    "Version": "2012-10-17",
4    "Statement": [
5      {
6        "Sid": "DenyPresigned",
7        "Action": [
8          "s3:Get*"
9        ],
10       "Effect": "Deny",
11       "Resource": "arn:aws:s3:::ayush-public-bucket/*",
12       "Condition": {
13         "StringEquals": {
14           "s3:authType": "REST-QUERY-STRING"
15         }
16       },
17       "Principal": "*"
18     }
19   ]
20 }
```

4. Create RDS subnet and launch RDS instance.

Create subnet

Specify your subnet's IP address block in CIDR format; for example, 10.0.0.0/24. IPv4 block sizes must be between a /16 net and a /64 CIDR block.

Name tag	<input type="text" value="ayush-SN-private-1"/>					
VPC*	<input type="text" value="vpc-0bce5df601296bb8a"/>					
Availability Zone	<input type="text" value="us-east-1c"/>					
VPC CIDRs	<table><thead><tr><th>CIDR</th><th>Status</th></tr></thead><tbody><tr><td>10.0.0.0/16</td><td>associated</td></tr></tbody></table>		CIDR	Status	10.0.0.0/16	associated
CIDR	Status					
10.0.0.0/16	associated					
IPv4 CIDR block*	<input type="text" value="10.0.3.0/24"/>					

Choose a database creation method [Info](#)

☒ Standard Create

You set all of the configuration options, including ones for availability, security, backups, and maintenance.

☐ Easy Create

Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type [Info](#)

☐ Amazon Aurora



☒ MySQL



☐ MariaDB



☐ PostgreSQL



☐ Oracle



☐ Microsoft SQL Server



Templates

Choose a sample template to meet your use case.

☐ Production

Use defaults for high availability and fast, consistent performance.

☐ Dev/Test

This instance is intended for development use outside of a production environment.

☒ Free tier

Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.

[Info](#)

Settings

DB instance identifier [Info](#)

Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens (1 to 15 for SQL Server). First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

DB instance identifier

Type a name for your DB instance. The name must be unique cross all DB instances owned by your AWS account in the current AWS Region.

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens (1 to 15 for SQL Server). First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

Master username [Info](#)

Type a login ID for the master user of your DB instance.

1 to 16 alphanumeric characters. First character must be a letter

☐ Auto generate a password

Amazon RDS can generate a password for you, or you can specify your own password

Master password [Info](#)

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), "(double quote) and @ (at sign).

Confirm password [Info](#)

Connectivity

Virtual Private Cloud (VPC)
Info

VPC that defines the virtual networking environment for this DB instance.

ayush-VPC (vpc-0bce5df601296bb8a)

Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change the VPC selection.

Additional connectivity configuration

Subnet group
Info

DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

Create new DB Subnet Group

Publicly accessible
Info

☐ Yes
Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.

☒ No
RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

5. what is parameter group and option group,ACL, Bucket policy, IAM Policy?

Parameter group: You manage your DB engine configuration by associating your DB instances with parameter groups. Amazon RDS defines parameter groups with default settings that apply to newly created DB instances. You can define your own parameter groups with customized settings. Then you can modify your DB instances to use your own parameter groups.

A *DB parameter group* acts as a container for engine configuration values that are applied to one or more DB instances.

Option Group: Some DB engines offer additional features that make it easier to manage data and databases, and to provide additional security for your database. Amazon RDS uses option groups to enable and configure these features. An *option group* can specify features, called options, that are available for a particular Amazon RDS DB instance. Options can have settings that specify how the option works. When you associate a DB instance with an option group, the specified options and option settings are enabled for that DB instance.

ACL: Amazon S3 access control lists (ACLs) enable you to manage access to buckets and objects. Each bucket and object has an ACL attached to it as a subresource. It defines which AWS accounts or groups are granted access and the type of access.

Bucket policy: A bucket policy is a resource-based AWS Identity and Access Management (IAM) policy. You add a bucket policy to a bucket to grant other AWS accounts or IAM users access permissions for the bucket and the objects in it. Object permissions apply only to the objects that the bucket owner creates.

IAM Policy: A policy is an entity that, when attached to an identity or resource, defines their permissions. You can use the AWS Management Console, AWS CLI, or AWS API to create customer managed policies in IAM. Customer managed policies are standalone policies that you administer in your own AWS account.

