TO THE NEW™

Doubt Resolving

Trainee Name : Chhavi Sharma

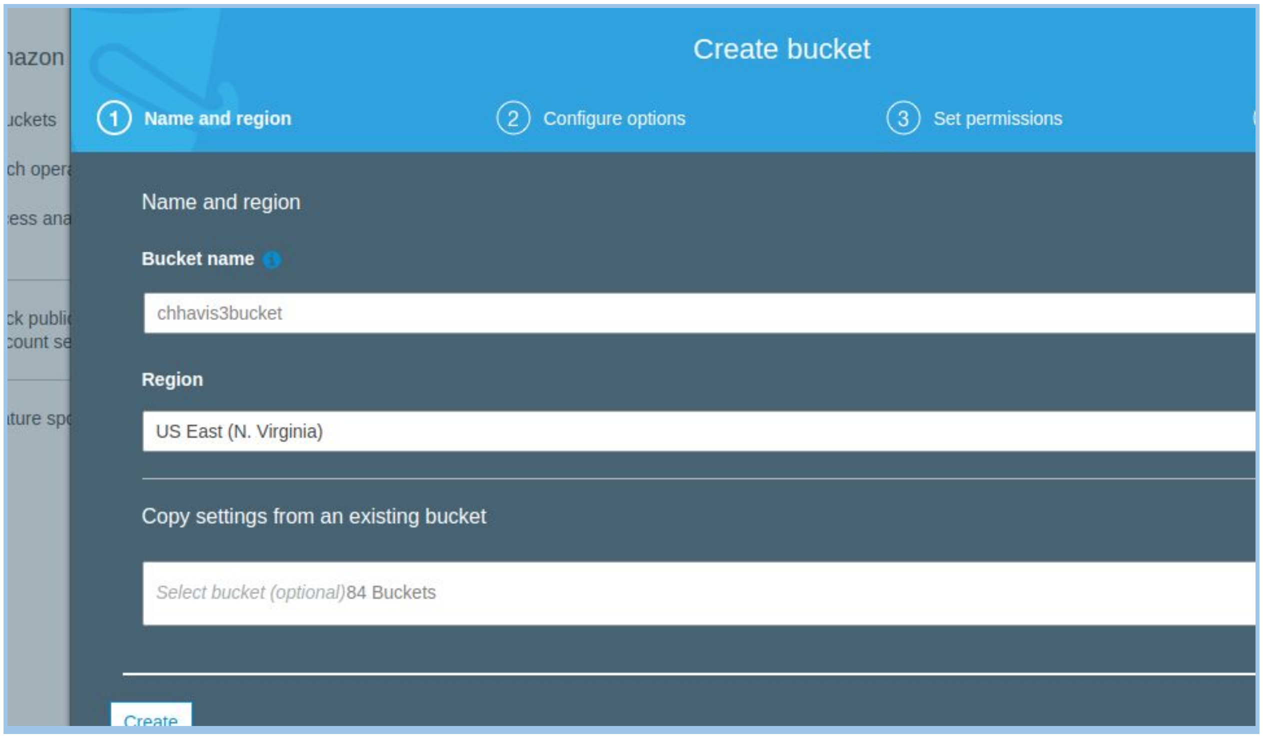
Newers ID : 4023

College : UPES

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

Ans.

Step 1: Create a S3 bucket.



Create bucket

Name and region

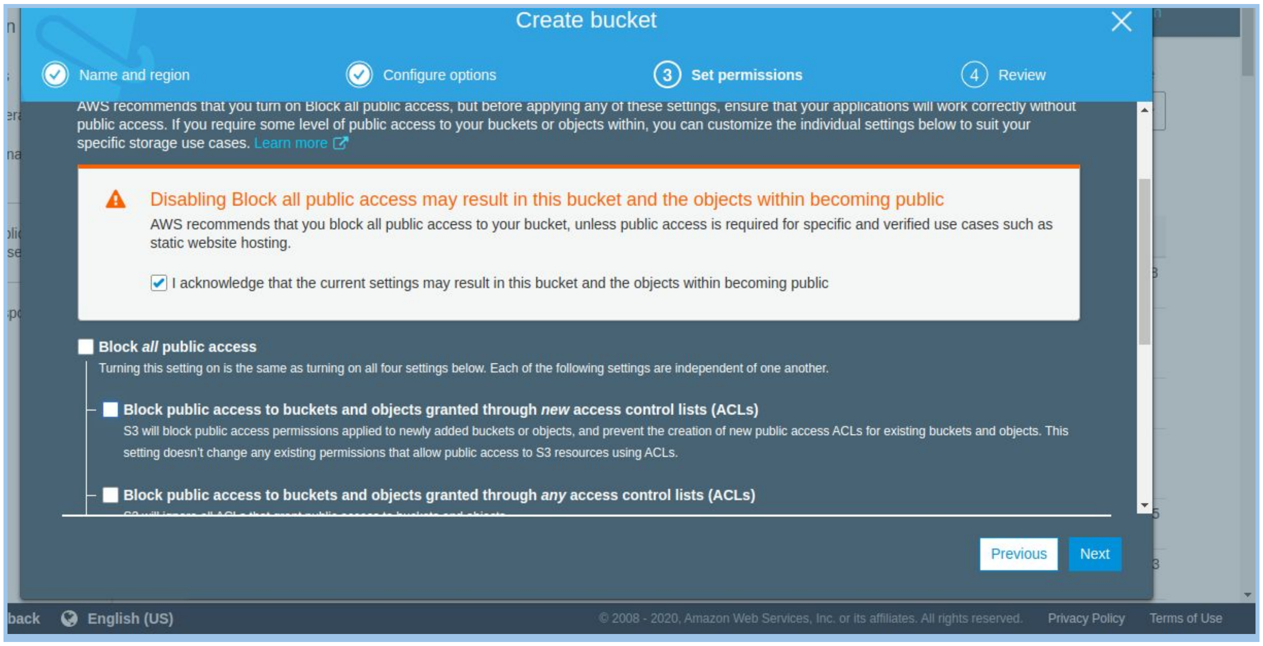
Bucket name: chhavis3bucket

Region: US East (N. Virginia)

Copy settings from an existing bucket

Select bucket (optional)84 Buckets

Step 2: Allow public access.



Create bucket

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

Disabling Block all public access may result in this bucket and the objects within becoming public

AWS recommends that you block all public access to your bucket, unless public access is required for specific and verified use cases such as static website hosting.

☑ I acknowledge that the current settings may result in this bucket and the objects within becoming public

□ **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.

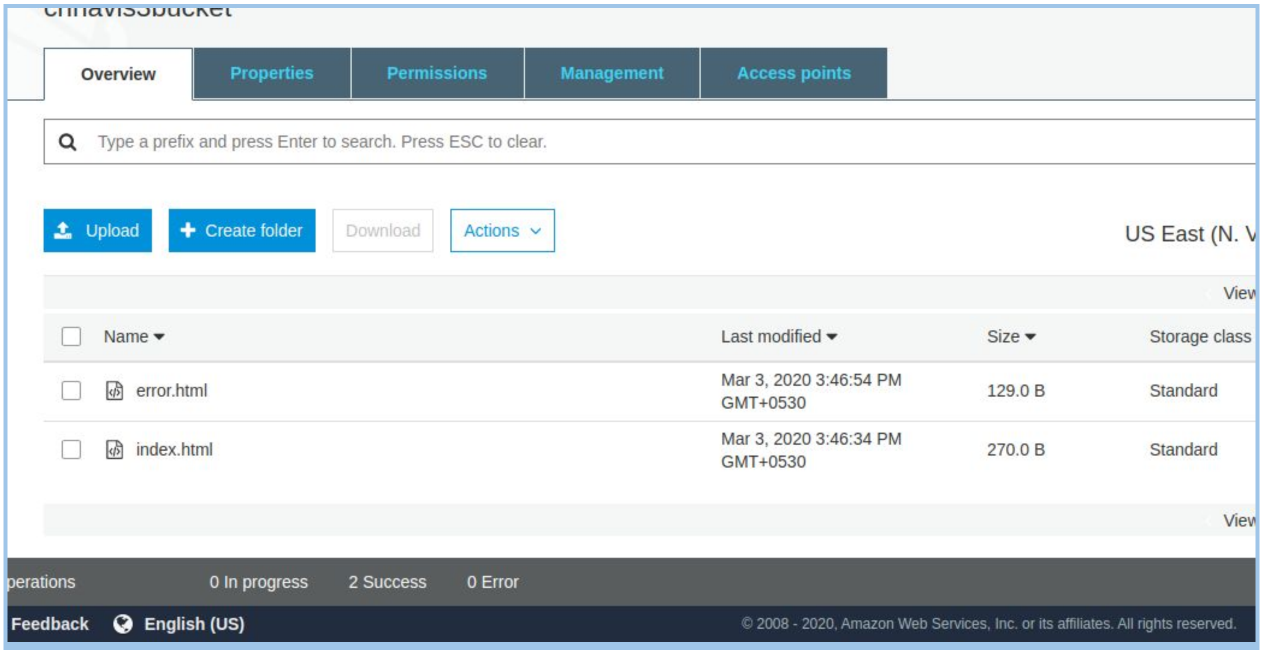
□ **Block public access to buckets and objects granted through *new* access control lists (ACLs)**

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)**

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

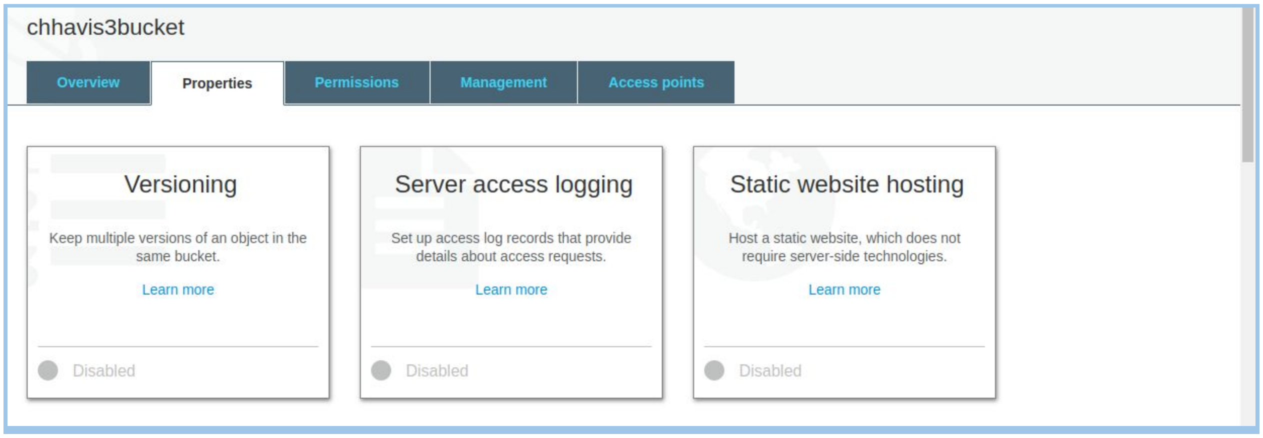
Step 3: Add index.html and error.html



	Name	Last modified	Size	Storage class
□	error.html	Mar 3, 2020 3:46:54 PM GMT+0530	129.0 B	Standard
□	index.html	Mar 3, 2020 3:46:34 PM GMT+0530	270.0 B	Standard

View
perations
0 In progress
2 Success
0 Error
Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Step 4: Go to the properties and select static website hosting.



Versioning

Keep multiple versions of an object in the same bucket.

Learn more

○ Disabled

Server access logging

Set up access log records that provide details about access requests.

Learn more

○ Disabled

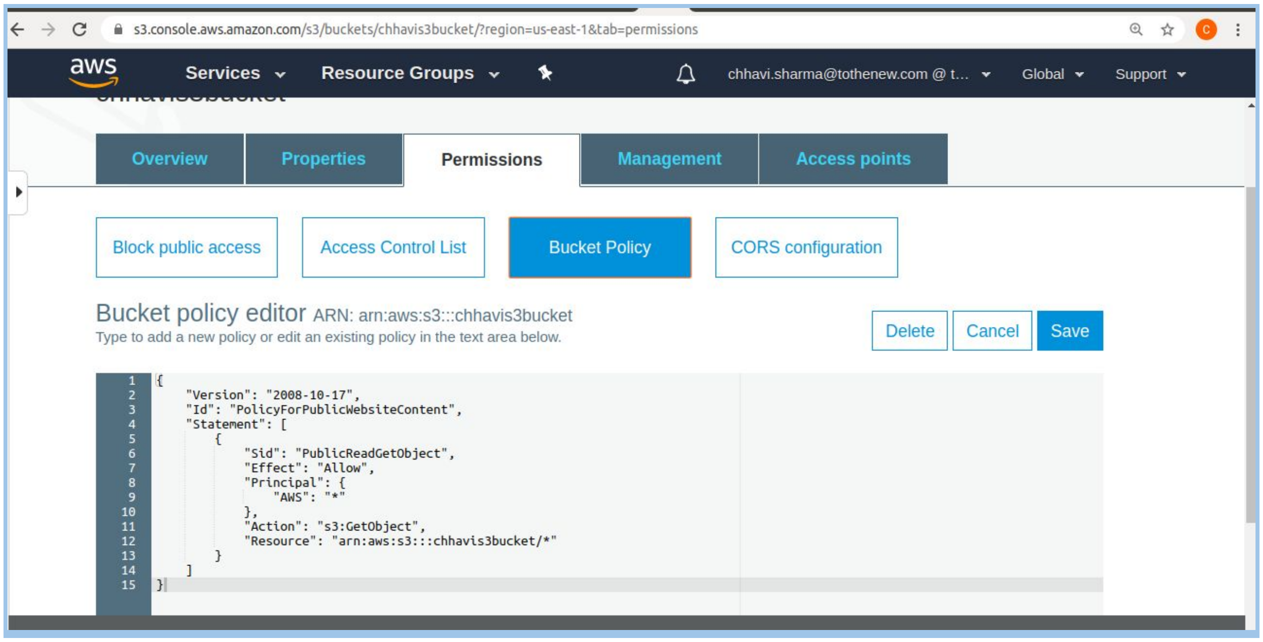
Static website hosting

Host a static website, which does not require server-side technologies.

Learn more

○ Disabled

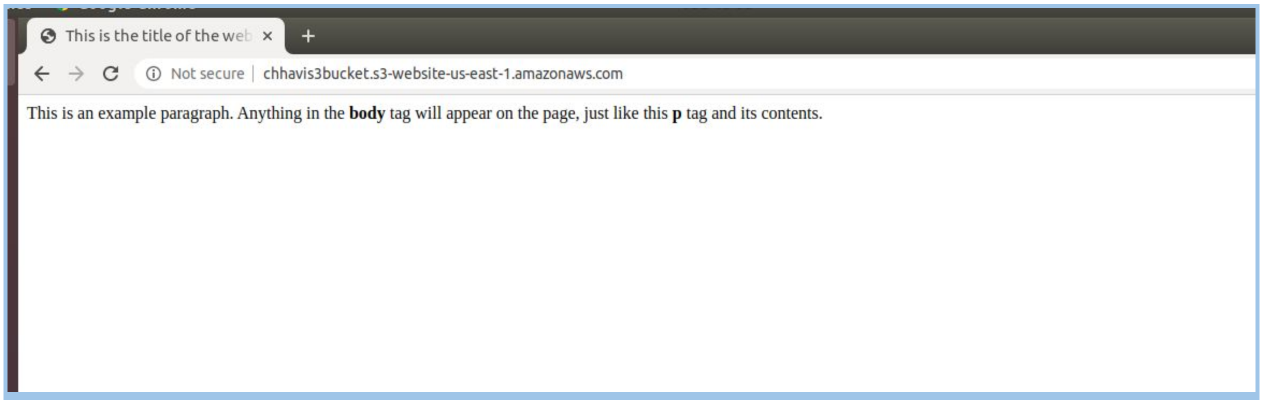
Step 5: Now create a policy and add public read access policy.



Bucket policy editor ARN: arn:aws:s3:::chhavis3bucket
Type to add a new policy or edit an existing policy in the text area below.

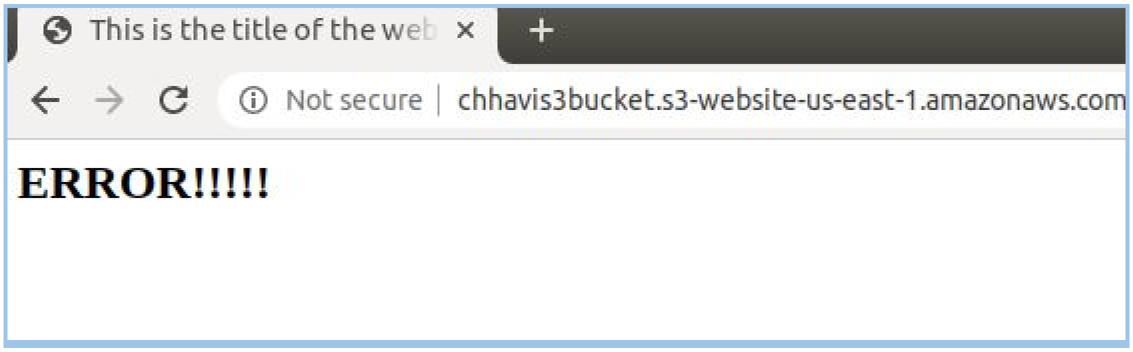
```
{
    "Version": "2008-10-17",
    "Id": "PolicyForPublicWebsiteContent",
    "Statement": [
        {
            "Sid": "PublicReadGetObject",
            "Effect": "Allow",
            "Principal": {
                "AWS": "*"
            },
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::chhavis3bucket/*"
        }
    ]
}
```

Step 6: Now copy the link and paste in the url.
Index.html will open by default.


This is the title of the web
Not secure
chhavis3bucket.s3-website-us-east-1.amazonaws.com

This is an example paragraph. Anything in the **body** tag will appear on the page, just like this **p** tag and its contents.

If you provide a non existing url, then an error page will be displayed.


This is the title of the web
Not secure
chhavis3bucket.s3-website-us-east-1.amazonaws.com

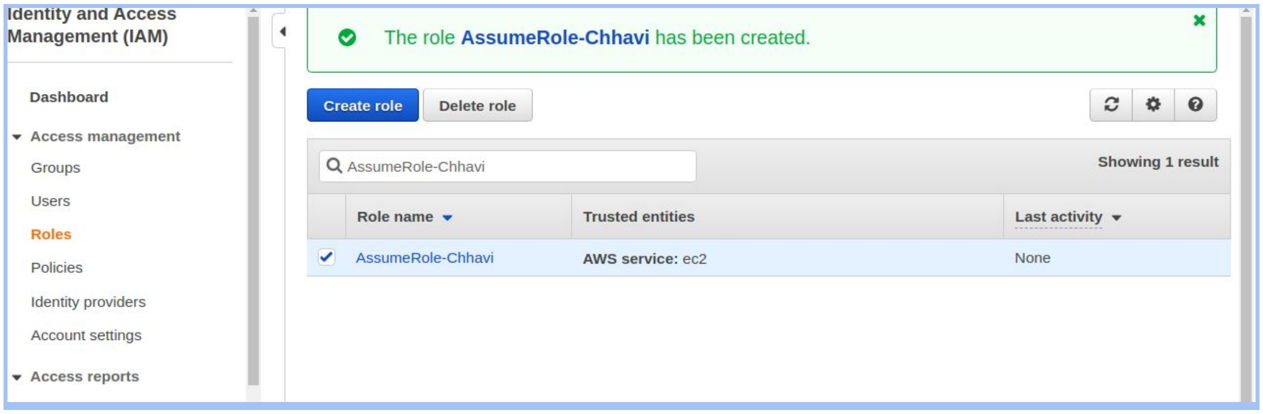
ERROR!!!!!

The index page is the first page that is served .The error.html is the page that is served whenever an error such as a non existing page is requested.

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

Ans.

Step 1: Create a new role.



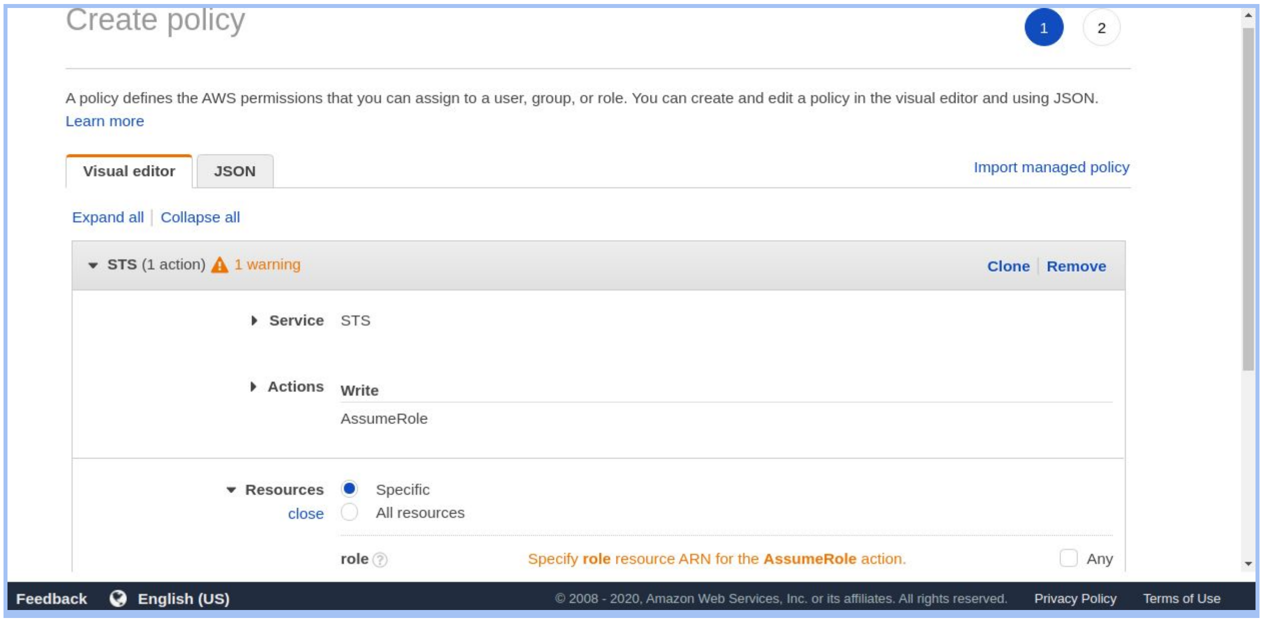
The role **AssumeRole-Chhavi** has been created.

AssumeRole-Chhavi

Showing 1 result

	Role name	Trusted entities	Last activity
☑	AssumeRole-Chhavi	**AWS service:** ec2	None

Step 2: Create a new policy.Select STS service.Select Assume role Action.



