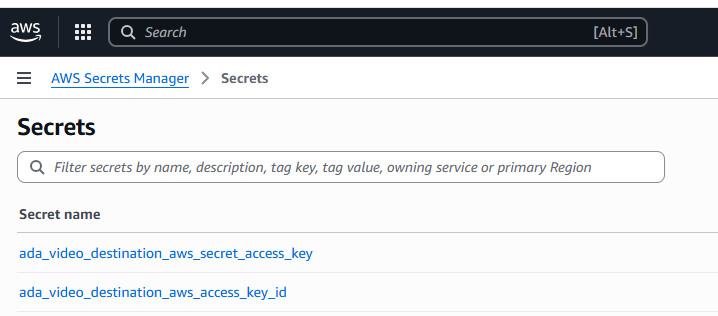
**How to Upload Video Files to the Destination S3 Bucket**

8/13/2025

# **Assumptions**

1. The destination AWS account credentials are stored in the Secret Manager in our current (default) AWS account.



1. The destination AWS account has an S3 bucket created. Bucket naming convention:

ada-video-storage-<12 digit destination account number>

sample name: ada-video-storage-173479170210

1. The credentials file will look like that:

[default]

aws\_access\_key\_id = <default key id>

aws\_secret\_access\_key = <default secret key>

[video\_destination]

aws\_access\_key\_id = <destination key id>

aws\_secret\_access\_key = <destination secret key>

# **Process**

The process includes two periodic processes:

* Requesting AWS credentials and setting the credentials file for the destination account (suggested every 5 min)
* Uploading video files to the destination S3 bucket account as soon as a new video is available

# **How to Get AWS Credentials for the Destination Account**

Run the following Python script from ~/.aws/credentials:

import json

import boto3

from botocore.exceptions import ClientError

region\_name = "us-west-1"

DEFAULT\_AWS\_ACCESS\_KEY\_ID = "<default key id>"

DEFAULT\_AWS\_SECRET\_ACCESS\_KEY = "<destination secret key> "

def get\_secret(secret\_name):

# Create a Secrets Manager client

session = boto3.session.Session()

client = session.client(

service\_name='secretsmanager',

region\_name=region\_name

)

try:

get\_secret\_value\_response = client.get\_secret\_value(

SecretId=secret\_name

)

except ClientError as e:

print(e)

return None

secret\_str = get\_secret\_value\_response['SecretString']

secret\_dict = json.loads(secret\_str)

return json.loads(secret\_str)[list(secret\_dict.keys())[0]]

def create\_aws\_credentials\_file():

aws\_secret\_access\_key = get\_secret("ada\_video\_destination\_aws\_secret\_access\_key")

aws\_access\_key\_id = get\_secret("ada\_video\_destination\_aws\_access\_key\_id")

if aws\_secret\_access\_key is None or aws\_access\_key\_id is None:

print("Error getting AWS Credentials")

return

with open("credentials", "w") as f:

f.write("[default]\n")

f.write("aws\_access\_key\_id = %s\n" % DEFAULT\_AWS\_ACCESS\_KEY\_ID)

f.write("aws\_secret\_access\_key = %s\n" % DEFAULT\_AWS\_SECRET\_ACCESS\_KEY)

f.write("\n")

f.write("[video\_destination]\n")

f.write("aws\_access\_key\_id = %s\n" % aws\_access\_key\_id)

f.write("aws\_secret\_access\_key = %s\n" % aws\_secret\_access\_key)

print("Credentials file has been created")

return

# **How to Upload a File to The Destination S3 Bucket**

Run the following Shell script:

chmod +x upload\_to\_s3.sh

./upload\_to\_s3.sh temp.txt

Script upload\_to\_s3.sh:

#!/bin/bash

# Variables

PROFILE="video\_destination"

BUCKET\_PREFIX="ada-video-storage"

# Check for file argument

if [[ -z "$1" ]]; then

echo "Usage: $0 <file-to-upload>"

exit 1

fi

FILE\_TO\_UPLOAD="$1"

# Check if file exists

if [[ ! -f "$FILE\_TO\_UPLOAD" ]]; then

echo "Error: File '$FILE\_TO\_UPLOAD' not found."

exit 1

fi

# Get AWS account number

ACCOUNT\_ID=$(aws sts get-caller-identity \

--query Account \

--output text \

--profile "$PROFILE")

# Check if we got an account ID

if [[ -z "$ACCOUNT\_ID" ]]; then

echo "Error: Could not retrieve AWS account number for profile '$PROFILE'."

exit 1

fi

# Construct bucket name

BUCKET\_NAME="${BUCKET\_PREFIX}-${ACCOUNT\_ID}"

echo "Uploading $FILE\_TO\_UPLOAD to s3://${BUCKET\_NAME}/$FILE\_TO\_UPLOAD ..."

# Upload the file

aws s3 --profile "$PROFILE" cp "$FILE\_TO\_UPLOAD" "s3://${BUCKET\_NAME}/$FILE\_TO\_UPLOAD"

# Check if upload succeeded

if [[ $? -eq 0 ]]; then

echo "Upload completed successfully."

else

echo "Upload failed."

exit 1

fi