

## Table of Contents *generated with [DocToc](#)*

- [IPCam SDK API Guide](#)
  - [1. 接口文档](#)
    - [1.1 分配句柄](#)
    - [1.2 初始化句柄](#)
    - [1.3 删除句柄](#)
    - [1.4 连接推流服务器](#)
    - [1.5 设置音频时间基准](#)
    - [1.6 设置视频时间基准](#)
    - [1.7 设置视频 PPS、SPS、SEI 参数](#)
    - [1.8 发送视频关键帧](#)
    - [1.9 发送非视频关键帧](#)
    - [1.10 发送音频数据](#)
    - [1.11 设置 AAC 音频配置数据](#)
  - [2. 使用示例](#)
    - [2.1 初始化](#)
    - [2.2 发送数据](#)
      - [2.2.1 发送视频数据](#)
      - [2.2.2 发送音频数据](#)
  - [2.3 销毁句柄释放内存](#)

# IPCam SDK API Guide

## 1. 接口文档

---

### 1.1 分配句柄

```
RtmpPubContext * RtmpPubNew(const char * _url, unsigned int _nTimeout, RtmpPubAudioType _nInputAudioType, RtmpPubAudioType _nOutputAudioType, RtmpPubTimeStampPolicy _nTimePolicy);
```

参数名	描述
url	RTMP 推流地址
nTimeout	发送及接收超时时间，单位：秒
nInputAudioType	音频输入类型，包括： RTMPPUBAUDIOAAC、RTMPPUBAUDIOG711A、RTMPPUBAUDIOG711U、 RTMPPUBAUDOPCM、RTMPPUBAUDIO_NONE
nOutputAudioType	值只能和 <i>nInputAudioType</i> 相同或者 RTMPPUBAUDIOAAC，输出音频不能为 RTMPPUBAUDO_PCM
nTimePolicy	RTMPPUBTIMESTAMPABSOLUTE：数据包使用绝对时间戳发送； RTNPPUBTIMESTAMPRELATIVE：数据包尽量使用相对时间戳发送，如果时间 发生回转，会发送绝对时间戳进行时间戳同步

其中，**RtmpPubAudioType** 的定义：

值	描述
RTMPPUBAUDIO_AAC	AAC 音频类型
RTMPPUBAUDIO_G711A	G711A 音频类型
RTMPPUBAUDIO_G711U	G711U音频类型
RTMPPUBAUDIO_PCM	PCM 16位音频类型
RTMPPUBAUDIO_NONE	无音频数据

其中，**RtmpPubTimeStampPolicy** 的定义：

值	描述
RTMPPUBTIMESTAMP_ABSOLUTE	绝对时间戳
RTMPPUBTIMESTAMP_RELATIVE	相对时间戳

返回值：

成功返回 **RtmpPubContext** 指针，失败返回NULL

## 1.2 初始化句柄

```
int RtmpPubInit(RtmpPubContext * _pRtmp);
```

初始化由 RtmpPubNew 分配的 RtmpPubContext \* 句柄

参数名	描述
_pRtmp	RtmpPubContext 指针

返回值：

成功返回0，失败返回 -1，失败时，需要调用RtmpPubDel释放资源

注意：

输入音频为 G711A/G711U 并且输出音频为 AAC 时，默认会以以下内容初始化句柄：

- 音频采样频率：8000Hz
- 通道数量：1
- 采样位宽：16位

### 1.3 删除句柄

```
void RtmpPubDel(RtmpPubContext * _pRtmp);
```

删除并释放 RtmpPubContext

参数名	描述
_pRtmp	RtmpPubContext 指针

### 1.4 连接推流服务器

```
int RtmpPubConnect(RtmpPubContext * _pRtmp);
```

参数名	描述
_pRtmp	RtmpContext 指针

返回值：

成功返回 0，失败返回 -1

### 1.5 设置音频时间基准

```
void RtmpPubSetAudioTimebase(RtmpPubContext * _pRtmp, unsigned int _nTimeStamp);
```

参数名	描述
_pRtmp	RtmpPubContext 指针
_nTimeStamp	音频时间基准，和第一个音频包的时间戳相同，后续音频包时间戳在此基准上递增

## 1.6 设置视频时间基准

```
void RtmpPubSetVideoTimebase(RtmpPubContext * _pRtmp, unsigned int _nTimeStamp);
```

参数名	描述
_pRtmp	RtmpPubContext 指针
_nTimeStamp	视频时间基准，和第一个视频包的时间戳相同，后续视频包时间戳在此基准上递增

## 1.7 设置视频 PPS、SPS、SEI 参数

发送 VIDEO 数据之前，必须至少设置 PPS 和 SPS

```
void RtmpPubSetPps(RtmpPubContext * _pRtmp, const char * _pData, unsigned int _nSize);
void RtmpPubSetSps(RtmpPubContext * _pRtmp, const char * _pData, unsigned int _nSize);
void RtmpPubSetSei(RtmpPubContext * _pRtmp, const char * _pData, unsigned int _nSize);
```