Create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. Learn more

STS (1 action) ⚠ 1 warning

Service STS

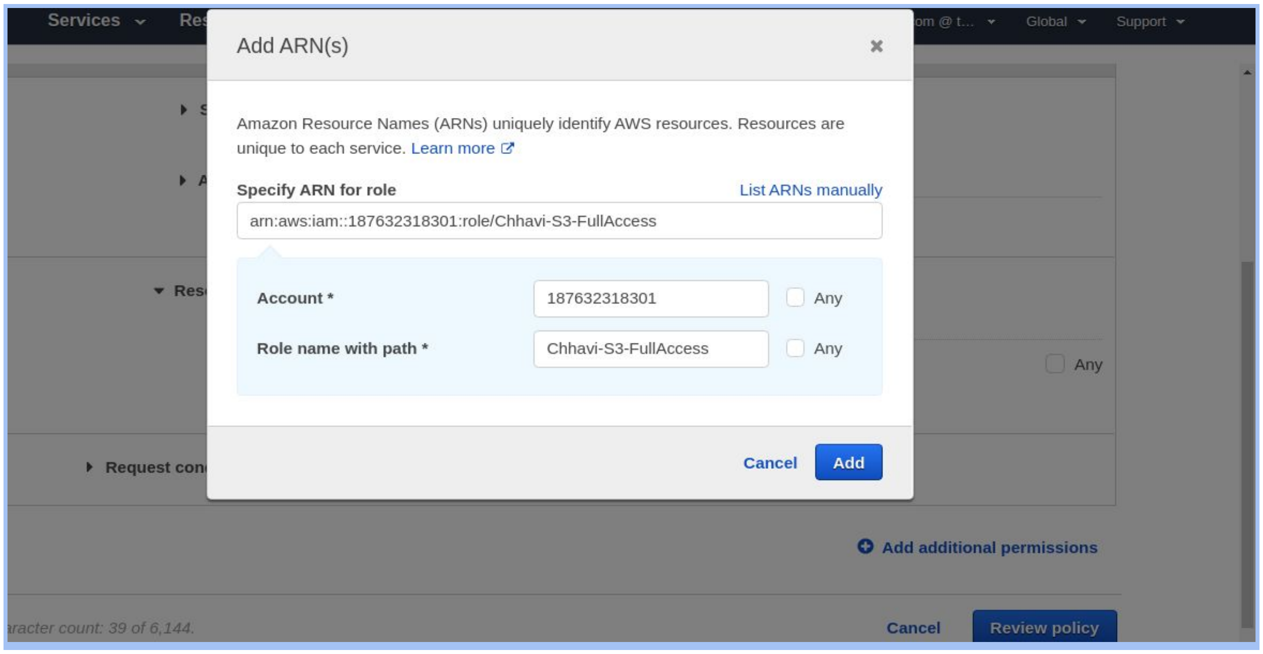
Actions **Write**
AssumeRole

Resources ● Specific
close ○ All resources

role Specify **role** resource ARN for the **AssumeRole** action. □ Any

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Step 3: Add previous policy’s ARN



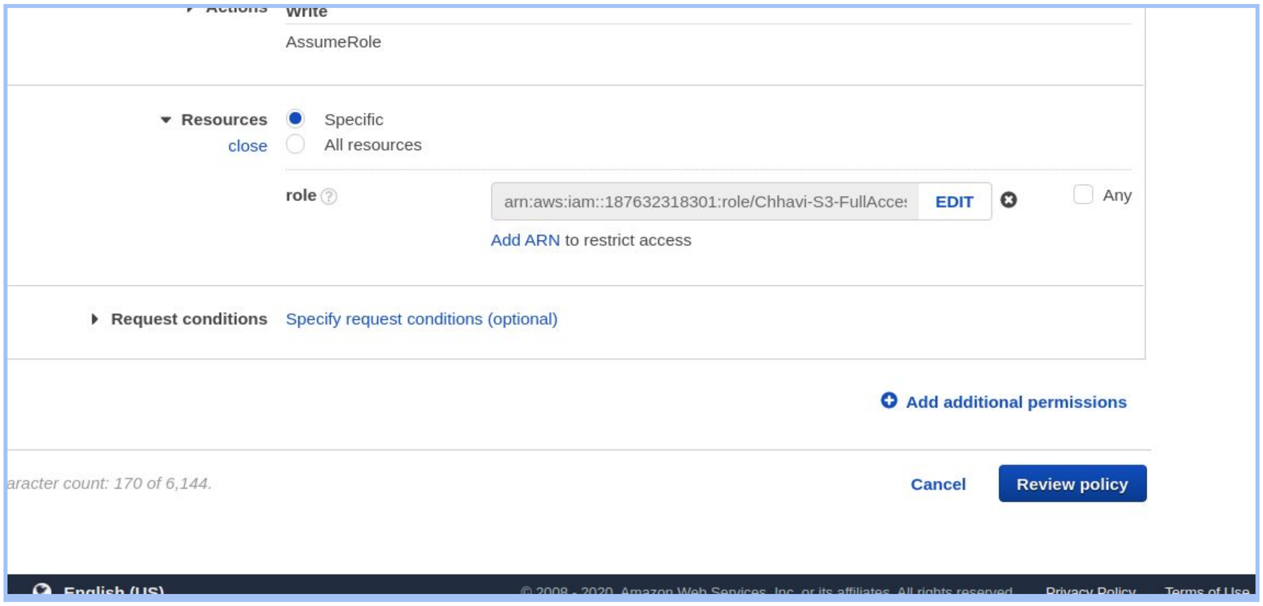
Amazon Resource Names (ARNs) uniquely identify AWS resources. Resources are unique to each service. Learn more

Specify ARN for role
arn:aws:iam::187632318301:role/Chhavi-S3-FullAccess

Account * 187632318301 □ Any

Role name with path * Chhavi-S3-FullAccess □ Any

aracter count: 39 of 6,144.



Actions
Write
Request conditions
Specify request conditions (optional)
Add additional permissions
Cancel
Review policy
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

AssumeRole

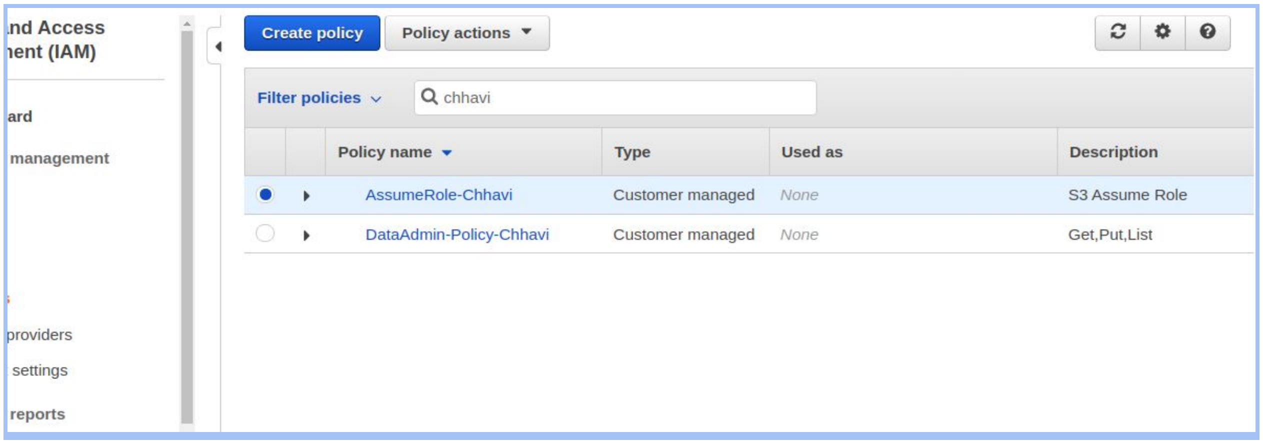
Resources ● Specific ○ All resources

role arn:aws:iam::187632318301:role/Chhavi-S3-FullAcces EDIT □ Any

Add ARN to restrict access

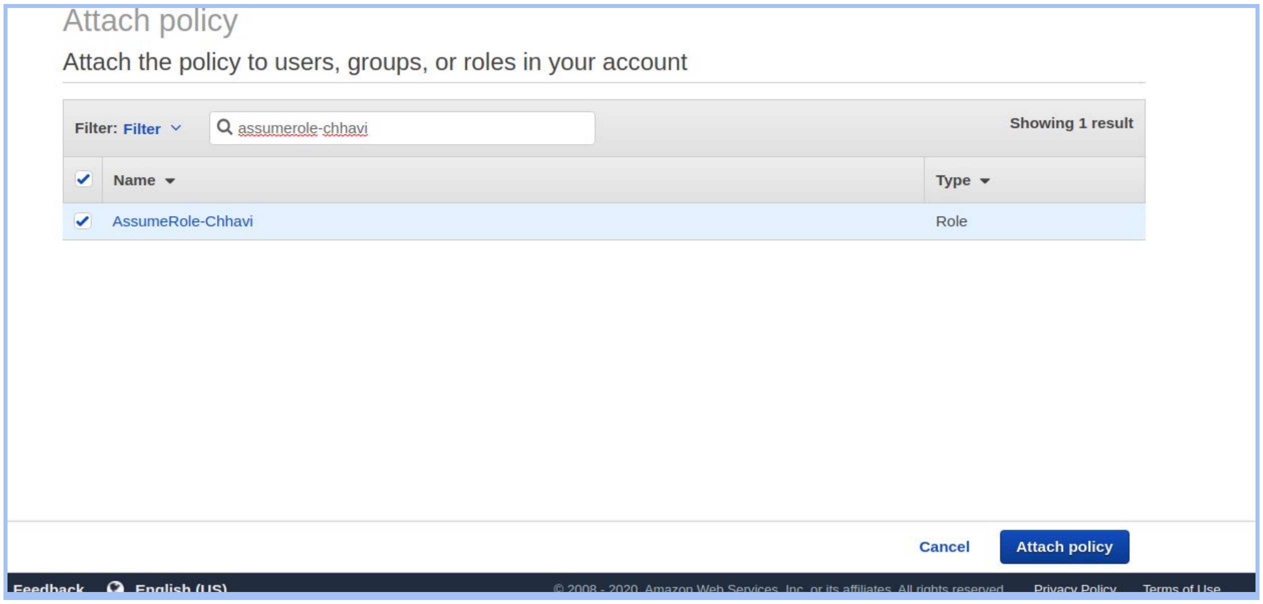
aracter count: 170 of 6,144.

Step 4: Attach new policy to new role created.



chhavi

		Policy name	Type	Used as	Description
●	▸	AssumeRole-Chhavi	Customer managed	*None*	S3 Assume Role
○	▸	DataAdmin-Policy-Chhavi	Customer managed	*None*	Get,Put,List



Attach policy

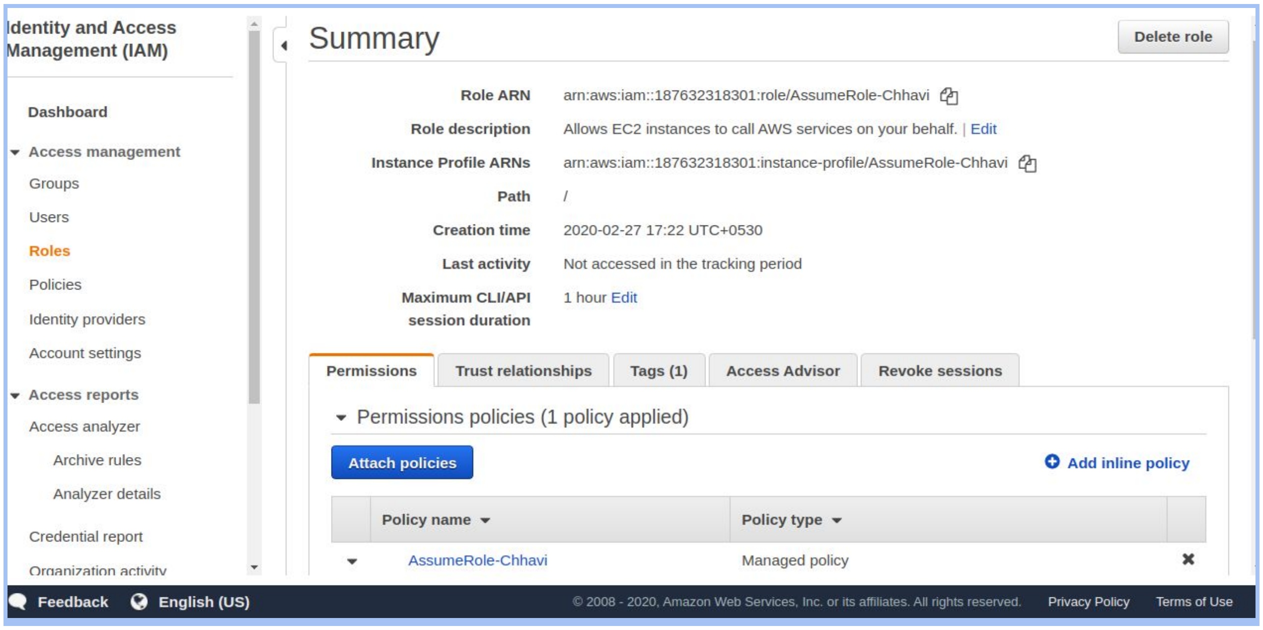
Attach the policy to users, groups, or roles in your account

Filter: Filter　assumerole-chhavi　Showing 1 result

☑	Name	Type
☑	AssumeRole-Chhavi	Role

Cancel
Attach policy
Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

Step 5: Check New role’s summary



Summary

Role ARN	arn:aws:iam::187632318301:role/AssumeRole-Chhavi
Role description	Allows EC2 instances to call AWS services on your behalf. | Edit
Instance Profile ARNs	arn:aws:iam::187632318301:instance-profile/AssumeRole-Chhavi
Path	/
Creation time	2020-02-27 17:22 UTC+0530
Last activity	Not accessed in the tracking period
Maximum CLI/API session duration	1 hour Edit

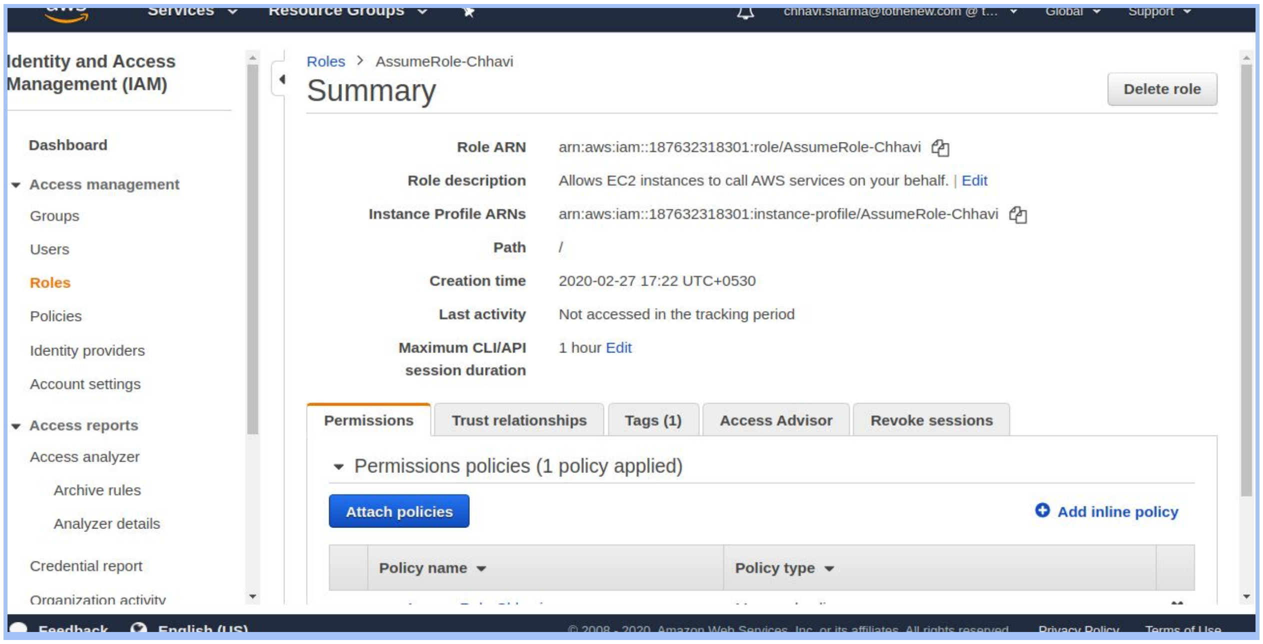
Permissions policies (1 policy applied)

Policy name	Policy type	
AssumeRole-Chhavi	Managed policy	✖

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

You can see the assumed role attached

Copy the ARN of assume role .



Summary

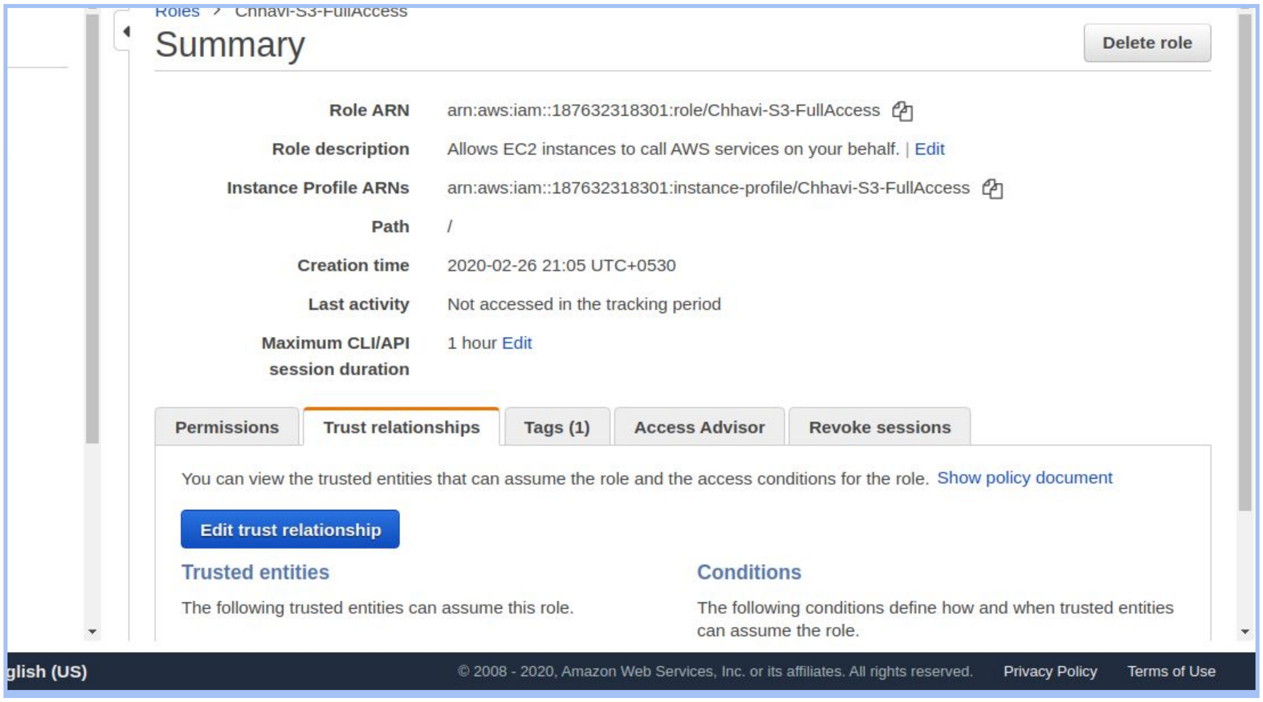
Role ARN	arn:aws:iam::187632318301:role/AssumeRole-Chhavi
Role description	Allows EC2 instances to call AWS services on your behalf. | Edit
Instance Profile ARNs	arn:aws:iam::187632318301:instance-profile/AssumeRole-Chhavi
Path	/
Creation time	2020-02-27 17:22 UTC+0530
Last activity	Not accessed in the tracking period
Maximum CLI/API session duration	1 hour Edit

Permissions policies (1 policy applied)

Policy name	Policy type	

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Goto Trust Relationship Tab in the previous policy.



Roles › Chhavi-S3-FullAccess

Summary

Role ARN	arn:aws:iam::187632318301:role/Chhavi-S3-FullAccess
Role description	Allows EC2 instances to call AWS services on your behalf. | Edit
Instance Profile ARNs	arn:aws:iam::187632318301:instance-profile/Chhavi-S3-FullAccess
Path	/
Creation time	2020-02-26 21:05 UTC+0530
Last activity	Not accessed in the tracking period
Maximum CLI/API session duration	1 hour Edit

You can view the trusted entities that can assume the role and the access conditions for the role. Show policy document

Trusted entities

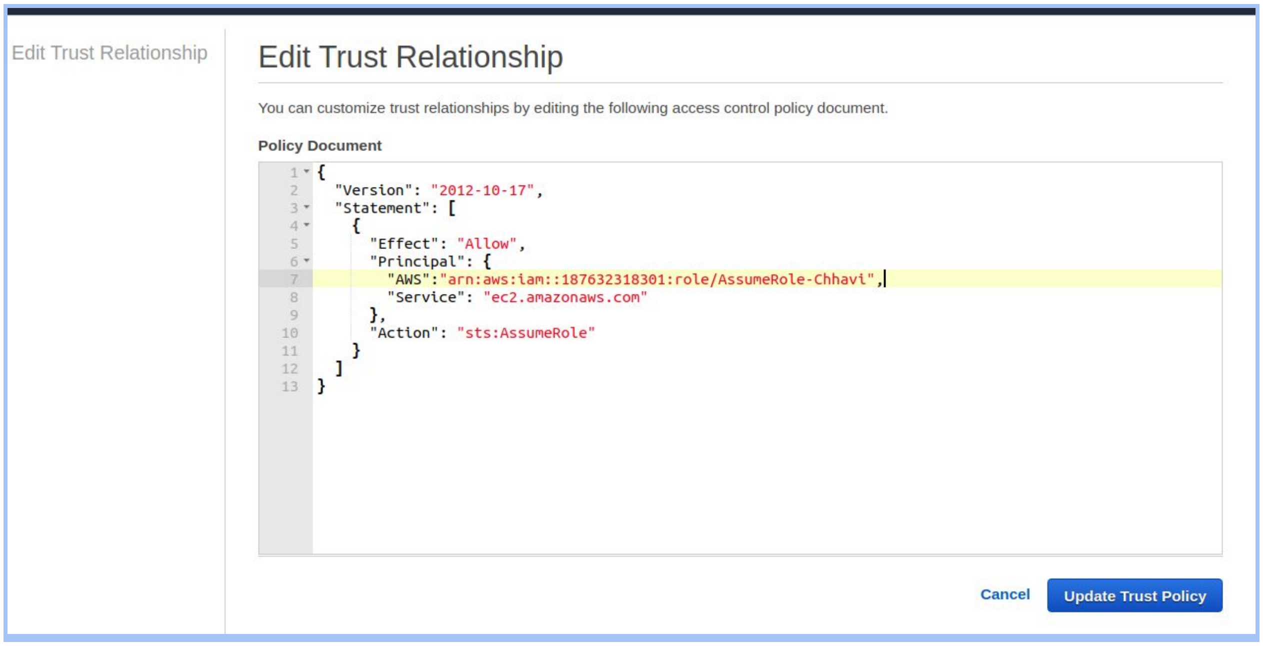
The following trusted entities can assume this role.

