TASK 4

**Step 1: Create an IAM Policy and User**

**Goal**: Restrict access to specific resources using IAM.

1. **Navigate to IAM Console**:
   * Go to **AWS Management Console** > **Services** > **IAM**.
   * **Screenshot 1**: Capture the IAM dashboard.  
     **Heading**: *IAM Dashboard Overview*.
2. **Create a Custom IAM Policy**:
   * In IAM, go to **Policies** > **Create Policy** > Switch to **JSON** tab.
   * Paste this policy (replace YOUR\_BUCKET\_NAME with your bucket name):

json

Copy

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Action": ["s3:GetObject", "s3:PutObject"],

"Resource": "arn:aws:s3:::YOUR\_BUCKET\_NAME/\*"

}

]

}

* + Name the policy (e.g., S3-ReadWrite-Access).
  + **Screenshot 2**: Capture the JSON policy editor.  
    **Heading**: *Custom IAM Policy JSON Configuration*.

1. **Create an IAM User**:
   * Go to **Users** > **Add user**.
   * Enter a username (e.g., SecureS3User).
   * Select **Programmatic access** and **AWS Management Console access**.
   * Attach the policy S3-ReadWrite-Access created earlier.
   * **Screenshot 3**: Capture the user summary page with the attached policy.  
     **Heading**: *IAM User Creation with Custom Policy*.

**Step 2: Create a Secure S3 Bucket**

**Goal**: Configure an S3 bucket with encryption, versioning, and blocking public access.

1. **Create an S3 Bucket**:
   * Go to **S3 Console** > **Create bucket**.
   * Enter a **unique bucket name** and ensure the region is unchanged.
   * **Screenshot 4**: Capture the bucket creation page (region visible).  
     **Heading**: *S3 Bucket Creation in Default Region*.
2. **Block Public Access**:
   * Under **Block Public Access settings**, check **Block all public access**.
   * **Screenshot 5**: Capture the public access blocking settings.  
     **Heading**: *S3 Bucket Public Access Block Configuration*.
3. **Enable Versioning**:
   * Go to the bucket’s **Properties** tab > **Bucket Versioning** > **Enable**.
   * **Screenshot 6**: Capture the versioning settings.  
     **Heading**: *S3 Bucket Versioning Enabled*.
4. **Enable Server-Side Encryption**:
   * Go to the bucket’s **Properties** tab > **Default encryption**.
   * Select **AWS Key Management Service (SSE-KMS)**.
   * Choose **AWS managed key (aws/s3)** or create a new KMS key (see Step 3).
   * **Screenshot 7**: Capture the encryption settings.  
     **Heading**: *S3 Bucket Default Encryption Configuration*.

**Step 3: Configure AWS KMS Encryption**

**Goal**: Create a KMS key for S3 encryption.

1. **Create a KMS Key**:
   * Go to **AWS KMS Console** > **Customer managed keys** > **Create key**.
   * Set **Key type** as **Symmetric** and **Usage** as **Encrypt and decrypt**.
   * Add a key alias (e.g., S3-Encryption-Key).
   * Assign the IAM user as a **key user** in the key policy.
   * **Screenshot 8**: Capture the KMS key policy configuration.  
     **Heading**: *KMS Key Policy with IAM User Permissions*.
2. **Apply KMS Key to S3 Bucket**:
   * Return to the S3 bucket’s **Default encryption** settings.
   * Select the KMS key you created (e.g., S3-Encryption-Key).
   * **Screenshot 9**: Capture the S3 bucket’s KMS key selection.  
     **Heading**: *S3 Bucket KMS Encryption Key Assignment*.

