

视频禁播与解禁

接口描述

- 1、通过视频id将视频修改为解禁状态或禁播状态，解禁状态的视频可以正常播放，禁播状态的视频无法播放，一次最多只能操作500个视频id

2、接口URL中的{userid}为点播账号userid，具体参考菜单【使用须知】->【获取密钥】

3、支持https协议

接口URL

http://api.polyv.net/v2/video/{userid}/forbidden

在线API调用

请求方式

POST

接口约束

- 1、接口同时支持HTTP 、HTTPS ， 建议使用HTTPS 确保接口安全，接口调用有频率限制，[详细请查看](#)
- 2、注意点： 只能修改“已发布”状态的视频为禁播状态， 只能修改“已禁播”状态的视频为已发布状态， 当请求中的视频id包含多种状态时， 只对符合条件的视频id进行状态修改操作，并返回成功；若没有符合条件的视频id则返回错误。

请求参数描述

参数名	必选	类型	说明
ptime	true	Long	当前时间的毫秒级时间戳，3分钟内有效

参数名	必选	类型	说明
sign	true	String	签名，为40位大写的SHA1值, 生成签名的secretkey密钥作为通信数据安全的关键信息，严禁保存在客户端直接使用，所有API都必须通过客户自己服务器中转调用POLYV服务器获取响应数据【详见 签名生成规则 】
vids	true	String	视频id，多个视频以英文逗号分隔，如 a2dc4f25179499d6586362672838cc2d_a,a2dc4f25179499d6586362672838cc2d_a 一次最多只能传500个视频id
forbidden	true	Integer	视频状态 1：禁播 0：解禁

示例

```
http://api.polyv.net/v2/video/1b448be323/forbidden
```

表单参数：

```
vids=1b448be3239c2ef0cb3ab9fd105f7fb2_1&forbidden=1&sign=5FBEB20C93D6FB6D715756CA0D25DBE96EFAD05A&ptime=1617156768546
```

响应参数描述

参数名	类型	说明
code	Integer	响应状态码，200为成功返回，非200为失败【详见 全局错误说明 】
status	String	响应状态文本信息

参数名	类型	说明
message	String	响应描述信息，当code为400或者500的时候，辅助描述错误原因
data	String	响应成功和失败都返回空字符串

Java请求示例

快速接入基础代码请[下载相关依赖源码](#)，[点击下载源代码](#)，下载后加入到自己的源码工程中即可。测试用例中的**HttpUtil.java** 和 **VodSignUtil.java** 都包含在下载文件中。

强烈建议您使用[点播Java SDK](#)完成API的功能对接，点播Java SDK 对API调用逻辑、异常处理、数据签名、HTTP请求线程池进行了统一封装和优化。

```
private static final Logger log = LoggerFactory.getLogger(VodVideoManageManagementTest.class);
/**
 *视频禁播与解禁
 */
@Test
public void testSetForbiddenStatus() throws Exception, NoSuchAlgorithmException {
    //公共参数,填写自己的实际参数
    String secretKey = super.secretKey;
    String userId = super.userId;
    String ptime = String.valueOf(System.currentTimeMillis());
    //业务参数
    String url = "http://api.polyv.net/v2/video/"+userId+"/forbidden";
    String vids="1b448be3239c2ef0cb3ab9fd105f7fb2_1";
    String forbidden=String.valueOf(1);

    //调用参数
    Map<String, String> requestMap = new HashMap<>();
    requestMap.put("userId", userId);
    requestMap.put("ptime", ptime);
    requestMap.put("vids", vids);
    requestMap.put("forbidden", forbidden);
    requestMap.put("sign", VodSignUtil.getSign(requestMap, secretKey));
    String response = HttpUtil.postFormBody(url, requestMap);
    log.debug("测试视频禁播与解禁,{", response);
    //do somethings
}
```

响应示例

系统全局错误说明详见[全局错误说明](#)

成功示例

```
{
  "code": 200,
  "status": "success",
  "message": "success",
  "data": ""
}
```

异常示例

签名错误

```
{
  "code": 400,
  "status": "error",
  "message": "the sign is not right.",
  "data": ""
}
```

时间戳过期

```
{
  "code": 400,
  "status": "error",
  "message": "ptime is too old",
  "data": ""
}
```

一次最多只能操作500个vid

```
{
  "code": 400,
  "status": "error",
  "message": "vid count can't over 500",
  "data": ""
}
```

只能修改“已发布”状态的视频为禁播状态

```
{
  "code": 400,
  "status": "error",
  "message": "只能修改“已发布”状态的视频为禁播状态",
  "data": ""
}
```

只能修改“已禁播”状态的视频为已发布状态

```
{
  "code": 400,
  "status": "error",
  "message": "只能修改“已禁播”状态的视频为已发布状态",
  "data": ""
}
```