CLOUD SECURITY & MANAGEMENT LAB

Name: Harsh Ranjan

SAP ID: 500097019

Roll no: R2142211262

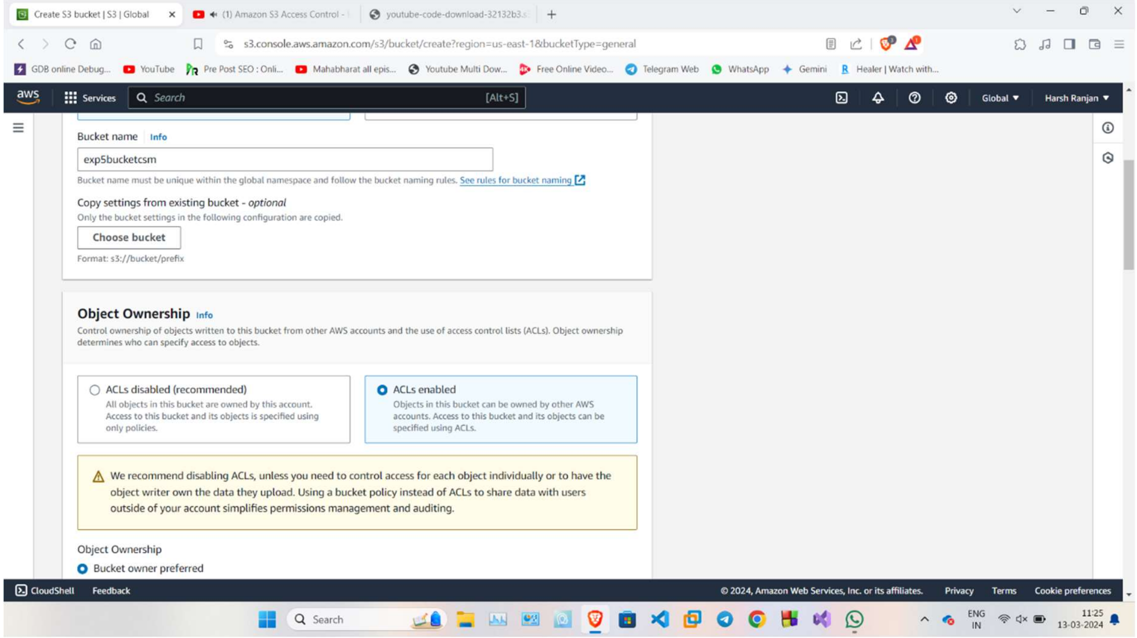
Batch :B7

SUBMISSION TO:

Ms. Avita Katal

EXP 5: To study and implement security features for Storage as a Service.

STEP 1: Click on the create bucket option and give the name of the bucket.



Bucket name Info

exp5bucketcsm

Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Choose bucket

Format: s3://bucket/prefix

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.

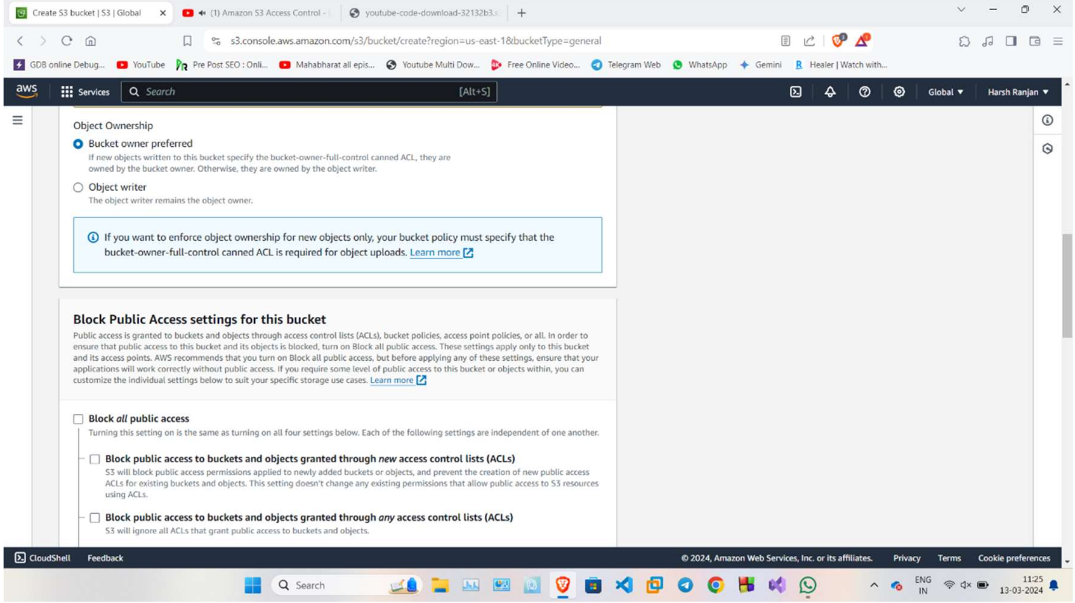
● ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

⚠ 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

STEP 2: Select The ACLs enable Options so that it allow administrators to define fine-grained permissions beyond traditional user-group-world permissions.



