

利用Cloudflare搭建ChatGPT API Proxy

利用Cloudflare搭建ChatGPT API Proxy

#chatgpt

#cloudflare

1、使用Cloudflare搭建免费ChatGPT API Proxy的原因

之所以要用Cloudflare搭建ChatGPT API Proxy，主要因为如下原因：

Cloudflare对免费账号提供最高每日10万次访问的额度，完全可以满足个人使用需求了，官方说明如下：

The image shows a side-by-side comparison of Cloudflare's 'Free' and 'Paid' plans. The 'Free' plan is for personal use and simple applications, costing \$0. It includes Workers & Pages Functions Bundled (up to 10ms CPU time per request, lowest latency after first request, up to 100,000 requests per day) and KV (Global low-latency key-value edge storage, up to 100,000 read operations per day, up to 1,000 write, delete, list operations per day). A link 'See Free Plan limits' is provided. The 'Paid' plan is for business use and scaling applications, starting at \$5/month. It includes Workers & Pages Functions Bundled (Always lowest latency, up to 50ms CPU time per request, includes monthly usage) and Workers & Pages Functions Unbound (Always lowest latency, up to 15 minutes duration per request, includes monthly usage). It also includes KV (Global low-latency key-value edge storage, includes monthly usage). Buttons for 'Your current plan' and 'Purchase Workers Paid' are shown.

Free	Paid
For personal use and simple applications	For business use and scaling applications
\$0	Pay-as-you-go Starting at \$5/month
Your current plan	Purchase Workers Paid
Workers & Pages Functions Bundled <ul style="list-style-type: none">Up to 10ms CPU time per requestLowest latency after first requestUp to 100,000 requests per day (UTC+0)	Workers & Pages Functions Bundled <ul style="list-style-type: none">Always lowest latencyUp to 50ms CPU time per requestIncludes monthly usage
KV <ul style="list-style-type: none">Global low-latency key-value edge storageUp to 100,000 read operations per dayUp to 1,000 write, delete, list operations per day	Workers & Pages Functions Unbound <ul style="list-style-type: none">Always lowest latencyUp to 15 minutes duration per requestIncludes monthly usage
See Free Plan limits	KV <ul style="list-style-type: none">Global low-latency key-value edge storageIncludes monthly usage

通过Cloudflare的work搭建的api proxy，在分配自有域名后，在国内可以正常访问，同时又可以基本不受限制的访问ChatGPT的api接口，从而就实现了在国内无需科学上网即可快速调用ChatGPT API接口的目的。

2、具体步骤

登录cloudflare创建账号登基本步骤就略过不作说明了，以下具体讲一下创建ChatGPT API Proxy的步骤：

2.1 创建worker

访问<https://dash.cloudflare.com/>，点击左侧菜单栏的“Workers”-“Overview”，如下图：

Overview

Build serverless applications and deploy instantly across the globe for exceptional performance, reliability, and scale.

Create a Service

[Workers documentation](#) 

点击右边蓝色"Create a Service"按钮即可创建自己的worker。

Service name

wild-violet-55e5



Your service will be deployed to: <https://wild-violet-55e5.colin4k.workers.dev>

Select a starter

Once deployed you'll be able to modify your code from the dashboard or the [Wrangler CLI](#).



HTTP handler

Worker that displays a simple Hello World response.



HTTP router

Worker that executes based on requests routed by URL patterns.

```
export default {
  async fetch(request, env) {
    return new Response("Hello world")
  }
}
```

Click to copy

Your service will be deployed to: [https://wild-violet-](https://wild-violet-55e5.colin4k.workers.dev)

Cancel

Create service

选择"HTTP handler", 无需做任何修改, 点击右下角蓝色按钮"Create service", 跳转到如下页面:

wild-violet-55e5

[Manage Service](#)

[Quick edit](#)

Preview wild-violet-					Free
Custom Domains	Routes	Cron Triggers	Email Triggers	Connected Workers	
0 View	1 View	0 View	0 View	0 View	

