

Follow5 API 开发手册

Ver0.71

www.follow5.com

目 录

1	基本概念.....	4
1.1	什么是Follow5API.....	4
1.2	Api_key申请.....	4
1.3	使用 API 的简单途径.....	5
1.4	访问限制Rate limiting.....	5
2	API说明.....	6
2.1	认证.....	6
2.2	分享.....	6
2.2.1	显示最新分享.....	6
2.2.2	显示用户和好友, 关注的人发送给所有人的分享.....	7
2.2.3	显示用户分享(用户分享所有人).....	7
2.2.4	显示用户和好友的私密分享.....	8
2.2.5	显示用户发送给自己的分享.....	8
2.2.6	获取一条指定的分享.....	9
2.2.7	发布分享.....	10
2.2.8	删除分享.....	11
2.2.9	发布分享给指定好友.....	11
2.2.10	发布分享到组.....	12
2.2.11	显示"@用户" 的分享.....	13
2.3	用户操作接口.....	14
2.3.1	显示用户好友列表.....	14
2.3.2	显示用户关注列表.....	15
2.3.3	显示用户被关注列表.....	15
2.3.4	显示用户信息.....	16
2.3.5	添加好友.....	17
2.3.6	删除好友.....	18
2.3.7	关注用户.....	18
2.3.8	取消关注用户.....	19
2.3.9	判断是否为好友.....	20
2.3.10	判断是否为关注.....	20
2.3.11	显示用户的组列表.....	21
2.3.12	显示用户指定组的成员列表.....	22
2.4	用户帐号验证接口.....	22
2.4.1	用户帐号验证.....	22
2.5	用户回复接口.....	23
2.5.1	获取每条分享的回复.....	23
2.5.2	发布分享回复.....	25
2.5.3	获取“回复我的”回复列表.....	25
2.6	悄悄话接口.....	26
2.6.1	发送悄悄话.....	26
2.6.2	删除悄悄话.....	27
2.6.3	显示悄悄话.....	28
3	错误码.....	30
4	常见问题.....	30

4.1	为什么API_key申请石沉大海没有反馈了?	30
4.2	为什么“来自”不是我事先指定的名称?	30
4.3	为什么有的时候我通过API发布的分享不能够通过 5 步大同分享到其他网站?	30
5	附录	31
5.1	Follow5 开放平台开发者协议	31
5.2	定义	31
5.3	Follow5 开放平台用户管理规则	31
5.4	开发者管理规则	31
5.5	开放平台服务的变更与中断.....	31
5.6	知识产权	32
5.7	协议修改	32
5.8	责任限制	32

1 基本概念

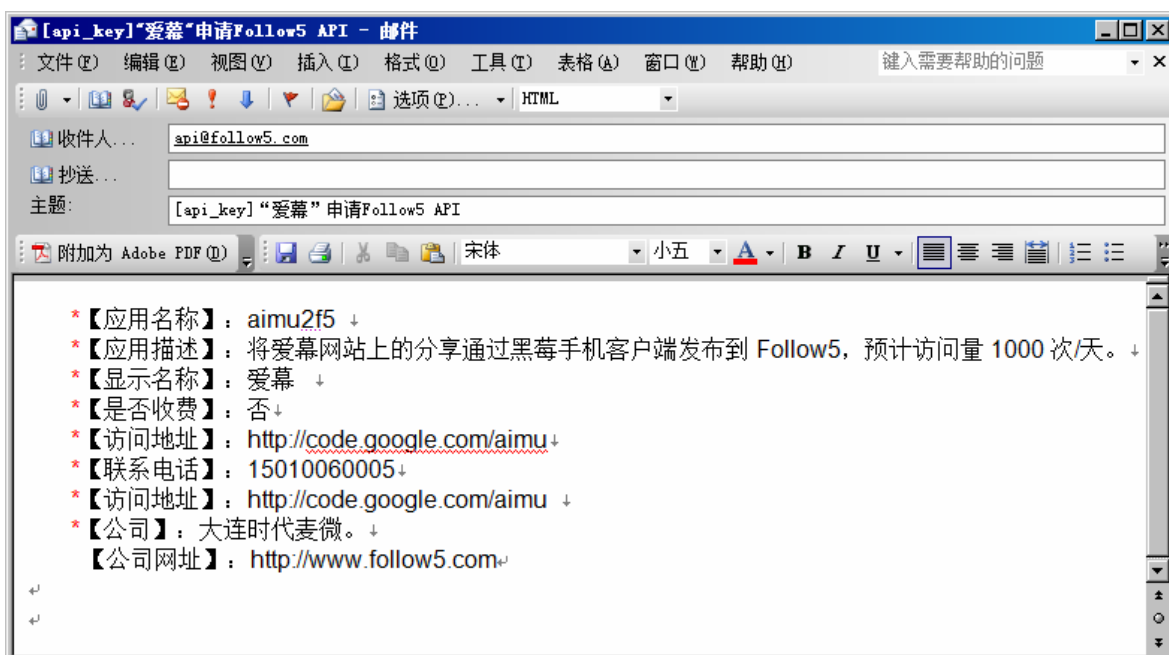
1.1 什么是Follow5API

Follow5 开放平台API是Follow5 平台为第三方开发者和合作伙伴开发的外部接口,所有Follow5 开放平台的用户或开发者在使用Follow5 开放平台API的时候,都必须遵循《follow5 开放平台开发者协议》,具体内容请[参考附录](#)。

1.2 Api_key申请

- Follow5 开放平台 API 中所有的 url 请求中必须带 api_key 参数。
- 开发者可以通过发送邮件到 api@follow5.com 来申请 api_key。
- 申请 api_key 的 邮件格式如下:
 - 邮件标题: [api_key] 申请标题
 - 邮件正文:
 - * 【应用名称】: 应用程序名称
 - * 【应用描述】: 表述应用程序功能概要、用途、访问量 估计
 - * 【显示名称】: 在 follow5.com 上 显示的名称 (follow5.com 上会 显示“通过 xxx 分享”)
 - * 【是否收费】: 是/否 (描述该应用程序是否收费)
 - * 【访问地址】: 可以下载应用程序或使用本 api 的 网站
 - * 【联系电话】: 填写常用的电话号码, Follow5 会通过电话方式核实申请人的真实性
 - * 【公司】: 应用程序的所属组织, 个人申请则填写 “个人” 既可。
 - 【公司网址】: 应用程序所属组织的网址