参数名	描述
_pRtmp	RtmpPubContext 指针
_pData	PPS/SPS/SEI 数据
_nSize	PPS/SPS/SEI 字节数

## 1.8 发送视频关键帧

```
int RtmpPubSendVideoKeyframe(RtmpPubContext * _pRtmp, const char * _pData, unsigned int _nSize, unsigned int _presentationTime);
```

发送视频关键帧

参数名	描述
_pRtmp	RtmpPubContext 指针
_pData	关键帧数据
_nSize	关键帧数据字节数
_presentationTime	pts

返回值：

成功返回 0，失败返回 -1，失败时，表明连接已经失效

## 1.9 发送非视频关键帧

```
int RtmpPubSendVideoInterframe(RtmpPubContext * _pRtmp, const char * _pData, unsigned int _nSize, unsigned int _presentationTime);
```

发送视频非关键帧数据

参数名	描述
_pRtmp	RtmpPubContext 指针
_pData	非关键帧数据
_nSize	非关键帧数据字节数
_presentationTime	pts

返回值：

成功返回 0，失败返回 -1，失败时，表明连接已经失效

## 1.10 发送音频数据

```
int RtmpPubSendAudioFrame(RtmpPubContext * _pRtmp, const char * _pData, unsigned int _nSize, int _nPresentationTime);
```

如果输入数据类型为 RTMP\_PUBAUDIO\_AAC，发送音频数据前需要先调用 RtmpPubSetAac 发送 AAC 音频配置数据

参数名	描述
_pRtmp	RtmpPubContext 指针
_pData	音频数据
_nSize	音频数据字节数
_presentationTime	pts

返回值：

成功返回 0，失败返回 -1，失败时，表明连接已经失效

### 1.11 设置 AAC 音频配置数据

```
void RtmpPubSetAac(RtmpPubContext * _pRtmp, const char * _pAacCfgRecord, unsigned int _nSize);
```

如果输入数据类型为 RTMPPUBAUDIO\_AAC，发送 AAC 数据前，需要先调用此函数设置 AAC Configuration Record

参数名	描述
_pRtmp	RtmpPubContext 指针
_pAacCfgRecord	AAC 配置数据
_nSize	配置数据字节数

## 2. 使用示例

示例中没有对返回值进行处理，实际使用时需要处理返回值。

### 2.1 初始化

```
// 分配句柄，读写超时时间为10s，输入音频类型为G711A，输出音频类型为AAC,并使用绝对时间戳策略
RtmpPubContext * pRtmpc = RtmpPubNew(serverUrl, 10, RTMP_PUB_AUDIO_G711A, RTMP_PUB_AUDIO_AAC, RTMP_PUB_TIMESTAMP_ABSOLUTE);
//初始化句柄
RtmpPubInit(pRtmpc);
//连接服务器
RtmpPubConnect(pRtmpc);
```

## 2.2 发送数据

### 2.2.1 发送视频数据

```
if (isVideoInit == false) {
    //初始化视频时间基准, presentationTime为第一个视频包时间戳
    RtmpPubSetVideoTimeBase(presentationTime)
    //设置SPS及PPS参数
    RtmpPubSetSps(pRtmpc, spsData, spsDataSize);
    RtmpPubSetPps(pRtmpc, ppsData, ppsDataSize);
    isVideoInit = true;
}
if (isKeyFrame == true) {
    //发送关键帧数据
    RtmpPubSendVideoKeyframe(pRtmpc, frameData, frameDataSize, presentationTime);
}
if (isInterFrame == true) {
    //发送非关键帧数据
    RtmpPubSendVideoKeyframe(pRtmpc, frameData, frameDataSize, presentationTime);
}
```

### 2.2.2 发送音频数据

- 输入音频类型为 AAC

```
if (isAudioInit == false) {
    //初始化音频时间基准, presentationTime和第一个音频包的时间戳相同
    RtmpPubSetAudioTimebase(pRtmpc, presentationTime);
    //设置AAC配置数据
    RtmpPubSetAac(pRtmpc, pAacCfgRecord, cfgSize);
    isAudioInit = true;
}
//发送音频数据
RtmpPubSendAudioFrame(pRtmpc, pAacData, aacDataSize, presentationTime);
```

- 输入音频类型为 G711A、G711U 或者 PCM 16位

```
if (isAudioInit == false) {
    //初始化音频时间基准, presentationTime和第一个音频包的时间戳相同
    RtmpPubSetAudioTimebase(pRtmpc, presentationTime);
    isAudioInit = true;
}
//发送音频数据
RtmpPubSendAudioFrame(pRtmpc, pAudioData, audioDataSize, presentationTime);
```

## 2.3 销毁句柄释放内存

```
//当程序退出、数据发送完成等情况释放资源  
RtmpPubDel(pRtmpc)
```