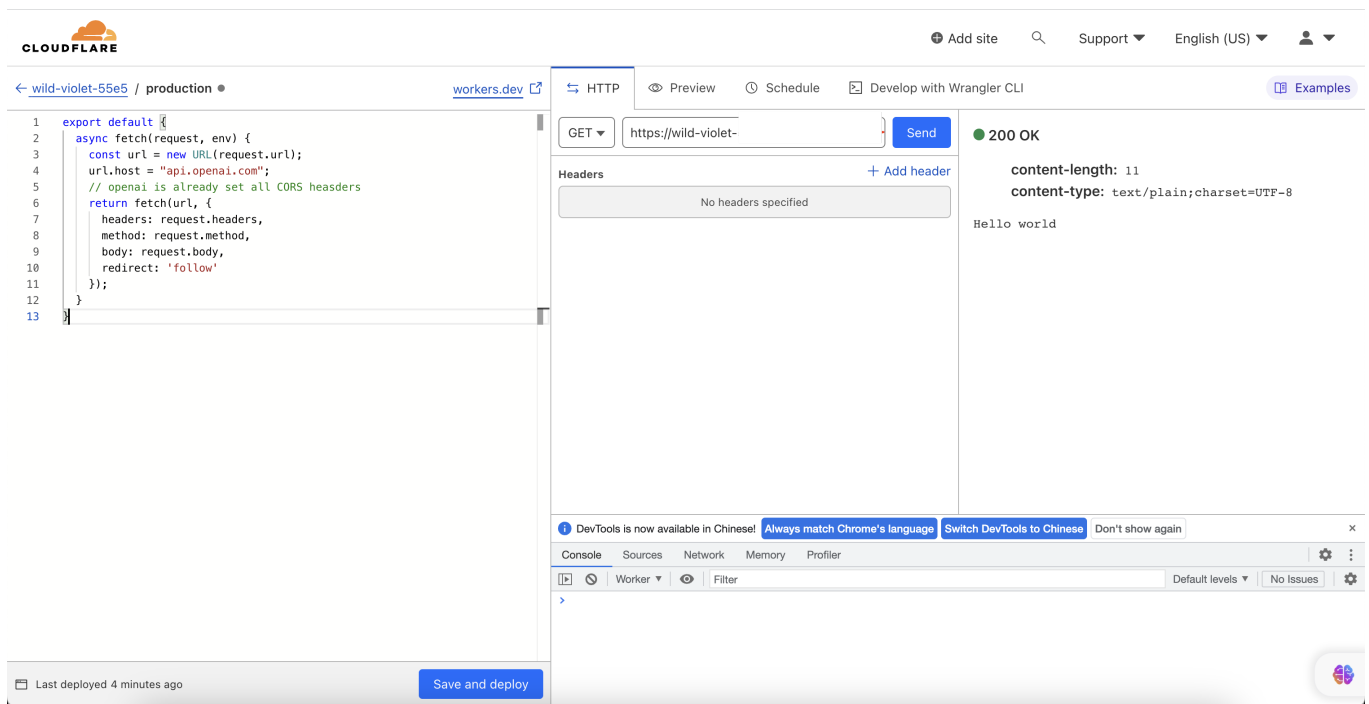
点击右上角的“Quick Edit”按钮，就会进入到代码编辑页面。

2.2 修改Worker的代码

worker创建完毕后，在左侧的代码编辑器中，删除现有的所有代码，然后复制粘贴以下内容到代码编辑器：

```
export default {
  async fetch(request, env) {
    const url = new URL(request.url);
    url.host = "api.openai.com";
    // openai is already set all CORS headers
    return fetch(url, {
      headers: request.headers,
      method: request.method,
      body: request.body,
      redirect: 'follow'
    });
  }
}
```

如下图：



最后点击编辑器右下角的 "Save and deploy" 按钮部署该代码，在弹出的对话框中继续选择 "Save and deploy" 确认部署。

2.3 绑定自有域名

之所以要绑定自有域名，原因是如果直接使用Cloudflare提供的域名，在国内如不科学上网是无法正常访问的，只要绑定一个自有域名就可以正常快速访问了。可以选择将自己在其他域名服务提供商申请的域名转入Cloudflare（点击 Cloudflare 侧边栏的 "Websites"，然后点击 "Add a Site" 按钮，根据提示将域名的 NS 记录指定到 Cloudflare），也可以直接在Cloudflare申请一个便宜的新域名（最便宜的一年大概3美刀）。转入或者购买域名的方法就不详细介绍了，网上有很多教程，这里讲一下已经在Cloudflare具备自有域名之后的步骤：

