

Python Examples

The following example uses video files on AWS in an S3 bucket and writes files out to a different AWS S3 Bucket.

```
import requests

url = "https://streamengine.igolgi.com/api/v0/job"

headers = {
    'Content-Type': 'application/json',
}

# Authentication credentials (username and password)
auth = ('INSERT_YOUR_API_KEY', 'a')

# Data to be sent in the POST request
data = {
    'videoProfiles': [
        {
            'output': 's3://igolgi-se-out/py2-testcwaudio3d.mp4',
            'width': 1920,
            'height': 1080,
            'video_bitrate_mode': 'cbr',
            'video_bitrate': 2250, # there are minimum bitrates see tables
            'video_format': 'progressive',
            'video_framerate': '1x',
            'audio_profiles': '1' # required if you want audio 0 - 7
        }
    ],
    'audioProfiles': [
        {
            'audio_bitrate': 256,
            'audio_channels': 2,
            'audio_codec': 'aac',
            'source_stream': 1 # required if you want audio 1 - 8
        }
    ]
}
```

```

    }],
    "master_variant_mode": False,
    "output_container": "mp4",
    "video_codec": "h.264",
    "input": "s3://igolgi-se-in/GOPR8297-2.7K-30-fps.MP4",
    "auto_detelecine_flag": False,
    "cloud_credentials": {
        "input": {
            "cloud_provider": "aws",
            "access_key": "INSERT_YOUR_AWS_ACCESS_KEY",
            "secret_key": "INSERT_YOUR_AWS_SECRET_KEY",
            "region": "us-east-2"
        },
        "output": {
            "cloud_provider": "aws",
            "access_key": "INSERT_YOUR_AWS_ACCESS_KEY",
            "secret_key": "INSERT_YOUR_AWS_SECRET_KEY",
            "region": "us-east-2"
        }
    }
}

# Disable SSL verification (not recommended for production)
response = requests.post(url, json=data, headers=headers, auth=auth)

# Print the response content
print(response.content.decode('utf-8'))

```

```

import requests
import sys
import json
from requests.auth import HTTPBasicAuth

if len(sys.argv) > 1:
    external_var = sys.argv[1]
else:
    print("Please provide a job ID as an argument.")
    sys.exit(1)

```

```

# Define the URL and API key
url = f"https://streamengine.igolgi.com/api/v0/job/{external_var}"
api_key = "INSERT_YOUR_API_KEY"

# Make the GET request
response = requests.get(url, auth=HTTPBasicAuth(api_key, ""))

# Check for a successful response
if response.status_code == 200:
    data = response.json()
    status = data.get("status", {})
    config = data.get("config", {})

    print("Job Status:")
    print("-----")
    print(f"Job ID: {status.get('job_id')}")
    print(f"State: {status.get('state')}")
    print("Progress:")
    print(f"  Transcode: {status.get('tcode_progress')}%")
    print(f"  Transfer: {status.get('xfer_progress')}%")
    print("Time:")
    print(f"  Transcode: {status.get('tcode_time')} seconds")
    print(f"  Transfer: {status.get('xfer_time')} seconds")
    print("Speed:")
    print(f"  Transcode: {status.get('tcode_speed')} x")
    print(f"  Transfer: {status.get('xfer_speed')} MB/s")
    print(f"Job Completed: {status.get('job_completed')}")

    print("\nJob Config:")
    print("-----")
    print(json.dumps(config, indent=2))

else:
    print(f"Failed to retrieve job. Status code: {response.status_code}")
    print("Response:", response.text)

```

Job Status:

Job ID: 4920

State: ready

Progress:

Transcode: 100%

Transfer: 0%

Time:

Transcode: 19.94 seconds

Transfer: 27.87 seconds

Speed:

Transcode: 0.0 x

Transfer: 0.0 MB/s

Job Completed: True

Job Config:

{
 "input": "s3://igolgi-se-in/GOPR8297-2.7K-30-fps.MP4",
 "videoProfiles": [
 {
 "output": "s3://igolgi-se-out/go-testcwaudio3f.mp4",
 "width": 1920,
 "height": 1080,
 "video_bitrate_mode": "cbr",
 "video_bitrate": 2301,
 "video_framerate": "1x",
 "audio_profiles": "1"
 }
],
 "audioProfiles": [
 {
 "audio_codec": "aac",
 "audio_channels": 2,
 "audio_bitrate": 256,
 "source_stream": 1
 }
],
 "output_container": "mp4",
 "separate_audio": false,

```
"segmented_output_hls": false,  
"segmented_output_dash": false,  
"video_codec": "h.264",  
"gop_length": 1,  
"scte35_passthrough": false,  
"audio_volume": 100,  
"h264_quality": "good",  
"video_aspect_ratio": "passthrough",  
"picture_transform": "none"  
}
```

Revision #1

Created 16 July 2024 22:25:42 by Admin

Updated 25 July 2024 16:43:34 by Admin