Conditions

The following conditions define how and when trusted entities can assume the role.

glish (US) © 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Edit the Trust Relationship of the previous role created.Add the ARN of the assume role created.

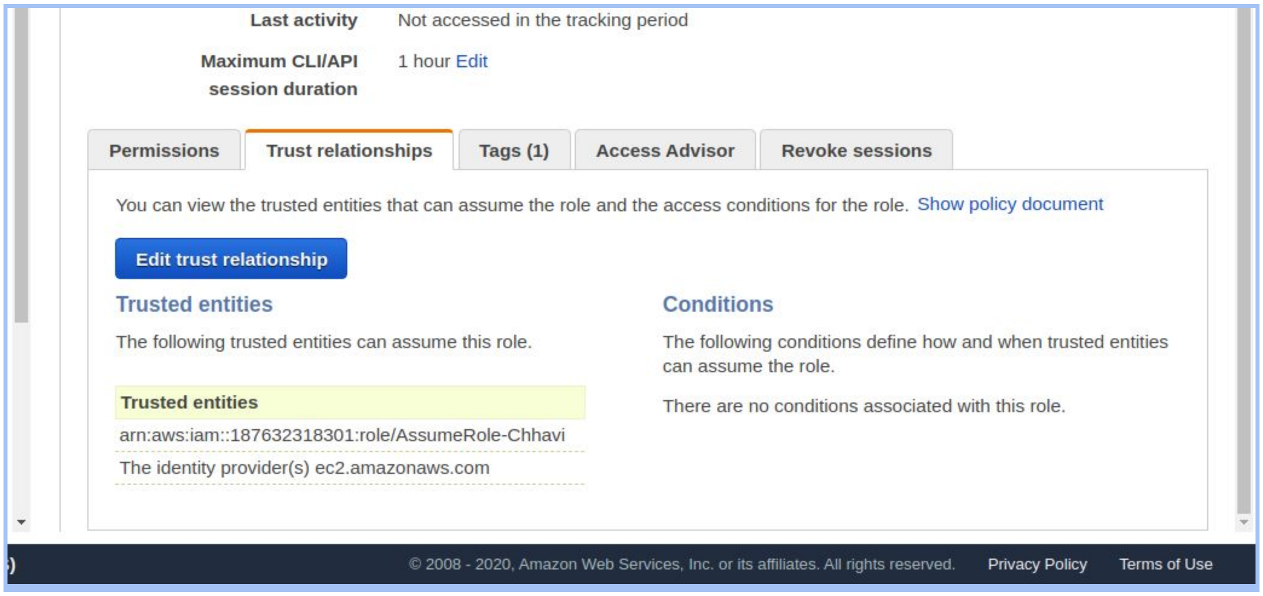


Edit Trust Relationship

You can customize trust relationships by editing the following access control policy document.

Policy Document

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS":"arn:aws:iam::187632318301:role/AssumeRole-Chhavi",
        "Service": "ec2.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```



Last activity: Not accessed in the tracking period

Maximum CLI/API session duration: 1 hour Edit

You can view the trusted entities that can assume the role and the access conditions for the role. Show policy document

Trusted entities

The following trusted entities can assume this role.

Trusted entities
arn:aws:iam::187632318301:role/AssumeRole-Chhavi
The identity provider(s) ec2.amazonaws.com

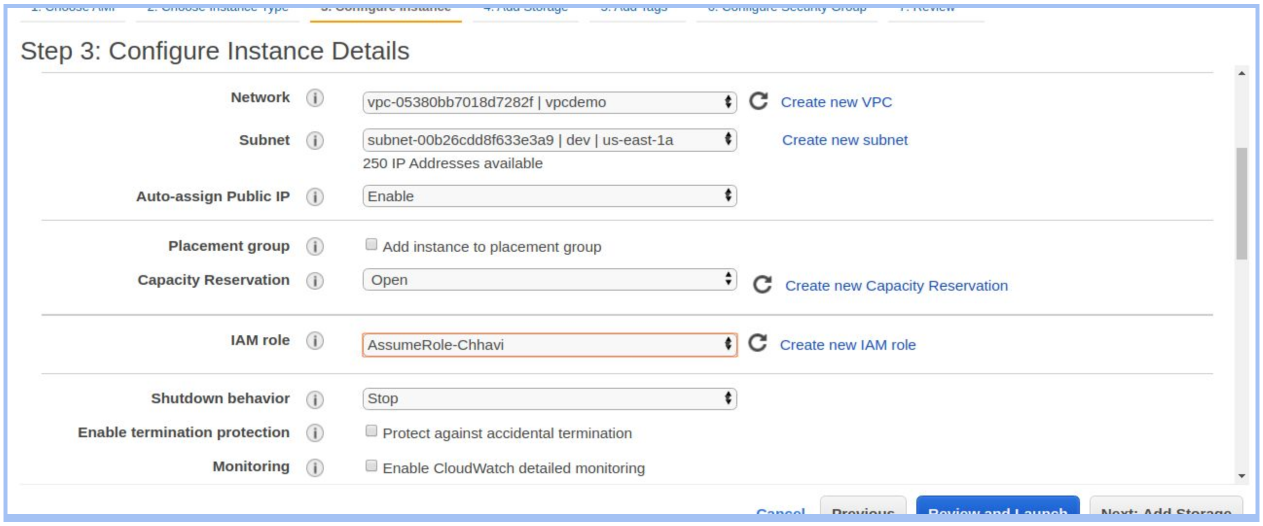
Conditions

The following conditions define how and when trusted entities can assume the role.

There are no conditions associated with this role.

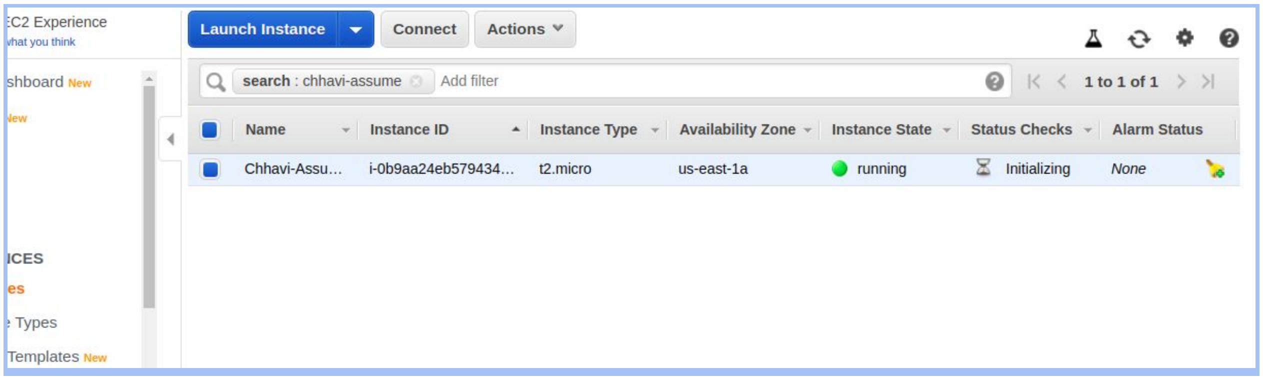
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Now create a new instance.Chhavi-AssumeRole-Instanceand attach AssumeRole-Chhavi to the instance in the configuration.



Step 3: Configure Instance Details

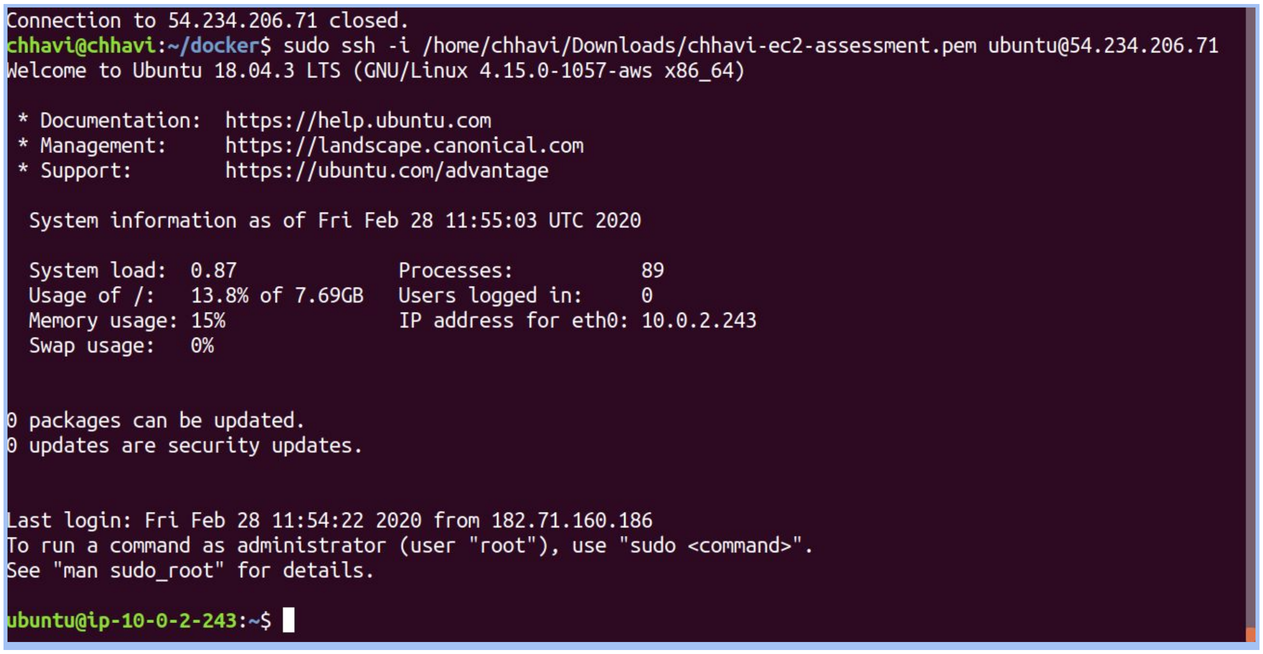
Network	vpc-05380bb7018d7282f | vpcdemo
Subnet	subnet-00b26cdd8f633e3a9 | dev | us-east-1a 250 IP Addresses available
Auto-assign Public IP	Enable
Placement group	□ Add instance to placement group
Capacity Reservation	Open
IAM role	AssumeRole-Chhavi
Shutdown behavior	Stop
Enable termination protection	□ Protect against accidental termination
Monitoring	□ Enable CloudWatch detailed monitoring



search : chhavi-assume   Add filter

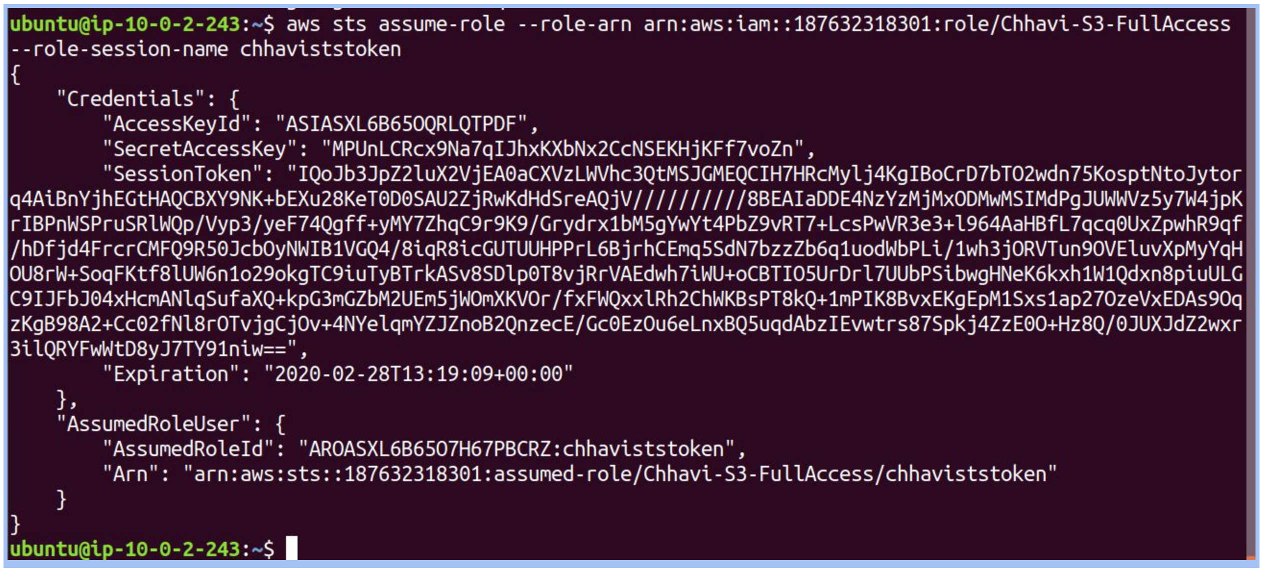
	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status
■	Chhavi-Assu...	i-0b9aa24eb579434...	t2.micro	us-east-1a	● running	Initializing	None

Ssh into the instance created.Install awscli.



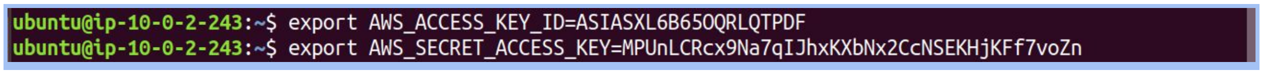
```
Connection to 54.234.206.71 closed.
chhavi@chhavi:~/docker$ sudo ssh -i /home/chhavi/Downloads/chhavi-ec2-assessment.pem ubuntu@54.234.206.71
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-1057-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Fri Feb 28 11:55:03 UTC 2020

  System load:  0.87              Processes:           89
  Usage of /:   13.8% of 7.69GB   Users logged in:     0
  Memory usage: 15%               IP address for eth0: 10.0.2.243
  Swap usage:   0%


0 packages can be updated.
0 updates are security updates.


Last login: Fri Feb 28 11:54:22 2020 from 182.71.160.186
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-10-0-2-243:~$ 
```

Generate an sts token.

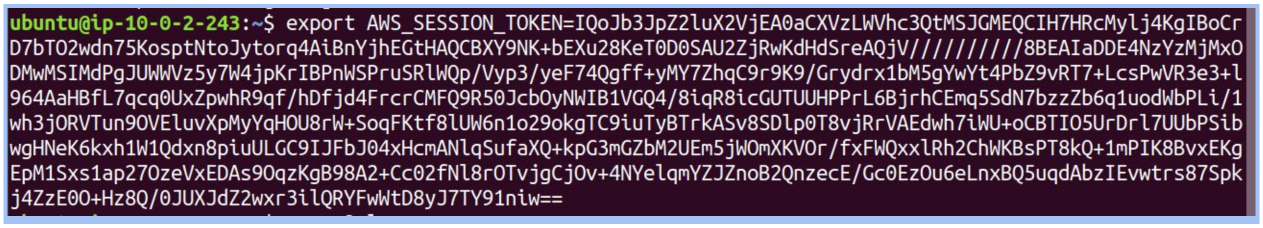


```
ubuntu@ip-10-0-2-243:~$ aws sts assume-role --role-arn arn:aws:iam::187632318301:role/Chhavi-S3-FullAccess
--role-session-name chhaviststoken
{
    "Credentials": {
        "AccessKeyId": "ASIASXL6B65OQRLQTPDF",
        "SecretAccessKey": "MPUnLCRcx9Na7qIJhxKXbNx2CcNSEKHjKFf7voZn",
        "SessionToken": "IQoJb3JpZ2luX2VjEA0aCXVzLWVhc3QtMSJGMEQCIH7HRcMylj4KgIBoCrD7bTO2wdn75KosptNtoJytor
q4AiBnYjhEGtHAQCBXY9NK+bEXu28KeT0D0SAU2ZjRwKdHdSreAQjV//////////8BEAIaDDE4NzYzMjMxODMwMSIMdPgJUWWVz5y7W4jpK
rIBPnWSPruSRlWQp/Vyp3/yeF74Qgff+yMY7ZhqC9r9K9/Grydrx1bM5gYwYt4PbZ9vRT7+LcsPwVR3e3+l964AaHBfL7qcq0UxZpwhR9qf
/hDfjd4FrcrCMFQ9R50JcbOyNWIB1VGQ4/8iqR8icGUTUUHPPrL6BjrhCEmq5SdN7bzzZb6q1uodWbPLi/1wh3jORVTun9OVEluvXpMyYqH
OU8rW+SoqFKtf8lUW6n1o29okgTC9iuTyBTrkASv8SDlp0T8vjRrVAEdwh7iWU+oCBTIO5UrDrl7UUbPSibwgHNeK6kxh1W1Qdxn8piuULG
C9IJFbJ04xHcmANlqSufaXQ+kpG3mGZbM2UEm5jWOmXKVOr/fxFWQxxlRh2ChWKBsPT8kQ+1mPIK8BvxEKgEpM1Sxs1ap27OzeVxEDAs9Oq
zKgB98A2+Cc02fNl8rOTvjgCjOv+4NYelqmYZJZnoB2QnzecE/Gc0EzOu6eLnxBQ5uqdAbzIEvwtrs87Spkj4ZzE0O+Hz8Q/0JUXJdZ2wxr
3ilQRYFwWtD8yJ7TY91niw==",
        "Expiration": "2020-02-28T13:19:09+00:00"
    },
    "AssumedRoleUser": {
        "AssumedRoleId": "AROASXL6B65O7H67PBCRZ:chhaviststoken",
        "Arn": "arn:aws:sts::187632318301:assumed-role/Chhavi-S3-FullAccess/chhaviststoken"
    }
}
ubuntu@ip-10-0-2-243:~$ 
```

Export access key id , secret access key and token.

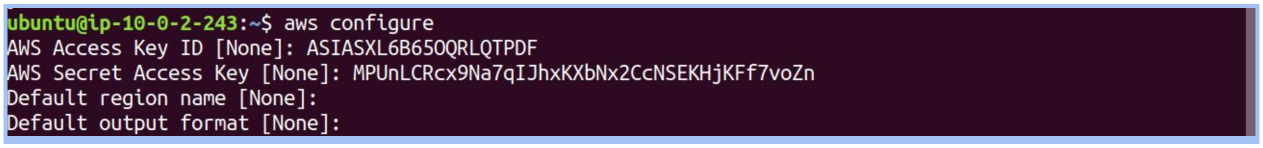


```
ubuntu@ip-10-0-2-243:~$ export AWS_ACCESS_KEY_ID=ASIASXL6B65OQRLQTPDF
ubuntu@ip-10-0-2-243:~$ export AWS_SECRET_ACCESS_KEY=MPUnLCRcx9Na7qIJhxKXbNx2CcNSEKHjKFf7voZn
```



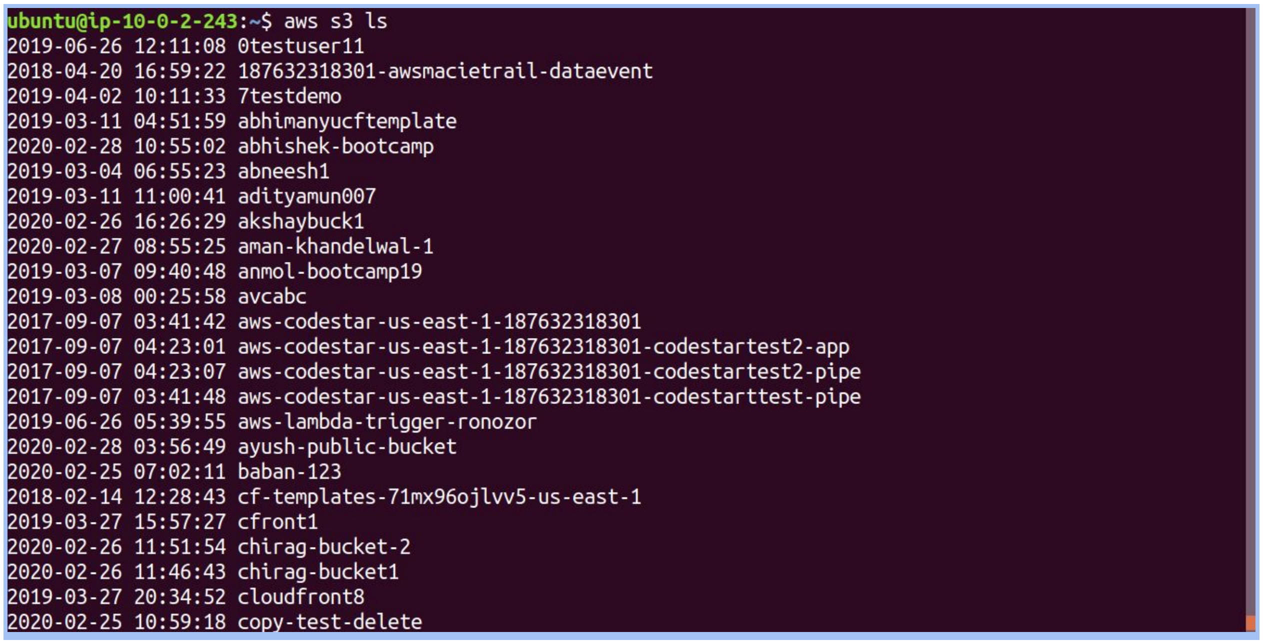
```
ubuntu@ip-10-0-2-243:~$ export AWS_SESSION_TOKEN=IQoJb3JpZ2luX2VjEA0aCXVzLWVhc3QtMSJGMEQCIH7HRcMylj4KgIBoCr
D7bTO2wdn75KosptNtoJytorq4AiBnYjhEGtHAQCBXY9NK+bEXu28KeT0D0SAU2ZjRwKdHdSreAQjV//////////8BEAIaDDE4NzYzMjMxO
DMwMSIMdPgJUWWVz5y7W4jpKrIBPnWSPruSRlWQp/Vyp3/yeF74Qgff+yMY7ZhqC9r9K9/Grydrx1bM5gYwYt4PbZ9vRT7+LcsPwVR3e3+l
964AaHBfL7qcq0UxZpwhR9qf/hDfjd4FrcrCMFQ9R50JcbOyNWIB1VGQ4/8iqR8icGUTUUHPPrL6BjrhCEmq5SdN7bzzZb6q1uodWbPLi/1
wh3jORVTun9OVEluvXpMyYqHOU8rW+SoqFKtf8lUW6n1o29okgTC9iuTyBTrkASv8SDlp0T8vjRrVAEdwh7iWU+oCBTIO5UrDrl7UUbPSib
wgHNeK6kxh1W1Qdxn8piuULGC9IJFbJ04xHcmANlqSufaXQ+kpG3mGZbM2UEm5jWOmXKVOr/fxFWQxxlRh2ChWKBsPT8kQ+1mPIK8BvxEKg
EpM1Sxs1ap27OzeVxEDAs9OqzKgB98A2+Cc02fNl8rOTvjgCjOv+4NYelqmYZJZnoB2QnzecE/Gc0EzOu6eLnxBQ5uqdAbzIEvwtrs87Spk
j4ZzE0O+Hz8Q/0JUXJdZ2wxr3ilQRYFwWtD8yJ7TY91niw==
```

Aws configure.



```
ubuntu@ip-10-0-2-243:~$ aws configure
AWS Access Key ID [None]: ASIASXL6B65OQRLQTPDF
AWS Secret Access Key [None]: MPUnLCRcx9Na7qIJhxKXbNx2CcNSEKHjKFf7voZn
Default region name [None]:
Default output format [None]:
```

Now write aws s3 ls.

