

Google Map api 国内正常使用该如何配置 (2021 最新)

作者: [pjcourse](#)

原文链接: <https://ld246.com/article/1614060148018>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

最近有客户要求给他们网站做地图方面的功能，由于某些原因，网站必须使用 Google map，而且希用 geocoding。大家知道 Google map API 调用国内已经访问不了，虽然网上有很多教程，什么换 ip 啊，把 maps.google.com 改成 maps.google.cn。但其实这些方法都是掉了牙的，早就不管。

今天我把我的安装方法分享出来。

申请 API 接口

首先，我们需要申请 Geocoding API 和 Google map API。你得有一个 Google 帐号，访问 Google map API console，根据自己需求来申请相关的 API。我申请的比较多。

Places API

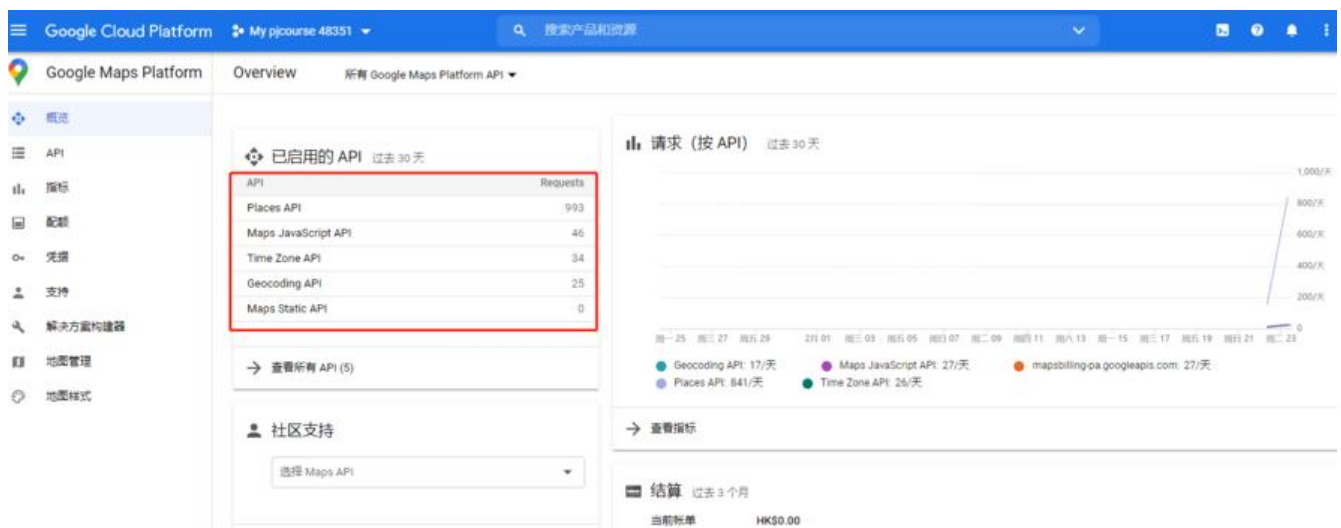
Maps JavaScript API

Time Zone API

Geocoding API

Maps Static API

你们可以根据各自需求来申请。也可以访问 www.pjcourse.com 看最后的应用效果

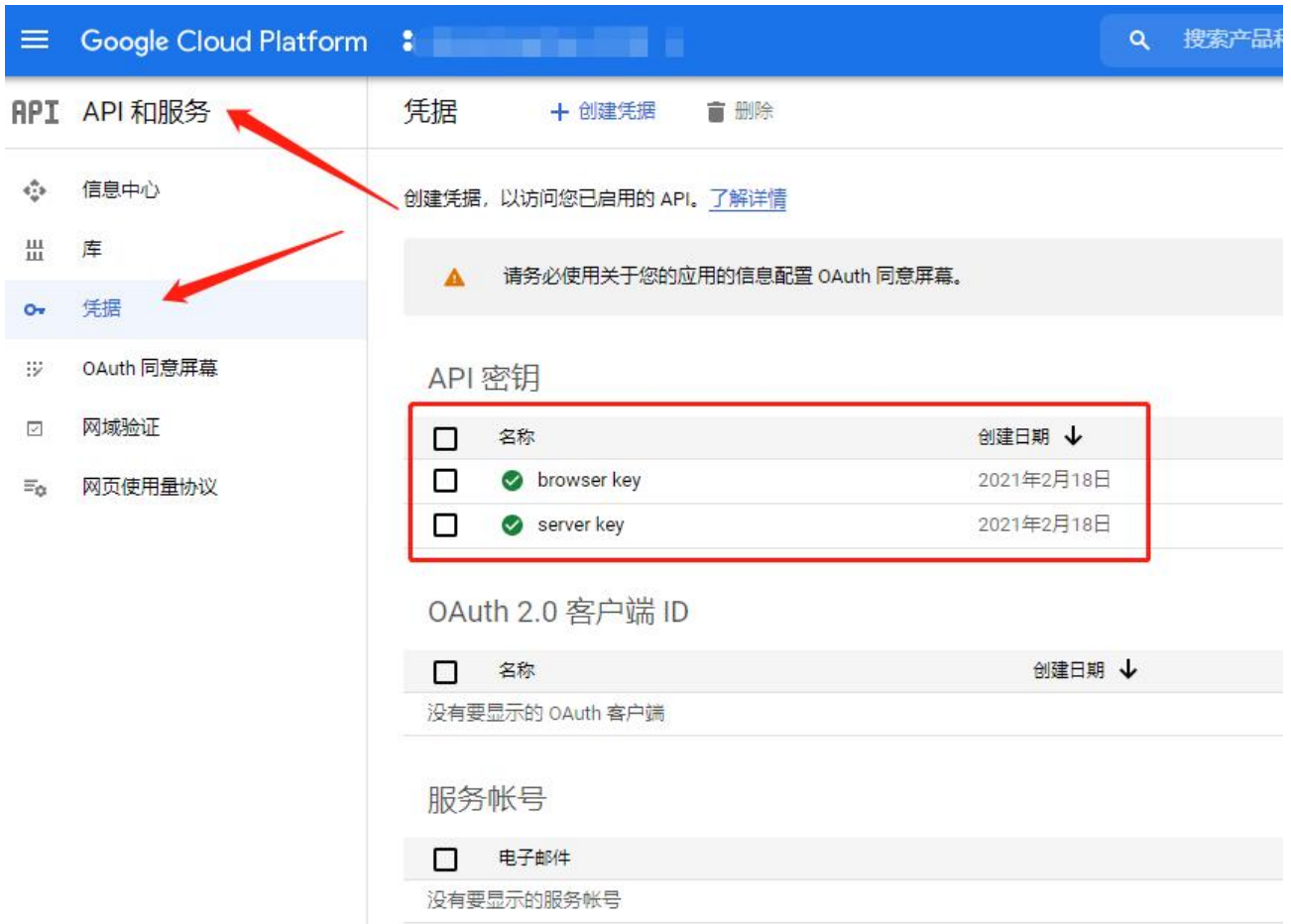


申请这个比较简单，

1.新建项目

2.搜索相应 API，申请

3.转到 API 和服务这一块，创建凭据。这些凭据就是 API key，也用来限制 API 的具体应用范围。



4.最后需要做结算。现在结算是免费试用阶段，申请的话，只要有一张信用卡就可以，因为已经没有中国地区的选项，所以地址选择香港。会扣除 8 港元，信用卡验证通过之后会退回。

这么一来，所以的申请算结束了。

配置子域名

我用的是 cloudflare，所以直接在上面新开两个子域名，maps.example.com，mapsapis.example.com。这里 example 替换成你自己的域名就可以。

配置 ssl，我用的是 let's encrypt，自动 90 天就会续签的。

```
sudo apt install certbot python3-certbot-nginx
sudo certbot --nginx -d example.com -d www.example.com
sudo systemctl status certbot.timer
```

输出结果如下，就说明自动续签正常。

- certbot.timer - Run certbot twice daily
Loaded: loaded (/lib/systemd/system/certbot.timer; enabled; vendor preset: enabled)
Active: active (waiting) since Mon 2020-05-04 20:04:36 UTC; 2 weeks 1 days ago
Trigger: Thu 2020-05-21 05:22:32 UTC; 9h left
Triggers: ● certbot.service

