



User Guide

Getting Started with Geyser Data

Your complete guide to creating your account and managing your cold data cloud archive.





Getting Started with Geyser Data

v25.11.26

Create an account	1
Activate & Login	4
Create a Cloud Tape Library	8
View and Manage Your Cloud Tape Library	10
Create Access and Secret Keys	12
Write Data to Your Cloud Tape Library	13
Browse Data in Your Cloud Tape Library	16
Read Data from Your Cloud Tape Library	17
Air Gap Feature	21
Tape Collections	21
Useful links	23
How to use Cyberduck with Geyser data	23
S3 Browser Setup Guide	23
Veeam Integration Guide	23
Cohesity Integration Guide	23
MinIO Integration Guide	23
Support	23

Create an account

- Go to <https://console.geyserdata.com>
- Click **Register** button (top right)
- Enter you Organization Name, Email Address, First Name and Last Name
- Click **Request Access**

Request Access to the platform

Organization

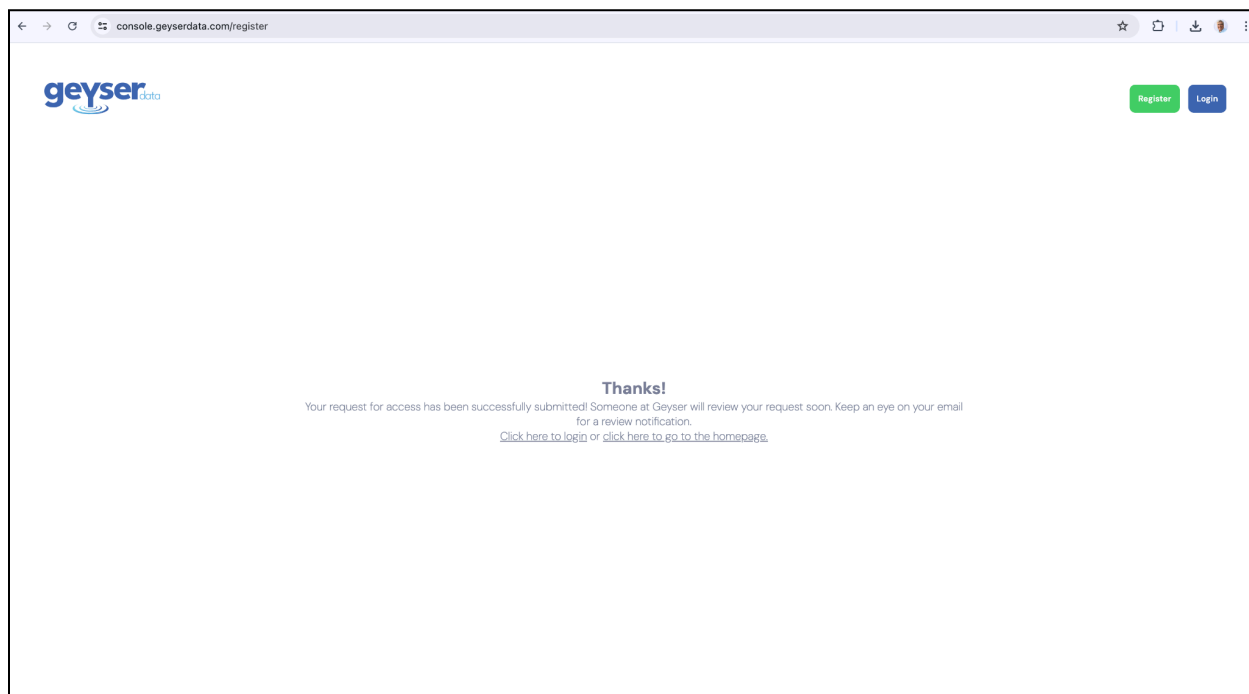
Email Address

First Name

Last Name

[Request Access](#)

- You will get a thank you screen.



Your request will be reviewed (typically within 24 hours)

If you don't receive a Welcome email within 24 hours, please check your spam folder or email to support@geyserdata.com

Activate & Login

- Click **Join** in your Welcome Email to set a password (8-100 characters).

Hello John Smith

Welcome to Geyser Data Cloud! We're excited to have you join our community of innovators who are leveraging the power of tape as a service to transform their businesses.

Ready to get started? Click the button below to join now and experience the future of data archive storage.

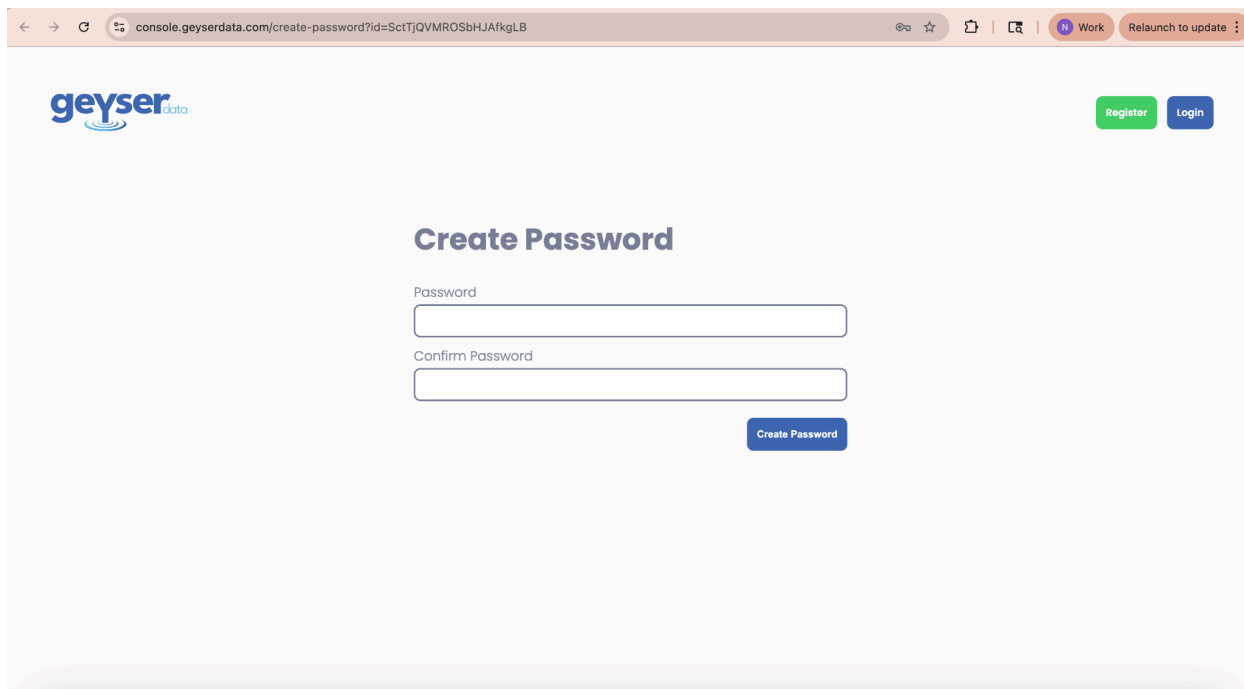
Join

We're here to help you every step of the way. If you have any questions or need assistance, please don't hesitate to reach out to our support team at support@geyserdata.com.

Welcome aboard!