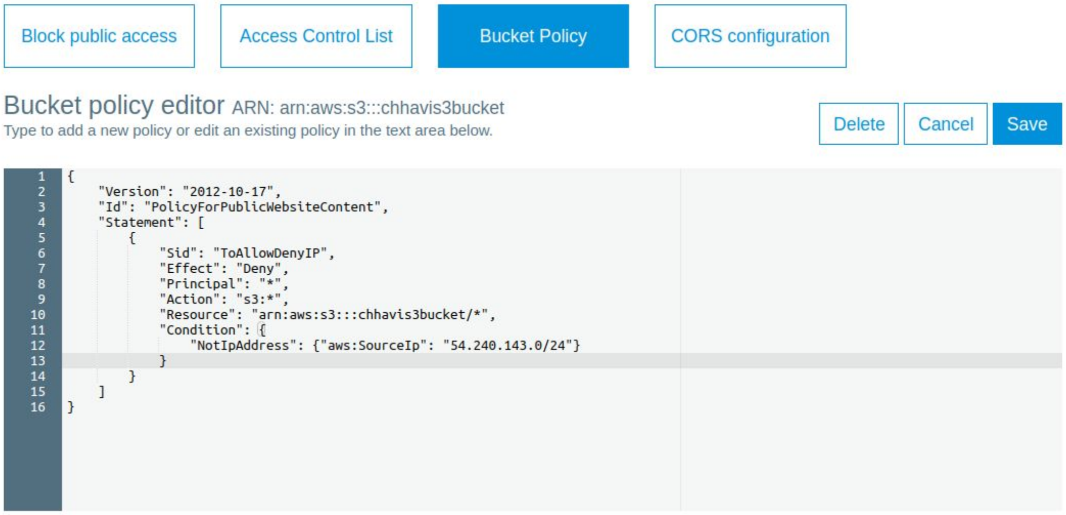


```
ubuntu@ip-10-0-2-243:~$ aws s3 ls
2019-06-26 12:11:08 0testuser11
2018-04-20 16:59:22 187632318301-awsmacietrail-dataevent
2019-04-02 10:11:33 7testdemo
2019-03-11 04:51:59 abhimanyucftemplate
2020-02-28 10:55:02 abhishek-bootcamp
2019-03-04 06:55:23 abneesh1
2019-03-11 11:00:41 adityamun007
2020-02-26 16:26:29 akshaybuck1
2020-02-27 08:55:25 aman-khandelwal-1
2019-03-07 09:40:48 anmol-bootcamp19
2019-03-08 00:25:58 avcabc
2017-09-07 03:41:42 aws-codestar-us-east-1-187632318301
2017-09-07 04:23:01 aws-codestar-us-east-1-187632318301-codestartest2-app
2017-09-07 04:23:07 aws-codestar-us-east-1-187632318301-codestartest2-pipe
2017-09-07 03:41:48 aws-codestar-us-east-1-187632318301-codestarttest-pipe
2019-06-26 05:39:55 aws-lambda-trigger-ronozor
2020-02-28 03:56:49 ayush-public-bucket
2020-02-25 07:02:11 baban-123
2018-02-14 12:28:43 cf-templates-71mx96ojlvv5-us-east-1
2019-03-27 15:57:27 cfront1
2020-02-26 11:51:54 chirag-bucket-2
2020-02-26 11:46:43 chirag-bucket1
2019-03-27 20:34:52 cloudfront8
2020-02-25 10:59:18 copy-test-delete
```

3. Block s3 access on the basis of

i. IP

Ans.



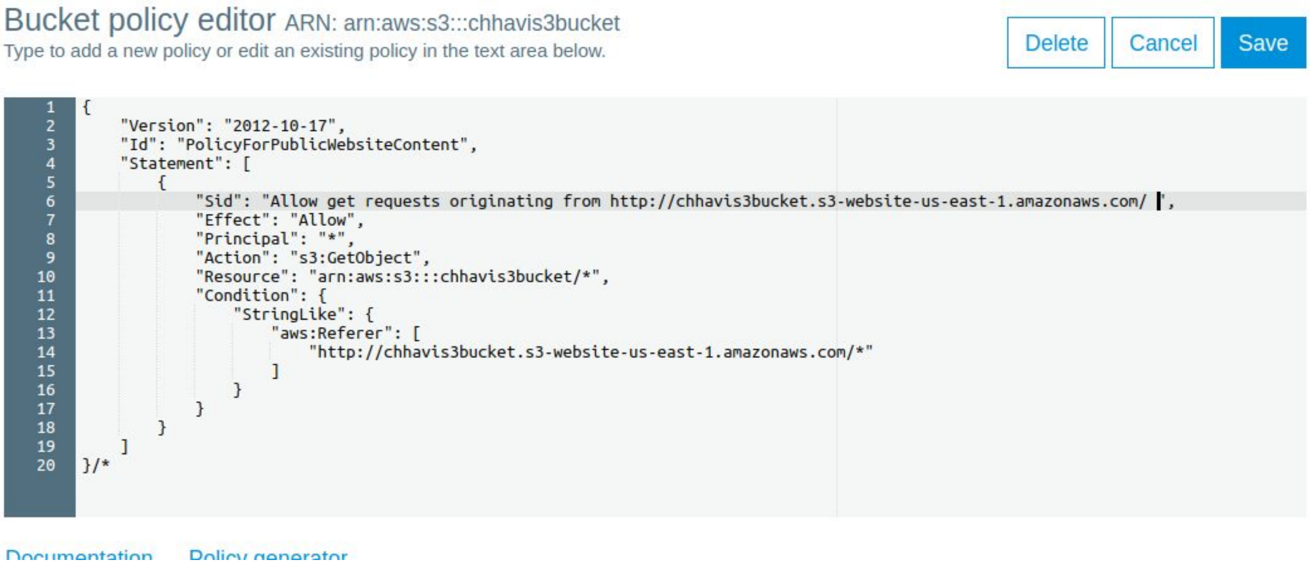
Bucket policy editor ARN: arn:aws:s3:::chhavis3bucket

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

```
{
    "Version": "2012-10-17",
    "Id": "PolicyForPublicWebsiteContent",
    "Statement": [
        {
            "Sid": "ToAllowDenyIP",
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3:*",
            "Resource": "arn:aws:s3:::chhavis3bucket/*",
            "Condition": {
                "NotIpAddress": {"aws:SourceIp": "54.240.143.0/24"}
            }
        }
    ]
}
```

ii. Domain

Ans.



Bucket policy editor ARN: arn:aws:s3:::chhavis3bucket

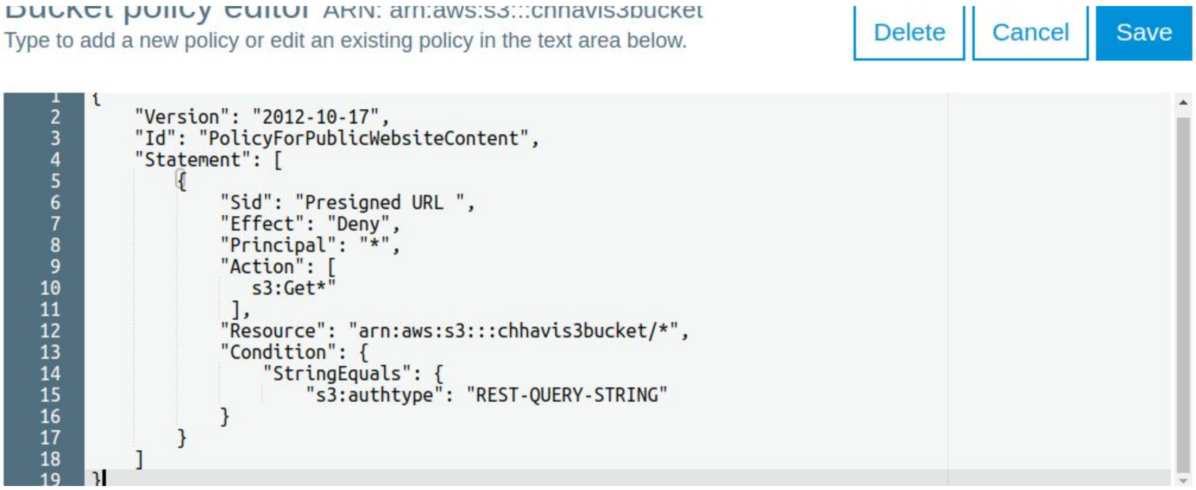
Type to add a new policy or edit an existing policy in the text area below.

```
{
    "Version": "2012-10-17",
    "Id": "PolicyForPublicWebsiteContent",
    "Statement": [
        {
            "Sid": "Allow get requests originating from http://chhavis3bucket.s3-website-us-east-1.amazonaws.com/ ",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::chhavis3bucket/*",
            "Condition": {
                "StringLike": {
                    "aws:Referer": [
                        "http://chhavis3bucket.s3-website-us-east-1.amazonaws.com/*"
                    ]
                }
            }
        }
    ]
}/*
```

iii. Pre-signed URL(Time based)

Ans. A Pre-signed URL is the one that you can provide to your users to grant temporary access to a specific S3 object. Using the URL, a user can either READ the object or WRITE an Object (or update an existing object). THe URL contains specific parameters which are set by your application.

1. Bucket: The bucket that the object is in (or will be in)
2. Key: The name of the object.
3. Expires: THe amount of time that the URL is valid for.



Bucket policy editor ARN: arn:aws:s3:::chhavis3bucket

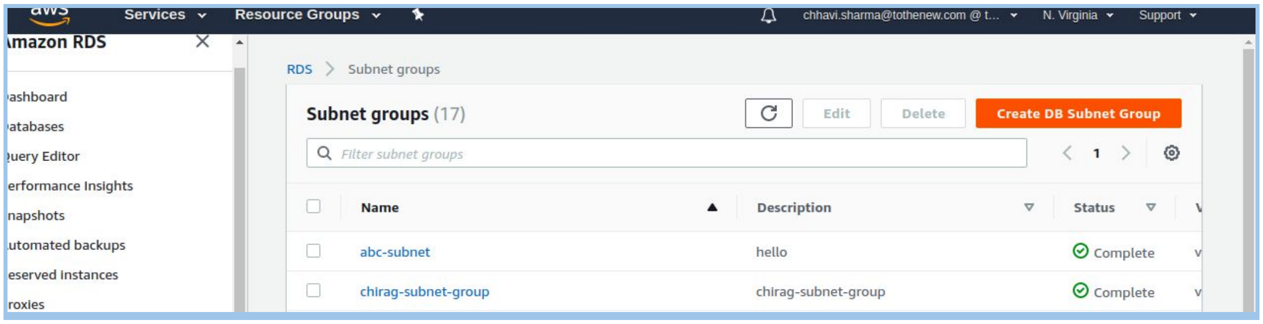
Type to add a new policy or edit an existing policy in the text area below.

```
{
    "Version": "2012-10-17",
    "Id": "PolicyForPublicWebsiteContent",
    "Statement": [
        {
            "Sid": "Presigned URL ",
            "Effect": "Deny",
            "Principal": "*",
            "Action": [
              s3:Get*"
             ],
            "Resource": "arn:aws:s3:::chhavis3bucket/*",
            "Condition": {
                "StringEquals": {
                    "s3:authtype": "REST-QUERY-STRING"
            }
        }
    ]
}
```

4. Create RDS subnet and launch RDS instance.What is parameter group and option group?

Ans.

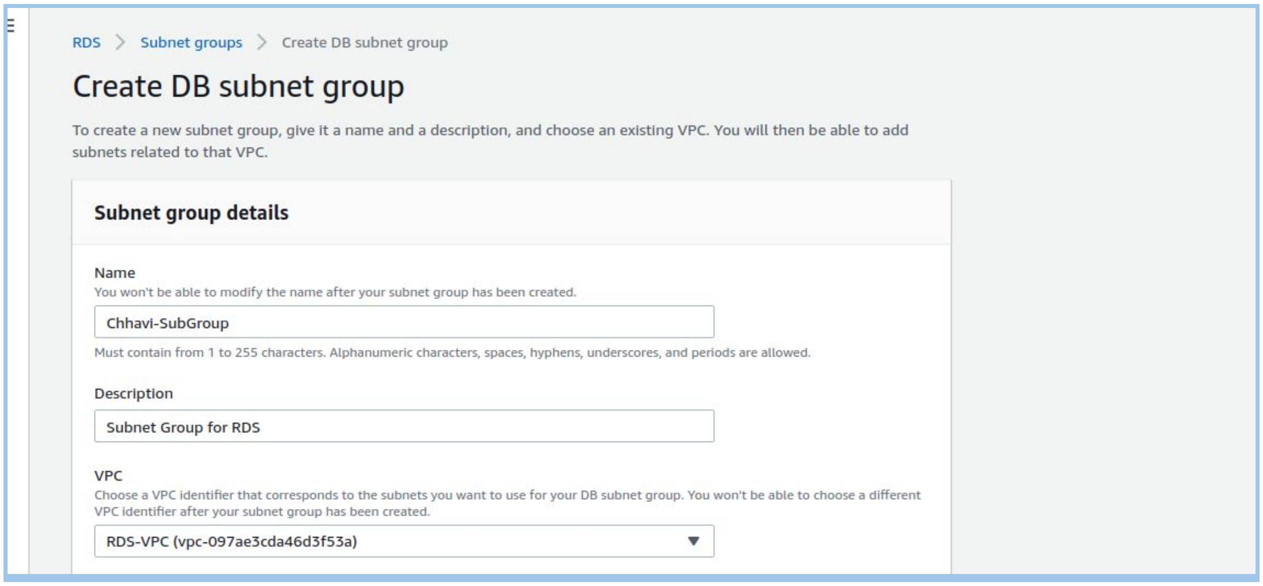
Go to Amazon RDS, build a db-subnet group



RDS > Subnet groups

Subnet groups (17)

□	Name	Description	Status
□	abc-subnet	hello	Complete
□	chirag-subnet-group	chirag-subnet-group	Complete



RDS > Subnet groups > Create DB subnet group

Create DB subnet group

To create a new subnet group, give it a name and a description, and choose an existing VPC. You will then be able to add subnets related to that VPC.

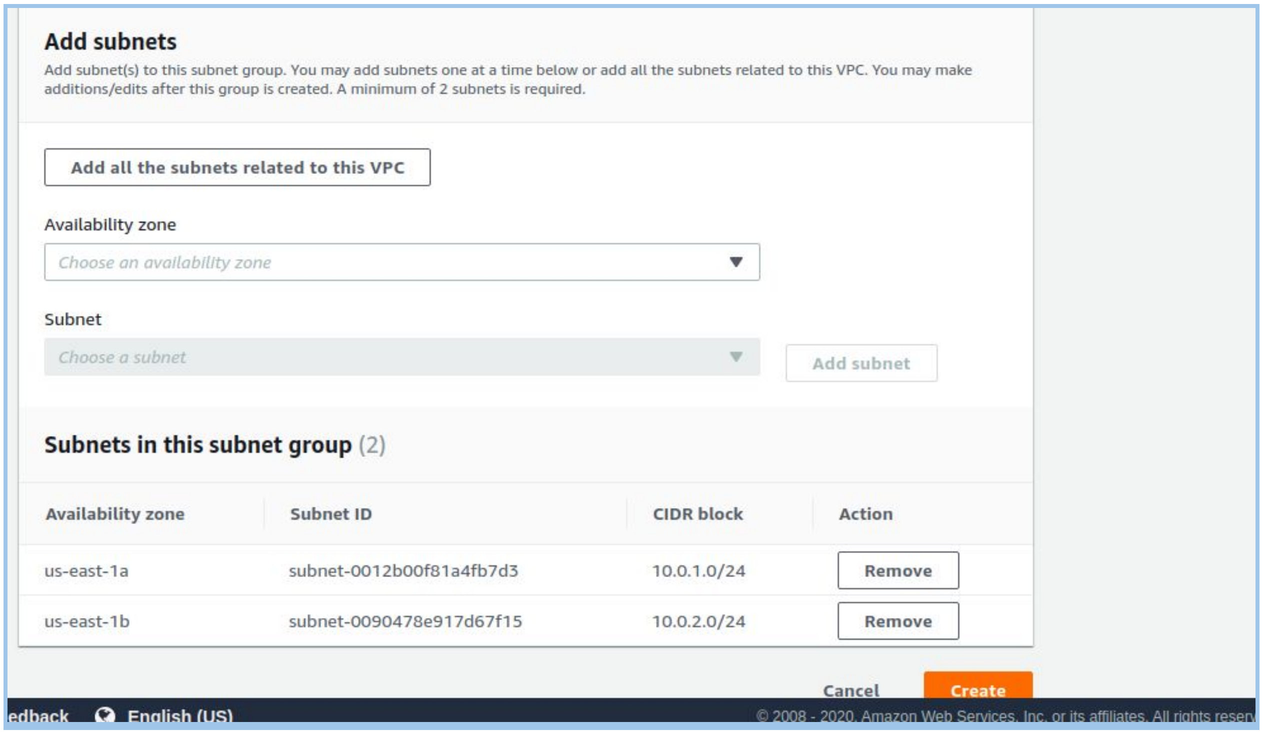
Subnet group details

Name
You won't be able to modify the name after your subnet group has been created.
Chhavi-SubGroup
Must contain from 1 to 255 characters. Alphanumeric characters, spaces, hyphens, underscores, and periods are allowed.

Description
Subnet Group for RDS

VPC
Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You won't be able to choose a different VPC identifier after your subnet group has been created.
RDS-VPC (vpc-097ae3cda46d3f53a)

In the Add subnets section, choose Add all the subnets related to this VPC.



Add subnets

Add subnet(s) to this subnet group. You may add subnets one at a time below or add all the subnets related to this VPC. You may make additions/edits after this group is created. A minimum of 2 subnets is required.

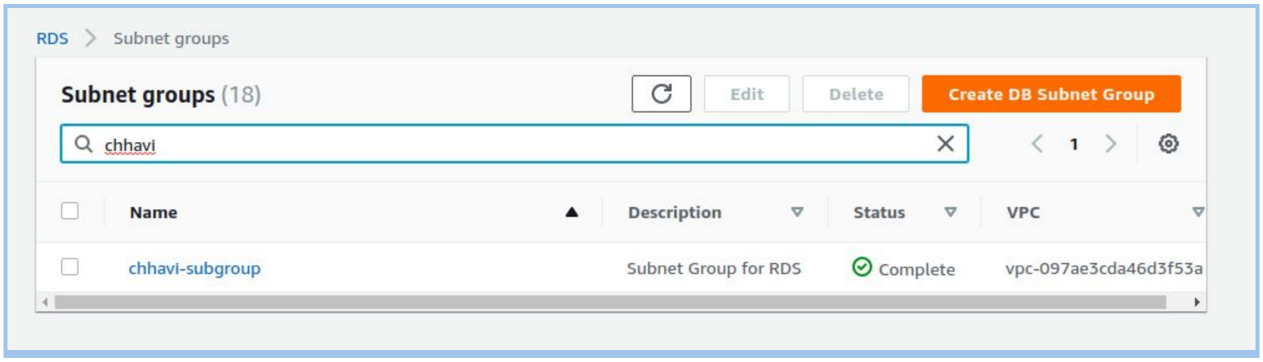
Availability zone: *Choose an availability zone*

Subnet: *Choose a subnet*

Subnets in this subnet group (2)

Availability zone	Subnet ID	CIDR block	Action
us-east-1a	subnet-0012b00f81a4fb7d3	10.0.1.0/24	Remove
us-east-1b	subnet-0090478e917d67f15	10.0.2.0/24	Remove

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserv



Subnet groups (18)

Search: chhavi

□	Name	Description	Status	VPC
□	chhavi-subgroup	Subnet Group for RDS	✓ Complete	vpc-097ae3cda46d3f53a

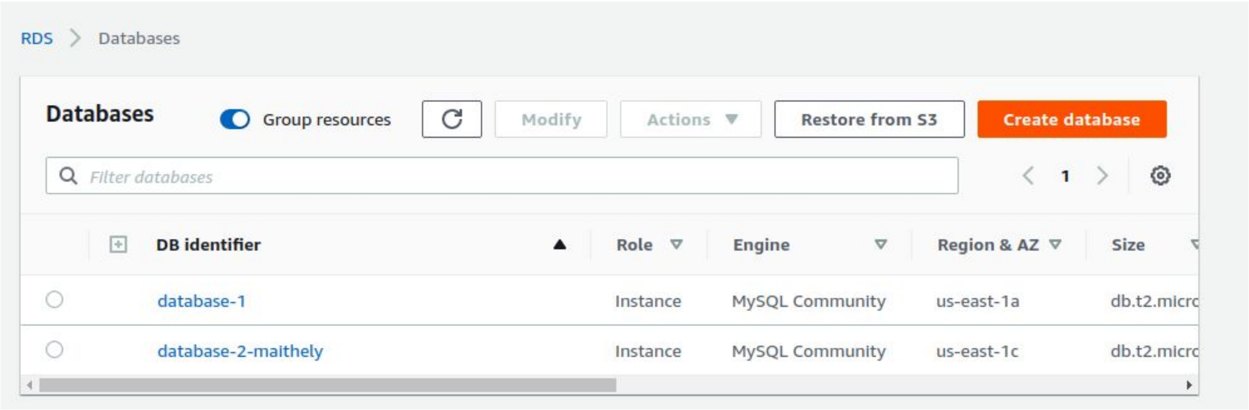
Create a VPC Security Group: Before you create your DB instance, you must create a VPC security group to associate with your DB instance. Choose the security group you created and edit inbound rules.Set the following values for your new inbound rule to allow MySQL traffic on port 3306 from your EC2 instance. If you do this, you can connect from your web server to your DB instance to store and retrieve data from your web application to your database.

