To delete recordings less than 2 MB in your Zoom account, you'll need to use Zoom's REST API to retrieve and manage recordings. The process involves:

1. Authenticating with Zoom's API using a JWT or OAuth token.
2. Fetching a list of recordings.
3. Filtering recordings by size.
4. Deleting recordings that are smaller than 2 MB.

Here’s a Python script to accomplish this:

**Requirements:**

1. **Install dependencies:**

bash

Copy code

pip install requests

1. **Set up a Zoom API app** in the Zoom Marketplace to obtain API credentials.

**Python Code:**

python

Copy code

import requests

# Replace these with your Zoom API credentials

API\_KEY = 'your\_api\_key'

API\_SECRET = 'your\_api\_secret'

TOKEN = 'your\_jwt\_or\_oauth\_token'

BASE\_URL = "https://api.zoom.us/v2"

def get\_recordings(user\_id, page\_token=None):

"""Fetch recordings for a specific user."""

url = f"{BASE\_URL}/users/{user\_id}/recordings"

params = {

"page\_size": 30, # Max items per page

"page\_token": page\_token,

}

headers = {

"Authorization": f"Bearer {TOKEN}"

}

response = requests.get(url, headers=headers, params=params)

response.raise\_for\_status()

return response.json()

def delete\_recording(meeting\_id, recording\_id):

"""Delete a specific recording."""

url = f"{BASE\_URL}/meetings/{meeting\_id}/recordings/{recording\_id}"

headers = {

"Authorization": f"Bearer {TOKEN}"

}

response = requests.delete(url, headers=headers)

if response.status\_code == 204:

print(f"Recording {recording\_id} deleted successfully.")

else:

print(f"Failed to delete recording {recording\_id}. Status code: {response.status\_code}")

def main():

# Replace with the Zoom user ID or "me" for your account

user\_id = "me"

print("Fetching recordings...")

page\_token = None

while True:

recordings\_data = get\_recordings(user\_id, page\_token)

for meeting in recordings\_data.get("meetings", []):

meeting\_id = meeting["id"]

for recording in meeting["recording\_files"]:

recording\_id = recording["id"]

file\_size\_mb = recording["file\_size"] / (1024 \* 1024) # Convert bytes to MB

if file\_size\_mb < 2:

print(f"Deleting recording {recording\_id} (Size: {file\_size\_mb:.2f} MB)...")

delete\_recording(meeting\_id, recording\_id)

# Check for more pages

page\_token = recordings\_data.get("next\_page\_token")

if not page\_token:

break

if \_\_name\_\_ == "\_\_main\_\_":

main()

**How It Works:**

1. **Authentication:**  
   Replace TOKEN with your JWT or OAuth token.
2. **Fetch Recordings:**  
   The script retrieves all recordings associated with your account using the /users/{userId}/recordings endpoint.
3. **Filter by Size:**  
   Each recording file is checked to see if its size is less than 2 MB.
4. **Delete Recordings:**  
   If a recording meets the criteria, the /meetings/{meetingId}/recordings/{recordingId} endpoint is called to delete it.

**Important Notes:**

* Ensure your Zoom account has API access and permissions to manage recordings.
* Test the script in a safe environment first to avoid unintended deletions.
* Modify the page size and rate-limit the requests if needed to comply with Zoom's API rate limits.