邮件示例



注: “*”为必填项, 完整的填写信息会有助于您顺利的申请 api_key

1.3 使用 API 的简单途径

如果你的系统中有 [cURL](#)，就可以通过非常简单的方式使用这些API了。

示例：

- 显示最新分享的消息

```
curl http://api.follow5.com/api/statuses/public_timeline.xml?api_key=123
```

- 显示用户的分享

```
curl -u loginname:password http://api.follow5.com/api/statuses/user_timeline.xml?api_key=123
```

- 显示用户和好友的信息

```
curl -u loginname:password http://api.follow5.com/api/statuses/friends_timeline.xml?api_key=123
```

- 发布消息

```
curl-u loginname:password -d status="notes"
```

```
http://api.follow5.com/api/statuses/update.xml?api\_key=123
```

1.4 访问限制Rate limiting

API 限制客户端每小时只能执行有限的请求。

REST API Rate Limiting

默认 REST API 的访问限制是每小时 300 次，限制分用户和 IP，未授权的访问次数限制主要针对 IP，登录后的请求访问限制主要针对用户。（由于目前 API 处于测试阶段，所有的 API 用户都将使用测试授权，以后我们会对权限进行补充）。

测试授权限制情况：

请求限制：单用户每小时 300 次

发表微博：单用户每小时最大 60 次

发表回复：单用户每小时最大 120 次

发表私信：单用户每小时最大 120 次

添加关注：每小时最多添加关注 120 个，每天最多关注 400 个

黑名单

我们希望 API 调用者都能遵循请求限制，过度频率的调用 API 会导致你的应用/IP 加入黑名单。加入黑名单之后，所有请求都会无任何返回。

2 API说明

2.1 认证

如需要验证用户，验证方式为 [HTTP Basic](#) 验证。

2.2 分享

2.2.1 显示最新分享

说明：查看最新分享

路径： [http://api.follow5.com/api/statuses/public_timeline.\[json|xml\]](http://api.follow5.com/api/statuses/public_timeline.[json|xml])

参数：

- `api_key` (必选) – `api_key`，每次请求必选传入此参数，否则用户将没有权限访问 `api`。
- `count` (可选) - 消息数，范围 1-20，默认为 20。

示例：http://api.follow5.com/api/statuses/public_timeline.xml?count=10&api_key=123

格式： xml, json

HTTP Method(s): GET

需要验证： NO

XML 格式：

```
<?xml version="1.0" encoding="UTF-8"?>
<statuses>
  <status>
    <created_at>留言时间</created_at>
    <id>留言ID</id>
    <text>留言内容</text>
    <source>分享方式</source>
    <receiver>接收人</receiver>
    <link>link</link>
    <link_cover>link封皮</link_cover>
    <image_address>图片地址</image_address>
    <longitude>经度</longitude>
    <latitude>纬度</latitude>
    <reply_count>回复数量</reply_count>
    <user>
      <id>用户ID</id>
      <name>用户名</name>
      <profile_image_url>头像地址</profile_image_url>
      <url>空间地址</url>
    </user>
  </status>
</statuses>
```

```
<set>
  <id>组ID</id>
  <name>组的名称</name>
</set>

</status>
... 更多 ...
</statuses>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.2.2 显示用户和好友，关注的人发送给所有人的分享

说明: 查看用户自己、好友、关注的人发送给所有人的分享

路径: [http://api.follow5.com/api/statuses/friends_timeline.\[json|xml\]](http://api.follow5.com/api/statuses/friends_timeline.[json|xml])

参数:

- **api_key** (必选) – **api_key**, 每次请求必选传入此参数, 否则用户将没有权限访问 **api**。
- **count** (可选) - 消息数, 范围 1-20, 默认为 20。
示例: http://api.follow5.com/api/statuses/friends_timeline.xml?count=10&api_key=123
- **page** (可选) - 页码, 从 1 开始
示例: http://api.follow5.com/api/statuses/friends_timeline.xml?page=1&api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: YES

XML 格式:

同 2.1.1 格式

JSON 格式:

同 2.1.1 格式

异常:

错误返回 HTTP 的 400

2.2.3 显示用户分享(用户分享所有人)

说明: 查看用户分享给所用人的分享

路径: [http://api.follow5.com/api/statuses/user_timeline.\[json|xml\]](http://api.follow5.com/api/statuses/user_timeline.[json|xml])

参数:

- **api_key** (必选) – **api_key**, 每次请求必选传入此参数, 否则用户将没有权限访问 **api**。
- **count** (可选) - 消息数, 范围 1-20, 默认为 20。
示例: http://api.follow5.com/api/statuses/user_timeline.xml?count=10&api_key=123
- **page** (可选) - 页码, 从 1 开始
示例: http://api.follow5.com/api/statuses/user_timeline.xml?page=1&api_key=123

- **id (必选)** – 用户 id, 没有此参数时需 验证用户。
示例: http://api.follow5.com/api/statuses/user_timeline.xml?id=100025&api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: YES

XML 格式:

同 2.1.1 格式

JSON 格式:

同 2.1.1 格式

异常:

错误返回 HTTP 的 400

2.2.4 显示用户和好友的私密分享

说明: 查看用户和其好友的私密分享

路径: [http://api.follow5.com/api/statuses/private_friend.\[json|xml\]](http://api.follow5.com/api/statuses/private_friend.[json|xml])

参数:

- **api_key (必选)** – api_key, 每次请求必选传入此参数, 否则用户将没有权限访问 api。
- **count (可选)** - 消息数, 范围 1-20, 默认为 20。
示例: http://api.follow5.com/api/statuses/private_friend.xml?count=10&api_key=123
- **page (可选)** - 页码, 从 1 开始
示例: http://api.follow5.com/api/statuses/private_friend.xml?page=1&api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: YES

XML 格式:

同 2.1.1 格式

JSON 格式:

同 2.1.1 格式

异常:

错误返回 HTTP 的 400

2.2.5 显示用户发送给自己的分享

说明: 查看用户发送给自己的分享

路径: [http://api.follow5.com/api/statuses/private_mine.\[json|xml\]](http://api.follow5.com/api/statuses/private_mine.[json|xml])

参数:

- **api_key (必选)** – api_key, 每次请求必选传入此参数, 否则用户将没有权限访问 api。
- **count (可选)** - 消息数, 范围 1-20, 默认为 20。
示例: http://api.follow5.com/api/statuses/private_mine.xml?count=10&api_key=123
- **page (可选)** - 页码, 从 1 开始
示例: http://api.follow5.com/api/statuses/private_mine.xml?page=1&api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: YES

XML 格式:

同 2.1.1 格式

JSON 格式:

同 2.1.1 格式

异常:

错误返回 HTTP 的 400

2.2.6 获取一条指定的分享

说明: 根据提供的 ID 查询留言, 如果该分享对象为好友、悄悄话、组的需要提供用户名和密码进行验证

路径: [http://api.follow5.com/api/statuses/show.\[json|xml\]?id=2345](http://api.follow5.com/api/statuses/show.[json|xml]?id=2345)

参数:

- **api_key (必选)** – api_key, 每次请求必选传入此参数, 否则用户将没有权限访问 api。
- **id (必选)** – 分享 id。

示例: http://api.follow5.com/api/statuses/show.xml?id=128222&api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: YES

XML 格式:

```
<status>
  <created_at>留言时间</created_at>
  <id>留言ID</id>
  <text>留言内容</text>
  <source>分享方式</source>
  <receiver>接收人</receiver>
  <link>link</link>
  <link_cover>link封皮</link_cover>
  <image_address>图片地址</image_address>
  <longitude>经度</longitude>
  <latitude>纬度</latitude>
  <reply_count>回复数量</reply_count>
  <user>
    <id>用户ID</id>
    <name>用户名</name>
    <profile_image_url>头像地址</profile_image_url>
    <url>空间地址</url>
  </user>
  <set>
    <id>组ID</id>
    <name>组的名称</name>
  </set>
</status>
```

```
</status>
```

JSON 格式:

同 2.1.1 格式

异常:

错误返回 HTTP 的 400

2.2.7 发布分享

说明: 根据身份验证进行发送分享

路径: [http://api.follow5.com/api/statuses/update.\[json|xml\]](http://api.follow5.com/api/statuses/update.[json|xml])

参数:

- **api_key** (必选) – **api_key**, 每次请求必选传入此参数, 否则用户将没有权限访问 **api**。
- **status** (必选) – 消息内容, 使用 **POST** 方式提交, **utf-8** 编码。
- **link** (可选) – 消息 **link**, 使用 **POST** 方式提交, **utf-8** 编码。支持视频, 音频, 图片
- **source** (可选) - 消息来源, 使用 **POST** 方式提交, 如果与 **follow5** 的数据库匹配, 网页上将以格式显示: **status**(消息内容) 通过 **source** 对应的 **API**, **source** 的获取在申请 **api_key** 的同时说明消息来源, 与 **api_key** 是对应

示例: <http://api.follow5.com/api/statuses/update.xml>

格式: xml, json

HTTP Method(s): POST

需要验证: YES

XML 格式:

```
<status>
  <created_at>留言时间</created_at>
  <id>留言ID</id>
  <text>留言内容</text>
  <source>分享方式</source>
  <receiver>接收人</receiver>
  <link>link</link>
  <link_cover>link封皮</link_cover>
  <image_address>图片地址</image_address>
  <longitude>经度</longitude>
  <latitude>纬度</latitude>
  <reply_count>回复数量</reply_count>
  <user>
    <id>用户ID</id>
    <name>用户名</name>
    <profile_image_url>头像地址</profile_image_url>
    <url>空间地址</url>
  </user>
  <set>
    <id>组ID</id>
    <name>组的名称</name>
```

```
</set>

</status>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.2.8 删除分享

说明: 根据 ID 和身份验证进行删除分享

路径: [http://api.follow5.com/api/statuses/destroy.\[json|xml\]](http://api.follow5.com/api/statuses/destroy.[json|xml])

参数:

- **api_key** (必选) – **api_key**, 每次请求必选传入此参数, 否则用户将没有权限访问 **api**。
- **id** (必选) – 分享 id。

示例: http://api.follow5.com/api/statuses/destroy.xml?id=123&api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: YES

XML 格式:

```
<status>true</status>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.2.9 发布分享给指定好友

说明: 根据身份验证进行发送分享到指定好友

路径: [http://api.follow5.com/api/statuses/update_friend.\[json|xml\]](http://api.follow5.com/api/statuses/update_friend.[json|xml])

参数:

- **api_key** (必选) – **api_key**, 每次请求必选传入此参数, 否则用户将没有权限访问 **api**。
- **status** (必选) – 消息内容, 使用 **POST** 方式提交, utf-8 编码。
- **fid** (必选) – 好友 ID 集合。(以 “,” 分隔, 例子 “100025,100026”)
- **link** (可选) – 消息 link, 使用 **POST** 方式提交, utf-8 编码。支持视频, 音频, 图片
- **source** (可选) - 消息来源, 使用 **POST** 方式提交, 如果与 **follow5** 的数据库匹配, 网页上将以格式显示: **status**(消息内容) 通过 **source** 对应的 API, **source** 的获取在申请 **api_key** 的同时说明消息来源, 与 **api_key** 是对应

示例: http://api.follow5.com/api/statuses/update_friend.xml

格式: xml, json

HTTP Method(s): POST

需要验证: YES

XML 格式:

```
<status>
  <created_at>留言时间</created_at>
  <id>留言ID</id>
  <text>留言内容</text>
  <source>分享方式</source>
  <receiver>接收人</receiver>
  <link>link</link>
  <link_cover>link封皮</link_cover>
  <image_address>图片地址</image_address>
  <longitude>经度</longitude>
  <latitude>纬度</latitude>
  <reply_count>回复数量</reply_count>
  <user>
    <id>用户ID</id>
    <name>用户名</name>
    <profile_image_url>头像地址</profile_image_url>
    <url>空间地址</url>
  </user>
  <set>
    <id>组ID</id>
    <name>组的名称</name>
  </set>
</status>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.2.10 发布分享到组

说明: 根据身份验证进行发送分享到组

路径: http://api.follow5.com/api/statuses/update_set .[json|xml]

参数:

- **api_key** (必选) – **api_key**, 每次请求必选传入此参数, 否则用户将没有权限访问 **api**。
- **status** (必选) – 消息内容, 使用 **POST** 方式提交, **utf-8** 编码。
- **setid** (必选) – 组 ID。
- **link** (可选) – 消息 **link**, 使用 **POST** 方式提交, **utf-8** 编码。支持视频, 音频, 图片
- **source** (可选) - 消息来源, 使用 **POST** 方式提交, 如果与 **follow5** 的数据库匹配, 网页上将以格式显示: **status**(消息内容) 通过 **source** 对应的 API, **source** 的获取在申请 **api_key** 的同时说明消息来源, 与 **api_key** 是对应

示例: http://api.follow5.com/api/statuses/update_set.xml

格式: xml, json

HTTP Method(s): POST

需要验证: YES

XML 格式:

```
<status>
  <created_at>留言时间</created_at>
  <id>留言ID</id>
  <text>留言内容</text>
  <source>分享方式</source>
  <receiver>接收人</receiver>
  <link>link</link>
  <link_cover>link封皮</link_cover>
  <image_address>图片地址</image_address>
  <longitude>经度</longitude>
  <latitude>纬度</latitude>
  <reply_count>回复数量</reply_count>
  <user>
    <id>用户ID</id>
    <name>用户名</name>
    <profile_image_url>头像地址</profile_image_url>
    <url>空间地址</url>
  </user>
  <set>
    <id>组ID</id>
    <name>组的名称</name>
  </set>
</status>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.2.11 显示 “@用户” 的分享

说明: 用户可通过 api 查看 @ 给自己的分享

路径: [http://api.follow5.com/api/statuses/mentions_me.\[json|xml\]](http://api.follow5.com/api/statuses/mentions_me.[json|xml])

参数:

- **api_key** (必选) – api_key, 每次请求必选传入此参数, 否则用户将没有权限访问 api。
- **count** (可选) - 消息数, 范围 1-20, 默认为 20。
示例: http://api.follow5.com/api/statuses/mentions_me.xml?count=10&api_key=123
- **page** (可选) - 页码, 从 1 开始
示例: http://api.follow5.com/api/statuses/mentions_me.xml?page=1&api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: YES

XML 格式:

同 2.1.1 格式

JSON 格式:

同 2.1.1 格式

异常:

错误返回 HTTP 的 400

2.3 用户操作接口

2.3.1 显示用户好友列表

说明: 没有 ID 参数或用户设隐私时需验证用户

路径: `http://api.follow5.com/api/users/friends.[json|xml]`

参数:

- **api_key (必选)** – `api_key`, 每次请求必选传入此参数, 否则用户将没有权限访问 `api`。
- **count (可选)** - 消息数, 范围 1-20, 默认为 20。
示例: `http://api.follow5.com/api/users/friends.xml?count=10&api_key=123`
- **page (可选)** - 页码, 从 1 开始
示例: `http://api.follow5.com/api/users/friends.xml?page=1&api_key=123`
- **id (可选)** – 用户 id, 没有此参数或用 户设隐私时需验证用户。
示例: `http://api.follow5.com/api/users/friends.xml?id=100025&api_key=123`

格式: xml, json

HTTP Method(s): GET

需要验证: NO

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
<users>
  <user>
    <id>用户id</id>
    <name>用户名称</name>
    <location>来自</location>
    <description>自我介绍</description>
    <sex>性别</sex>
    <profile_image_url>头像地址</profile_image_url>
    <url>空间地址</url>
    <created_at>注册日期</created_at>
  </user>
  ... 更多 ...
</users>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.3.2 显示用户关注列表

说明: 没有 ID 参数时需验证用户, 显示该用户的关注列表

路径: `http://api.follow5.com/api/users/followers.[json|xml]`

参数:

- `api_key` (必选) – `api_key`, 每次请求必选传入此参数, 否则用户将没有权限访问 `api`。
- `count` (可选) - 消息数, 范围 1-20, 默认为 20。
示例: `http://api.follow5.com/api/users/followers.xml?count=10&api_key=123`
- `page` (可选) - 页码, 从 1 开始
示例: `http://api.follow5.com/api/users/followers.xml?page=1&api_key=123`
- `id` (可选) – 用户 `id`, 没有此参数时需 验证用户。
示例: `http://api.follow5.com/api/users/followers.xml?id=100025&api_key=123`

格式: `xml, json`

HTTP Method(s): `GET`

需要验证: `NO`

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
<users>
  <user>
    <id>用户id</id>
    <name>用户名称</name>
    <location>来自</location>
    <description>自我介绍</description>
    <sex>性别</sex>
    <created_at>注册日期</created_at>
    <profile_image_url>头像地址</profile_image_url>
    <url>空间地址</url>
  </user>
  ... 更多 ...
</users>
```

JSON 格式:

节点名同 `xml`

异常:

错误返回 HTTP 的 400

2.3.3 显示用户被关注列表

说明: 没有 ID 参数时需验证用户, 显示该用户的被关注列表

路径: `http://api.follow5.com/api/users/followed.[json|xml]`

参数:

- **api_key** (必选) – **api_key**，每次请求必选传入此参数，否则用户将没有权限访问 **api**。
- **count** (可选) - 消息数，范围 1-20，默认为 20。
示例：http://api.follow5.com/api/users/followed.xml?count=10&api_key=123
- **page** (可选) - 页码，从 1 开始
示例：http://api.follow5.com/api/users/followed.xml?page=1&api_key=123
- **id** (可选) – 用户 **id**，没有此参数时需验证用户。
示例：http://api.follow5.com/api/users/followed.xml?id=100025&api_key=123

格式： xml, json

HTTP Method(s): GET

需要验证： NO

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
<users>
  <user>
    <id>用户id</id>
    <name>用户名称</name>
    <location>来自</location>
    <description>自我介绍</description>
    <sex>性别</sex>
    <created_at>注册日期</created_at>
    <profile_image_url>头像地址</profile_image_url>
    <url>空间地址</url>
  </user>
  ... 更多 ...
</users>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.3.4 显示用户信息

说明：没有 ID 参数或者设置隐私时需验证用户，显示该用户的信息

路径：[http://api.follow5.com/api/users/show.\[json|xml\]](http://api.follow5.com/api/users/show.[json|xml])

参数:

- **api_key** (必选) – **api_key**，每次请求必选传入此参数，否则用户将没有权限访问 **api**。
- **id** (可选) – 用户 **id**，没有此参数或者设置隐私时需验证用户。
示例：http://api.follow5.com/api/users/show.xml?id=100025&api_key=123

格式： xml, json

HTTP Method(s): GET

需要验证： NO

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
```



```
<user>
  <id>用户id</id>
  <name>用户名称</name>
  <location>来自</location>
  <description>自我介绍</description>
  <profile_image_url>头像地址</profile_image_url>
  <url>空间地址</url>
  <sex>性别</sex>
  <birthday>生日</birthday>
  < mobile >手机
</mobile>
  <qq>qq帐号</qq>
  <msn>msn帐号</msn>
  <email>email地址</email>
  <created_at>注册日期</created_at>
  <favourites_count>收藏数量</favourites_count>
  <followers_count>关注我的数量</followers_count>
  <following_count>我关注的数量</following _count>
    <friends_count>好友数量</friends_count>
    <statuses_count>分享数量</statuses_count>
</user>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.3.5 添加好友

说明: 需验证用户进行添加好友,对方用户需要身份验证,成功发送申请

路径: [http://api.follow5.com/api/friendships/create.\[json|xml\]](http://api.follow5.com/api/friendships/create.[json|xml])

参数:

- **api_key** (必选) – **api_key**, 每次请求必选传入此参数, 否则用户将没有权限访问 **api**。
- **id**(必选) – 添加用户 **id**。

示例: <http://api.follow5.com/api/friendships/create.xml>

格式: xml, json

HTTP Method(s): POST

需要验证: YES

XML 格式:

```
成功添加好友
<?xml version="1.0" encoding="UTF-8"?>
<user>
  <id>用户id</id>
  <name>用户名称</name>
  <location>来自</location>
```

```
<description>自我介绍</description>
<sex>性别</sex>
<profile_image_url>头像地址</profile_image_url>
<url>空间地址</url>
<created_at>注册日期</created_at>
</user>
成功发送申请
<user>true</user>
失败
<user>false</user>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.3.6 删除好友

说明: 需验证用户进行删除好友

路径: [http://api.follow5.com/api/friendships/destroy.\[json|xml\]](http://api.follow5.com/api/friendships/destroy.[json|xml])

参数:

- api_key (必选) – api_key, 每次请求必选传入此参数, 否则用户将没有权限访问 api。
- id(必选) – 删除用户 id。

示例: <http://api.follow5.com/api/friendships/destroy.xml>

格式: xml, json

HTTP Method(s): POST

需要验证: YES

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
成功
<user>true</user>
失败
<user>false</user>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.3.7 关注用户

说明: 需验证用户进行关注用户

路径: [http://api.follow5.com/api/follow/create.\[json|xml\]](http://api.follow5.com/api/follow/create.[json|xml])

参数:

- **api_key** (必选) – **api_key**, 每次请求必选传入此参数, 否则用户将没有权限访问 **api**。
 - **id**(必选) – 添加用户 **id**.
- 示例: <http://api.follow5.com/api/follow/create.xml>

格式: xml, json

HTTP Method(s): POST

需要验证: YES

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
成功
<user>
  <id>用户id</id>
  <name>用户名称</name>
  <location>来自</location>
  <description>自我介绍</description>
  <sex>性别</sex>
  <profile_image_url>头像地址</profile_image_url>
  <url>空间地址</url>
  <created_at>注册日期</created_at>
</user>

失败
<user>false</user>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.3.8 取消关注用户

说明: 需验证用户进行取消关注用户

路径: [http://api.follow5.com/api/follow/destroy.\[json|xml\]](http://api.follow5.com/api/follow/destroy.[json|xml])

参数:

- **api_key** (必选) – **api_key**, 每次请求必选传入此参数, 否则用户将没有权限访问 **api**。
 - **id**(必选) – 取消用户 **id**.
- 示例: <http://api.follow5.com/api/follow/destroy.xml>

格式: xml, json

HTTP Method(s): POST

需要验证: YES

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
成功
```

```
<user>true</user>
```

失败

```
<user>false</user>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.3.9 判断是否为好友

说明: 如果 id 和 fid 是好友返回 true, 否则返回 false

路径: [http://api.follow5.com/api/friendships/exists.\[json|xml\]](http://api.follow5.com/api/friendships/exists.[json|xml])

参数:

- api_key (必选) – api_key, 每次请求必选传入此参数, 否则用户将没有权限访问 api。
- id(必选) – 用户 id.
- fid(必选) – 好友 fid

示例: <http://api.follow5.com/api/friendships/exists.xml?id=100026&fid=100025>

格式: xml, json

HTTP Method(s): GET

需要验证: NO

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
```