Create a DB Instance in the VPC

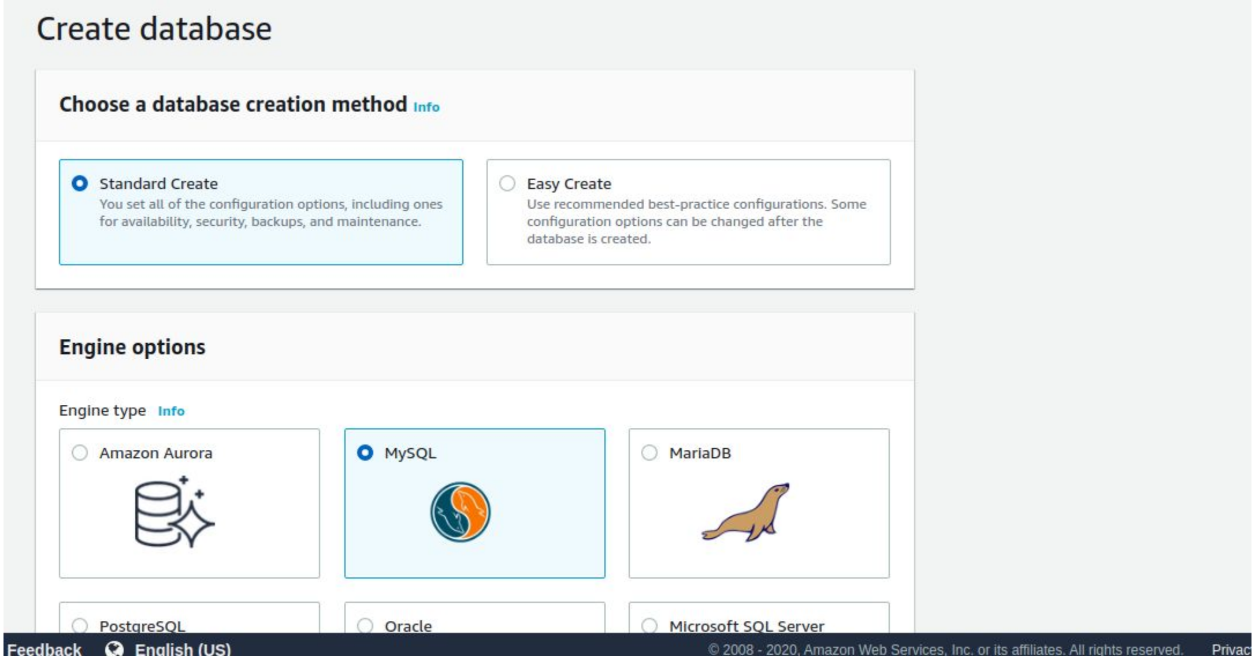
Databases &gt; Choose Create database &gt; In Choose a database creation method, choose Standard Create

*Use the VPC name, the DB subnet group, and the VPC security group you created in the previous steps.

*If you want your DB instance in the VPC to be publicly accessible, you must enable the VPC attributes DNS hostnames and DNS resolution.



	DB identifier	Role	Engine	Region & AZ	Size
○	database-1	Instance	MySQL Community	us-east-1a	db.t2.micr
○	database-2-maithely	Instance	MySQL Community	us-east-1c	db.t2.micr



Create database

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

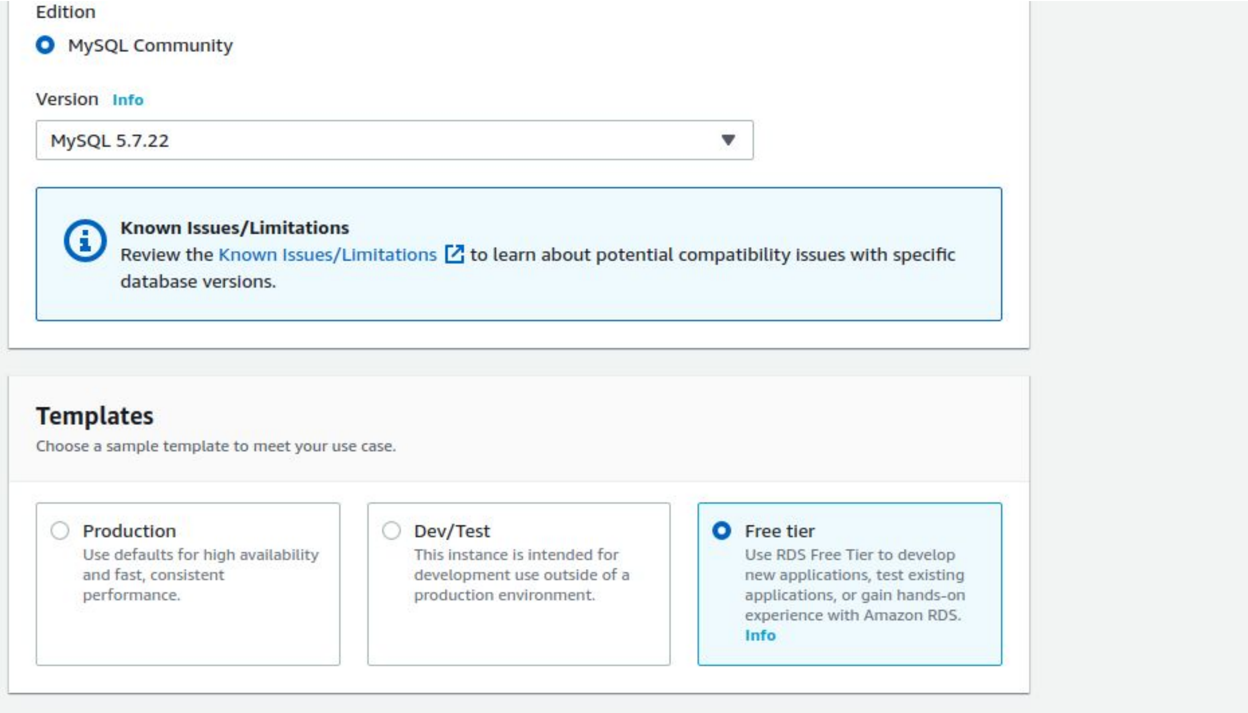
○ MariaDB

○ PostgreSQL

○ Oracle

○ Microsoft SQL Server

Feedback
English (US)
© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privac



Edition

● MySQL Community

Version Info

MySQL 5.7.22 ▼

Known Issues/Limitations
Review the Known Issues/Limitations to learn about potential compatibility issues with specific database versions.

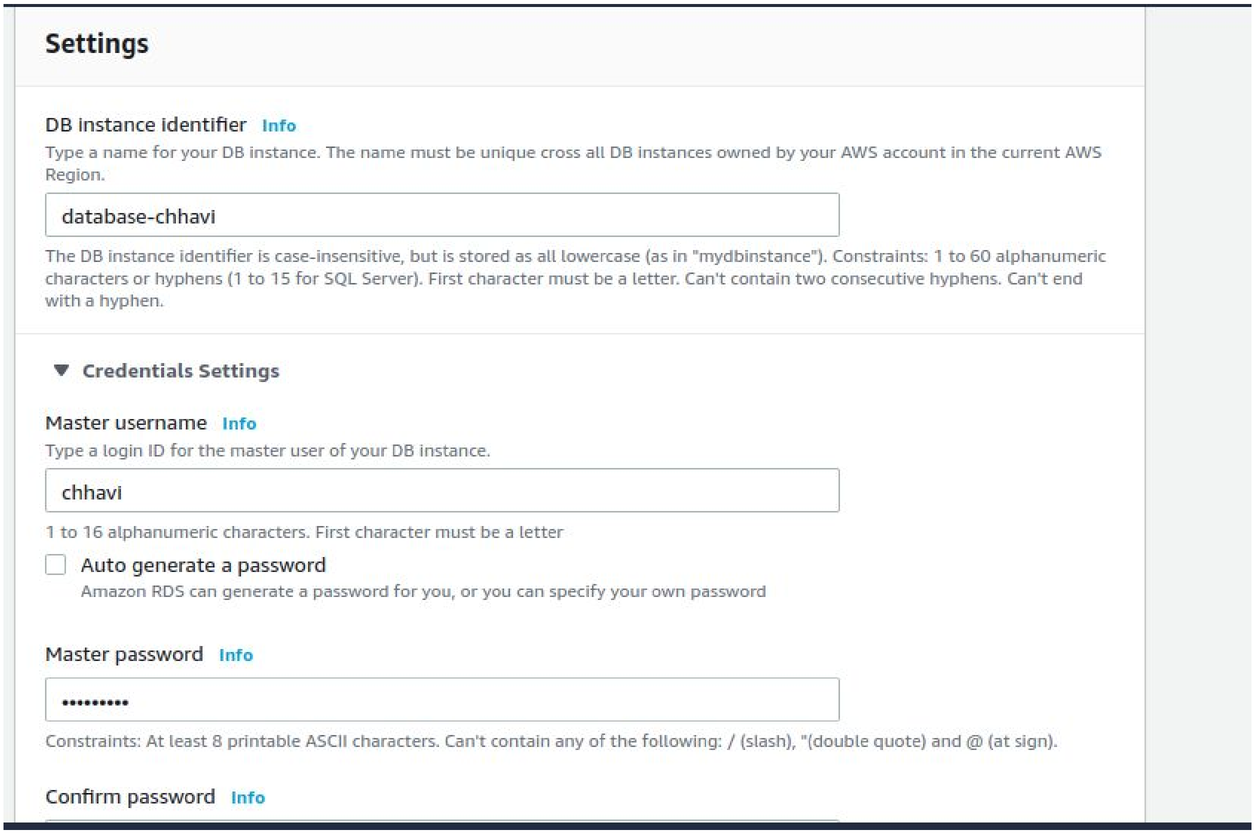
Templates

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

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.

database-chhavi

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.

▼ Credentials Settings

Master username

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

chhavi

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

□ Auto generate a password

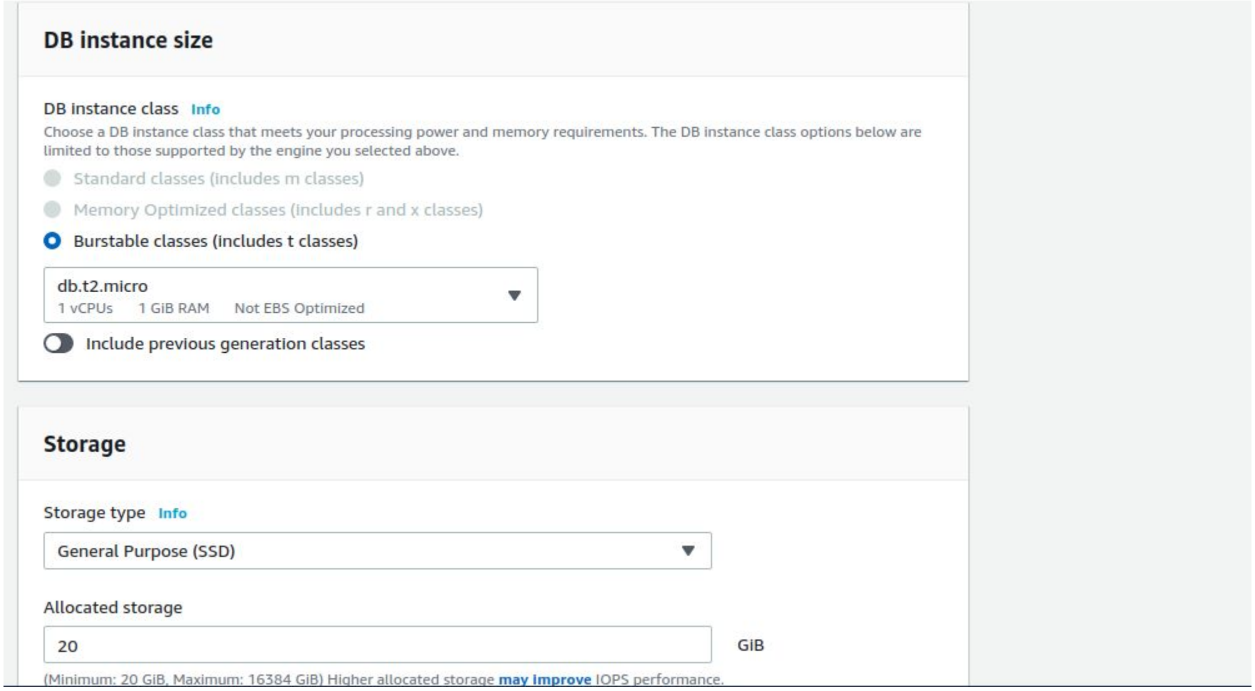
Amazon RDS can generate a password for you, or you can specify your own password

Master password

•••••••••

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

Confirm password



DB instance size

DB instance class

Choose a DB instance class that meets your processing power and memory requirements. The DB instance class options below are limited to those supported by the engine you selected above.

○ Standard classes (includes m classes)

○ Memory Optimized classes (includes r and x classes)

● Burstable classes (includes t classes)

db.t2.micro
1 vCPUs 1 GiB RAM Not EBS Optimized ▼

Include previous generation classes

Storage

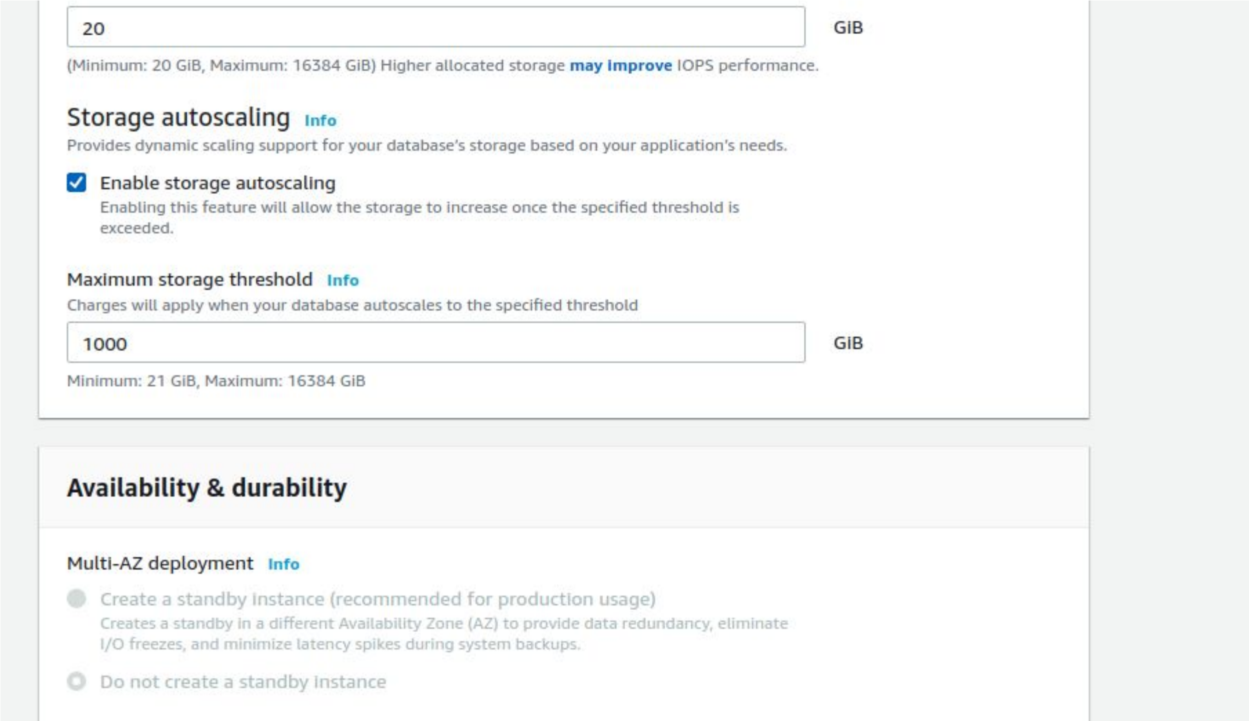
Storage type

General Purpose (SSD) ▼

Allocated storage

20 GiB

(Minimum: 20 GiB, Maximum: 16384 GiB) Higher allocated storage **may improve** IOPS performance.



20 GiB

(Minimum: 20 GiB, Maximum: 16384 GiB) Higher allocated storage **may improve** IOPS performance.

Storage autoscaling Info

Provides dynamic scaling support for your database's storage based on your application's needs.

☑ Enable storage autoscaling
Enabling this feature will allow the storage to increase once the specified threshold is exceeded.

Maximum storage threshold Info

Charges will apply when your database autoscales to the specified threshold

1000 GiB

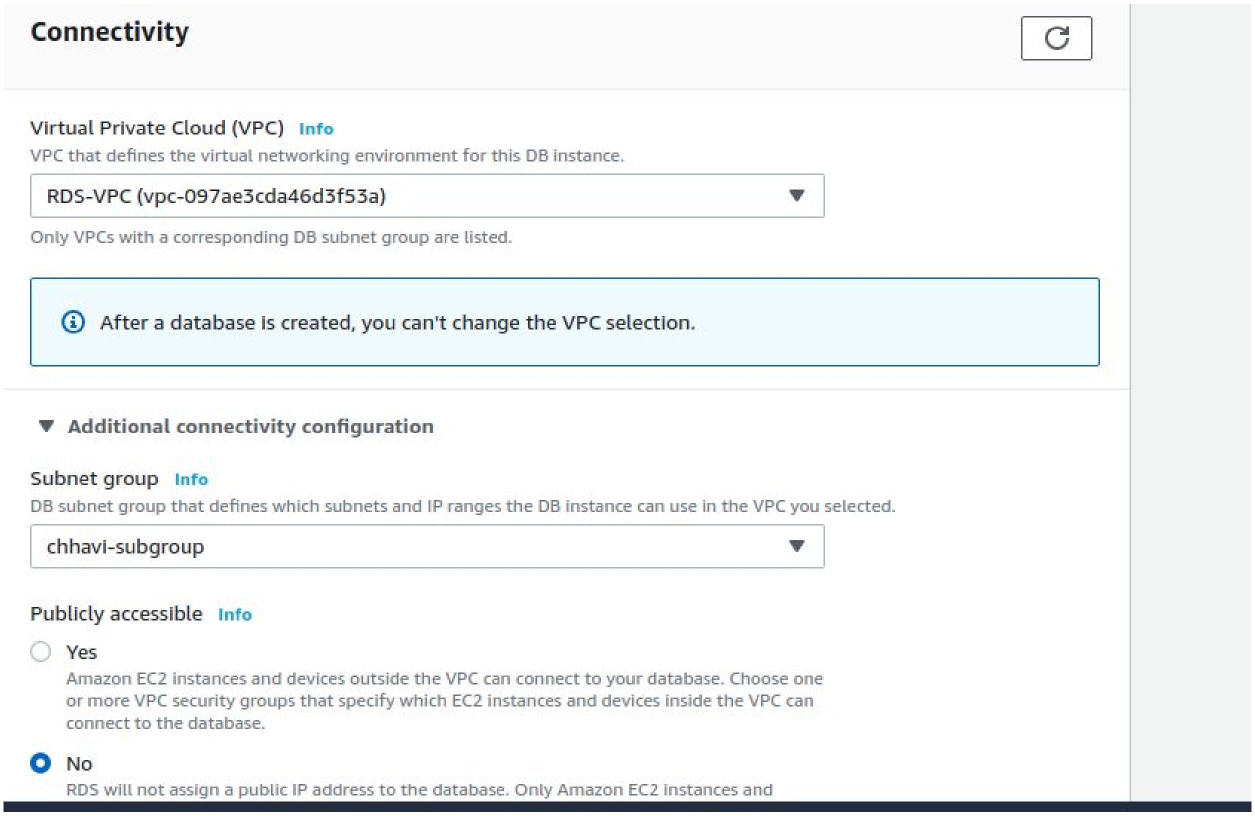
Minimum: 21 GiB, Maximum: 16384 GiB

Availability & durability

Multi-AZ deployment Info

○ Create a standby instance (recommended for production usage)
Creates a standby in a different Availability Zone (AZ) to provide data redundancy, eliminate I/O freezes, and minimize latency spikes during system backups.

○ Do not create a standby instance



Connectivity

Virtual Private Cloud (VPC) Info

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

RDS-VPC (vpc-097ae3cda46d3f53a) ▼

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

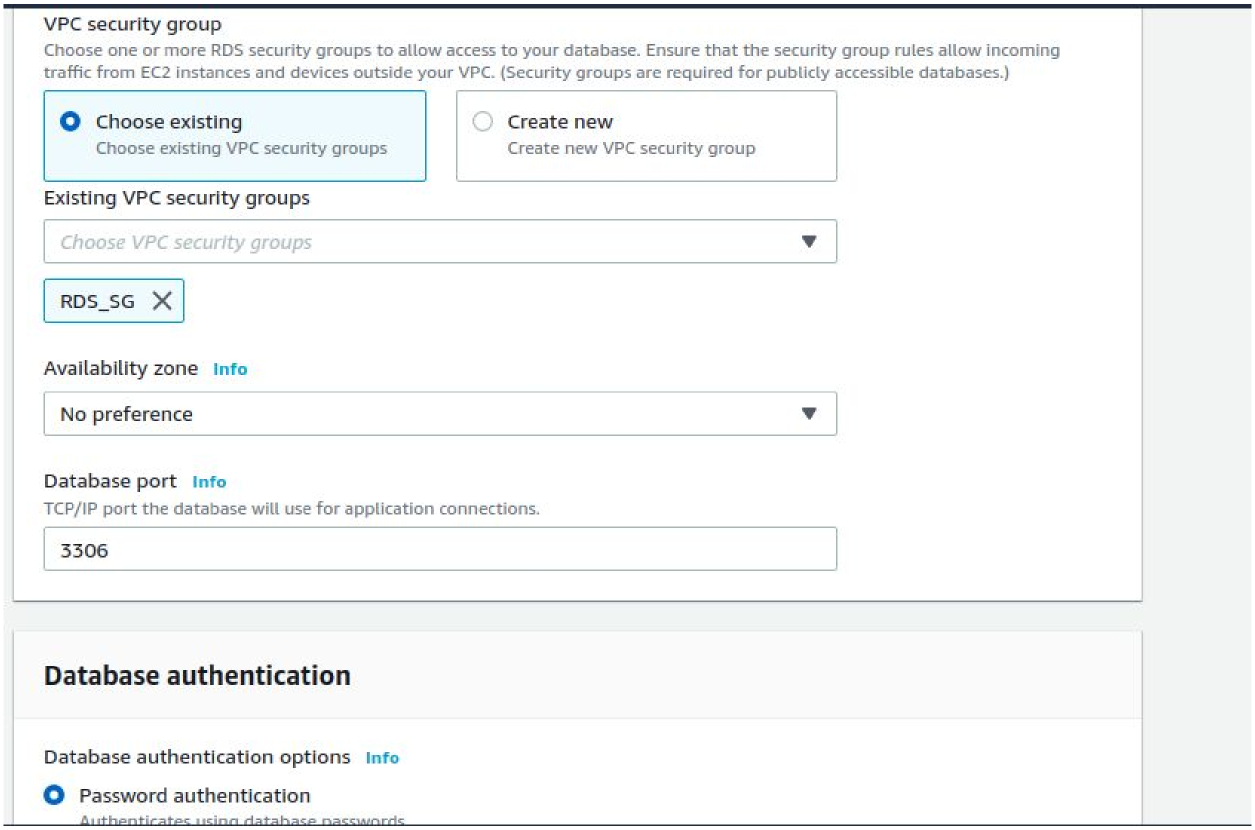
DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

chhavi-subgroup ▼

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



VPC security group

Choose one or more RDS security groups to allow access to your database. Ensure that the security group rules allow incoming traffic from EC2 instances and devices outside your VPC. (Security groups are required for publicly accessible databases.)