The Geyser Data Cloud Team

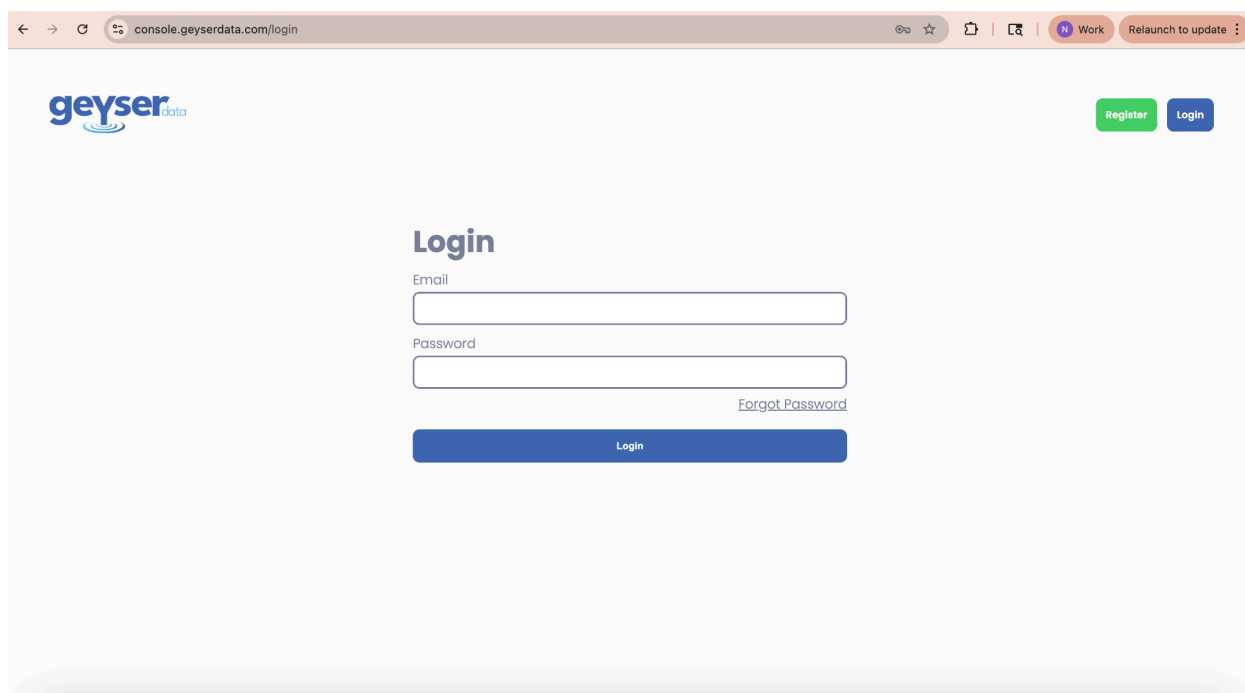
- You will be redirected to the Create Password page



The screenshot shows a web browser window with the URL `console.geyserdata.com/create-password?id=SctJTQVMROSbHJAfkgLB`. The page features the Geyser Data logo in the top left and 'Register' and 'Login' buttons in the top right. The main heading is 'Create Password'. Below it are two input fields: 'Password' and 'Confirm Password'. A blue 'Create Password' button is positioned to the right of the 'Confirm Password' field.

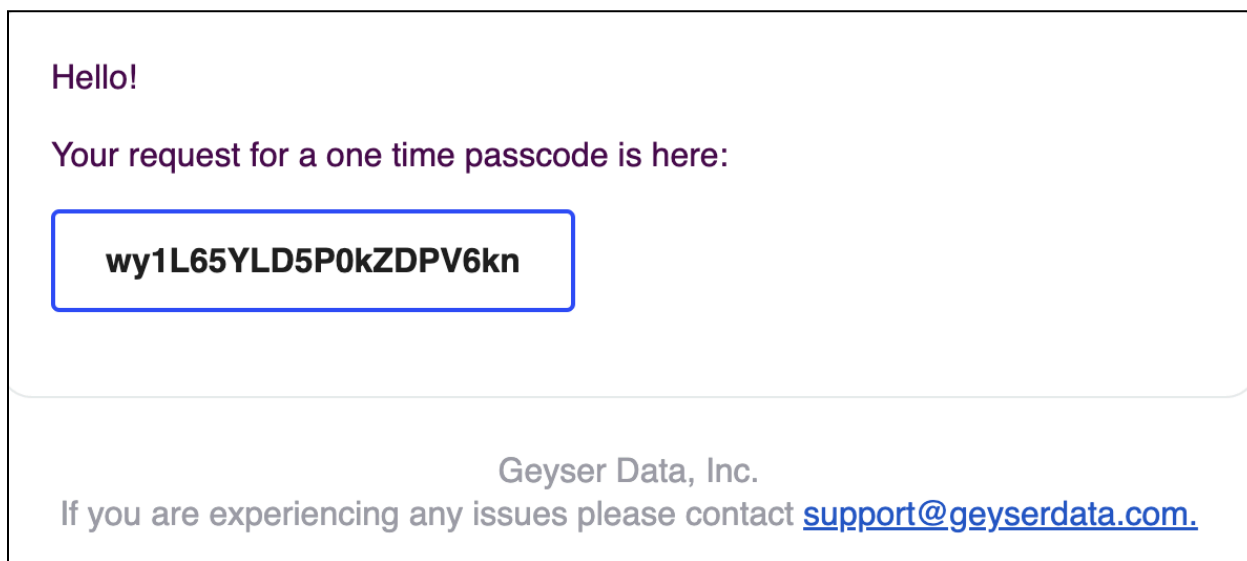
After setting your password you can go ahead and log in!

- Enter your email address and password, then click **Login**.