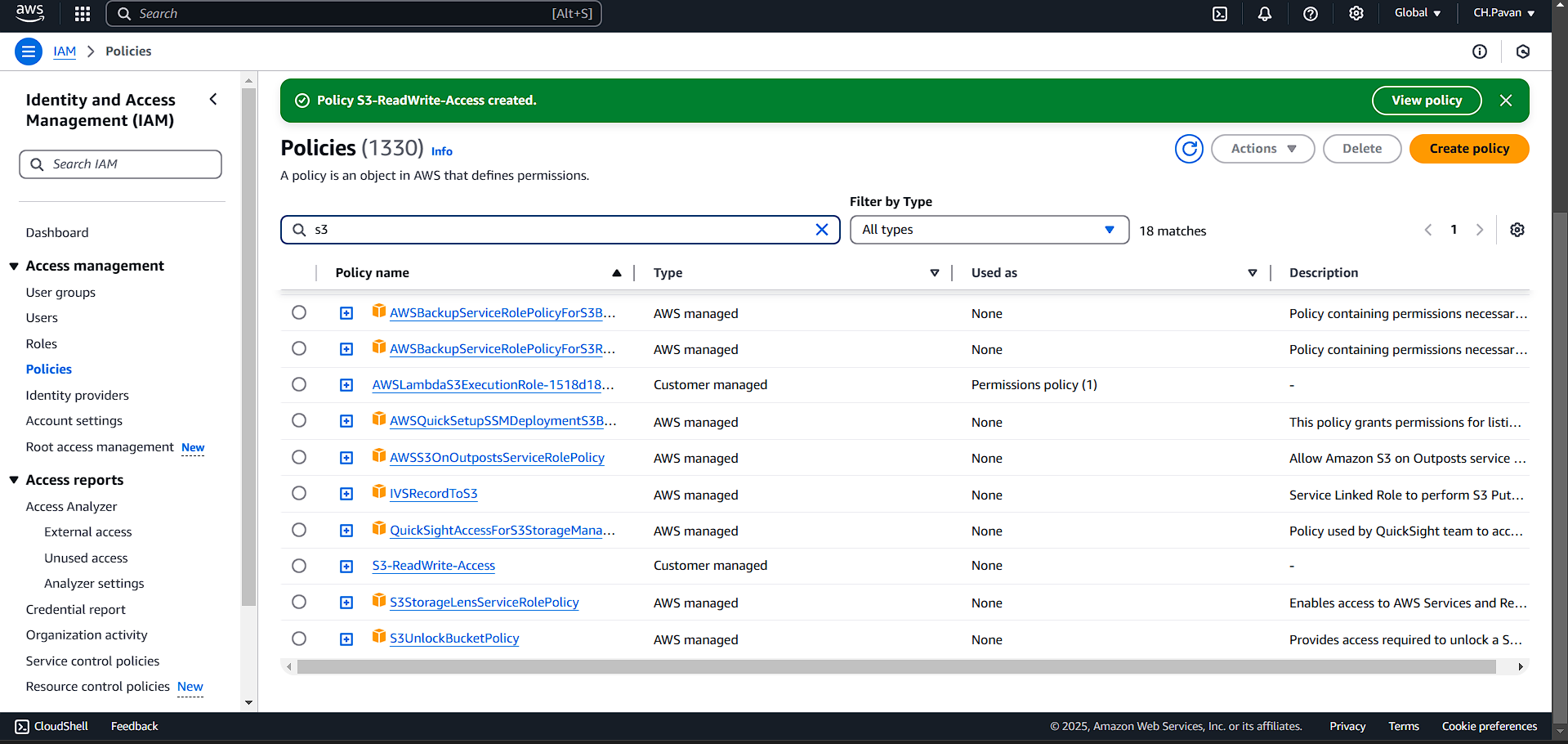
**Step 4: Test and Validate**

**Goal**: Verify security policies and encryption.

1. **Upload a Test File**:
   * Use the IAM user credentials to log in to the AWS Console.
   * Upload a file to the S3 bucket.
   * **Screenshot 10**: Capture the successful upload.  
     **Heading**: *File Upload to S3 Using Restricted IAM User*.
2. **Check Encryption Status**:
   * Select the uploaded file in S3 > **Properties** > **Server-Side Encryption**.
   * Confirm encryption is enabled with the KMS key.
   * **Screenshot 11**: Capture the file’s encryption details.  
     **Heading**: *S3 Object Encryption Status with KMS*.

**Final Deliverable: Report Structure**

1. **Introduction**: Explain the purpose of securing AWS resources.
2. **IAM Configuration**: Include Screenshots 1-3.
3. **S3 Security**: Include Screenshots 4-7.
4. **KMS Encryption**: Include Screenshots 8-9.
5. **Validation**: Include Screenshots 10-11.
6. **Conclusion**: Summarize how IAM, S3, and KMS enhance security.

S3 Read Write  


Created a User

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Bucket Visionary setting

A screenshot of a computer

AI-generated content may be incorrect.

Default Encryption

A screenshot of a computer

AI-generated content may be incorrect.

Created Key User

A screenshot of a computer

AI-generated content may be incorrect.

File Uploaded

A screenshot of a computer

AI-generated content may be incorrect.

Server side Encryption

A screenshot of a computer

AI-generated content may be incorrect.

**Final Deliverable: Report Structure**

1. **Introduction**: Explain the purpose of securing AWS resources.
2. **IAM Configuration**: Include Screenshots 1-3.
3. **S3 Security**: Include Screenshots 4-7.
4. **KMS Encryption**: Include Screenshots 8-9.
5. **Validation**: Include Screenshots 10-11.
6. **Conclusion**: Summarize how IAM, S3, and KMS enhance security.