● Choose existing
Choose existing VPC security groups

○ Create new
Create new VPC security group

Existing VPC security groups

Choose VPC security groups ▼

RDS_SG ✕

Availability zone Info

No preference ▼

Database port Info

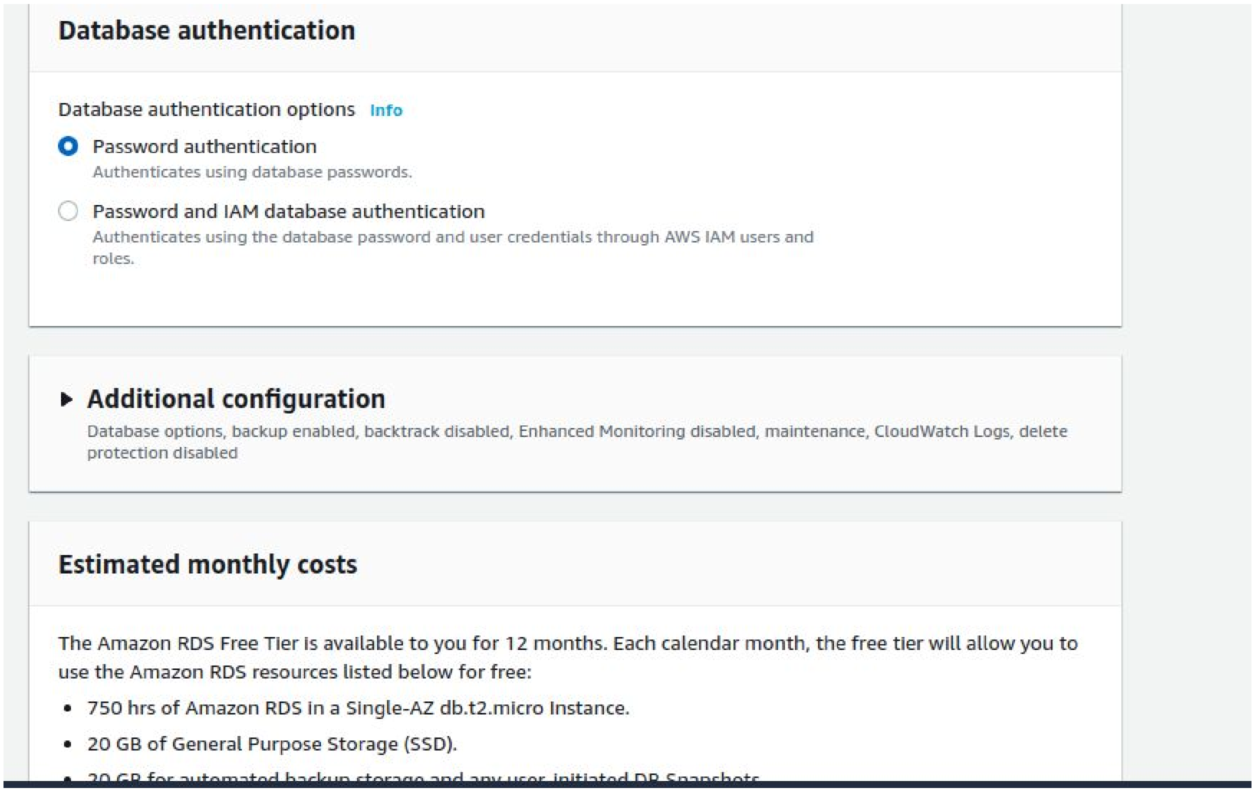
TCP/IP port the database will use for application connections.

3306

Database authentication

Database authentication options Info

● Password authentication
Authenticates using database passwords



Database authentication

Database authentication options Info

● Password authentication
Authenticates using database passwords.

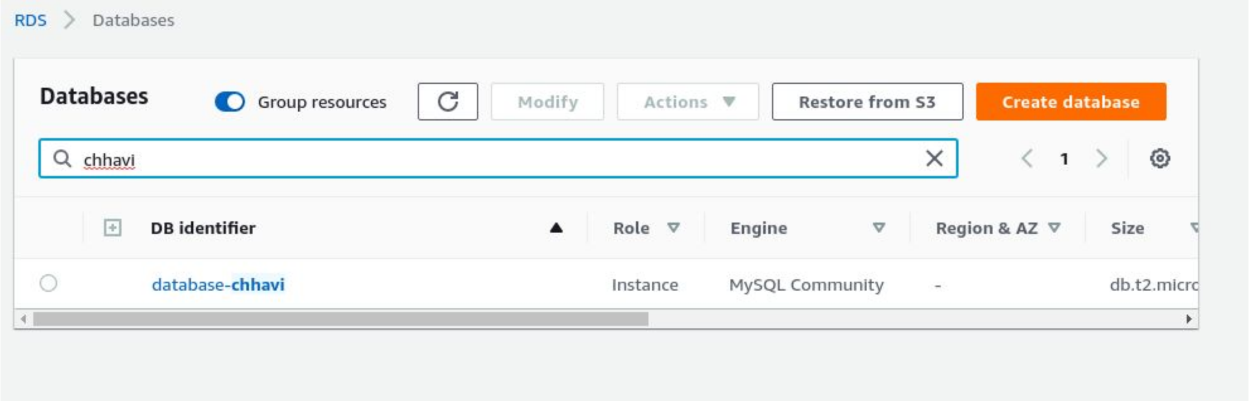
○ Password and IAM database authentication
Authenticates using the database password and user credentials through AWS IAM users and roles.

▸ **Additional configuration**
Database options, backup enabled, backtrack disabled, Enhanced Monitoring disabled, maintenance, CloudWatch Logs, delete protection disabled

Estimated monthly costs

The Amazon RDS Free Tier is available to you for 12 months. Each calendar month, the free tier will allow you to use the Amazon RDS resources listed below for free:

- 750 hrs of Amazon RDS in a Single-AZ db.t2.micro Instance.
- 20 GB of General Purpose Storage (SSD).
- 20 GB for automated backup storage and any user-initiated DB Snapshots



DB identifier ▲	Role	Engine	Region & AZ	Size
○ database-chhavi	Instance	MySQL Community	-	db.t2.micro

Parameter group :For AWS RDS instances, you manage your database engine configuration through the use of parameters in a DB parameter group. DB parameter groups act as a container for engine configuration values that are applied to one or more DB instances.
Option Group: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.
Amazon RDS supports options for the following database engines:
1.MariaDB
2.Microsoft SQL Server
3.MySQL
4.Oracle

5. ACL, Bucket policy, IAM Policy.
Ans.
ACL : Access Control Lists
ACLs are used to define other users' access permissions for your file and folder objects. The Access Permissions that you set using the ACL determine what a user can and cannot do with your file and folder objects. For example, you can set permissions on a file object to let one user read the contents of a file (read access) and let another user make changes to the file (write access). In Amazon S3 you will first add grants to objects and then set the permissions for the grant.
There are 4 types of grants:
1. **An Owner grant** - which defines the permissions the owner of the object has.
2. **Authenticated Users** – which are all Amazon S3 storage users that have an account with S3.
3. **Public** – which means any anonymous user that you have provided the URL to.
4. **Email-ID** – which is an email address of specific S3 customers that have S3 accounts, not

general public emails. The email given must match exactly the email address the S3 user signed up with and can only match one user account.

Bucket Policy

Bucket Policies are similar to IAM policies in that they allow access to resources via a JSON script. However, Bucket policies are applied to Buckets in S3, where as IAM policies are assigned to user/groups/roles and are used to govern access to any AWS resource through the IAM service. When a bucket policy is applied the permissions assigned apply to all objects within the Bucket. The policy will specify which ‘principles’ (users) are allowed to access which resources. The use of Principles within a Bucket policy differs from IAM policies, Principles within IAM policies are defined by who is associated to that policy via the user and group element. As Bucket policies are assigned to Buckets, there is this need of an additional requirement of ‘Principles’.

IAM Policy

A policy is an entity that, when attached to an identity or resource, defines their permissions. A policy that is attached to an identity in IAM is known as an identity-based policy. Identity-based policies can include AWS managed policies, customer managed policies, and inline policies. AWS managed policies are created and managed by AWS. You can use them, but you can’t manage them. An inline policy is one that you create and embed directly to an IAM group, user, or role. Inline policies can’t be reused on other identities or managed outside of the identity where it exists.

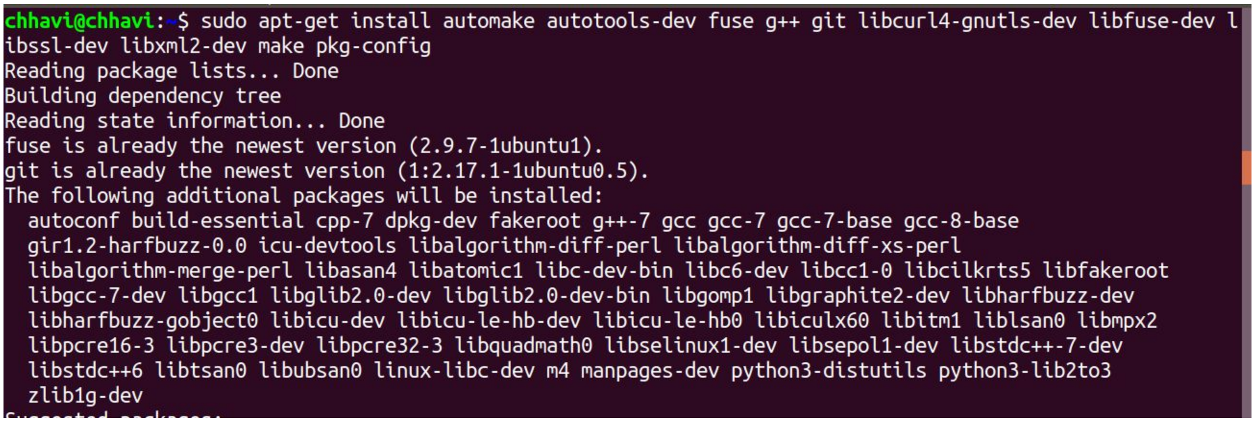
6. Mount S3 to an EC2 instance.

Ans.

A S3 bucket can be mounted in a AWS instance as a file system known as S3fs. S3fs is a FUSE file-system that allows you to mount an Amazon S3 bucket as a local file-system. It behaves like a network attached drive, as it does not store anything on the Amazon EC2, but users can access the data on S3 from EC2 instance.

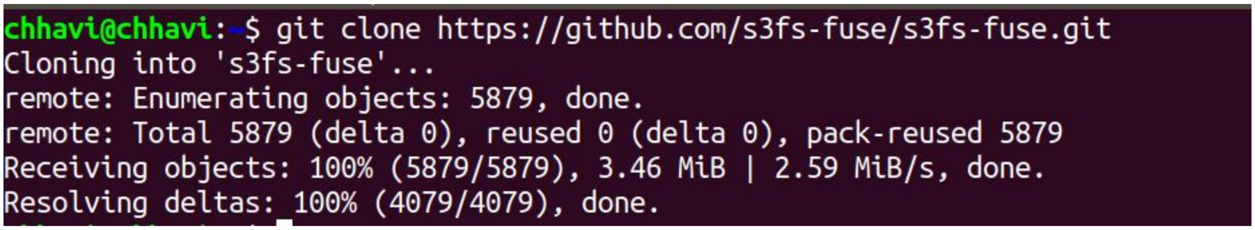
Filesystem in Userspace (FUSE) is a simple interface for userspace programs to export a virtual file-system to the Linux kernel. It also aims to provide a secure method for non privileged users to create and mount their own file-system implementations.

Step 1 : Install the dependencies.



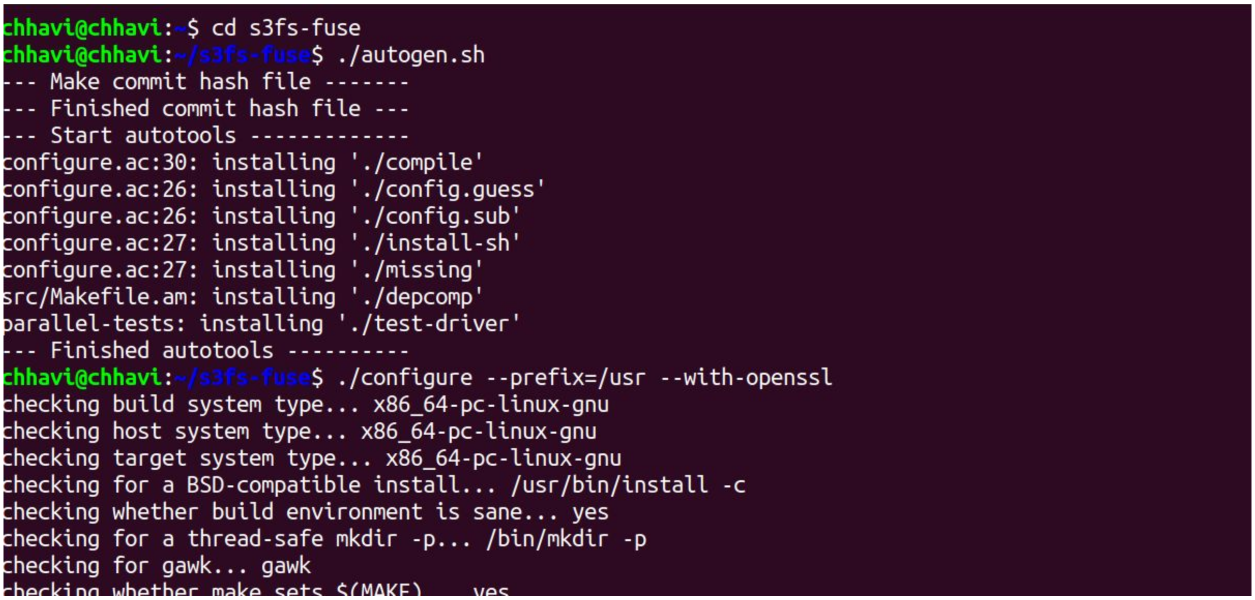
```
chhavi@chhavi:~$ sudo apt-get install automake autotools-dev fuse g++ git libcurl4-gnutls-dev libfuse-dev l
ibssl-dev libxml2-dev make pkg-config
Reading package lists... Done
Building dependency tree
Reading state information... Done
fuse is already the newest version (2.9.7-1ubuntu1).
git is already the newest version (1:2.17.1-1ubuntu0.5).
The following additional packages will be installed:
  autoconf build-essential cpp-7 dpkg-dev fakeroot g++-7 gcc gcc-7 gcc-7-base gcc-8-base
  gir1.2-harfbuzz-0.0 icu-devtools libalgorithm-diff-perl libalgorithm-diff-xs-perl
  libalgorithm-merge-perl libasan4 libatomic1 libc-dev-bin libc6-dev libcc1-0 libcilkrts5 libfakeroot
  libgcc-7-dev libgcc1 libglib2.0-dev libglib2.0-dev-bin libgomp1 libgraphite2-dev libharfbuzz-dev
  libharfbuzz-gobject0 libicu-dev libicu-le-hb-dev libicu-le-hb0 libiculx60 libitm1 liblsan0 libmpx2
  libpcre16-3 libpcre3-dev libpcre32-3 libquadmath0 libselinux1-dev libsepol1-dev libstdc++-7-dev
  libstdc++6 libtsan0 libubsan0 linux-libc-dev m4 manpages-dev python3-distutils python3-lib2to3
  zlib1g-dev
Suggested packages:
```

Step 2 : Clone S3fs source code from git.

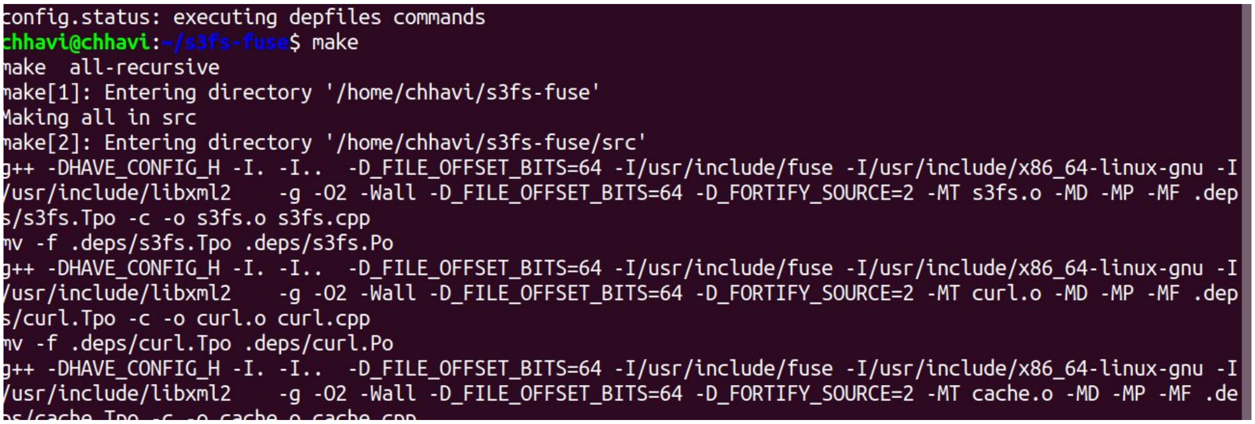


```
chhavi@chhavi:~$ git clone https://github.com/s3fs-fuse/s3fs-fuse.git
Cloning into 's3fs-fuse'...
remote: Enumerating objects: 5879, done.
remote: Total 5879 (delta 0), reused 0 (delta 0), pack-reused 5879
Receiving objects: 100% (5879/5879), 3.46 MiB | 2.59 MiB/s, done.
Resolving deltas: 100% (4079/4079), done.
```

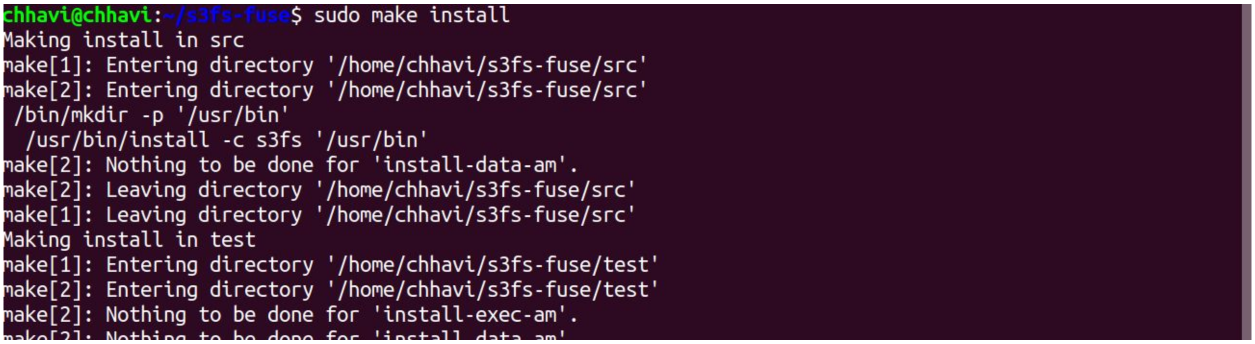
Step 3 : Now change to source code directory, and compile and install the code .



```
chhavi@chhavi:~$ cd s3fs-fuse
chhavi@chhavi:~/s3fs-fuse$ ./autogen.sh
--- Make commit hash file -------
--- Finished commit hash file ---
--- Start autotools -------------
configure.ac:30: installing './compile'
configure.ac:26: installing './config.guess'
configure.ac:26: installing './config.sub'
configure.ac:27: installing './install-sh'
configure.ac:27: installing './missing'
src/Makefile.am: installing './depcomp'
parallel-tests: installing './test-driver'
--- Finished autotools ----------
chhavi@chhavi:~/s3fs-fuse$ ./configure --prefix=/usr --with-openssl
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
checking target system type... x86_64-pc-linux-gnu
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
```

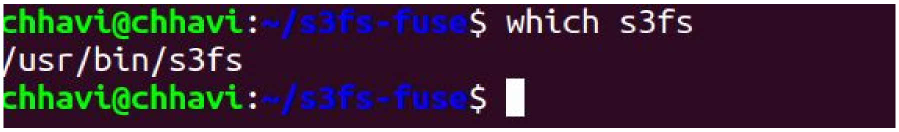


```
config.status: executing depfiles commands
chhavi@chhavi:~/s3fs-fuse$ make
make  all-recursive
make[1]: Entering directory '/home/chhavi/s3fs-fuse'
Making all in src
make[2]: Entering directory '/home/chhavi/s3fs-fuse/src'
g++ -DHAVE_CONFIG_H -I. -I..  -D_FILE_OFFSET_BITS=64 -I/usr/include/fuse -I/usr/include/x86_64-linux-gnu -I
/usr/include/libxml2    -g -O2 -Wall -D_FILE_OFFSET_BITS=64 -D_FORTIFY_SOURCE=2 -MT s3fs.o -MD -MP -MF .dep
s/s3fs.Tpo -c -o s3fs.o s3fs.cpp
mv -f .deps/s3fs.Tpo .deps/s3fs.Po
g++ -DHAVE_CONFIG_H -I. -I..  -D_FILE_OFFSET_BITS=64 -I/usr/include/fuse -I/usr/include/x86_64-linux-gnu -I
/usr/include/libxml2    -g -O2 -Wall -D_FILE_OFFSET_BITS=64 -D_FORTIFY_SOURCE=2 -MT curl.o -MD -MP -MF .dep
s/curl.Tpo -c -o curl.o curl.cpp
mv -f .deps/curl.Tpo .deps/curl.Po
g++ -DHAVE_CONFIG_H -I. -I..  -D_FILE_OFFSET_BITS=64 -I/usr/include/fuse -I/usr/include/x86_64-linux-gnu -I
/usr/include/libxml2    -g -O2 -Wall -D_FILE_OFFSET_BITS=64 -D_FORTIFY_SOURCE=2 -MT cache.o -MD -MP -MF .de
ps/cache.Tpo -c -o cache.o cache.cpp
```



```
chhavi@chhavi:~/s3fs-fuse$ sudo make install
Making install in src
make[1]: Entering directory '/home/chhavi/s3fs-fuse/src'
make[2]: Entering directory '/home/chhavi/s3fs-fuse/src'
 /bin/mkdir -p '/usr/bin'
  /usr/bin/install -c s3fs '/usr/bin'
make[2]: Nothing to be done for 'install-data-am'.
make[2]: Leaving directory '/home/chhavi/s3fs-fuse/src'
make[1]: Leaving directory '/home/chhavi/s3fs-fuse/src'
Making install in test
make[1]: Entering directory '/home/chhavi/s3fs-fuse/test'
make[2]: Entering directory '/home/chhavi/s3fs-fuse/test'
make[2]: Nothing to be done for 'install-exec-am'.
make[2]: Nothing to be done for 'install-data-am'.
```

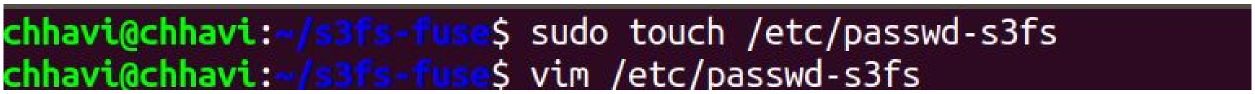
Step 4 : Check where s3fs command is placed in O.S.