好友

```
<friends>true</friends>
```

不是好友

```
<friends>false</friends>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.3.10 判断是否为关注

说明: 如果 id 和 fid 是关注关系返回 true, 否则返回 false

路径: [http://api.follow5.com/api/follow/exists.\[json|xml\]](http://api.follow5.com/api/follow/exists.[json|xml])

参数:

- api_key (必选) – api_key, 每次请求必选传入此参数, 否则用户将没有权限访问 api。
- id(必选) – 用户 id.
- fid(必选) – 被关注 fid

示例: http://api.follow5.com/api/follow/exists.xml?id=100026&fid=100025&api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: NO

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
已经关注
<followed>true</followed>
没有关注
<followed>false </followed>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.3.11 显示用户的组列表

说明: 查看用户的组列表

路径: [http://api.follow5.com/api/set/list.\[json|xml\]](http://api.follow5.com/api/set/list.[json|xml])

参数:

- **api_key (必选)** – api_key, 每次请求必选传入此参数, 否则用户将没有权限访问 api。

示例: http://api.follow5.com/api/set/list.xml?api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: YES

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
<sets>
  <set>
    <id>用户id</id>
    <name>用户名称</name>
  </set>
  ... 更多 ...
</sets>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.3.12 显示用户指定组的成员列表

说明: 查看用户的组列表

路径: `http://api.follow5.com/api/set/account_list.[json|xml]`

参数:

- `api_key` (必选) – `api_key`, 每次请求必选传入此参数, 否则用户将没有权限访问 `api`。
- `id` (必选) – 组 `id`。
- `count` (可选) – 成员数, 范围 1-20, 默认为 20。
示例: `http://api.follow5.com/api/set/account_list.xml?count=10&api_key=123&id=xxx`
- `page` (可选) - 页码, 从 1 开始
示例: `http://api.follow5.com/api/set/account_list.xml?page=1&api_key=123&id=xxx`

格式: `xml, json`

HTTP Method(s): `GET`

需要验证: `YES`

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
<set>
  <id>用户id</id>
  <name>用户名称</name>
  <member_count>成员总数</member_count>
  <users>
    <user>
      <id>用户id</id>
      <name>用户名称</name>
      <profile_image_url>头像地址</profile_image_url>
      <url>空间地址</url>
      <nick_name>好友昵称</nick_name>
    </user>
    ... 更多 ...
  </users>
</set>
```

JSON 格式:

节点名同 `xml`

异常:

错误返回 HTTP 的 400

2.4 用户帐号验证接口

2.4.1 用户帐号验证

说明：根据身份验证，看看用户是否注册

路径：[http://api.follow5.com/api/users/verify_credentials.\[json|xml\]](http://api.follow5.com/api/users/verify_credentials.[json|xml])

参数：

- `api_key` (必选) – `api_key`，每次请求必选传入此参数，否则用户将没有权限访问 `api`。

格式：`xml, json`

HTTP Method(s): `POST`

需要验证：`YES`

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
<user>
  <id>用户id</id>
  <name>用户名称</name>
  <location>来自</location>
  <description>自我介绍</description>
  <profile_image_url>头像地址</profile_image_url>
  <url>空间地址</url>
  <sex>性别</sex>
  <birthday>生日</birthday>
  <mobile>手机</mobile>
  <qq>qq帐号</qq>
  <msn>msn帐号</msn>
  <email>email地址</email>
  <created_at>注册日期</created_at>
  <favourites_count>收藏数量</favourites_count>
  <followers_count>关注我的数量</followers_count>
  <following_count>我关注的数量</following_count>
    <friends_count>好友数量</friends_count>
    <statuses_count>分享数量</statuses_count>
    <following>是否关注</following>
    <friend>是否好友</friend>
</user>
```

JSON 格式:

节点名同 `xml`

异常：

错误返回 HTTP 的 `400`

2.5 用户回复接口

2.5.1 获取每条分享的回复

说明：留言类型为私信或者好友，进行身份验证

路径：[http://api.follow5.com/api/statuses/reply_timeline.\[json|xml\]](http://api.follow5.com/api/statuses/reply_timeline.[json|xml])

参数：

- **api_key** (必选) – **api_key**, 每次请求必选传入此参数, 否则用户将没有权限访问 **api**。
- **id** (必选) – 分享 **id**。
示例: http://api.follow5.com/api/statuses/reply_timeline.xml?id=100025&api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: YES

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
<status>
  <created_at>留言时间</created_at>
  <id>留言ID</id>
  <text>留言内容</text>
  <source>分享方式</source>
  <receiver>接收人</receiver>
  <link>link</link>
  <link_cover>link封皮</link_cover>
  <image_address>图片地址</image_address>
  <longitude>经度</longitude>
  <latitude>纬度</latitude>
  <reply_count>回复数量</reply_count>
  <user>
    <id>用户ID</id>
    <name>用户名</name>
    <profile_image_url>头像地址</profile_image_url>
    <url>空间地址</url>
  </user>
  <relpyes>
    <reply>
      <created_at>留言时间</created_at>
      <id>留言ID</id>
      <text>留言内容</text>
      <source>分享方式</source>
      <user>
        <id>用户ID</id>
        <name>用户名</name>
        <profile_image_url>头像地址</profile_image_url>
        <url>空间地址</url>
      </user>
    </reply>
    ... 更多 ... </relpyes>
</status>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.5.2 发布分享回复

说明：根据身份验证进行发送分享回复。

路径：[http://api.follow5.com/api/statuses/reply.\[json|xml\]](http://api.follow5.com/api/statuses/reply.[json|xml])

参数：

- **api_key** (必选) – **api_key**，每次请求必选传入此参数，否则用户将没有权限访问 **api**。
- **id** (必选) – 分享 **id**。
- **status** (必选) – 消息内容，使用 **POST** 方式提交，**utf-8** 编码。
- **source** (可选) - 消息来源，使用 **POST** 方式提交，如果与 **follow5** 的数据库匹配，网页上将以此格式显示：**status**(消息内容) 通过 **source** 对应的 **API**，**source** 的获取在申请 **api_key** 的同时说明消息来源，与 **api_key** 是对应。

示例：<http://api.follow5.com/api/statuses/reply.xml>

格式：**xml**, **json**

HTTP Method(s): **POST**

需要验证：**YES**

XML 格式：