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

Block Public Access settings for this bucket

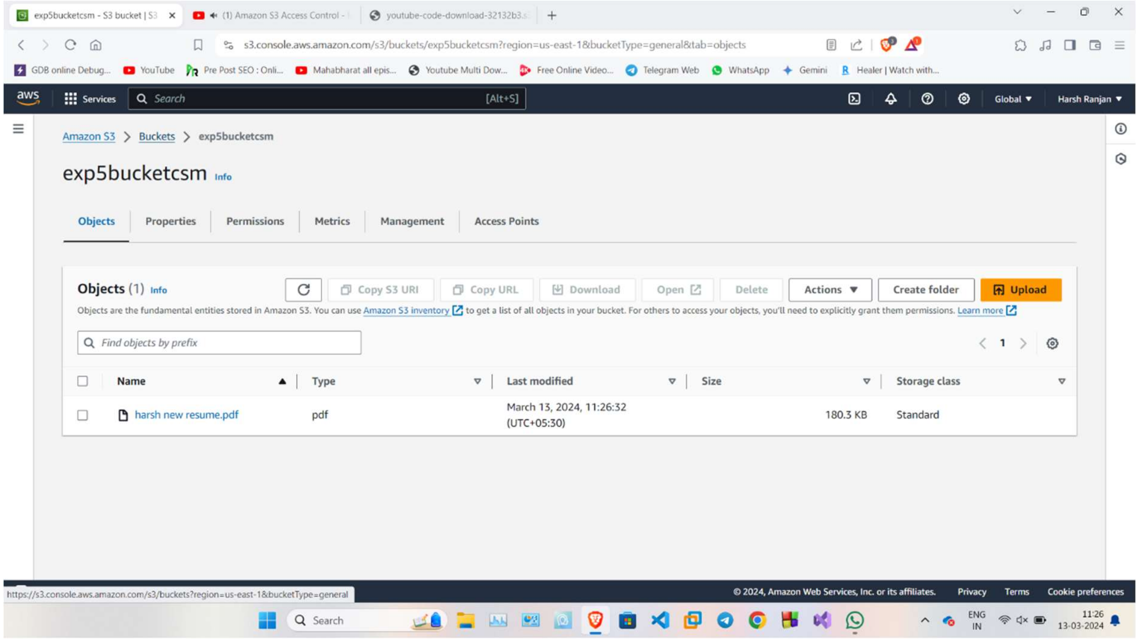
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access 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.

STEP 3: After Creating the Bucket click on the upload option to add files or folder in s3 bucket.



exp5bucketcsm Info

Objects (1) Info

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

Name	Type	Last modified	Size	Storage class
harsh new resume.pdf	pdf	March 13, 2024, 11:26:32 (UTC+05:30)	180.3 KB	Standard

STEP 4: To grant read access to everyone for an object in an S3 bucket, follow these steps:

• Navigate to the S3 management console and click on the bucket containing the object.

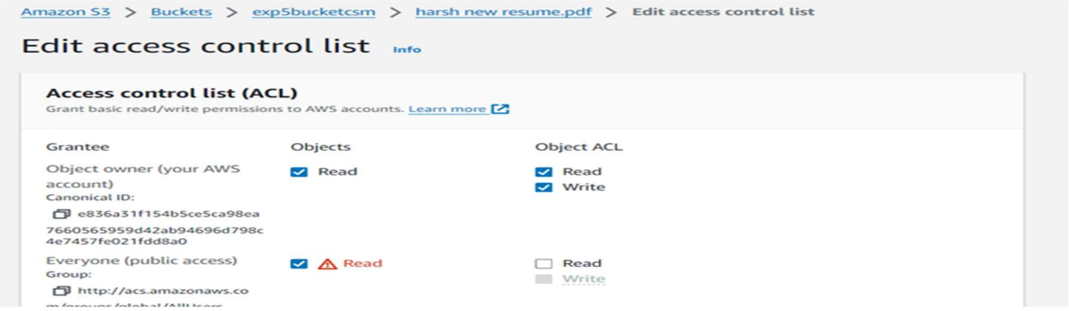
• Locate the object you want to modify and click on it to select it.

• Under the "Access control list (ACL)" section, locate the permission settings.

• Click on the "Edit ACL" button or similar option to modify the ACL settings.

• Ensure the permission for "Read" or "Read object" is enabled for the "Everyone" entity.

• Save your changes.