6. Mount S3 to an EC2 instance.

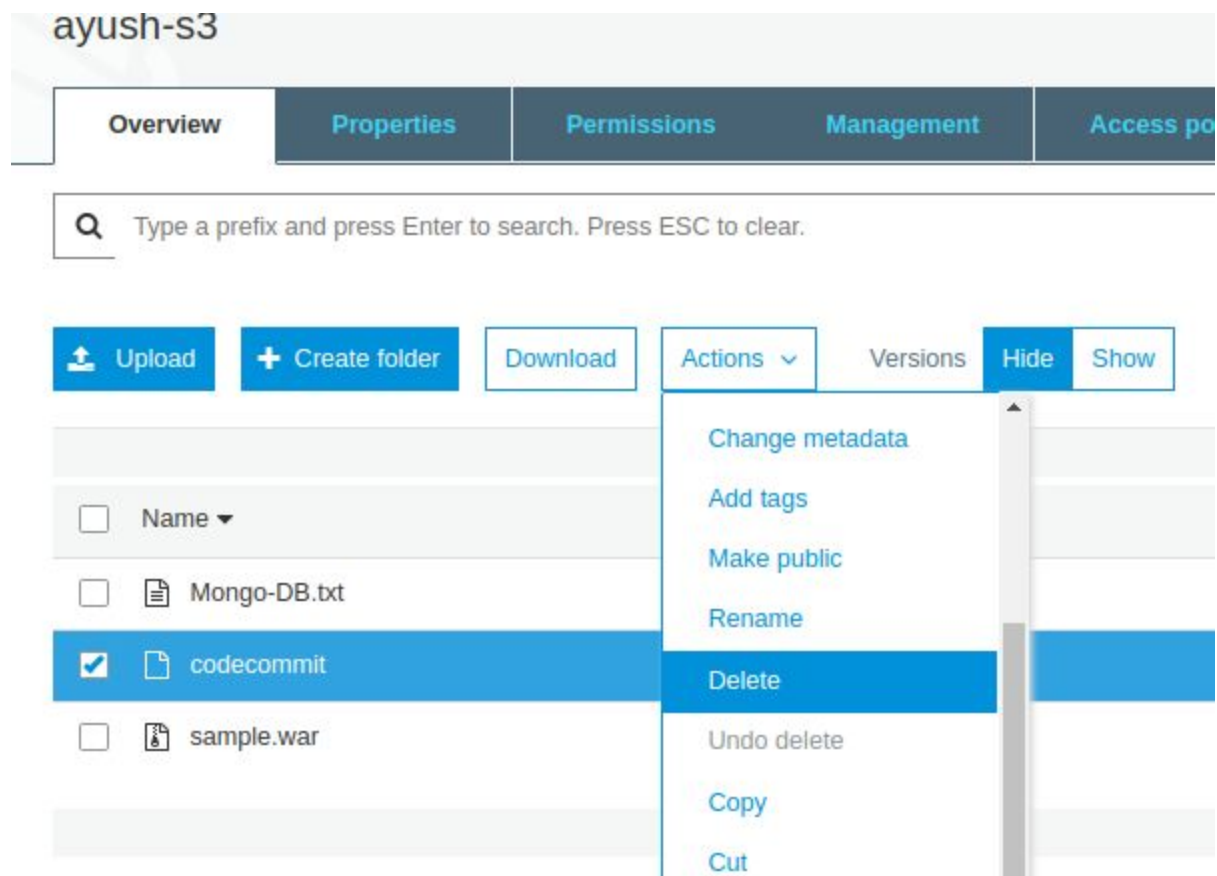
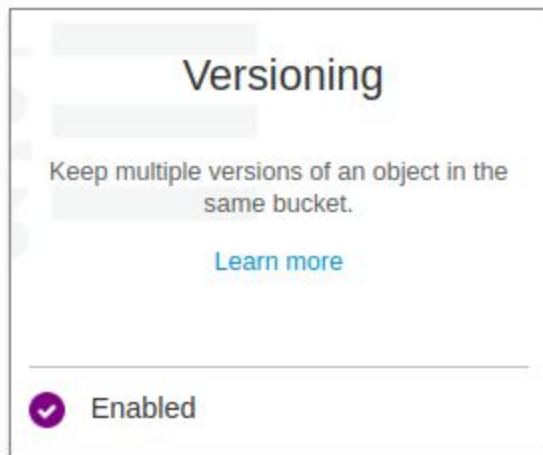
<https://www.tothenew.com/blog/mounting-s3-bucket-into-an-ec2-instance/>