```
<reply>
  <created_at>留言时间</created_at>
  <id>留言ID</id>
  <text>留言内容</text>
  <source>分享方式</source>
  <user>
    <id>用户ID</id>
    <name>用户名</name>
    <profile_image_url>头像地址</profile_image_url>
    <url>空间地址</url>
  </user>
</reply>
```

JSON 格式：

节点名同 **xml**

异常：

错误返回 **HTTP** 的 **400**

2.5.3 获取“回复我的”回复列表

说明：获取“回复我的”回复列表，包括用户的 **ID**、头像、回复的内容、

路径：[http://api.follow5.com/api/statuses/replies_timeline.\[json|xml\]](http://api.follow5.com/api/statuses/replies_timeline.[json|xml])

参数：

- **api_key** (必选) – **api_key**，每次请求必选传入此参数，否则用户将没有权限访问 **api**。
- **count** (可选) - 消息数，范围 **1-20**，默认为 **20**。

示例：http://api.follow5.com/api/statuses/replies_timeline.xml?count=10&api_key=123

- **page** (可选) - 页码，从 **1** 开始

示例: http://api.follow5.com/api/statuses/replies_timeline.xml?page=1&api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: YES

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
<relpyes>
  <reply>
    <created_at>回复时间</created_at>
    <id>回复的ID</id>
    <text>回复的内容</text>
    <source>回复方式</source>
    <original_id>原文ID</original_id>
    <user>
      <id>用户ID</id>
      <name>用户名</name>
      <profile_image_url>头像地址</profile_image_url>
      <url>空间地址</url>
    </user>
  </reply>
  ... 更多 ...
</relpyes>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.6 悄悄话接口

2.6.1 发送悄悄话

说明: 发送悄悄话给指定的好友

路径: [http://api.follow5.com/api/direct_messages/new.\[json|xml\]](http://api.follow5.com/api/direct_messages/new.[json|xml])

参数:

- **api_key** (必选) – **api_key**, 每次请求必选传入此参数, 否则用户将没有权限访问 **api**。
- **status** (必选) – 消息内容, 使用 **POST** 方式提交, utf-8 编码。
- **fid** (必选) – 接受悄悄话的好友 ID。
- **link** (可选) – 消息 link, 使用 **POST** 方式提交, utf-8 编码。支持视频, 音频, 图片
- **source** (可选) - 消息来源, 使用 **POST** 方式提交, 如果与 **follow5** 的数据库匹配, 网页上将以格式显示: **status**(消息内容) 通过 **source** 对应的 API, **source** 的获取在申请 **api_key** 的同时说明消息来源, 与 **api_key** 是对应

示例: http://api.follow5.com/api/direct_messages/new.xml

格式: xml, json

HTTP Method(s): POST

需要验证: YES

XML 格式:

```
<status>
  <created_at>留言时间</created_at>
  <id>留言ID</id>
  <text>留言内容</text>
  <source>分享方式</source>
  <link>link</link>
  <link_cover>link封皮</link_cover>
  <image_address>图片地址</image_address>
  <longitude>经度</longitude>
  <latitude>纬度</latitude>
  <sender>
    <id>用户ID</id>
    <name>用户名</name>
    <profile_image_url>头像地址</profile_image_url>
    <url>空间地址</url>
  </sender>
  <recipient>
    <id>悄悄话接收人用户ID</id>
    <name>悄悄话接收人用户名</name>
    <profile_image_url>悄悄话接收人头像地址</profile_image_url>
    <url>悄悄话接收人空间地址</url>
  </recipient>
</status>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

没权限返回 HTTP 的 403

2.6.2 删除悄悄话

说明: 删除悄悄话

路径: [http://api.follow5.com/api/direct_messages/destroy.\[json|xml\]](http://api.follow5.com/api/direct_messages/destroy.[json|xml])

参数:

- **api_key** (必选) – **api_key**, 每次请求必选传入此参数, 否则用户将没有权限访问 **api**。
- **id** (必选) – 悄悄话 **id**。
示例: http://api.follow5.com/api/direct_messages/destroy.xml?id=123&api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: YES

XML 格式:

```
<status>true</status>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

2.6.3 显示悄悄话

说明: 查看好友发送给我的悄悄话以及我发送给别人的悄悄话

路径: [http://api.follow5.com/api/destroy_messages.\[json|xml\]](http://api.follow5.com/api/destroy_messages.[json|xml])

参数:

- **api_key (必选)** – api_key, 每次请求必选传入此参数, 否则用户将没有权限访问 api。
- **count (可选)** - 消息数, 范围 1-20, 默认为 20。
示例: http://api.follow5.com/api/destroy_messages.xml?count=10&api_key=123
- **page (可选)** - 页码, 从 1 开始
示例: http://api.follow5.com/api/destroy_messages.xml?page=1&api_key=123

格式: xml, json

HTTP Method(s): GET

需要验证: YES

XML 格式:

```
<?xml version="1.0" encoding="UTF-8"?>
<statuses>
  <status>
    <created_at>留言时间</created_at>
    <id>留言ID</id>
    <text>留言内容</text>
    <source>分享方式</source>
    <link>link</link>
    <link_cover>link封皮</link_cover>
    <image_address>图片地址</image_address>
    <longitude>经度</longitude>
    <latitude>纬度</latitude>
    <sender>
      <id>用户ID</id>
      <name>用户名</name>
      <profile_image_url>头像地址</profile_image_url>
      <url>空间地址</url>
```

```
</sender>
<recipient>
  <id>悄悄话接收人用户ID</id>
  <name>悄悄话接收人用户名</name>
  <profile_image_url>悄悄话接收人头像地址</profile_image_url>
  <url>悄悄话接收人空间地址</url>
</recipient>