Amazon S3 > Buckets > exp5bucketcsm > harsh new resume.pdf > Edit access control list

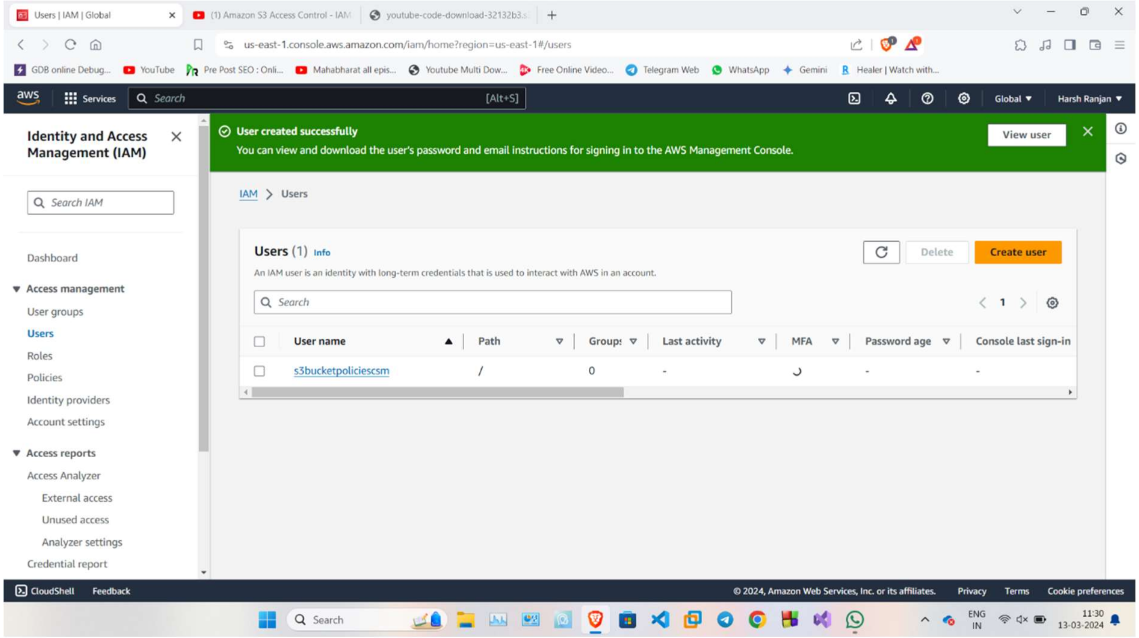
Edit access control list Info

Access control list (ACL)

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

Grantee	Objects	Object ACL
Object owner (your AWS account) Canonical ID: e836a31f154b5ce5ca98ea7660565959d42ab94696d798c4e7457fe021fdd8a0	☑ Read	☑ Read ☑ Write
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	☑ ⚠ Read	□ Read □ Write

STEP 5: Create a IAM User and set the Custom Password.



User created successfully

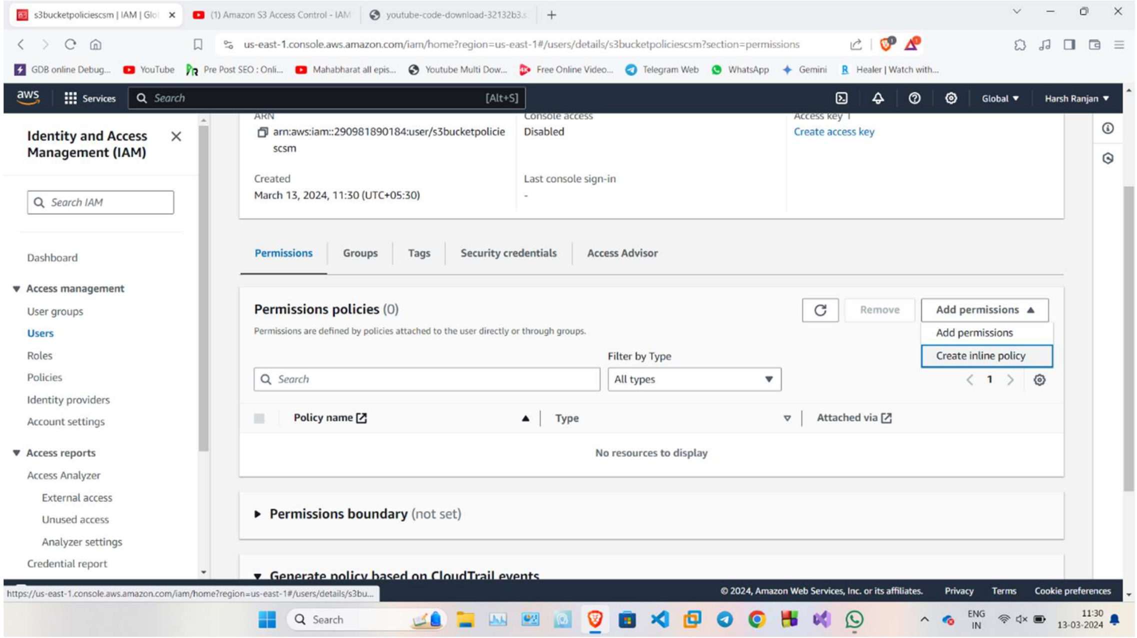
You can view and download the user's password and email instructions for signing in to the AWS Management Console.