在 Cloudflare Workers 的管理界面中，点击 Triggers选项卡，然后点击 Custom Domians中的 Add Custom Domain按钮以绑定域名。

proxy

[Manage Service](#)[Quick edit](#)

Preview

Free

[proxy](#) [.dev](#)

Custom Domains

1

[View](#)

Routes

1

[View](#)

Cron Triggers

0

[View](#)


Email Triggers

0

[View](#)

Connected Workers

0

[View](#)via  Dashboard by

13 days ago

[Metrics](#)[Triggers](#)[Logs](#)[Deployments](#)

Beta

[Settings](#)

Custom Domains

Connect your Worker to a custom domain. [Using Custom Domains](#)[Add Custom Domain](#)

Domain

Certificate

[proxy.](#)[↗](#)

Active

...

绑定后，需要在该域名的管理界面添加一条A类型域名记录，该记录指向的IP地址无所谓（例如我输入2.2.2.2），关键是“Proxy status”字段需为“Proxied”。

←

Overview

Analytics & Logs

DNS

Records

Settings

Email

SSL/TLS

Security

Access

Speed

Caching

Workers Routes

Rules

Network

Search DNS Records

▼ Add filter

Q

Search

⊕ Add record

proxy.apiproxy.win points to 2.2.2.2 and has its traffic proxied through Cloudflare.

Type

A

Name (required)

proxy

IPv4 address (required)

2.2.2.2

Proxy status

☒ Proxied

TTL

Auto

Record Attributes

Documentation

The information provided here will not impact DNS record resolution and is only meant for your reference.

Comment

Enter your comment here (up to 50 characters).

Cancel

Save

Type	Name	Content	Proxy status	TTL	Actions
A	api	2.2.2.2	<input checked="" type="checkbox"/> Proxied	Auto	Edit
Worker			<input checked="" type="checkbox"/> Proxied	Auto	Edit
Worker			<input checked="" type="checkbox"/> Proxied	Auto	Edit

点击右下角"Save"蓝色按钮，就大功告成了。

2.4 测试

最后来验证一下刚才部署的proxy是否好用，在Linux/MacOS下可以执行以下命令来验证：

```
#将下面api.openai.com替换为第3步设置的域名，将sk-xxx替换为自己的api key
curl --location 'https://api.openai.com/v1/chat/completions' \
--header 'Authorization: Bearer sk-xxxxxxxxxxxxxxxx' \
--header 'Content-Type: application/json' \
--data '{
  "model": "gpt-3.5-turbo",
  "messages": [{"role": "user", "content": "Hello!"}]
}'
```

注意要替换上面脚本中url中的域名为前面设置的自有域名，例如你设置的域名为proxy.myapi.cc，则可以将上面的"https://api.openai.com/v1/chat/completions" 修改为 "https://proxy.myapi.cc/v1/chat/completions"，同时也需要将脚本中的sk-xxxx替换为你自己的key。

2.5 环境设置

好了，现在你的自有ChatGPT API Proxy已经创建好了，后续使用的话有两种方法，一种是配置到环境变量中，很多第三方插件或者工具都会自动读取环境变量，另一种是直接在使用插件或工具中修改。现在先具体说明一下第一种方法，第二种后续会结合具体的应用来说明。

假设你的ChatGPT API Proxy的地址是proxy.myapi.cc，则设置如下（之所以要设置以下两个环境变量，是因为不同的插件或工具读取的变量名不同，这两个是比较常用的，如果你看到有其他的也可以自行根据其需求再做设置）

```
#Linux/MacOS
export OPENAI_API_BASE=https://proxy.myapi.cc/v1
export OPENAI_API_HOST=https://proxy.myapi.cc
#Windows
set OPENAI_API_BASE=https://proxy.myapi.cc/v1
set OPENAI_API_HOST=https://proxy.myapi.cc
```