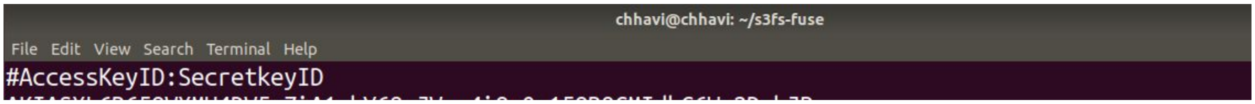
```
chhavi@chhavi:~/s3fs-fuse$ which s3fs
/usr/bin/s3fs
chhavi@chhavi:~/s3fs-fuse$
```

Step 5 : Get the access key and secret key from your aws account.

Step 6 : Create a new file in /etc with the name passwd-s3fs and Paste the access key and secret key .



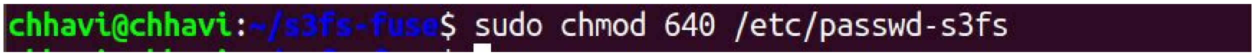
```
chhavi@chhavi:~/s3fs-fuse$ sudo touch /etc/passwd-s3fs
chhavi@chhavi:~/s3fs-fuse$ vim /etc/passwd-s3fs
```



chhavi@chhavi: ~/s3fs-fuse
File Edit View Search Terminal Help

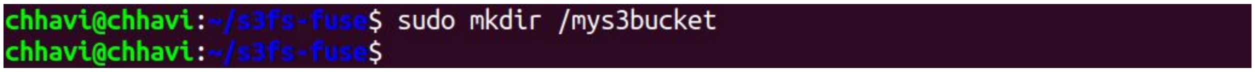
```
#AccessKeyID:SecretkeyID
```

Step 7 : Change the permission of file.

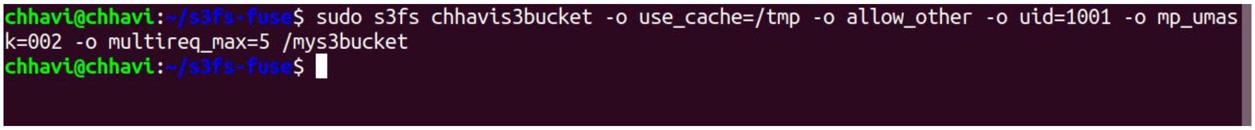


```
chhavi@chhavi:~/s3fs-fuse$ sudo chmod 640 /etc/passwd-s3fs
```

Step 8 :Now create a directory or provide the path of an existing directory and mount S3bucket in it.

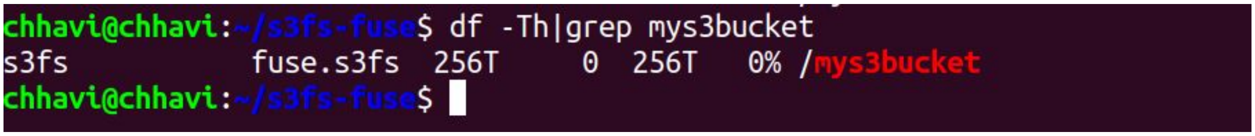


```
chhavi@chhavi:~/s3fs-fuse$ sudo mkdir /mys3bucket
chhavi@chhavi:~/s3fs-fuse$
```



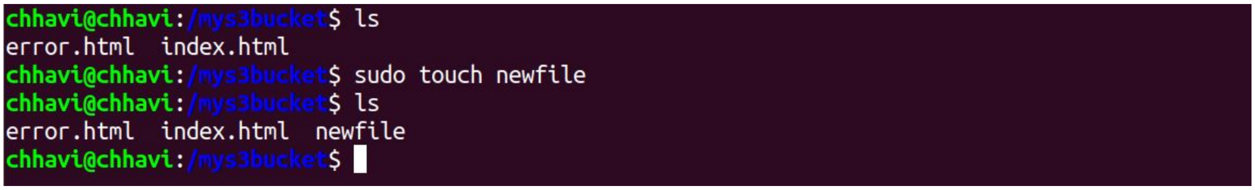
```
chhavi@chhavi:~/s3fs-fuse$ sudo s3fs chhavis3bucket -o use_cache=/tmp -o allow_other -o uid=1001 -o mp_umas
k=002 -o multireq_max=5 /mys3bucket
chhavi@chhavi:~/s3fs-fuse$ 
```

Step 9 : Check the mounted s3 bucket.

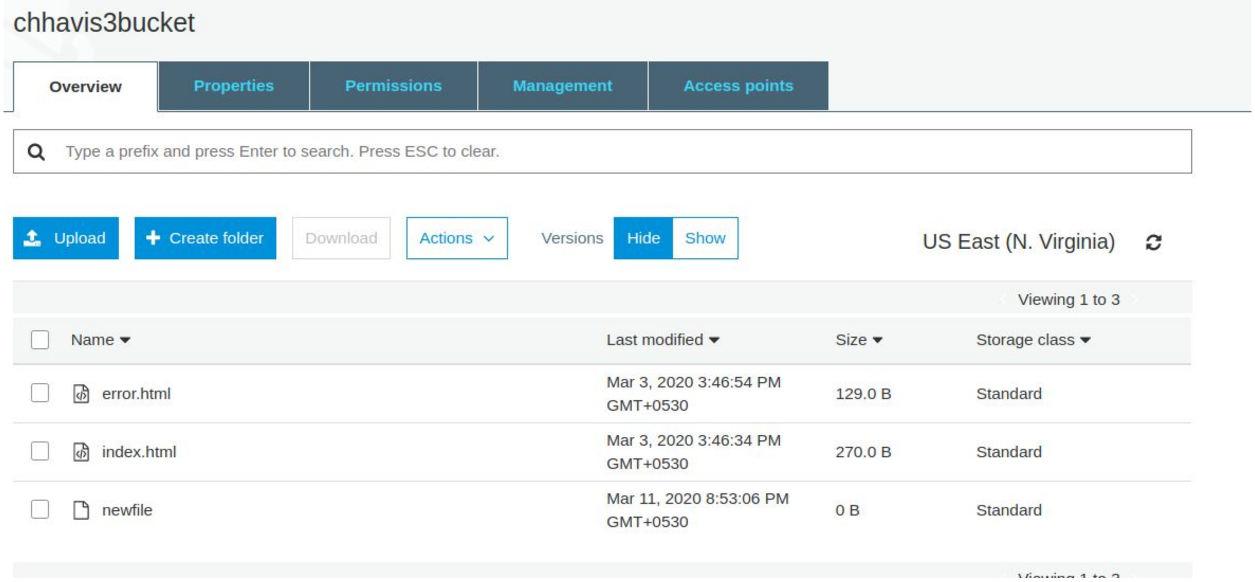


```
chhavi@chhavi:~/s3fs-fuse$ df -Th|grep mys3bucket
s3fs           fuse.s3fs  256T     0  256T    0% /mys3bucket
chhavi@chhavi:~/s3fs-fuse$ 
```

Step 10 :Now we can test this by creating a file locally. This file should also be reflected in your bucket in s3.



```
chhavi@chhavi:/mys3bucket$ ls
error.html  index.html
chhavi@chhavi:/mys3bucket$ sudo touch newfile
chhavi@chhavi:/mys3bucket$ ls
error.html  index.html  newfile
chhavi@chhavi:/mys3bucket$ 
```



chhavis3bucket

US East (N. Virginia)

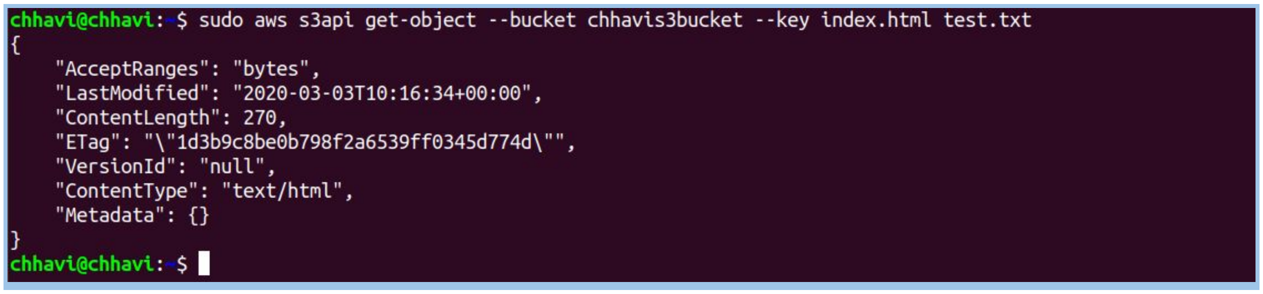
Viewing 1 to 3

	Name	Last modified	Size	Storage class
□	error.html	Mar 3, 2020 3:46:54 PM GMT+0530	129.0 B	Standard
□	index.html	Mar 3, 2020 3:46:34 PM GMT+0530	270.0 B	Standard
□	newfile	Mar 11, 2020 8:53:06 PM GMT+0530	0 B	Standard

7. Change content type using s3.

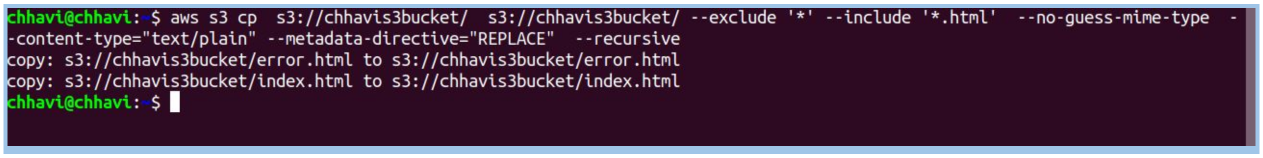
Ans.

Before Changing.



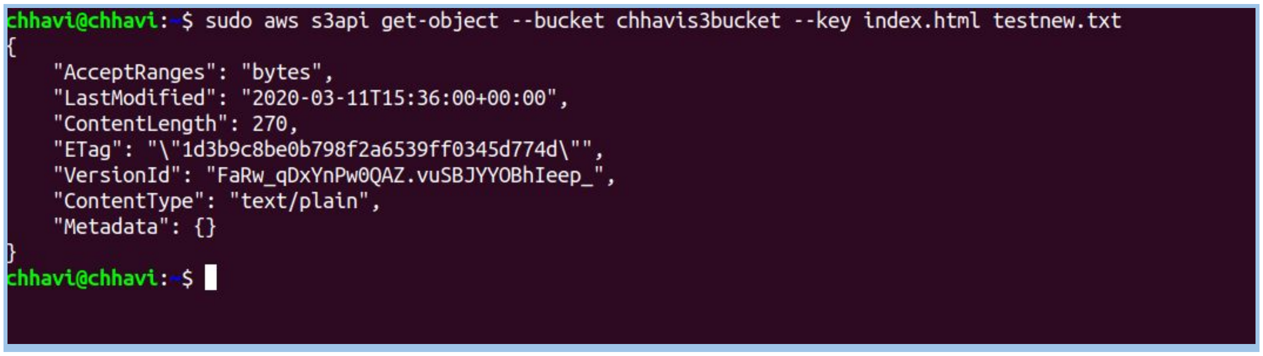
```
chhavi@chhavi:~$ sudo aws s3api get-object --bucket chhavis3bucket --key index.html test.txt
{
    "AcceptRanges": "bytes",
    "LastModified": "2020-03-03T10:16:34+00:00",
    "ContentLength": 270,
    "ETag": "\"1d3b9c8be0b798f2a6539ff0345d774d\"",
    "VersionId": "null",
    "ContentType": "text/html",
    "Metadata": {}
}
chhavi@chhavi:~$ 
```

Changing the content type to text/plain



```
chhavi@chhavi:~$ aws s3 cp  s3://chhavis3bucket/  s3://chhavis3bucket/ --exclude '*' --include '*.html'  --no-guess-mime-type  -
-content-type="text/plain" --metadata-directive="REPLACE"  --recursive
copy: s3://chhavis3bucket/error.html to s3://chhavis3bucket/error.html
copy: s3://chhavis3bucket/index.html to s3://chhavis3bucket/index.html
chhavi@chhavi:~$ 
```

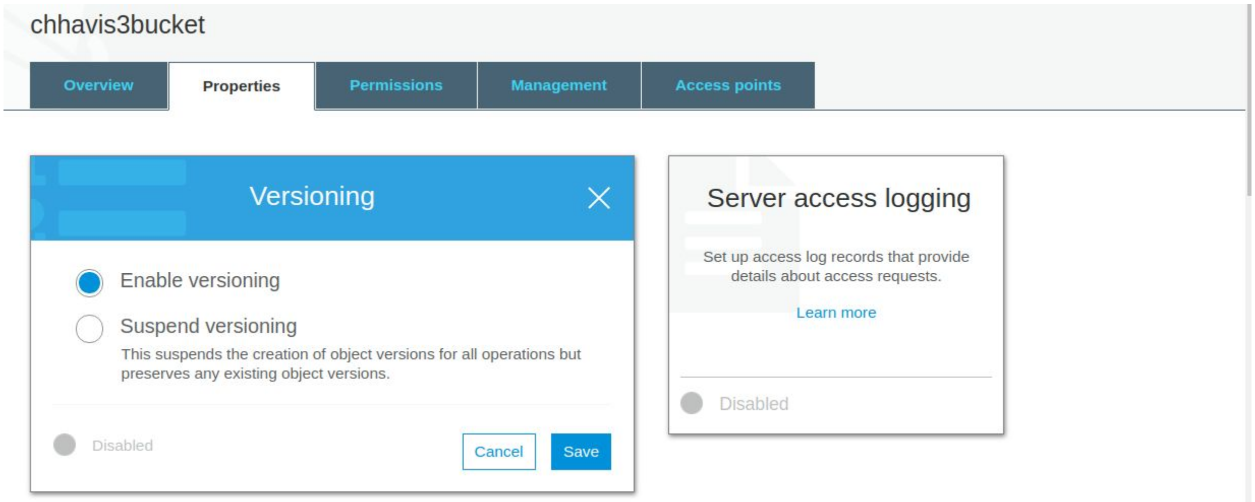
After Changing the content type.



```
chhavi@chhavi:~$ sudo aws s3api get-object --bucket chhavis3bucket --key index.html testnew.txt
{
    "AcceptRanges": "bytes",
    "LastModified": "2020-03-11T15:36:00+00:00",
    "ContentLength": 270,
    "ETag": "\"1d3b9c8be0b798f2a6539ff0345d774d\"",
    "VersionId": "FaRw_qDxYnPw0QAZ.vuSBJYYOBhIeep_",
    "ContentType": "text/plain",
    "Metadata": {}
}
chhavi@chhavi:~$ 
```

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

Ans.



chhavis3bucket

Versioning

● Enable versioning

○ Suspend versioning

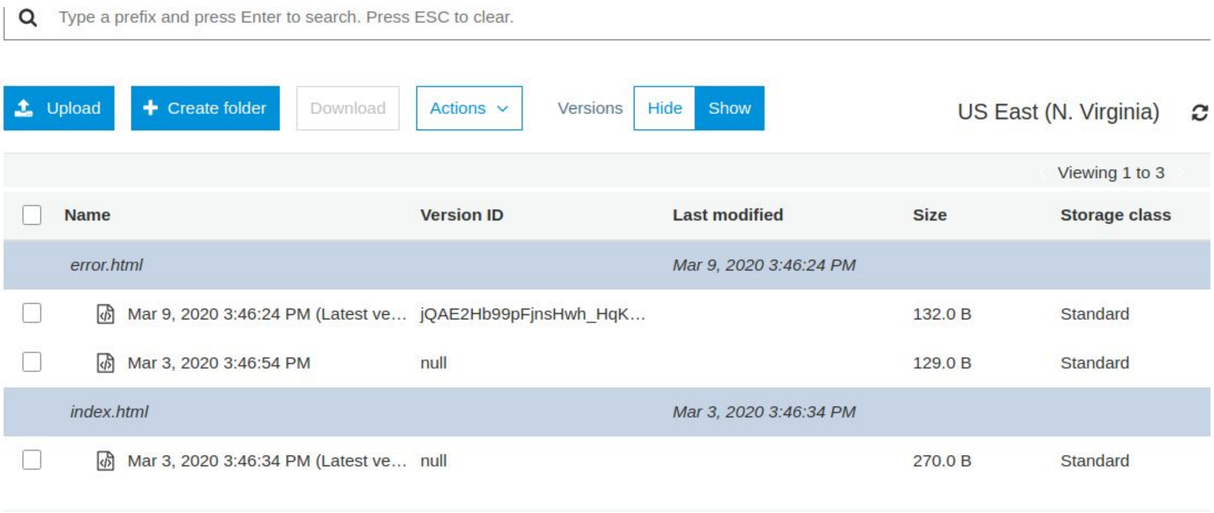
This suspends the creation of object versions for all operations but preserves any existing object versions.

Disabled

Server access logging

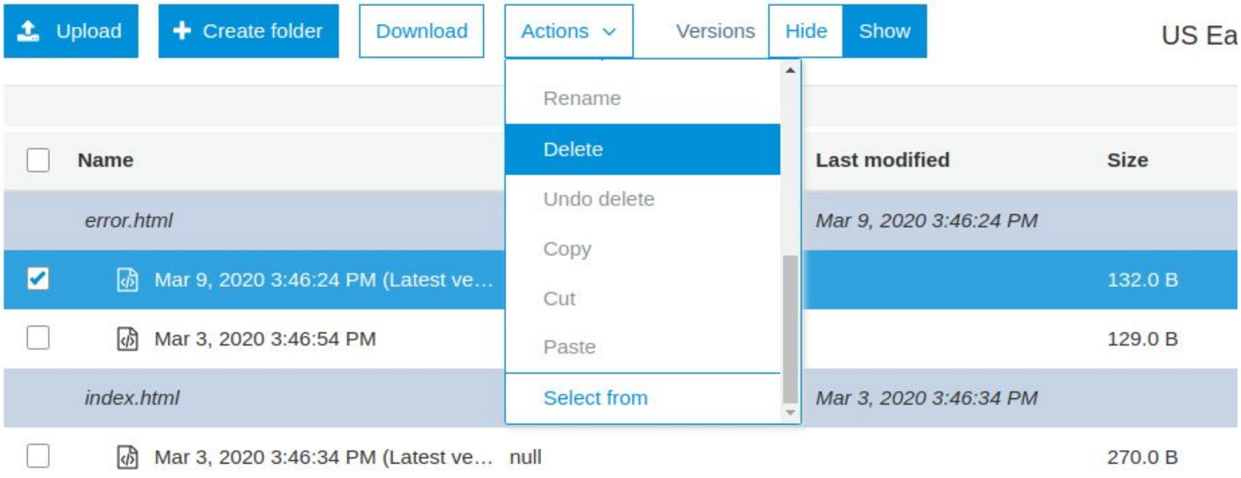
Set up access log records that provide details about access requests.

Disabled



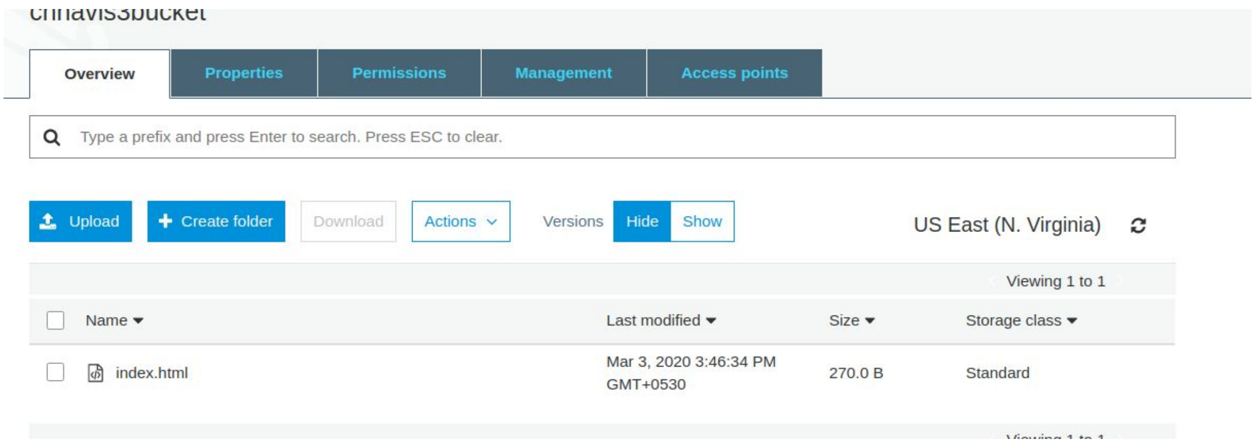
☐	Name	Version ID	Last modified	Size	Storage class
	error.html		*Mar 9, 2020 3:46:24 PM*		
☐	Mar 9, 2020 3:46:24 PM (Latest ve…	jQAE2Hb99pFjnsHwh_HqK…		132.0 B	Standard
☐	Mar 3, 2020 3:46:54 PM	null		129.0 B	Standard
	index.html		*Mar 3, 2020 3:46:34 PM*		
☐	Mar 3, 2020 3:46:34 PM (Latest ve…	null		270.0 B	Standard

Delete the new version to rollback.(You need to delete the delete node as well.)

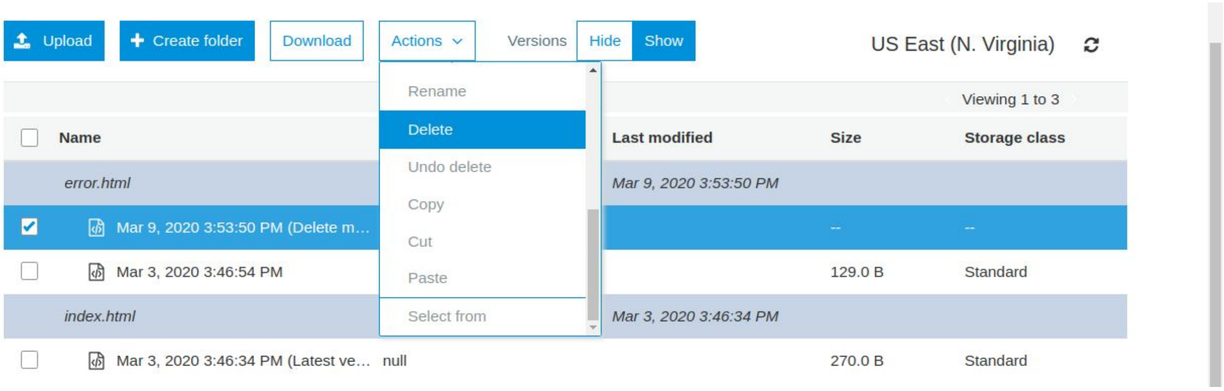


☐	Name	Last modified	Size
	error.html	*Mar 9, 2020 3:46:24 PM*	
☑	Mar 9, 2020 3:46:24 PM (Latest ve…		132.0 B
☐	Mar 3, 2020 3:46:54 PM		129.0 B
	index.html	*Mar 3, 2020 3:46:34 PM*	
☐	Mar 3, 2020 3:46:34 PM (Latest ve… null		270.0 B

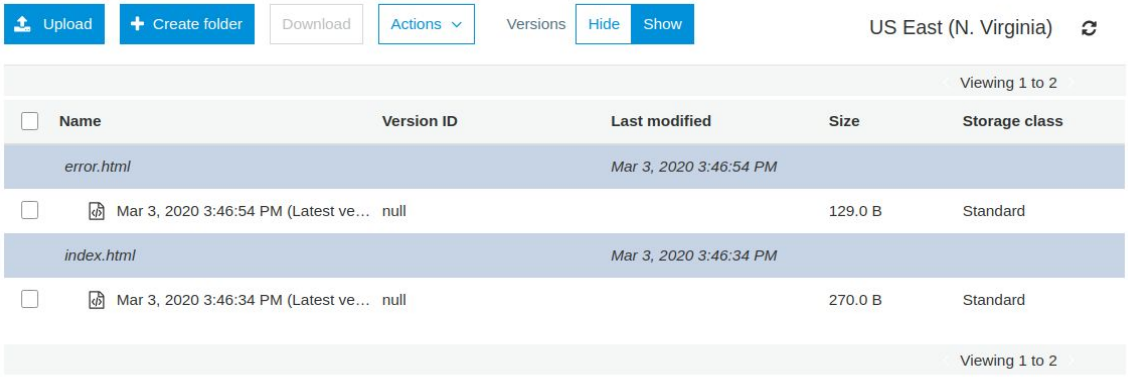
Now if you delete the original file. Show versions. You get the deleted nodes. Delete the new version. You will get back the old version.