Users (1)

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

User name	Path	Groups	Last activity	MFA	Password age	Console last sign-in
s3bucketpoliciescsm	/	0	-		-	-

STEP 6: Add an Inline Policy and give the name of the policy.



ARN: arn:aws:iam::290981890184:user/s3bucketpoliciescsm

Console access: Disabled

Access key 1: Create access key

Created: March 13, 2024, 11:30 (UTC+05:30)

Last console sign-in: -

Permissions policies (0)

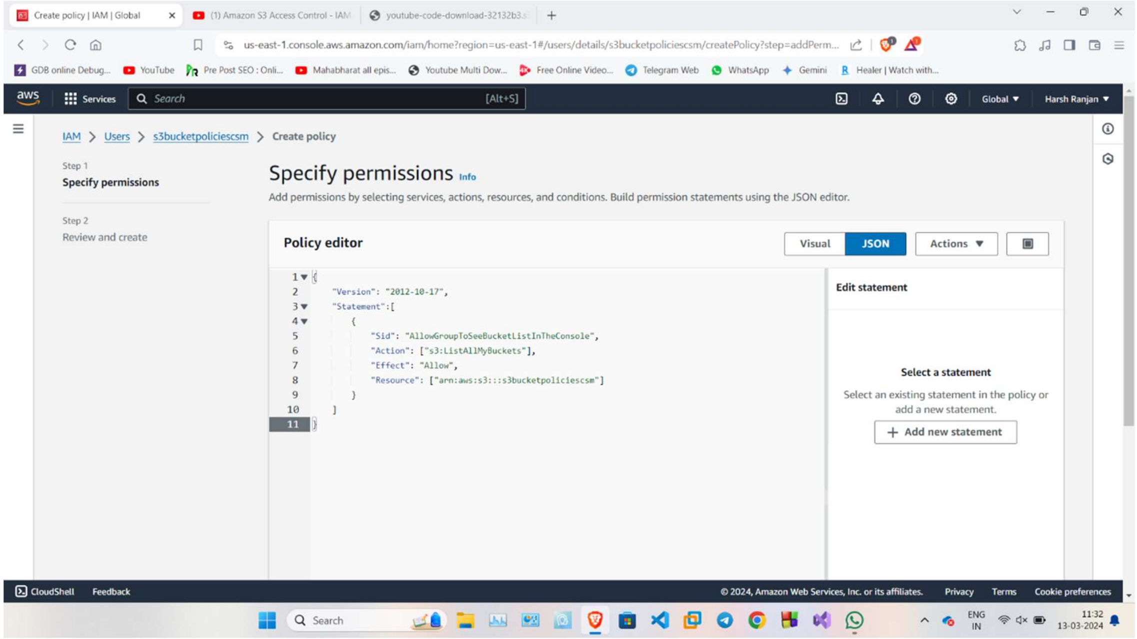
Permissions are defined by policies attached to the user directly or through groups.

Policy name	Type	Attached via

No resources to display

us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/users/details/s3bucketpoliciescsm?section=permissions
Permissions
Groups
Tags
Security credentials
Access Advisor
Remove
Add permissions
Add permissions
Create inline policy
Filter by Type
All types
Permissions boundary (not set)
Generate policy based on CloudTrail events

STEP 7: Attach a user policy which allow users to the list buckets and copy this Json code in the policy editor.