7. Change content type using s3.

```
ayush@ayush:~$ aws s3 cp \  
> s3://ayush-s3/ \  
> s3://ayush-s3/ \  
> --exclude '*' \  
> --include '*.txt' \  
> --no-guess-mime-type \  
> --content-type="text/plain" \  
> --metadata-directive="REPLACE" \  
> --recursive  
copy: s3://ayush-s3/Mongo-DB.txt to s3://ayush-s3/Mongo-DB.txt  
ayush@ayush:~$
```

```
ayush@ayush:~$ aws s3api get-object --bucket ayush-s3 --key Mongo-DB.txt data.txt  
{  
  "AcceptRanges": "bytes",  
  "LastModified": "Sun, 01 Mar 2020 16:44:55 GMT",  
  "ContentLength": 1089,  
  "ETag": "\"d9d87d8114436642a9b84ca0e5f55345\"",  
  "ContentType": "text/plain",  
  "Metadata": {}  
}
```

8. Retrive previous version of S3(enable versioning).



Upload	Create folder	Download	Actions ▾	Versions	Hide	Show
<input type="checkbox"/> Name ▾	Last modified ▾					
<input type="checkbox"/> Mongo-DB.txt	Mar 1, 2020 10:14:55 PM GMT+0530					
<input type="checkbox"/> codecommit	Mar 1, 2020 6:17:22 PM GMT+0530					
<input type="checkbox"/> sample.war	Mar 1, 2020 6:17:37 PM GMT+0530					

9. S3 VPC endpoint.

Create Endpoint

A VPC endpoint allows you to securely connect your VPC to another service.

An interface endpoint is powered by [PrivateLink](#), and uses an elastic network interface (ENI) as an entry point for traffic destined to the service.

A gateway endpoint serves as a target for a route in your route table for traffic destined for the service.

- Service category**
- ☒ AWS services
 - ☐ Find service by name
 - ☐ Your AWS Marketplace services

Service Name com.amazonaws.us-east-1.s3 ⓘ

<input type="text" value="search : s3"/> Add filter		
Service Name	Owner	Type
<input checked="" type="radio"/> com.amazonaws.us-east-1.s3	amazon	Gateway

VPC*

vpc-0bce5df601296bb8a


Configure route tables

A rule with destination **pl-63a5400a (com.amazonaws.us-east-1.s3)** and a target with this endpoints' ID (e.g. vpce-1234567) tables you select below.

Subnets associated with selected route tables will be able to access this endpoint.

rtb-057d656b47af151b9


	Route Table ID	Main	Associated With
<input checked="" type="checkbox"/>	rtb-057d656b47af151b9	Yes	3 subnets

 **Warning**

When you use an endpoint, the source IP addresses from your instances in your affected subnets for accessing the region will be private IP addresses, not public IP addresses. Existing connections from your affected subnets to the public IP addresses may be dropped. Ensure that you don't have critical tasks running when you create or modify

[Endpoints](#) > Create Endpoint

Create Endpoint

 The following VPC Endpoint was created:

VPC Endpoint ID [vpce-0f5f8108aeea5dce0](#)

Close

10. CORS, Enable CORS for 2 specific website.

Cross-origin resource sharing (CORS) defines a way for client web applications that are loaded in one domain to interact with resources in a different domain. With CORS support, you can build rich client-side web applications with Amazon S3 and selectively allow cross-origin access to your Amazon S3 resources.

```
<?xml version="1.0" encoding="UTF-8"?>
<CORSConfiguration
xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <CORSRule>
    <AllowedOrigin>https://website-1.com</AllowedOrigin>
    <AllowedOrigin>https://website-2.com</AllowedOrigin>
    <AllowedMethod>GET</AllowedMethod>
    <AllowedMethod>POST</AllowedMethod>
    <AllowedMethod>PUT</AllowedMethod>
    <MaxAgeSeconds>3000</MaxAgeSeconds>
    <AllowedHeader>Authorization</AllowedHeader>
  </CORSRule>
</CORSConfiguration>
```