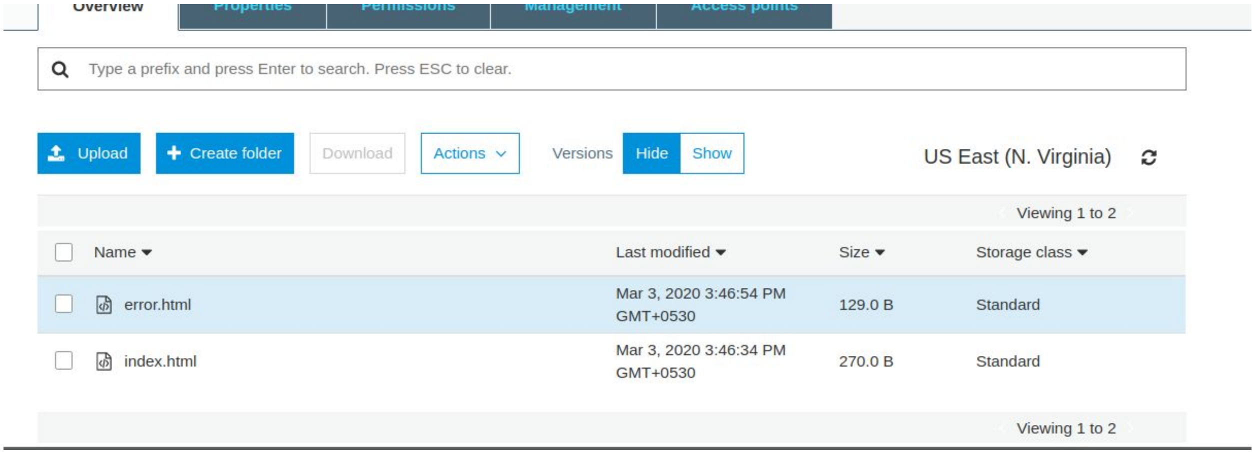
☐	Name	Last modified	Size	Storage class
☐	index.html	Mar 3, 2020 3:46:34 PM GMT+0530	270.0 B	Standard



Name		Last modified	Size	Storage class
error.html		*Mar 9, 2020 3:53:50 PM*		
☑ Mar 9, 2020 3:53:50 PM (Delete m…			--	--
□ Mar 3, 2020 3:46:54 PM			129.0 B	Standard
index.html		*Mar 3, 2020 3:46:34 PM*		
□ Mar 3, 2020 3:46:34 PM (Latest ve…	null		270.0 B	Standard



Name	Version ID	Last modified	Size	Storage class
error.html		*Mar 3, 2020 3:46:54 PM*		
□ Mar 3, 2020 3:46:54 PM (Latest ve…	null		129.0 B	Standard
index.html		*Mar 3, 2020 3:46:34 PM*		
□ Mar 3, 2020 3:46:34 PM (Latest ve…	null		270.0 B	Standard

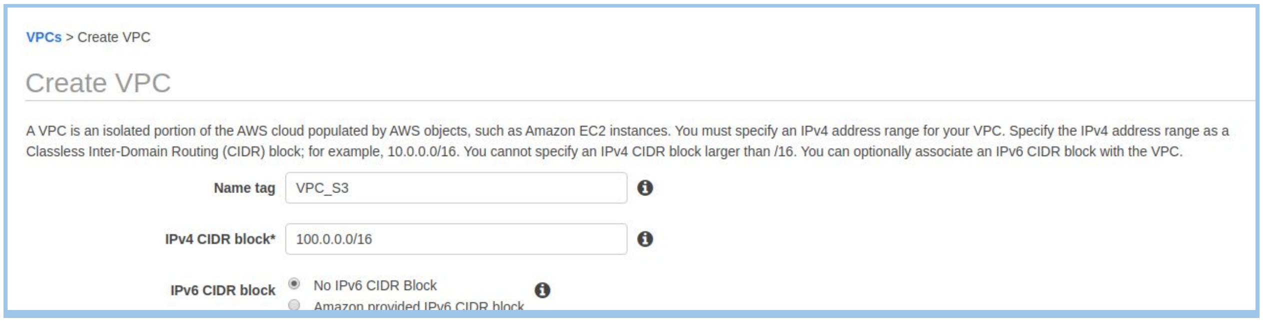


Name	Last modified	Size	Storage class
□ error.html	Mar 3, 2020 3:46:54 PM GMT+0530	129.0 B	Standard
□ index.html	Mar 3, 2020 3:46:34 PM GMT+0530	270.0 B	Standard

9. S3 VPC endpoint.

Ans.

Create a VPC with subnets.



VPCs > Create VPC

Create VPC

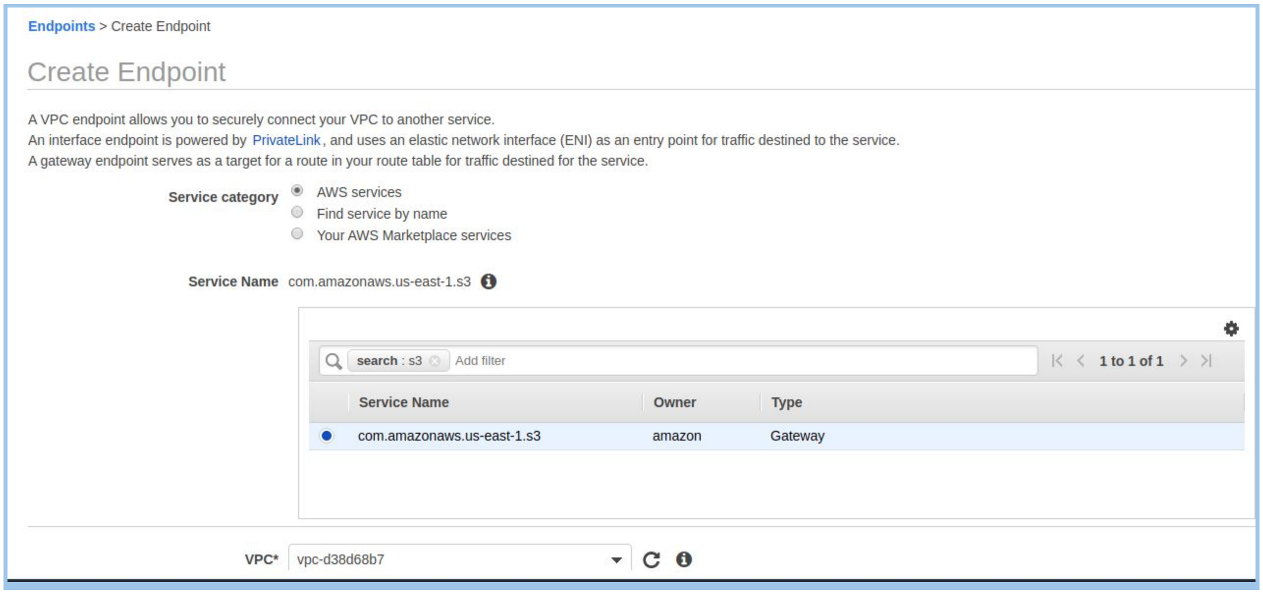
A VPC is an isolated portion of the AWS cloud populated by AWS objects, such as Amazon EC2 instances. You must specify an IPv4 address range for your VPC. Specify the IPv4 address range as a Classless Inter-Domain Routing (CIDR) block; for example, 10.0.0.0/16. You cannot specify an IPv4 CIDR block larger than /16. You can optionally associate an IPv6 CIDR block with the VPC.

Name tag: VPC_S3

IPv4 CIDR block*: 100.0.0.0/16

IPv6 CIDR block: ● No IPv6 CIDR Block
○ Amazon provided IPv6 CIDR block

From the navigation pane, choose Endpoints.-> Create Endpoints.For Service category, verify that AWS services is selected and for the Service Name, select the service name that includes "s3". Service name in the US East (N. Virginia) Region is com.amazonaws.us-east-1.s3.



Endpoints > Create 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

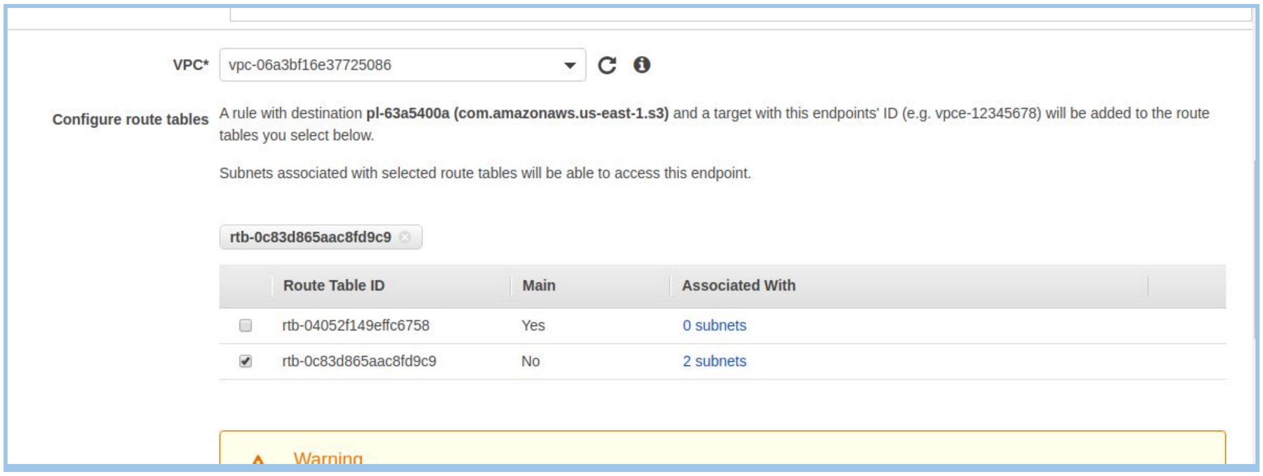
search : s3 Add filter 1 to 1 of 1

	Service Name	Owner	Type
●	com.amazonaws.us-east-1.s3	amazon	Gateway

VPC*: vpc-d38d68b7

select the VPC that you want to use.

For Configure route tables, select the route tables based on the associated subnets that you want to be able to access the endpoint.



VPC* vpc-06a3bf16e37725086

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-12345678) will be added to the route tables you select below.

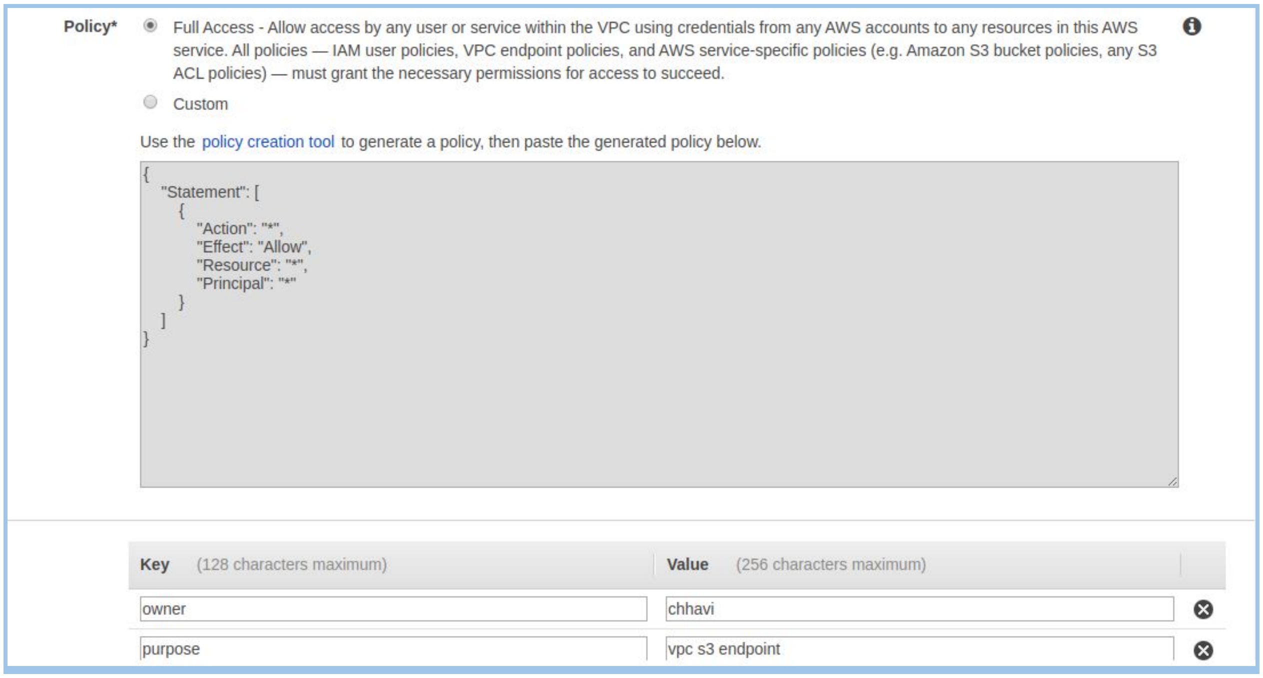
Subnets associated with selected route tables will be able to access this endpoint.

rtb-0c83d865aac8fd9c9

	Route Table ID	Main	Associated With
☐	rtb-04052f149effc6758	Yes	0 subnets
☑	rtb-0c83d865aac8fd9c9	No	2 subnets

Warning

For Policy, verify that Full Access is selected.



Policy* ● Full Access - Allow access by any user or service within the VPC using credentials from any AWS accounts to any resources in this AWS service. All policies — IAM user policies, VPC endpoint policies, and AWS service-specific policies (e.g. Amazon S3 bucket policies, any S3 ACL policies) — must grant the necessary permissions for access to succeed.

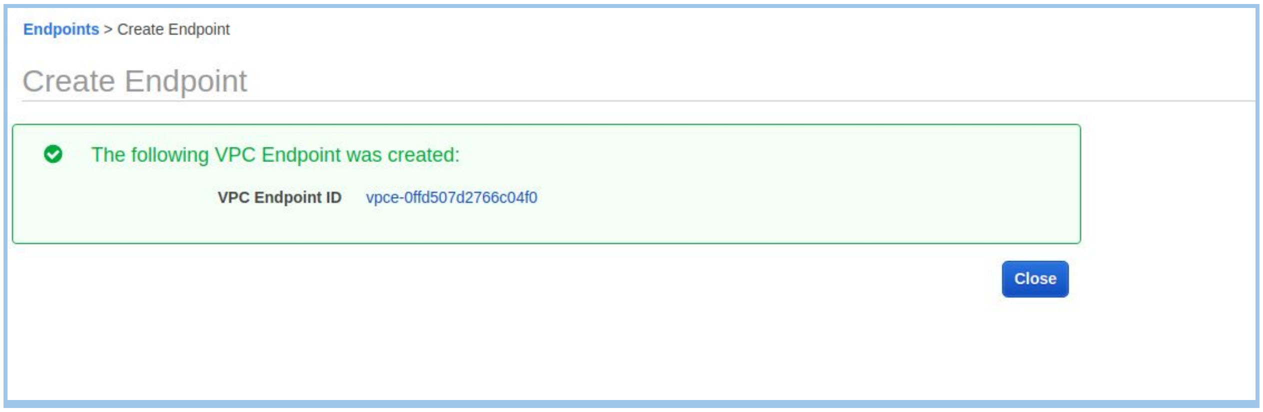
○ Custom

Use the policy creation tool to generate a policy, then paste the generated policy below.

```
{
  "Statement": [
    {
      "Action": "*",
      "Effect": "Allow",
      "Resource": "*",
      "Principal": "*"
    }
  ]
}
```

Key (128 characters maximum)	**Value** (256 characters maximum)
owner	chhavi
purpose	vpc s3 endpoint

Take note of the VPC Endpoint ID. You need this ID for a later step.



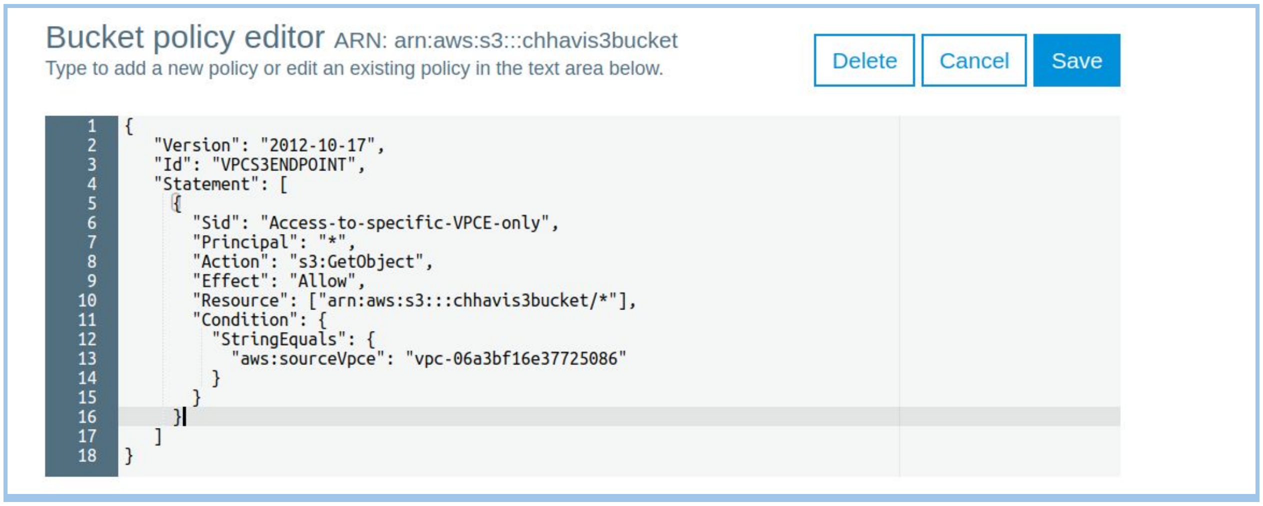
Endpoints > Create Endpoint

Create Endpoint

The following VPC Endpoint was created:

VPC Endpoint ID vpce-0ffd507d2766c04f0

Add a bucket policy that allows access from the VPC endpoint

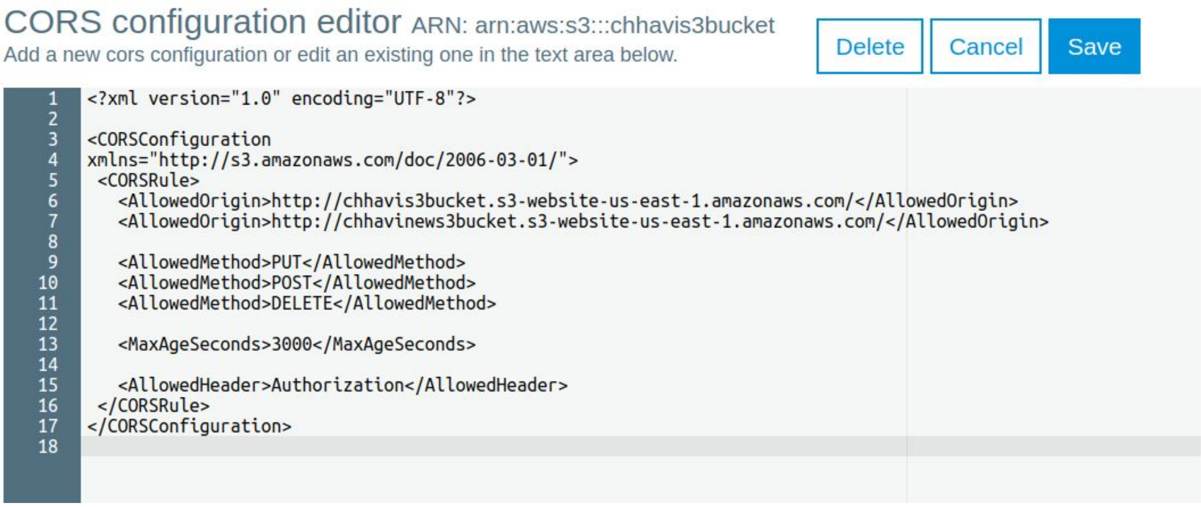

Bucket policy editor ARN: arn:aws:s3:::chhavis3bucket
Type to add a new policy or edit an existing policy in the text area below.
Delete
Cancel
Save

```
{
    "Version": "2012-10-17",
    "Id": "VPCS3ENDPOINT",
    "Statement": [
      {
        "Sid": "Access-to-specific-VPCE-only",
        "Principal": "*",
        "Action": "s3:GetObject",
        "Effect": "Allow",
        "Resource": ["arn:aws:s3:::chhavis3bucket/*"],
        "Condition": {
          "StringEquals": {
            "aws:sourceVpce": "vpc-06a3bf16e37725086"
          }
        }
      }
    ]
}
```

10. CORS, Enable CORS for 2 specific websites.

Ans.

Cross-Origin Resource Sharing (CORS) is a mechanism that uses additional HTTP headers to tell browsers to give a web application running at one origin, access to selected resources from a different origin. A web application executes a cross-origin HTTP request when it requests a resource that has a different origin (domain, protocol, or port) from its own.


CORS configuration editor ARN: arn:aws:s3:::chhavis3bucket
Add a new cors configuration or edit an existing one in the text area below.
Delete
Cancel
Save

```
<?xml version="1.0" encoding="UTF-8"?>

<CORSConfiguration
xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
 <CORSRule>
   <AllowedOrigin>http://chhavis3bucket.s3-website-us-east-1.amazonaws.com/</AllowedOrigin>
   <AllowedOrigin>http://chhavinews3bucket.s3-website-us-east-1.amazonaws.com/</AllowedOrigin>

   <AllowedMethod>PUT</AllowedMethod>
   <AllowedMethod>POST</AllowedMethod>
   <AllowedMethod>DELETE</AllowedMethod>

   <MaxAgeSeconds>3000</MaxAgeSeconds>

   <AllowedHeader>Authorization</AllowedHeader>
 </CORSRule>
</CORSConfiguration>
```