IAM > Users > s3bucketpoliciescsm > Create policy

Specify permissions

Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

Policy editor

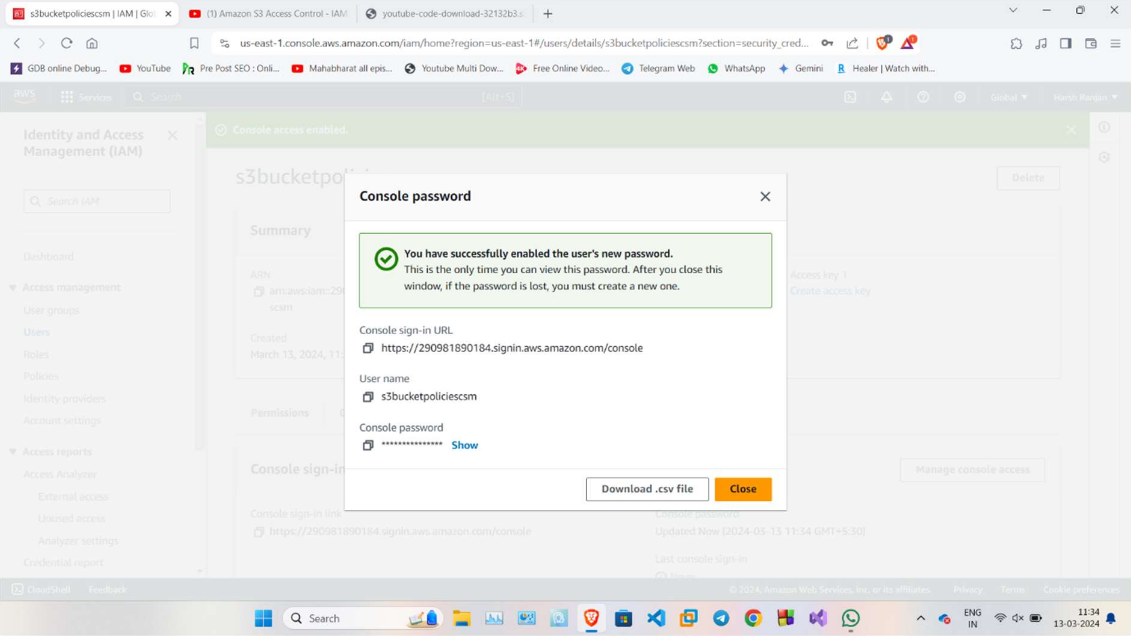
```
{
    "Version": "2012-10-17",
    "Statement":[
        {
            "Sid": "AllowGroupToSeeBucketListInTheConsole",
            "Action": ["s3:ListAllMyBuckets"],
            "Effect": "Allow",
            "Resource": ["arn:aws:s3:::s3bucketpoliciescsm"]
        }
    ]
}
```

Edit statement

Select a statement

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

STEP 8: After changing the permission in the policy Login to the IAM user and refresh the screen we will be able to see the buckets in the IAM user.



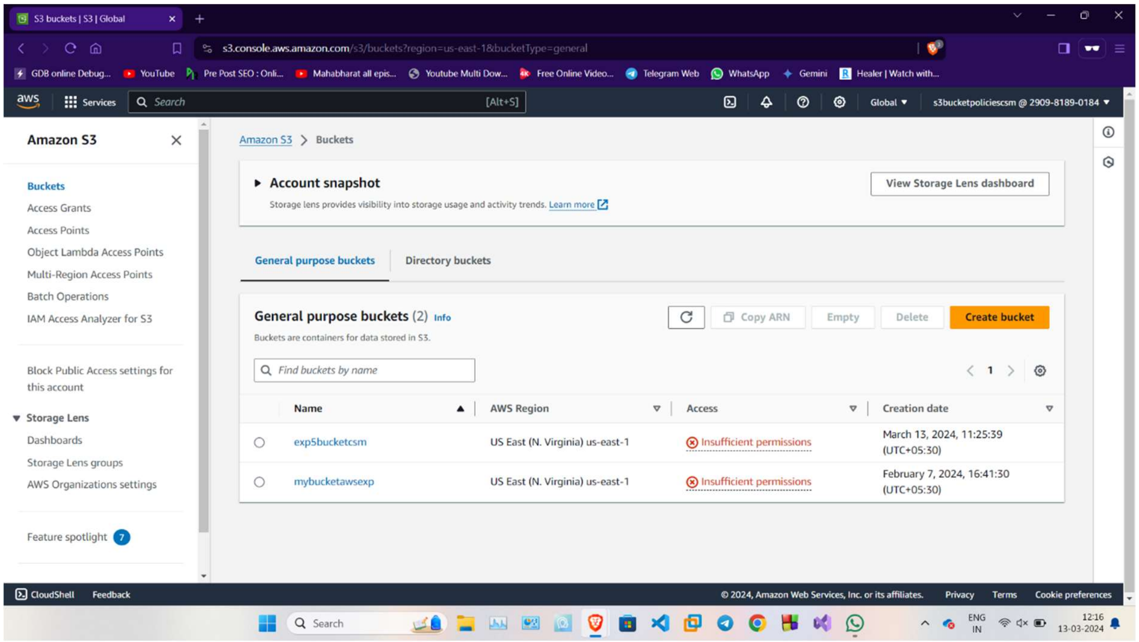
Console password

You have successfully enabled the user's new password.
This is the only time you can view this password. After you close this window, if the password is lost, you must create a new one.