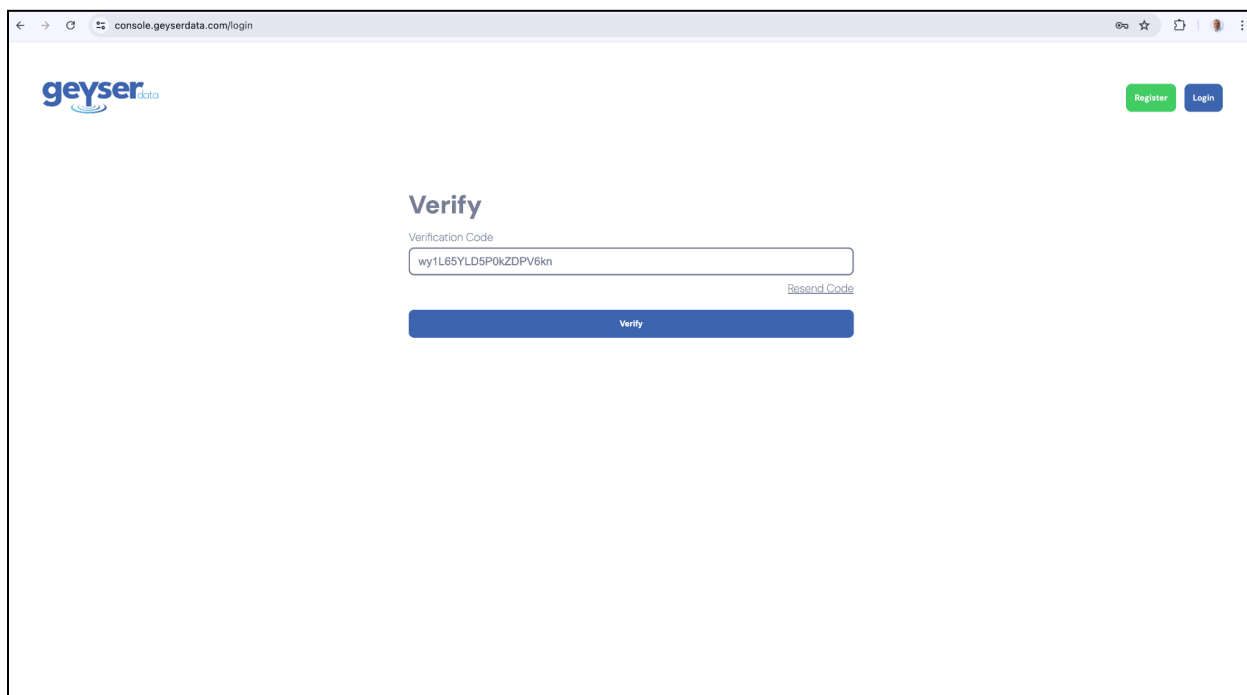


The screenshot shows a web browser window with the URL `console.geyserdata.com/login`. The page features the Geyser Data logo in the top left and 'Register' and 'Login' buttons in the top right. The main heading is 'Login'. Below it are two input fields: 'Email' and 'Password'. A blue 'Login' button is positioned below the 'Password' field. A link labeled 'Forgot Password' is located to the right of the 'Password' field.

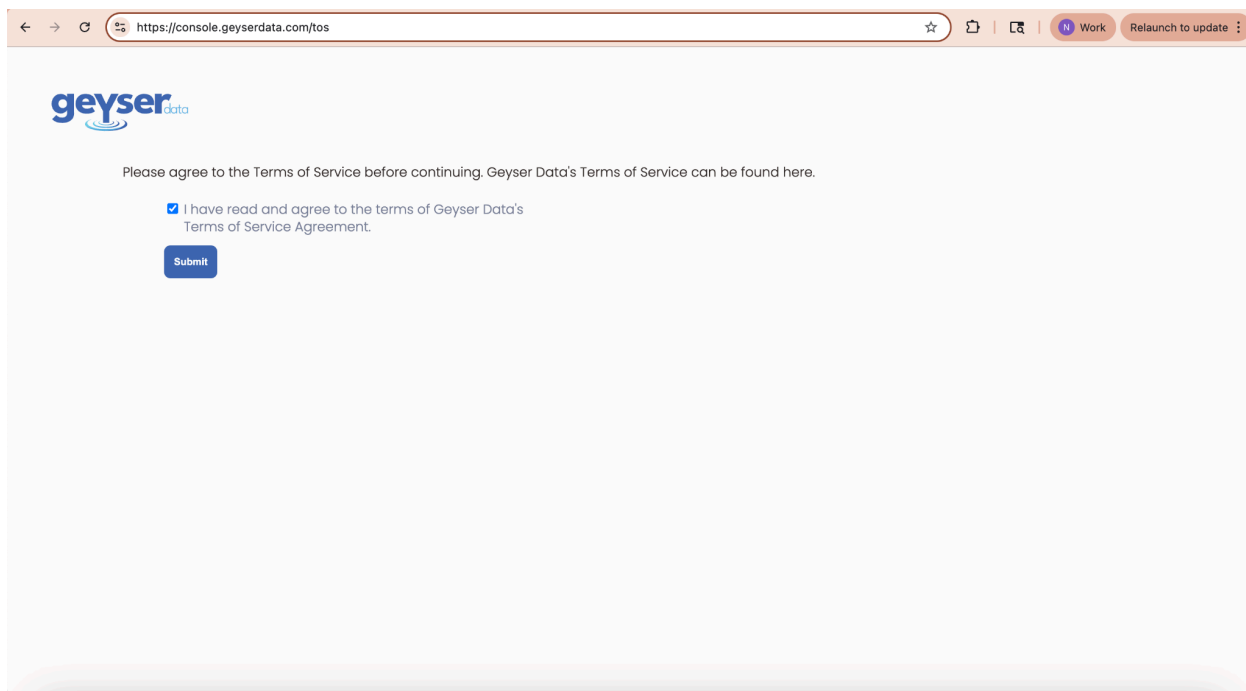
- Enter the **verification code** sent to your email (two-factor authentication).



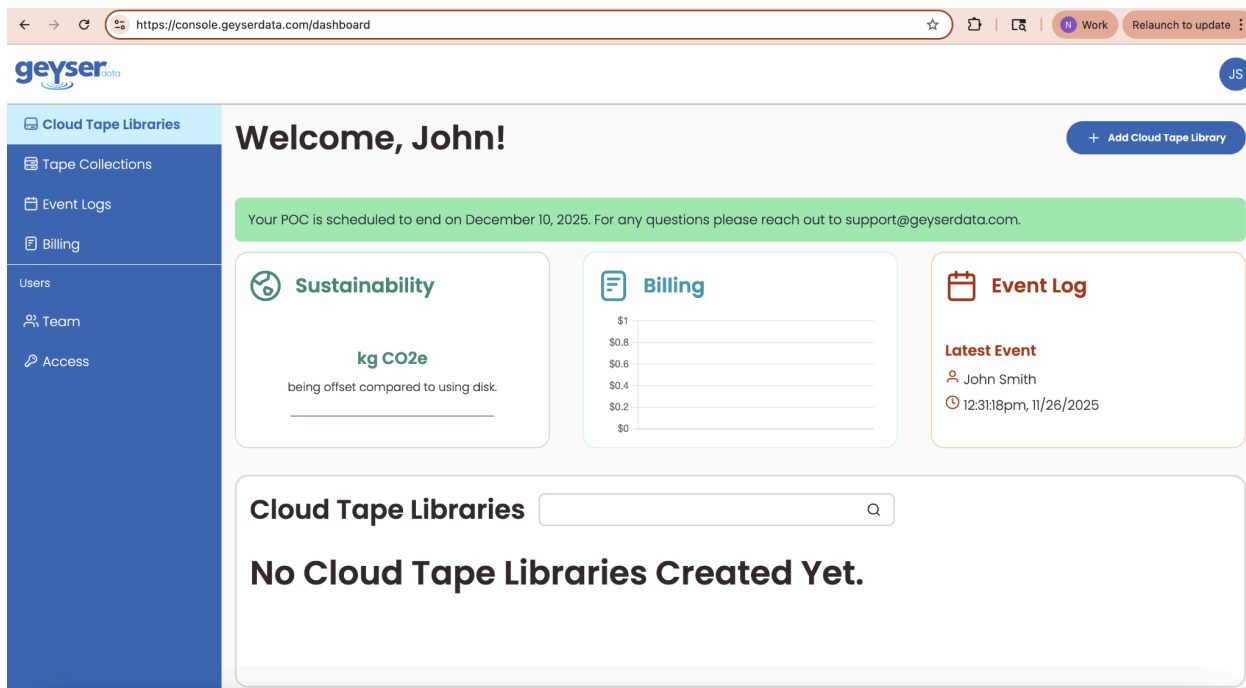
- Copy the **verification code** from the email, paste into the Verify screen and click **Verify**.



- Finally, agree to our terms of service.



- Congratulations! You are now on your Geyser Data dashboard.