</status>
... 更多 ...
</statuses>
```

JSON 格式:

节点名同 xml

异常:

错误返回 HTTP 的 400

3 错误码

- * **200 成功**: 操作成功!
- * **400 错误的请求**: 请求是无效的。
- * **401 未经授权**: 访问由于凭据无效被拒绝。
- * **403 禁止**: 请求被拒绝。未授权。
- * **404 找不到**: 请求的 URI 无效或请求的资源无效。
- * **500 内部服务器错误**: 因为意外情况, 服务器不能完成请求。
- * **502 错误网关**: follow5 的是关闭或正在升级。
- * **503 服务不可用**: 在 follow5 的 服务器, 请求超载, 服务器无法处理请求。

4 常见问题

4.1 为什么API_key申请石沉大海没有反馈了?

大多数情况下是因为您没有正确的按照格式填写申请邮件, 详细请参考API文档的“[1.2 API_KEY申请](#)”

4.2 为什么“来自”不是我事先指定的名称?

对于“来自”的显示, follow5 一直慎重的进行审核, 只有分享数量、分享的质量达到一定的要求后, 才能显示。一般情况下, 申请 API 的 1~2 周后, 就会放开“来自”的显示。

4.3 为什么有的时候我通过API发布的分享不能够通过 5 步大同分享到其他网站?

这种情况一般是由如下原因造成的。

- 大量的广告信息想要进行同步
- 被对方的网站投诉
- 网络原因 (对方网站服务不稳定或由于 5 步大同用户数量过大)

5 附录

感谢您对 Follow5 开放平台的兴趣，通过 Follow5 开放平台您可以调用 Follow5 开放的 API 实现您所期望的应用并通过 Follow5 提供的资源实现应用的价值。但是出于我们对 Follow5 用户的服务承诺以及对自身安全的考虑，我们需要您遵守以下条款，如果您未能遵守以下任何一款协议，我们将无法授予您调用 Follow5 开放平台 API 的资格。

5.1 Follow5 开放平台开发者协议

本服务协议缔约方为 Follow5 开放平台与 Follow5 开放平台用户，凡使用 Follow5 开放平台的用户视为自动认可本协议。

5.2 定义

- 1) **Follow5 开放平台用户**：所有直接或间接使用 Follow5 开放平台及第三方所基于 Follow5 开放平台的应用的单位或者个人。
- 2) **开发者**：Follow5 用户中通过有效申请流程申请并通过验证的可以基于 Follow5 开放平台进行开发的单位或者个人。
- 3) **应用**：开发者基于 Follow5 开放平台所开发的软件或者服务。
- 4) **APIkey**：开发者通过申请获得由 Follow5 开放平台授予的应用程序接入帐户和密钥。

5.3 Follow5 开放平台用户管理规则

- 1) Follow5 开放平台用户应同时遵守《Follow5 服务协议》。
- 2) 所有 Follow5 开放平台的用户，有权利使用自己的 APIkey 登陆 Follow5 开放平台并对 APIkey 的安全负全部责任，同时对以其 APIkey 进行的所有活动和事件负全责。
- 3) Follow5 开放平台用户不得以任何形式擅自转让或授权他人使用自己的 APIkey。
- 4) 同意接收来自 Follow5 开放平台及其合作伙伴发出的邮件、信息。

5.4 开发者管理规则

- 1) 开发者须通过 Follow5 开放平台提供的通道获取合法的 API key。
- 2) 开发者声明并保证为其提供的应用或附属产品的应用内容及方向真实性、合法性、准确性、完整性，且应用上线之前需要 Follow5 开放平台的审核，Follow5 开放平台将依据自身判断决定该应用是否获得审批。
- 3) Follow5 开放平台有权对开发者提供的应用进行定期或不定期审核，且有权利对违反 Follow5 政策的应用在不通知的情况下予以终止。
- 4) 如开发者违反上述条款，或其提供的应用未能通过 Follow5 开放平台的审核，则 Follow5 开放平台有权随时终止开发者使用 Follow5 开放平台服务的资格，并追究开发者的违约责任。

5.5 开放平台服务的变更与中断

- 1) 鉴于网络服务的特殊性，Follow5 开放平台用户同意 Follow5 开放平台会变更、中断部分或全部的的服务，并删除（不再保存）Follow5 开放平台用户在使用“服务”中提交的任何资料，而无需通知 Follow5 开放平台用户，也无需对任何 Follow5 开放平台用户或任何第三方承担任何责任。
- 2) Follow5 开放平台需要定期或不定期地对提供服务的平台或相关的设备进行检修或者维护，如因此类情况而造成服务在合理时间内的中断，Follow5 开放平台无需为此承担任何责任。

5.6 知识产权

- 1) Follow5 开放平台上所有内容，包括但不限于数据、著作、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由 Follow5 开放平台或其他权利人依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经 Follow5 开放平台或其他权利人书面同意任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表本网站程序或内容。
- 2) Follow5 开放平台用户应对其使用或提供的包括但不限于开发工具、数据资源之知识产权承担全部责任。

5.7 协议修改

- 1) Follow5 开放平台有权随时修改本服务协议的任何条款，且有权将本服务协议项下的权利及义务转让予第三方，一旦本服务协议的任何内容发生变动，Follow5 开放平台将会通过适当方式向 Follow5 开放平台用户提示修改内容。
- 2) 如果不同意 Follow5 开放平台对本服务协议相关条款所做的修改及（或）转让，Follow5 开放平台用户有权停止使用网络服务。如果用户继续使用网络服务，则视为用户接受 Follow5 开放平台对本服务协议相关条款所做的修改及（或）协议的转让。

5.8 责任限制

Follow5 开放平台用户明确理解并同意，如因其违反有关法律或者本服务协议之规定，使 Follow5 开放平台遭受任何损失，受到任何第三方的索赔，或任何行政管理部的处罚，应对 Follow5 开放平台提供补偿，包括合理的律师费用。