Console sign-in URL
https://290981890184.signin.aws.amazon.com/console

User name
s3bucketpoliciescsm

Console password
************** Show


S3 buckets | S3 | Global
s3.console.aws.amazon.com/s3/buckets?region=us-east-1&bucketType=general
Amazon S3
Buckets
Access Grants
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
Storage Lens groups
AWS Organizations settings
Feature spotlight 7
Amazon S3 > Buckets
Account snapshot
View Storage Lens dashboard
General purpose buckets
Directory buckets
Copy ARN
Empty
Delete
Create bucket
Find buckets by name
s3bucketpoliciescsm @ 2909-8189-0184
Global
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Storage lens provides visibility into storage usage and activity trends. Learn more

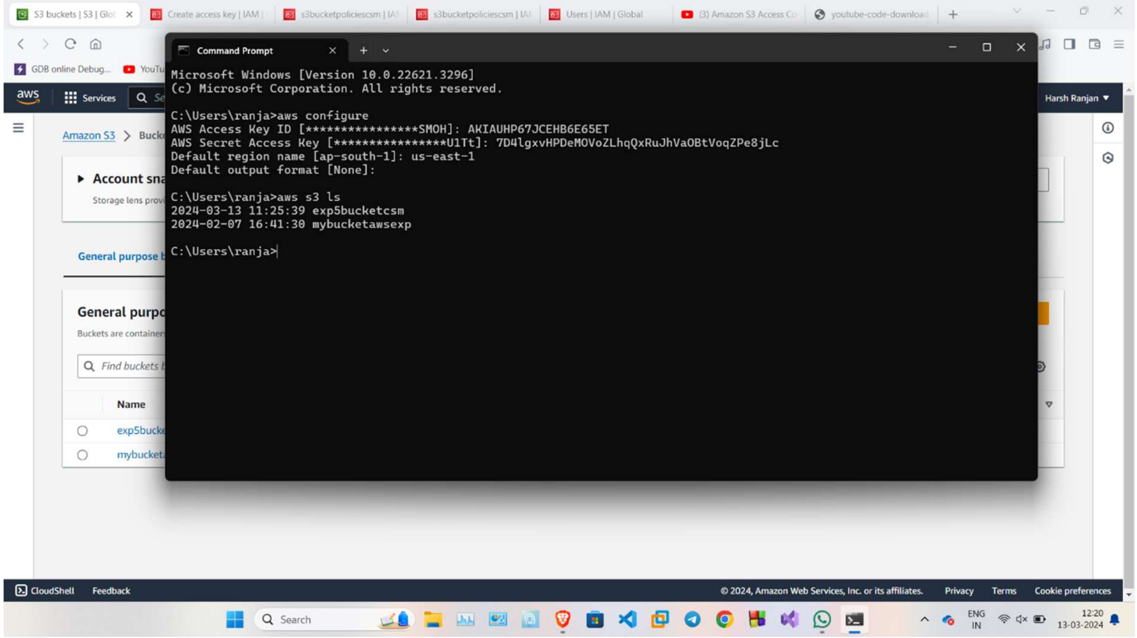
General purpose buckets (2) Info

Buckets are containers for data stored in S3.

	Name	AWS Region	Access	Creation date
○	exp5bucketcsm	US East (N. Virginia) us-east-1	Insufficient permissions	March 13, 2024, 11:25:39 (UTC+05:30)
○	mybucketawsexp	US East (N. Virginia) us-east-1	Insufficient permissions	February 7, 2024, 16:41:30 (UTC+05:30)