Create a Cloud Tape Library

- On the dashboard, click on **+ Add Cloud Tape Library** (top right corner)

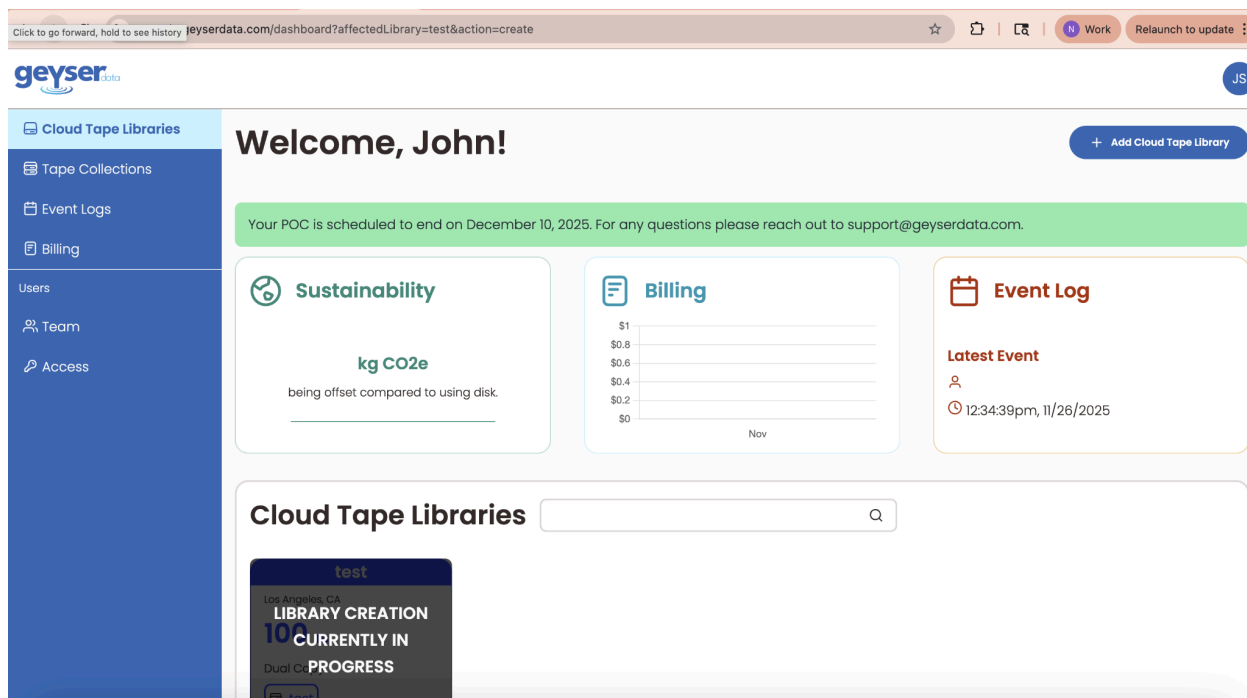
The screenshot shows the Geyser Data dashboard. The left sidebar contains navigation links: Cloud Tape Libraries, Tape Collections, Event Logs, Billing, Users, Team, and Access. The main content area displays a welcome message for 'John!', a sustainability widget showing 0 kg CO2e, a billing widget with a line graph, and an event log showing the latest event at 2:24:21pm on 11/26/2025. A blue button labeled '+ Add Cloud Tape Library' is located in the top right corner, with an arrow pointing to it from the instruction above.

The screenshot shows the 'Create a New Cloud Tape Library' form. The form includes the following fields and options:

- Cloud Tape Library Name:** A text input field with the placeholder 'Enter name'.
- Region:** A dropdown menu currently showing 'LA1 - Los Angeles, CA'.
- Capacity (TB):** A text input field with the placeholder 'Enter capacity needed (TBs)'.
- Tape Collection:** A dropdown menu currently showing 'Create New Tape Collection'.
- Options:**
 - ☒ Library Color
 - ☐ Object Versioning
 - ☐ Object Locking
 - ☐ Veeam Enabled
 - ☒ Dual Copy
 - ☐ Single Copy
 - ☐ Compression
 - ☐ Encryption

A blue button labeled 'Create Cloud Tape Library' is located at the bottom of the form.

- Fill out the form:
 - Enter Cloud Tape Library Name
 - Choose Region from the dropdown.
 - Provisioned Capacity in TBs. (The capacity rounds up to the next full Tape)
 - Color: Choose a color to visually organize your libraries.
 - Object Versioning: Enabling the bucket to retain multiple copies with the same file name and path.
 - Object Locking: Enabling a lock on any object so it cannot be deleted.
 - Veeam Enabled: Enabling this bucket for the Veeam Archive tier.
 - Tape Collection: Default to Create a new tape collection (see Tape Collections section for more details)
 - Copy Type: Select Single Copy or Dual Copy. (For production environments we recommend dual copy, ie the data is written to two tapes)
 - Optional: Enable Compression, Encryption (These are available at additional cost)
- Click **Create Cloud Tape Library**
- Watch for the “Library Creation Currently in Progress” message on the dashboard - setup takes less than 1 minute.



Click to go forward, hold to see history | geysersdata.com/dashboard?affectedLibrary=test&action=create

geyser data JS

Cloud Tape Libraries

Wellcome, John!

+ Add Cloud Tape Library

Your POC is scheduled to end on December 10, 2025. For any questions please reach out to support@geysersdata.com.

Sustainability

kg CO2e

being offset compared to using disk

Billing

\$1
\$0.8
\$0.6
\$0.4
\$0.2
\$0

Nov

Event Log

Latest Event

12:34:39pm, 11/26/2025

Cloud Tape Libraries

test

Los Angeles, CA

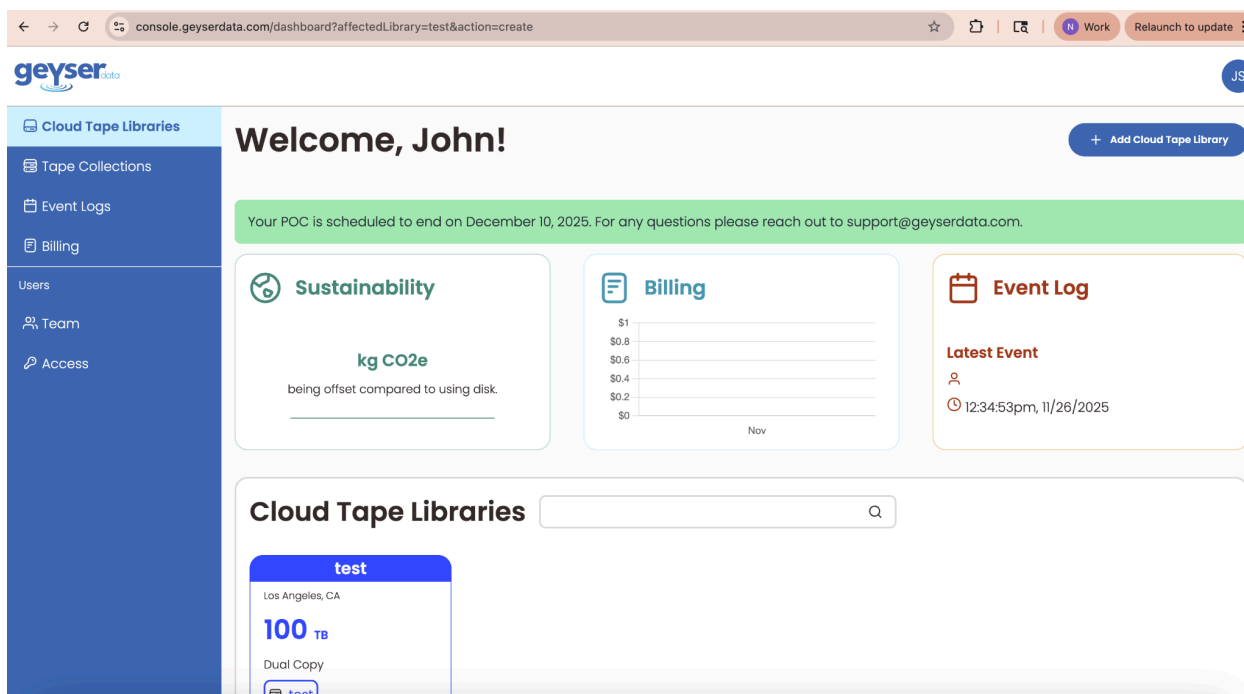
LIBRARY CREATION

100% CURRENTLY IN PROGRESS

Dual Copy

View and Manage Your Cloud Tape Library

- Once the Cloud Tape Library is created, it will show up as a tile on the Dashboard screen.



