

# API Monuv Câmeras em Nuvem

Endpoint: <https://app.monuv.com.br>

## Validação de usuário/senha

Caminho: /api/authenticate

Método: POST

Parâmetros:

- email - Email do usuário na Monuv
- password - Senha na Monuv
- w - Código de whitelabel (somente para clientes que possuam whitelabel) - não obrigatório

Tipo do retorno: JSON

Retorno: Objeto com os campos:

- token - O token retornado deve ser usado nas próximas chamadas a API
- logo - Link para o logo da empresa associada ao usuário

## Listagem de câmeras

Caminho: api/cameras

Método: GET

Parâmetros:

- token - obtido no momento do login
- p - Formato do stream - Passar p=hls para streams no formato HLS da Apple

Tipo do retorno: JSON

Retorno: Array de câmeras

Campos do objeto câmera retornado no array:

- **id** - Id da câmera
- **description** - Descrição da câmera (nome)
- **camera\_app** - Aplicação onde a câmera está instalada
- **live\_url** - URL para obtenção do vídeo ao vivo (essa url deve ser passada para o player de vídeo)
- **server\_url** - URL do servidor onde a câmera está instalada
- **thumb\_url** - URL para o thumbnail da câmera
- **digest** - Digest para as chamadas a API da câmera
- **hash** - Hash usado em chamadas a vídeos da câmera
- **status** - (ONLINE, OFFLINE, ERRO CONEXÃO)
- **client\_id** - Id do cliente da câmera
- **client\_logo** - Link para o logo do cliente da câmera

## Obter player da câmera

Caminho: `api/camera/{id}/player`

Método: GET

Parâmetros:

- token - Token obtido no método de validação de usuário/senha
- id - Id da câmera
- ts - timestamp linux da gravação (passar 0 para vídeo ao vivo)
- delay = delay em segundos para trás em relação ao timestamp passado (Ex: para ver 10 segundos antes, passar delay=10)

Tipo de retorno: HTML com o player da câmera

## Obter timeline de vídeos

Caminho: `/camera/{id}/timeline`

Método: GET

Parâmetros:

- token - Token obtido no método de validação de usuário/senha
- id - ID da câmera (passado na URL)
- digest - Digest da câmera - Obtido no método de listar câmeras
- timeStamp - timestamp (com milissegundos) do momento que você quer o timeline. Se for vídeo ao vivo, é o timestamp do horário atual. Se for gravado, é o timestamp da data/horário da busca.

Tipo do Retorno: JSON

Retorno: Array com os campos:

- CODE - Código de retorno - 0 - SUCESSO, <> 0 - FALHA
- DATA - array de timeline (campos descritos abaixo)
- MSG - Em caso de erro, mensagem de retorno

Array de timeline:

- ini - timestamp do início do vídeo (com milissegundos)
- dur - duração do vídeo (em milissegundos)
- file - Nome do arquivo do vídeo

## Obter gravações

Caminho: `/camera/{id}/recordings`

Método: GET

Parâmetros:

- token - Token obtido no método de validação de usuário/senha
- id - ID da câmera (passado na URL)
- digest - Digest da câmera - Obtido no método de listar câmeras
- timeStamp - timestamp (com milissegundos) do momento que você quer as gravações
- p - Formato do stream - Passar p=hls para streams no formato HLS da Apple

Tipo do Retorno: JSON

Retorno: Array com os campos:

- CODE - Código de retorno - 0 - SUCESSO, <> 0 - FALHA

- DATA - array de recordings (campos descritos abaixo)
- MSG - Em caso de erro, mensagem de retorno

Array de recordings:

- ini - timestamp do início do vídeo (com milisegundos)
- start\_date - Data de início do vídeo (YYYY-MM-DD HH:ii:ss)
- dur - duração do vídeo (em milisegundos)
- file - Nome do arquivo do vídeo
- url - Url para o vídeo (essa url deve ser passada para o player de vídeo)

## Listagem de imagens hora a hora

Caminho: api/camera/{id}/thumbs-hourly

Método: GET

Parâmetros:

- id - ID da câmera (passado na URL)
- ts - Timestamp (segundos) do dia da consulta
- digest - Digest da câmera - Obtido no método de listar câmeras

Tipo do retorno: JSON

Retorno: Array de links de thumbnails

- **INVALID\_USER** - usuário inválido
- **INVALID\_CAMERA** - câmera não encontrada ou acesso não permitido
- **ACCESS\_DENIED** - acesso não permitido

## Listagem de imagens minuto a minuto

Caminho: api/camera/{id}/thumbs-minutes

Método: GET

Parâmetros:

- id - ID da câmera (passado na URL)
- ts - Timestamp (segundos) da hora da consulta
- digest - Digest da câmera - Obtido no método de listar câmeras

Tipo do retorno: JSON

Retorno: Array de links de thumbnails

- **INVALID\_USER** - usuário inválido
- **INVALID\_CAMERA** - câmera não encontrada ou acesso não permitido
- **ACCESS\_DENIED** - acesso não permitido

## Obter url do vídeo ao vivo da câmera

Caminho: `api/camera/{id}/live-url`

Método: GET

Parâmetros:

- token - Token obtido no método de validação de usuário/senha
- id - Id da câmera

Tipo de retorno: JSON

Retorno

- code - Código de retorno (0 para sucesso)
- msg - Mensagem de retorno da API
- data - Url para o vídeo ao vivo da câmera

## Criar download de vídeo

Caminho: `v2/api/drives`

Método: POST

Parâmetros:

- token - Token obtido no método de validação de usuário/senha
- camera\_id - Id da câmera
- description - descrição do download a ser gerado
- start\_at - timestamp (milissegundos) de início
- end\_at - timestamp (milissegundos) de término

Tipo de retorno: JSON

Retorno

- CODE - Código de retorno (0 para sucesso)
- MSG - Mensagem de retorno da API
- DATA - objeto com os seguintes campos:
  - camera\_id - Id da câmera
  - description - descrição do download
  - start\_date - início do vídeo no formato YYYY-MM-DD HH:MM:SS
  - end\_date - término do vídeo no formato YYYY-MM-DD HH:MM:SS
  - create\_user\_id - Id do usuário que criou o download
  - update\_user\_id - Id do usuário que atualizou o download
  - status - código 0 na criação
  - updated\_at - data de atualização do download
  - created\_at - data de criação do download
  - id - Id do download
  - camera\_name - descrição da câmera
  - last\_thumb - Thumbnail do vídeo gerado
  - StartDateF - início do vídeo no formato DD/MM/YYYY HH:MM
  - EndDateF - término do vídeo no formato DD/MM/YYYY HH:MM
  - UriDrive - URL do vídeo gerado

## Listar downloads de vídeo

Caminho: v2/api/drives

Método: GET

Parâmetros:

- token - Token obtido no método de validação de usuário/senha
- ids - IDs dos Downloads de vídeos separados por vírgula, limite de 100 IDs (se nenhum for fornecido retornará todos os downloads gerados pelo usuário)

Tipo de retorno: JSON

Retorno

- CODE - Código de retorno (0 para sucesso)
- MSG - Mensagem de retorno da API
- DATA - array de objetos de Downloads de vídeo com os seguintes campos:
  - id - Id do download
  - description - descrição do download
  - url - URL do vídeo gerado
  - startDateF - início do vídeo no formato DD/MM/YYYY HH:MM
  - endDateF - término do vídeo no formato DD/MM/YYYY HH:MM
  - tsStart - timestamp de início
  - tsEnd - timestamp de término
  - camera\_name - descrição da câmera
  - status - código 1 se gerado com sucesso ou código 2 se houve problema
  - last\_thumb - Último thumbnail da câmera gerado

## Apagar download de vídeo

Caminho: v2/api/drives/delete/{id}

Método: POST

Parâmetros:

- token - Token obtido no método de validação de usuário/senha
- id - Id do Download de vídeo

Tipo de retorno: JSON

Retorno

- CODE - Código de retorno (0 para sucesso)
- MSG - Mensagem de retorno da API