STEP 9: Open the cmd and try to list the bucket using aws console command

- aws configure
- aws s3 ls



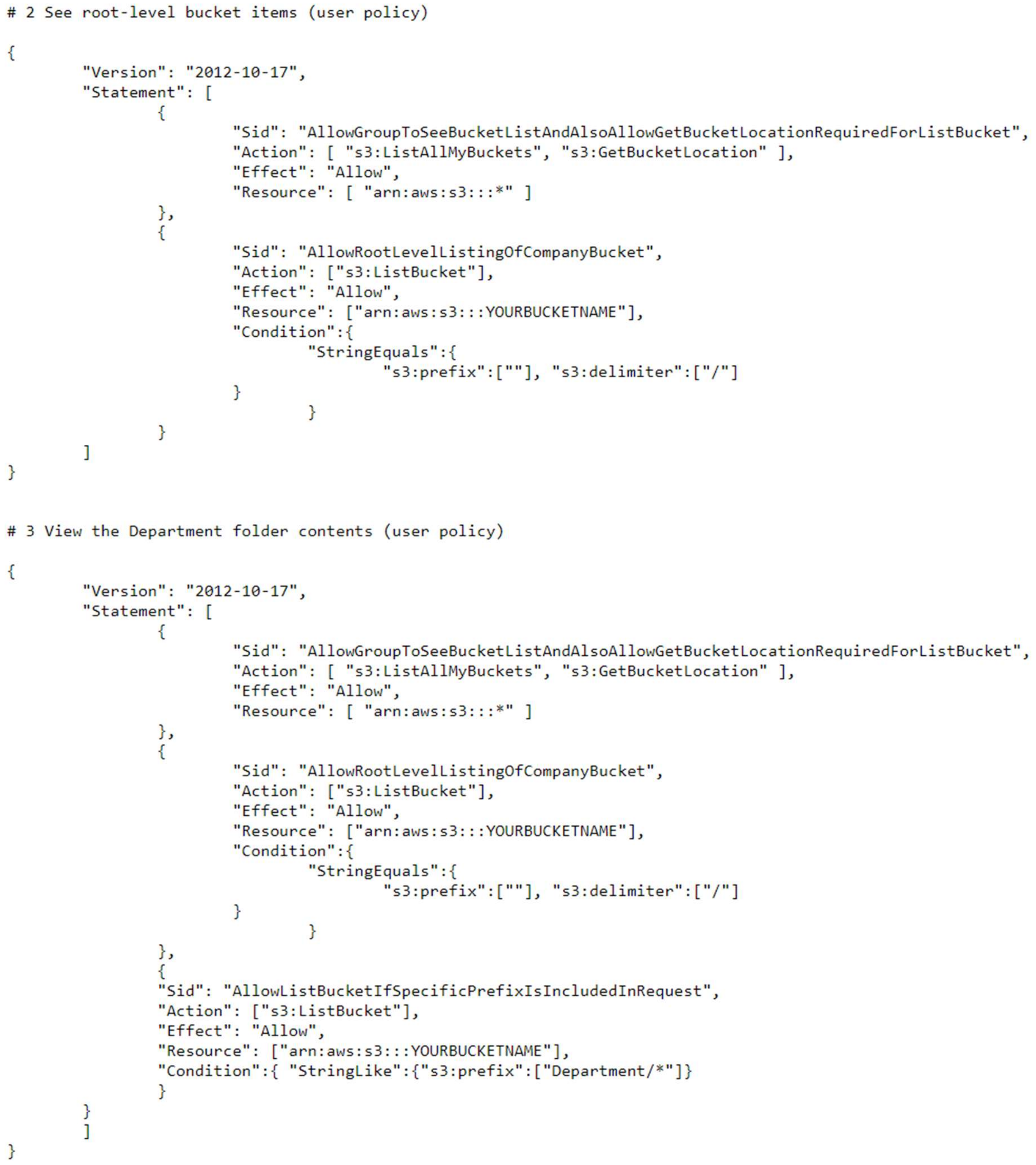
```
Microsoft Windows [Version 10.0.22621.3296]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ranja>aws configure
AWS Access Key ID [****************SMOH]: AKIAUHP67JCEHB6E65ET
AWS Secret Access Key [****************U1Tt]: 7D4lgxvHPDeMOVoZLhqQxRuJhVaOBtVoqZPe8jLc
Default region name [ap-south-1]: us-east-1
Default output format [None]:

C:\Users\ranja>aws s3 ls
2024-03-13 11:25:39 exp5bucketcsm
2024-02-07 16:41:30 mybucketawsexp

C:\Users\ranja>
```

STEP 10: Change the policies and then figure out the changes.



```
# 2 See root-level bucket items (user policy)

{
        "Version": "2012-10-17",
        "Statement": [
                {
                        "Sid": "AllowGroupToSeeBucketListAndAlsoAllowGetBucketLocationRequiredForListBucket",
                        "Action": [ "s3:ListAllMyBuckets", "s3:GetBucketLocation" ],
                        "Effect": "Allow",
                        "Resource": [ "arn:aws:s3:::*" ]
                },
                {
                        "Sid": "AllowRootLevelListingOfCompanyBucket",
                        "Action": ["s3:ListBucket"],
                        "Effect": "Allow",
                        "Resource": ["arn:aws:s3:::YOURBUCKETNAME"],
                        "Condition":{
                                "StringEquals":{
                                        "s3:prefix":[""], "s3:delimiter":["/"]
                        }
                                }
                }
        ]
}


# 3 View the Department folder contents (user policy)

{
        "Version": "2012-10-17",
        "Statement": [
                {
                        "Sid": "AllowGroupToSeeBucketListAndAlsoAllowGetBucketLocationRequiredForListBucket",
                        "Action": [ "s3:ListAllMyBuckets", "s3:GetBucketLocation" ],
                        "Effect": "Allow",
                        "Resource": [ "arn:aws:s3:::*" ]
                },
                {
                        "Sid": "AllowRootLevelListingOfCompanyBucket",
                        "Action": ["s3:ListBucket"],
                        "Effect": "Allow",
                        "Resource": ["arn:aws:s3:::YOURBUCKETNAME"],
                        "Condition":{
                                "StringEquals":{
                                        "s3:prefix":[""], "s3:delimiter":["/"]
                        }
                                }
                },
                {
                "Sid": "AllowListBucketIfSpecificPrefixIsIncludedInRequest",
                "Action": ["s3:ListBucket"],
                "Effect": "Allow",
                "Resource": ["arn:aws:s3:::YOURBUCKETNAME"],
                "Condition":{ "StringLike":{"s3:prefix":["Department/*"]}
                }
        }
        ]
}
```