- Click on the **Cloud Tape Library tile** to access the View Cloud Tape Library screen. This screen provides a comprehensive overview of your library:
 - Information about the Cloud Tape Library
 - Tape Collection settings
 - s3Url - the s3 endpoint used for directing S3 commands through CLI/API or application integrations

console.geyserdata.com/cloudTapeLibrary/03dea299-5a0e-4924-8258-a29bc8097baf

geyserdata JS

- Cloud Tape Libraries
- Tape Collections
- Event Logs
- Billing

Users

- Team
- Access

There are no access keys associated with this user. You will need an access key to write to this Cloud Tape Library. In order to manage your access keys, [click here](#) or click "Access" in the sidebar.

Delete Edit Airgap Browse

Cloud Tape Library Details

Cloud Tape Library Name test	Tape Collection Name test
Region Los Angeles, CA	Data Usage (TB) 100
<input type="checkbox"/> Object Versioning	<input type="checkbox"/> Compression
<input type="checkbox"/> Object Locking	<input type="checkbox"/> Encryption
Color ●	<input checked="" type="checkbox"/> Dual Copy
S3 Endpoint URL https://al.geyserdata.com/test-03dea299-5a0e-4924-8258-a29bc8097baf	

Cloud Integrations

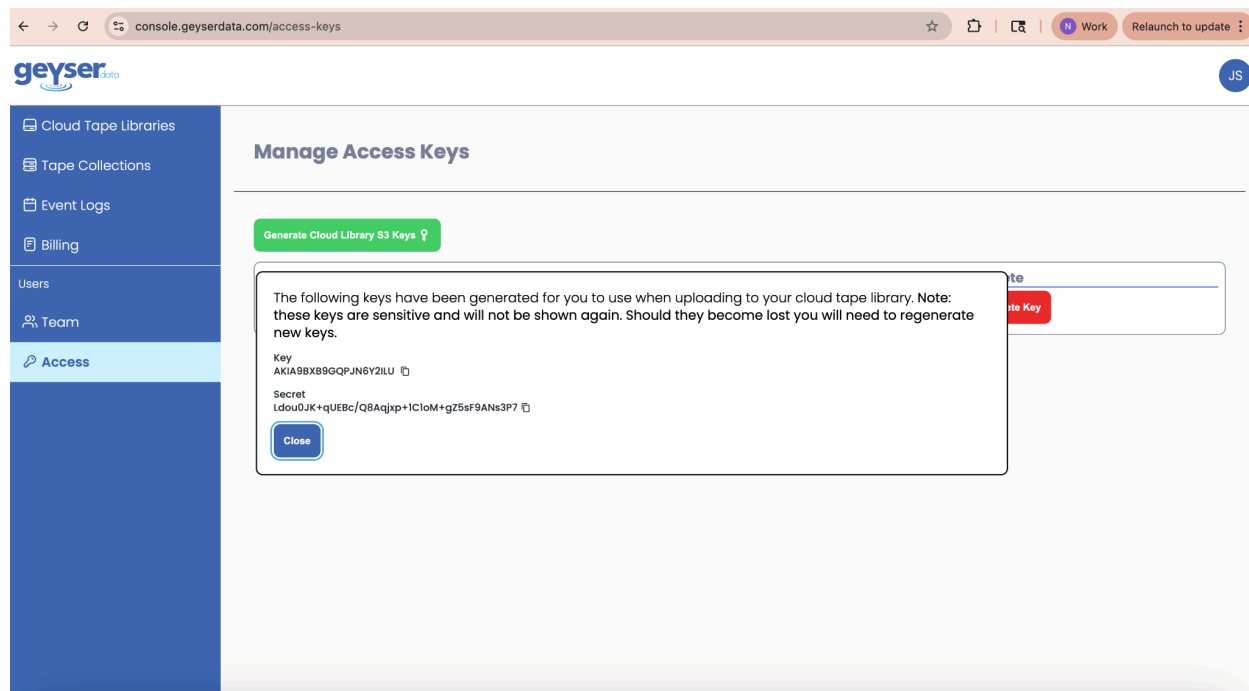
To enable cloud integration, select the integration type you'd like to restore content from this Cloud Tape Library, then enter credentials and bucket name.

[Enable AWS Restore and Add AWS Bucket](#)
[Enable Wasabi Restore and Add Wasabi Bucket](#)
[Enable Oracle Restore and Add Oracle Bucket](#)

- If you see a red banner, you need to generate Access/Secret keys first.

Create Access and Secret Keys

- If prompted, click **Click Here** in the red banner or click **Access** (left panel)
- Click **Generate Cloud Library S3 keys**
- **Copy BOTH Access and Secret Keys to a password manager or secure file.**
For security reasons, you cannot view these keys again—regenerate as needed.



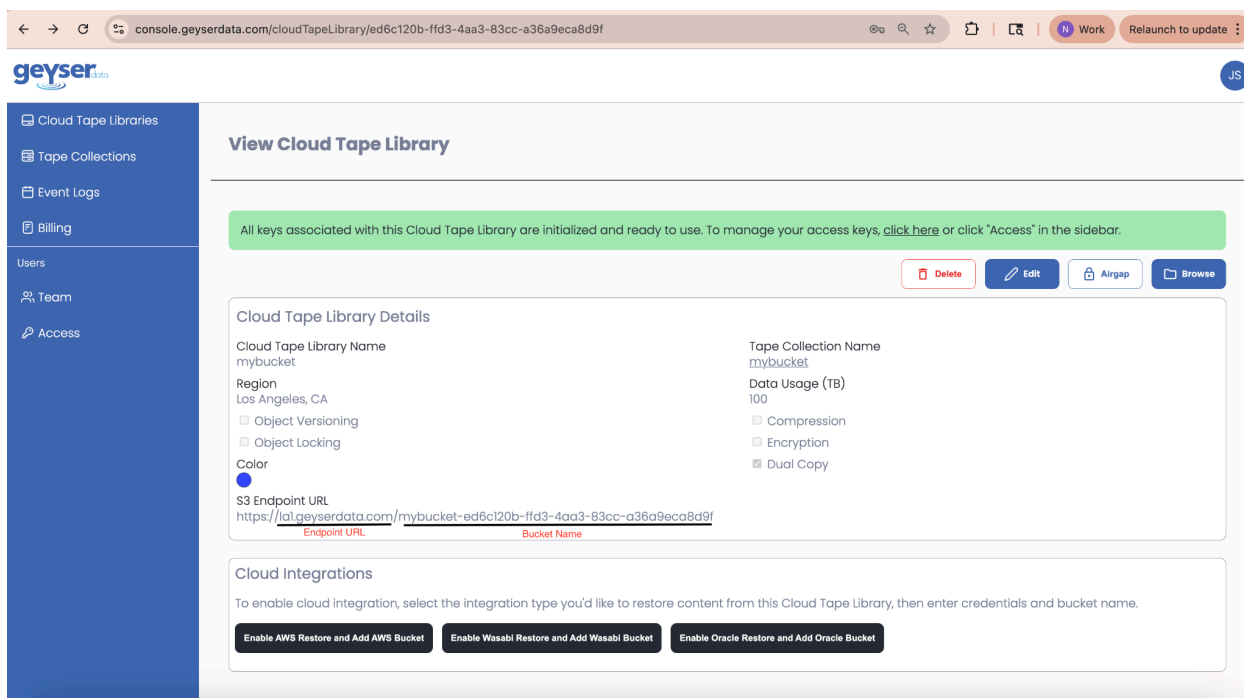
- You are ready to start writing data into your cloud tape library!

