# **Cloud Video API Document**

baseUrl = "https://open.eye4.cn"

Unified request header**：**

headers["AccessKey"] = "6p\*\*\*\*V"; // Please use your AccessKey

headers["SecretKey"] = "P1\*\*\*K9"; // Please use your SecretKey

**Corded electric camera：**get cloud storage summary(date and corresponding number of videos)

Request method：post

Request URL：$path/D004/summary/show (path and licensenkey parameter gets from motion detection cloud storage api)

Request parameter：

{  
 "licenseKey": licenseKey,  
 "uid": uid,  
 "timeZone": zone,  
}

|  |  |  |
| --- | --- | --- |
| Paramter | Type | Description |
| licenseKey | String | Authorization Information |
| uid | String | camera device id |
| timeZone | String | FlutterNativeTimezone.getLocalTimezone() |

Example of result data：

{“20220224”: 28, “20220225”: 58, “20220226”: 4, “20220227”: 0, “20220228”: 130, “20220301”: 336, “20220302”: 1332, “20220303”: 0}

**Corded electric camera：**get cloud video data for a specified day

Request method：post

Request URL：$path/D004/group/show （path and licensenkey parameter gets from motion detection cloud storage api）

Request parameter：

{  
 "licenseKey": licencekey,  
 "uid": uid,  
 "date": time,  
 "timeZone": zone,  
}

|  |  |  |
| --- | --- | --- |
| parameter | Type | Description |
| licenseKey | String | Authorization Information |
| uid | String | camera device id |
| date | String | Date，Such as：2023-12-09 |
| timeZone | String | FlutterNativeTimezone.getLocalTimezone() |

Example of result data：

[

{

"start": 42629,

"end": 42891,

"duration": 262,

"original": [

{

"key": "2023-11-29:03\_50\_29\_06",

"hour": "11",

"type": "h264",

"eventMark": "1",

"start\_index": 42629,

"end\_index": 42635

},

{

"key": "2023-11-29:03\_50\_35\_06",

"hour": "11",

"type": "h264",

"eventMark": "1",

"start\_index": 42635,

"end\_index": 42641

}

]

}

]

**Corded electric camera：**get cloud video data at a specified time

Request method：post

Request URL：$path/D004/file/url （path and licensenkey parameter gets from motion detection cloud storage api）

Request parameter：

{  
 "licenseKey": licenseKey,  
 "uid": uid,  
 "time": time,  
}

|  |  |  |
| --- | --- | --- |
| Parameter | Type | Description |
| licenseKey | String | Authorization Information |
| uid | String | camera device id |
| time | List<String> | video time，such as：["2023-11-29:03\_50\_29\_06"] |

Example of result data：

[{

"name": "VE0005622QHOW\_2023-11-29:03\_50\_29\_06",

"url": "<http://d004-vstc.eye4.cn/VE0005622QHOW_2023-11-29:03_50_29_06?e=1701264785&token=l5gvKghs6BCqoVtQJOkLwykc7JtTnXvUCGgl2AzZ:3QHpPvjYp2LZrQ0tpCjG1AqzYmk=>"

}]

**Corded electric camera：**get video cover data from cloud storage

Request method：post

Request URL：$path/D004/cover （path and licensenkey parameter gets from motion detection cloud storage api）

Request parameter：

{  
 "licenseKey": licenseKey,  
 "uid": uid,  
 "url": url,  
}

|  |  |  |
| --- | --- | --- |
| Parameter | Type | Description |
| licenseKey | String | Authorization Information |
| uid | String | camera device id |
| url | String | first url at video data list |

**Low-power camera：**Get cloud video data

Request method：post

Request URL：/push/fileid

Request parameter：

jsonEncode({  
 "fileid": fileId, //gets from Message  
 "type": fileType //"D009"  
})

|  |  |  |
| --- | --- | --- |
| Parameter | Type | Description |
| fileid | String | file id, gets from Message |
| fileType | String | file Type: ”D009“ |

Example of result data：

[

{

"file\_name": "<http://d015-z0.eye4.cn//tmp/HTB0005151PQSU_2023-12-08-07-57-50_01_0?e=1702277606&token=l5gvKghs6BCqoVtQJOkLwykc7JtTnXvUCGgl2AzZ:gL0UVClR9oFAcpSuuh534cqGk2k=>",

"file\_Type": "video"

},

{

"file\_name": "<http://d015-z0.eye4.cn//tmp/HTB0005151PQSU_2023-12-08-07-57-55_01_1?e=1702277606&token=l5gvKghs6BCqoVtQJOkLwykc7JtTnXvUCGgl2AzZ:JUzTf-lgT0vSZlVGtV899ddOg50=>",

"file\_Type": "video"

}

]

Note：The device records in segments, each lasting 5 seconds, and the total length of a complete video should be the sum of the durations of multiple videos stitched together from the array.

The video source is: NetworkVideoSource(urls)