4 Get and put objects in the Department folder (user policy)

```
{
        "Version": "2012-10-17",
        "Statement":[
        {
                "Sid": "AllowGroupToSeeBucketListAndAlsoAllowGetBucketLocationRequiredForListBucket",
                "Action": [ "s3:ListAllMyBuckets", "s3:GetBucketLocation" ],
                "Effect": "Allow",
                "Resource": [ "arn:aws:s3:::*" ]
        },
        {
                "Sid": "AllowRootLevelListingOfCompanyBucket",
                "Action": ["s3:ListBucket"],
                "Effect": "Allow",
                "Resource": ["arn:aws:s3:::YOURBUCKETNAME"],
                "Condition":{
                        "StringEquals":{
                                "s3:prefix":[""], "s3:delimiter":["/"]
                }
                        }
        },
        {
                "Sid":"AllowListBucketIfSpecificPrefixIsIncludedInRequest", "Action":["S3:ListBucket"],
                "Effect":"Allow",
                "Resource": ["arn:aws:s3:::YOURBUCKETNAME"],
                "Condition":{
                        "StringLike":{"s3:prefix":["Department/*"]
                        }
                }
        },
{
                "Sid":"AllowUserToReadWriteObjectDataInDepartmentFolder",
                "Action":["s3:GetObject", "s3:PutObject"],
                "Effect":"Allow",
                "Resource":["arn:aws:s3:::YOURBUCKETNAME/Department/*"]
        }
        ]
}
```

5 Explicitly grant access to Paul to list the Confidential folder (Bucket Policy) - use with policy 2 above

```
{
        "Version": "2012-10-17",
        "Id": "Policy1561964929358",
        "Statement":[
        {
                "Sid": "Stmt1561964454052",
                "Effect": "Allow",
                "Principal": {
                        "AWS": "arn:aws:iam::138422235973:user/Paul"
                },
                "Action": "s3:*",
                "Resource": "arn:aws:s3:::YOURBUCKETNAME",
                "Condition": {
                        "StringLike": {
                                "s3:prefix": "Confidential/*"
                        }
                }
        }
        ]
}
```

Q1: What is the purpose of ACL in S3?

ANSWER:

The purpose of Access Control Lists (ACLs) in Amazon S3 is to define who can access buckets and objects within buckets and what level of access they have. ACLs provide a simple way to manage access control by allowing you to grant permissions to predefined groups of users and grant specific permissions to individual AWS accounts.

Q2: When you should use ACL instaed of IAM Policy for managing access to S3 resources?

ANSWER:

You should use ACLs instead of IAM Policies for managing access to S3 resources when you need to grant access to specific individual objects within a bucket to AWS accounts that do not have IAM identities, such as third-party AWS accounts.

Q3: How does IAM differ from ACL in terms of managing access to S3 resources?

ANSWER:

IAM (Identity and Access Management) and ACLs both serve the purpose of managing access to S3 resources, but they differ in their scope and granularity. IAM is a centralized service that manages user identities and their permissions across the entire AWS ecosystem, whereas ACLs are specific to individual S3 buckets and objects. IAM provides more fine-grained control over permissions and allows for more complex access policies compared to ACLs.

Q4: Describe the principle of Least Privilege context of IAm and S3 sceurity.

ANSWER:

The principle of Least Privilege in the context of IAM and S3 security dictates that users or roles should only be granted the minimum level of permissions necessary to perform their intended tasks. This principle helps minimize the potential impact of security breaches or mistakes by limiting access to only what is required for legitimate operations.

Q5: What is Bucket Policy? How does it differ from ACL and IAM?

ANSWER:

A Bucket Policy is a resource-based policy that applies to an entire S3 bucket and allows you to define rules for access control at the bucket level. It differs from ACLs and IAM in that it can provide more granular control over access permissions, including conditions based on IP addresses, HTTP headers, and other attributes.

Q6: How would you integrate IAM policy, bucket policy and ACL to achieve comprehensive access control for S3 resources?

ANSWER:

To achieve comprehensive access control for S3 resources, you can integrate IAM policies, bucket policies, and ACLs as follows:

- Use IAM policies to manage access control for users, groups, and roles within your AWS account.
- Utilize bucket policies to define rules for access control at the bucket level, including cross-account access and conditions.
- Apply ACLs to individual objects within buckets to grant specific permissions to AWS accounts that do not have IAM identities, or when you need to grant access to specific objects.

Q7: What are the pitfalls for ACL, Bucket policy and IAM?

ANSWER:

Pitfalls for ACLs, Bucket Policies, and IAM include:

- Overly permissive policies: Granting too many permissions can lead to security vulnerabilities.
- Complexity: Managing multiple layers of access control can become complex and difficult to maintain.
- Misconfigurations: Incorrectly configured policies or ACLs can result in unintended access or denial of access to resources.
- Lack of auditing: Without proper monitoring and auditing, it can be challenging to identify and remediate security issues.