Write Data to Your Cloud Tape Library

Geyser Data Cloud Tape Library supports standard S3 Glacier protocols.

You'll need:

- Endpoint URL and Bucket Name



- Access and Secret keys

Sample Python Script to Upload a File

To use this script you need Python version 3 and install the AWS boto3 library with the command:

```
pip install boto3
```

To use this script you call it from the cli using:

```
python upload_file_to_s3.py <filename>
```

Copy the script below and paste it on a file upload_file_to_s3.py The content of the upload_to_s3.py is below.

Change the value of the following variables to match your own values:

bucket_name, endpoint_url, access_key, secret_key

```
import boto3 # type: ignore
import sys
import os

def upload_file_to_s3(filename,bucket_name, endpoint_url, access_key, secret_key):
    # Create a session with the specified parameters
    session = boto3.session.Session()

    # Create an S3 client with custom configurations
    s3_client = session.client(
        service_name='s3',
        aws_access_key_id=access_key,
        aws_secret_access_key=secret_key,
        endpoint_url=endpoint_url,
        verify=False # Ignore SSL certificates
    )

    # Get the basename of the file to use as the key
    key = os.path.basename(filename)

    try:
        # Upload the file
        s3_client.upload_file(Filename=filename, Bucket=bucket_name, Key=key)
        print(f"File '{filename}' uploaded to bucket '{bucket_name}' successfully.")
    except Exception as e:
        print(f"Error uploading file: {e}")

if __name__ == "__main__":
    if len(sys.argv) != 2:
        print("Usage: python3 upload_file_to_s3.py <filename>")
        sys.exit(1)
    # Variables
    filename = sys.argv[1]
    bucket_name = 'ctluserdemo-b428b54d-fbec-49b1-92ad-ceb520d777cb'
    endpoint_url = 'https://boulderlab.geyserdata.com'
    access_key = 'AKIA47CR2UE7WGLJIRVP'
```

```
secret_key = 'qN8cbYbqf/bUMYIn4iZd3LKFs0Z8hLrBWGaejW3'  
  
upload_file_to_s3(filename, bucket_name, endpoint_url, access_key, secret_key)
```

To upload the file test.bin run:

```
python3 upload_file_to_s3.py test.bin
```


Browse Data in Your Cloud Tape Library

- Go to the Dashboard, select your library.
- Click on **Browse**

The screenshot shows the 'View Cloud Tape Library' interface. On the left is a sidebar with navigation options: Cloud Tape Libraries, Tape Collections, Event Logs, Billing, Users, Team, and Access. The main content area has a title 'View Cloud Tape Library' and a green notification bar stating: 'All keys associated with this Cloud Tape Library are initialized and ready to use. To manage your access keys, [click here](#) or click "Access" in the sidebar.' Below this are buttons for Delete, Edit, Airgap, and Browse. The 'Browse' button is highlighted with a black arrow. Under 'Cloud Tape Library Details', the 'Tape Collection Name' is 'mybucket'. Other details include Region (Los Angeles, CA), Object Versioning, Object Locking, Color (blue), and S3 Endpoint URL. The 'Cloud Integrations' section at the bottom provides instructions on how to enable integrations with AWS, Wasabi, or Oracle.

- You will see the files uploaded to your cloud tape library

The screenshot shows the 'Browse Cloud Tape Library' interface. The sidebar is the same as in the previous screenshot. The main content area has a title 'Browse Cloud Tape Library' and shows the selected library 'mybucket' with its ID. There is an 'Upload File' button in the top right. Below it is a search bar labeled 'File name prefix search:'. A table lists the files in the library:

File Name	Versions	Size	Location	Last Modified
awesome.gif		57.9 KB	CACHE	2025-11-26T22:21:10.851+0000

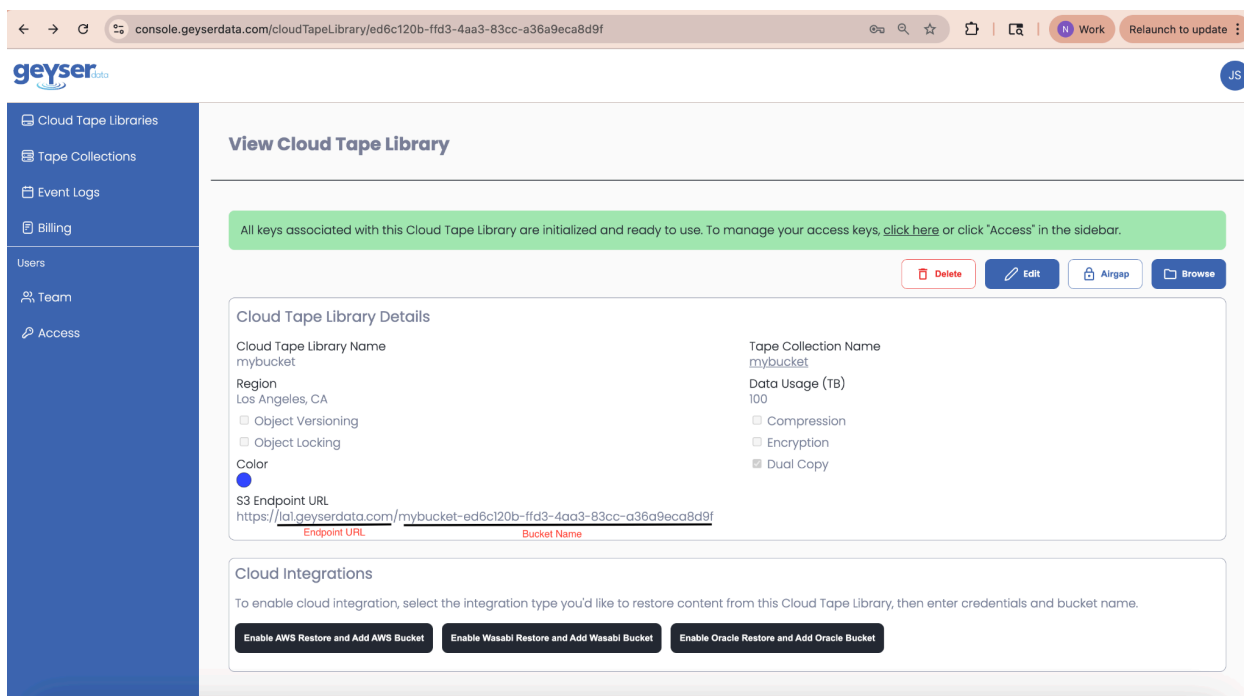
Below the table, there is a 'Download File' button and a trash icon for the file 'awesome.gif'.

Read Data from Your Cloud Tape Library

Geyser Data Cloud Tape Library supports standard S3 Glacier protocols.