安装必要的模块

我的配置环境是 Ubuntu 20.04, nginx。

1. 安装 replace-filter-nginx-module 模块

安装之前, 首先需要安装 sregex

```
git clone https://github.com/agentzh/sregex
cd sregex/
make
make install
```

下载 replace-filter-nginx-module

```
git clone https://github.com/agentzh/replace-filter-nginx-module
nginx -V
```

这里用到 nginx -V。主要是把 nginx 的模块全部显示出来, 等会需要重新编译。

```
wget https://nginx.org/download/nginx-1.18.0.tar.gz
tar xvf nginx-1.18.0.tar.gz
cd nginx-1.18.0/
./configure --with-cc-opt='-g -O2 -fdebug-prefix-map=/build/nginx-5J5hor/nginx-1.18.0=. -f
tack-protector-strong -Wformat -Werror=format-security -fPIC -Wdate-time -D_FORTIFY_SO
RCE=2' --with-ld-opt='-Wl,-Bsymbolic-functions -Wl,-z,relro -Wl,-z,now -fPIC' --prefix=/usr/s
are/nginx --conf-path=/etc/nginx/nginx.conf --http-log-path=/var/log/nginx/access.log --err
r-log-path=/var/log/nginx/error.log --lock-path=/var/lock/nginx.lock --pid-path=/run/nginx.
id --modules-path=/usr/lib/nginx/modules --http-client-body-temp-path=/var/lib/nginx/bo
dy --http-fastcgi-temp-path=/var/lib/nginx/fastcgi --http-proxy-temp-path=/var/lib/nginx/pr
oxy --http-scgi-temp-path=/var/lib/nginx/scgi --http-uwsgi-temp-path=/var/lib/nginx/uwsgi -
with-debug --with-compat --with-pcre-jit --with-http_ssl_module --with-http_stub_status_m
odule --with-http_realip_module --with-http_auth_request_module --with-http_v2_module --wi
th-http_dav_module --with-http_slice_module --with-threads --with-http_addition_module --w
ith-http_gunzip_module --with-http_gzip_static_module --with-http_image_filter_module=dyn
amic --with-http_sub_module --with-http_xslt_module=dynamic --with-stream=dynamic --wit
h-stream_ssl_module --with-mail=dynamic --with-mail_ssl_module --add-module=/root/repla
ce-filter-nginx-module
make
```

这里注意一下, --add-module=/root/replace-filter-nginx-module。需要添加进去。其它的配置项, 参考你们自己的 nginx -V 参数结果。

一般重新编译的时候, 都会有一堆报错。这主要是和你的模块配置参数有关, 你只要把相应的模块安上就可以。比如我遇到以下这些:

pcre

```
sudo apt-get install libpcre3 libpcre3-dev
```

gd lib

```
apt install libgd-dev
```

openssl

```
sudo apt-get install libssl-dev
```

最后，把 nginx 做个备份，再替换掉。

```
cp /usr/sbin/nginx /usr/sbin/nginx.bak
cp ./objs/nginx /usr/sbin/
```

配置 nginx

在/etc/nginx/sites-enabled 目录下，新建一个配置 maps.example.com.conf。

```
server {# default_server;# default_server;
    server_name maps.example.com mapsapis.example.com;

    location /maps/ {
        default_type text/javascript;
        proxy_set_header Accept-Encoding "";
        proxy_pass https://maps.googleapis.com/maps/;

        replace_filter_max_buffered_size 500k;
        replace_filter_last_modified keep;
        replace_filter_types text/javascript application/javascript;

        replace_filter maps.googleapis.com mapsapis.example.com ig;
    }

    location /maps-api-v3/ {
        proxy_pass https://maps.googleapis.com/maps-api-v3/;
    }

    listen [::]:443 ssl http2;
    listen 443 ssl http2;

    ssl_certificate /etc/letsencrypt/live/example.com/fullchain.pem; # managed by Certbot
    ssl_certificate_key /etc/letsencrypt/live/example.com/privkey.pem; # managed by Certb
```

至此，所有的配置已经完成。测试了一下，直接通过访问自己的子域名，就可以调用 maps.googleapis.com 的地图接口了。



文章来源: www.pjcourse.com