

Making Body-Worn Camera Videos Accessible for TRULEO Audio Analysis

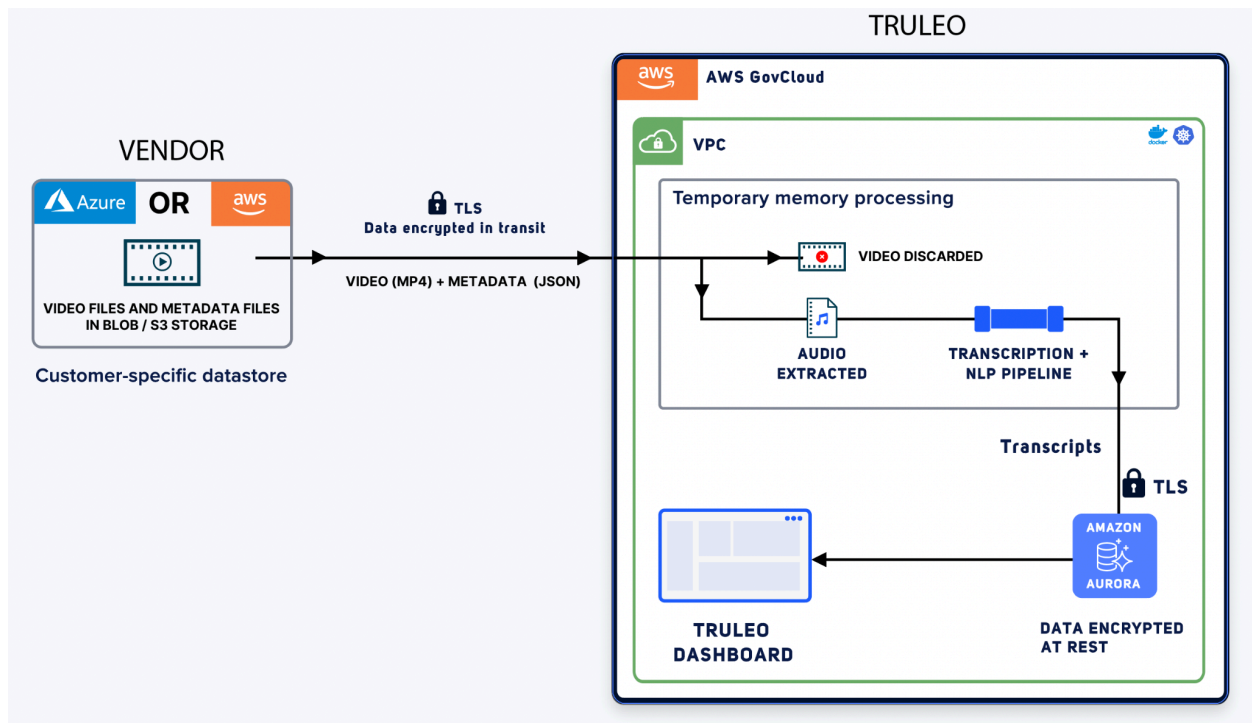
TRULEO's Audio Analytics platform is deployed in AWS GovCloud and securely accesses videos from a vendor or customer's AWS GovCloud or Azure Government Cloud environment. This document outlines how a vendor can make videos available for TRULEO to analyze.

1. Data Flow

TRULEO accesses a department's videos through a secure HTTPS connection to a cloud storage container or bucket (Azure Blob Storage or AWS S3) that a vendor provisions specifically for that customer. Every 12 hours, TRULEO streams all videos that it has not previously analyzed within the data store, analyzes just the audio portion, and stores only transcripts and metadata in a customer-specific datastore.

TRULEO **does not** copy the video from the vendor environment and loses access once the video is gone from the data store. TRULEO requires the video to be present for 60 days, during which a department user can stream the video through the TRULEO application. TRULEO has read-only access to the storage and does not insert transcripts or delete files to or from the vendor data store.

2. Architecture



3. Truleo Requirements for Storage Access

1. Customer-specific Azure Blob Storage Container in Azure Government (Hot Tier) or AWS S3 Standard Bucket in AWS GovCloud
2. Customer-specific read-only access key for storage account
3. Video files in MP4 format with a separate JSON metadata file per video
4. A folder per video containing the video file (MP4) and metadata file (JSON)
5. Data persistence for 60 days, after which data can be deleted from the data store by the vendor

4. File Structure Requirements

4.1 Metadata File Requirements

Officer properties

Required

- id: string
 - Must uniquely identify officer on external system
- first_name: string
- last_name: string
- badge_number: string
- email: string
 - Necessary in order to link Truleo user accounts with Officers

Video properties

Required

- id: string
 - Must uniquely identify video on external system
- officer_id: string
 - Identifier for the officer the video is for
- recording_time: ISO8601 formatted date/time string
 - The timestamp of when the recording took place
- filename: string
 - Filename of the video file
- mimetype: string
 - Mimetype of the video file (e.g. "video/mp4")
- size: number
 - Size of video file in bytes
- duration: number
 - Duration of video in seconds
- url: string
 - URL where the video file can be downloaded

Video properties (continued)

Optional

- latitude: number
- longitude: number
- deletion_time: ISO8601 formatted date/time string
 - Date and time of when the video will be deleted from the external system

4.2 Example Folder Structure

VIDEO-1

- VIDEO-1.mp4
- VIDEO-1.json

VIDEO-2

- VIDEO-2.mp4
- VIDEO-2.json

4.3 Sample metadata file (VIDEO-1.json)

```
{
  "VideoProperties": {
    "id": "VIDEO-1",
    "officer_id": "JDOE",
    "recording_time": "2024-04-09 12:43:13.000"
    "filename": "VIDEO-1.mp4",
    "mimetype": "video/mp4",
    "size": 757998754,
    "duration": 1231,
    "latitude": 43.6652308,
    "longitude": -116.6807927,
    "deletion_time": "2051-08-25 13:55:40.000"
  },
  "OfficerProperties": {
    "id": "JDOE",
    "first_name": "JOHN",
    "last_name": "DOE"
    "badge_number": "JDOE",
    "email": "john.doe@some_customer.gov"
  }
}
```