You'll need:

- Endpoint URL and Bucket Name



- Access and Secret keys

Sample Python Script to Upload a File

To use this script you need Python version 3 and install the AWS boto3 library with the command:

```
pip install boto3
```

To use this script you can call it with the cli using:

```
python download_file_from_glacier.py <filename> <local_directory_path>
```

The code first calls the Glacier Restore API on the particular file, and waits until the file is restored to disk. Once the file is on disk, it immediately calls the GET to download the file.

Copy the script below and paste it on a file download_file_from_glacier.py The content of the download_file_from_glacier.py is below.

Change the value of the following variables to match your own values:

bucket_name, endpoint_url, access_key, secret_key

```
import boto3 # type: ignore
import sys
import os
import time
from botocore.config import Config # type: ignore
from botocore.exceptions import ClientError # type: ignore

def initiate_restore_from_glacier(s3_client, bucket_name, key):
    try:
        # Initiate restore request
        response = s3_client.restore_object(
            Bucket=bucket_name,
            Key=key,
            RestoreRequest={
                'Days': 1,
                'GlacierJobParameters': {
                    'Tier': 'Standard'
                }
            }
        )
        print(f"Initiated restore for '{key}' from Glacier.")
    except ClientError as e:
        print(f"Error initiating restore: {e}")
        return False
    return True

def check_restore_status(s3_client, bucket_name, key):
    try:
        response = s3_client.head_object(Bucket=bucket_name, Key=key)
        print(f"\n Response = {response}")
        # Parse the restoration status
        restore_status = response.get('Restore', "")

        if 'ongoing-request="true"' in restore_status:
            print("Restoration is still in progress.")
```

```

        return False
    elif 'ongoing-request="false"' in restore_status:
        # Extract expiry date
        expiry_date = restore_status.split('expiry-date="')[1].split('"')[0]
        print(f"Restoration is complete. Restored object will be available until {expiry_date}.")
        return True
    else:
        print("Restoration information is not available or object is not in Glacier.")
        return True

except ClientError as e:
    print(f"Error checking restore status: {e}")
    return False

def download_file_from_s3(filename, bucket_name, endpoint_url, access_key, secret_key, download_path):
    # Create a session with the specified parameters
    session = boto3.session.Session()

    # Create a configuration with a long timeout
    config = Config(
        connect_timeout=300, # 5 minutes
        read_timeout=300    # 5 minutes
    )

    # Create an S3 client with custom configurations
    s3_client = session.client(
        service_name='s3',
        aws_access_key_id=access_key,
        aws_secret_access_key=secret_key,
        endpoint_url=endpoint_url,
        verify=False, # Ignore SSL certificates
        config=config
    )

    key = filename

    # Initiate the restore process
    if initiate_restore_from_glacier(s3_client, bucket_name, key):

```

```
# Check the restore status periodically
while True:
    if check_restore_status(s3_client, bucket_name, key):
        break
    print("Waiting for 30 seconds before checking the restore status again...")
    time.sleep(30)

try:
    # Download the file
    s3_client.download_file(Bucket=bucket_name, Key=key, Filename=os.path.join(download_path, filename))
    print(f"File '{filename}' downloaded from bucket '{bucket_name}' successfully.")
except Exception as e:
    print(f"Error downloading file: {e}")

if __name__ == "__main__":
    if len(sys.argv) != 3:
        print("Usage: python download_file_from_glacier.py <filename> <download_path>")
        sys.exit(1)

    filename = sys.argv[1]
    download_path = sys.argv[2]
    bucket_name = 'latestsinglecopy-774c5929-a972-4e59-b300-de4087b6b664'
    endpoint_url = 'https://la1.geyserdata.com'
    access_key = 'AKZA27CR2U87Q4Z26MVG'
    secret_key = 'r7gI0bzOSgkdfa9bUA0KZFk7+wbdNG/YkGX1gscy'

    start_time = time.time()

    download_file_from_s3(filename, bucket_name, endpoint_url, access_key, secret_key, download_path)
    end_time = time.time()

    elapsed_time = end_time - start_time
    print(f"Total execution time: {elapsed_time:.4f} seconds")
```

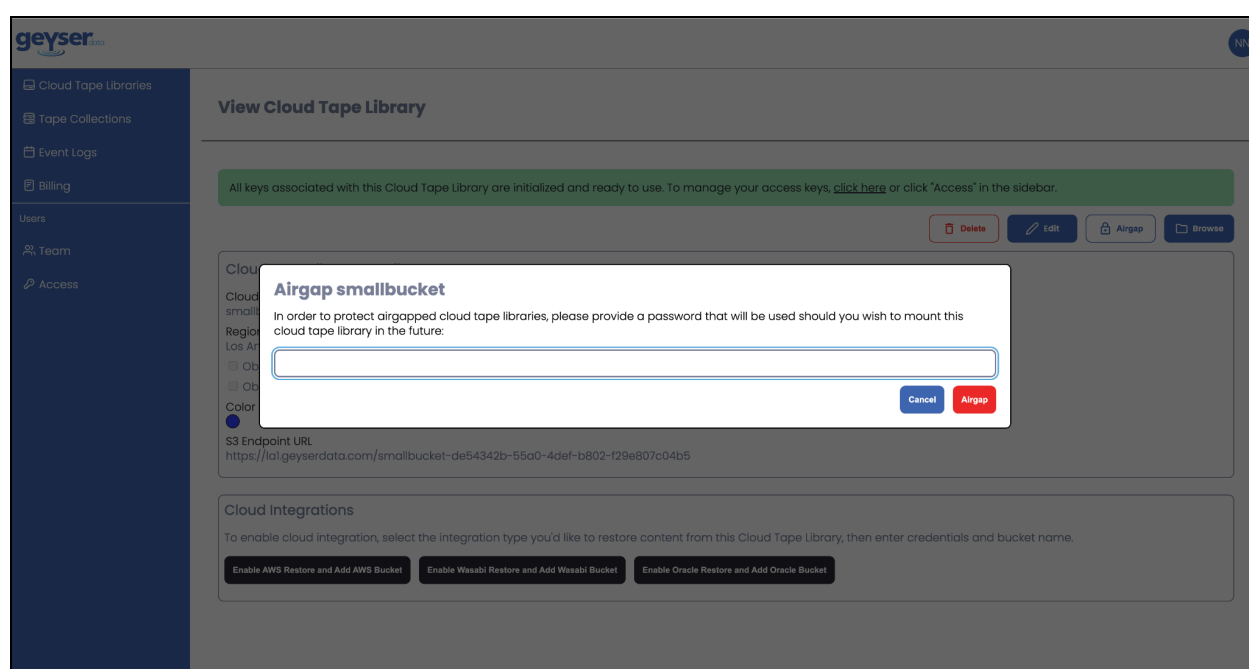
To restore and download a file:

```
python3 download_file_from_glacier.py test.bin ./
```

Air Gap Feature

Geyser Data enables you to Air Gap a Cloud Tape Library for enhanced security. When a Cloud Tape Library is Air gapped, its tape cartridges are unmounted from the virtual tape drives, preventing any reads or writes and providing strong protection against unauthorized access or cyber attacks.

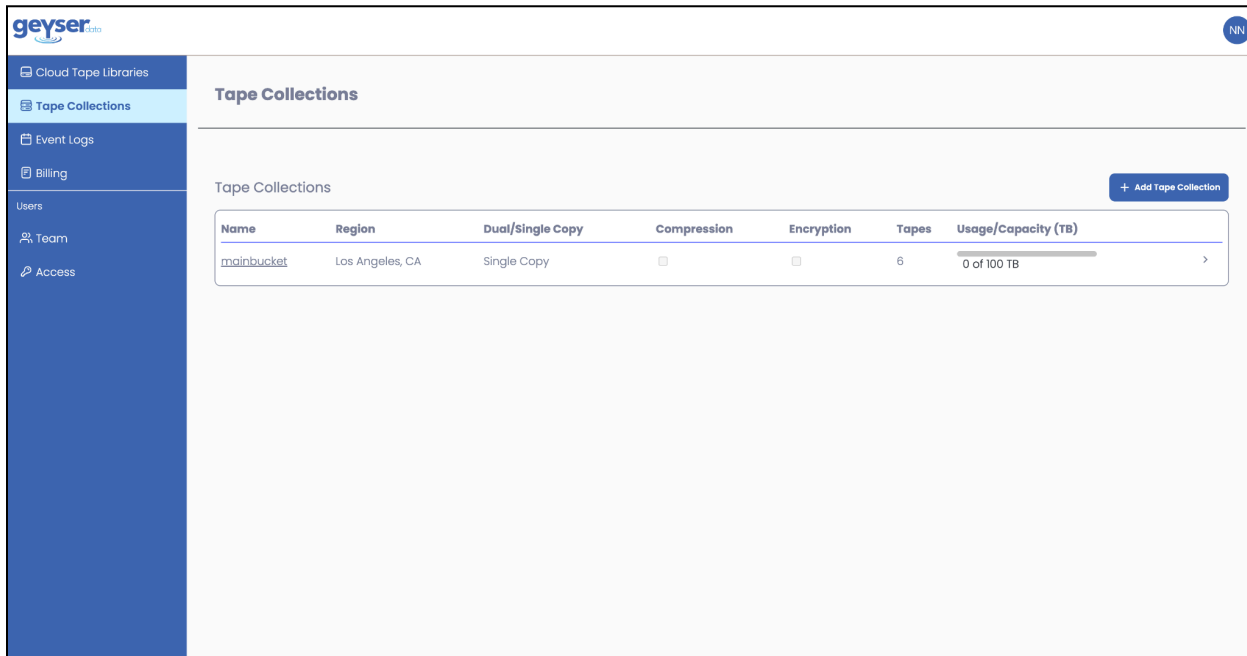
To Air Gap a cloud tape library, simply click the **Air Gap** button and enter a unique password for restoration. Be sure to record this password in a secure location, as you will need it to restore access; without it, the library cannot be remounted.



To restore access, repeat the process: select the Cloud Tape Library, click **Air Gap** and enter the same air gap password to re-enable read and write capabilities.

Tape Collections

A Tape collection is a set of Physical Tape Cartridges mapped to one or more Cloud Tape Libraries (i.e. buckets). Creating multiple Tape Collections helps separate workloads so that data for each is stored contiguously on tape, enabling more efficient data retrieval. For optimal performance, assign a dedicated Tape Collection to each Cloud Tape Library. However, for smaller buckets, you can make several Cloud Tape Libraries onto the same Tape Collection.



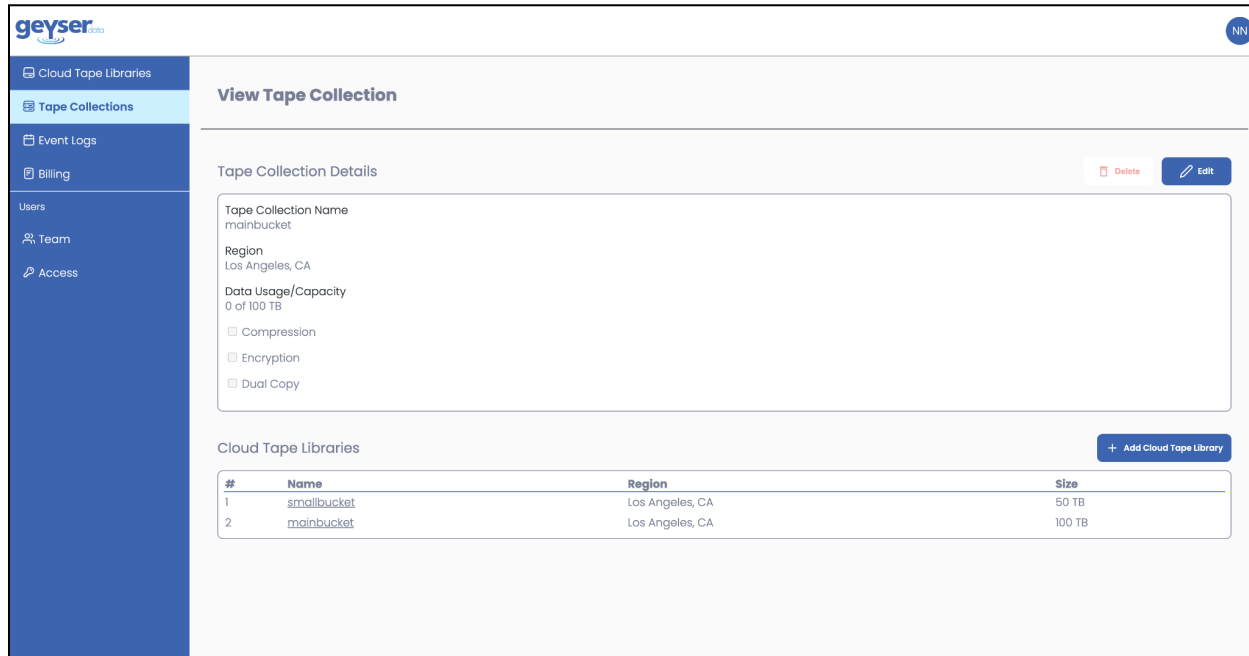
The screenshot shows the Geyser Tape Collections management interface. On the left is a sidebar with navigation links: Cloud Tape Libraries, Tape Collections (highlighted), Event Logs, Billing, and a Users section containing Team and Access. The main content area is titled 'Tape Collections' and features a table with the following data:

Name	Region	Dual/Single Copy	Compression	Encryption	Tapes	Usage/Capacity (TB)
mainbucket	Los Angeles, CA	Single Copy	<input type="checkbox"/>	<input type="checkbox"/>	6	0 of 100 TB

There is an '+ Add Tape Collection' button in the top right corner of the table area.

When you create your first Tape Collection along with the first Cloud Tape Library, you can later add additional libraries to that collection. In the Add Cloud Tape Library screen, simply select the desired Tape Collection.

✓ Create New Tape Collection
mainbucket (100TB)



View Tape Collection

Tape Collection Details Delete Edit

Tape Collection Name
mainbucket

Region
Los Angeles, CA

Data Usage/Capacity
0 of 100 TB

☐ Compression

☐ Encryption

☐ Dual Copy

Cloud Tape Libraries + Add Cloud Tape Library

#	Name	Region	Size
1	smallbucket	Los Angeles, CA	50 TB
2	mainbucket	Los Angeles, CA	100 TB

For example, a single Tape Collection named “mainbucket” may have 2 Cloud Tape Libraries mapped to it: “main bucket” and “small bucket.”

Useful links

[How to use Cyberduck with Geyser data](#)

[S3 Browser Setup Guide](#)

[Veeam Integration Guide](#)

[Cohesity Integration Guide](#)

[MinIO Integration Guide](#)

Support

Have questions or need help?

Contact: support@geyserdata.com