

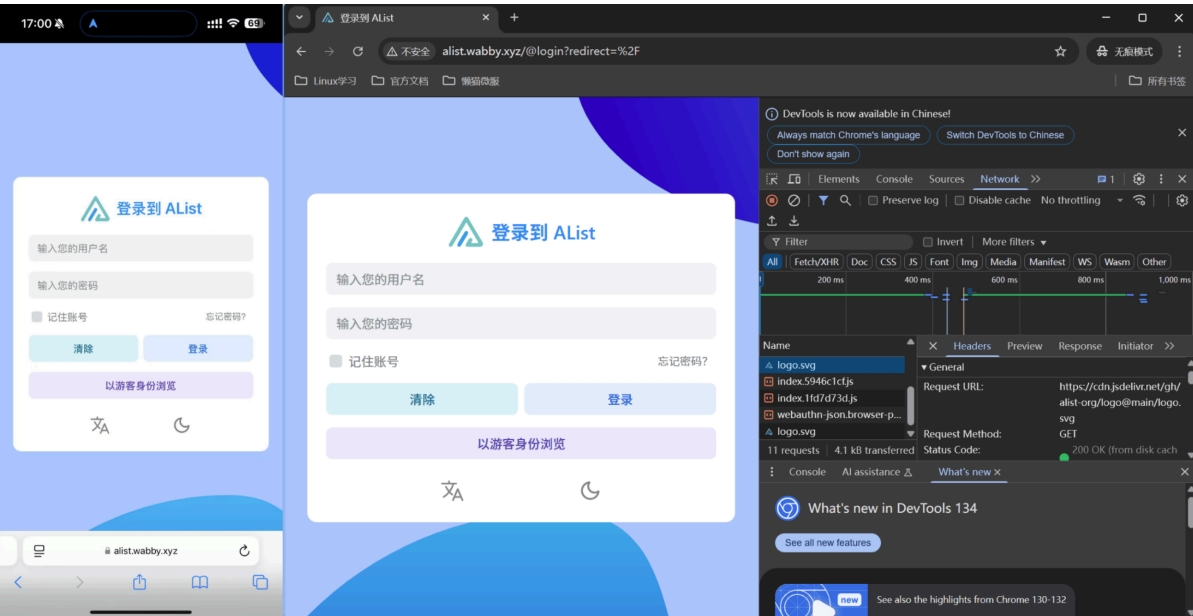
# CloudFlare 配置指南和端口转发技巧

## CloudFlare 配置指南和端口转发技巧

具体实现

- 1、登录CloudFlare
- 2、域名托管
  - 1、托管域名配置
  - 2、等待生效
  - 3、托管完成
- 3、配置转发
  - 1、微服关联Cloudflare
    - 1、创建隧道
    - 2、连接隧道
  - 2、配置应用转发
    - 1、获取应用端口
    - 2、配置转发
    - 3、测试Cloudflare应用与转发应用是否正常访问
  - 4、配置转发域名
    - 1、添加公共主机名
    - 2、公共主机名配置
- 4、测试访问
- 5、其他注意事项

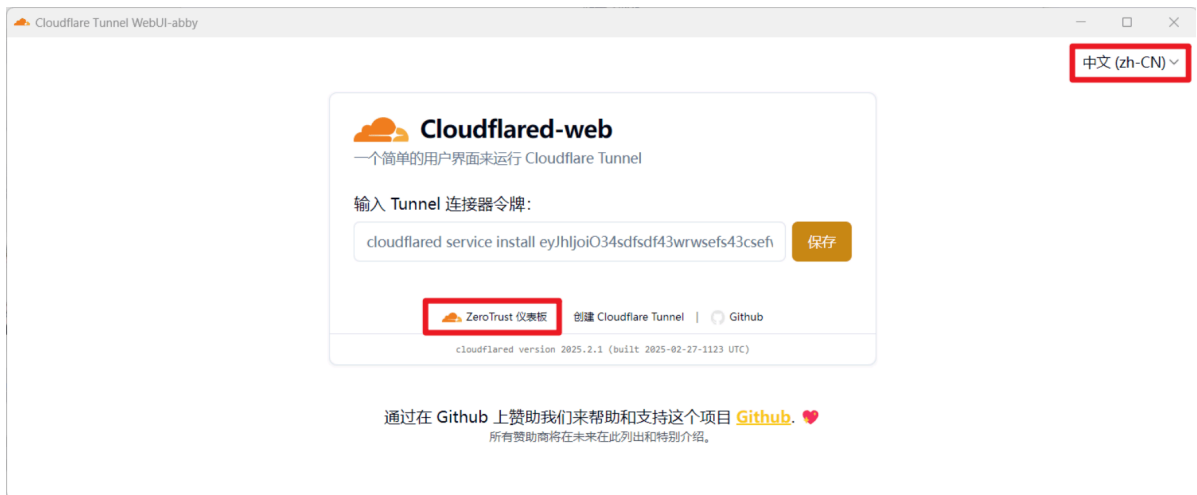
**需求背景：**用户需把微服内的应用通过域名托管和端口转发的方式，实现公网访问微服内的应用，实现下图界面



## 具体实现

### 1、登录CloudFlare

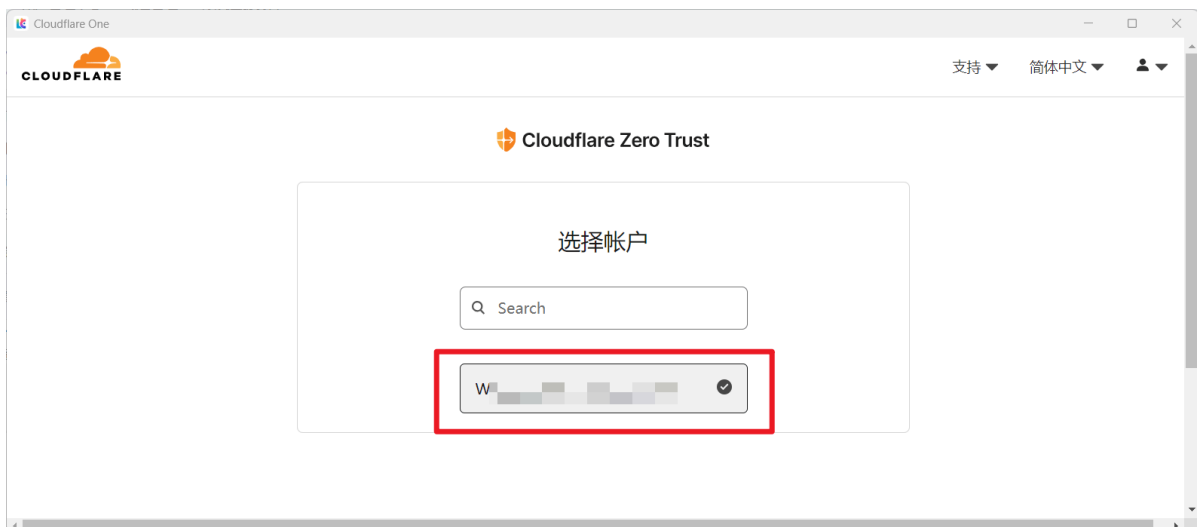
- 在微服上打开应用商店——>下载CloudFlared并打开——>点击右上角切换语言“中文简体”——>点击底部“ZeroTrust仪表盘”



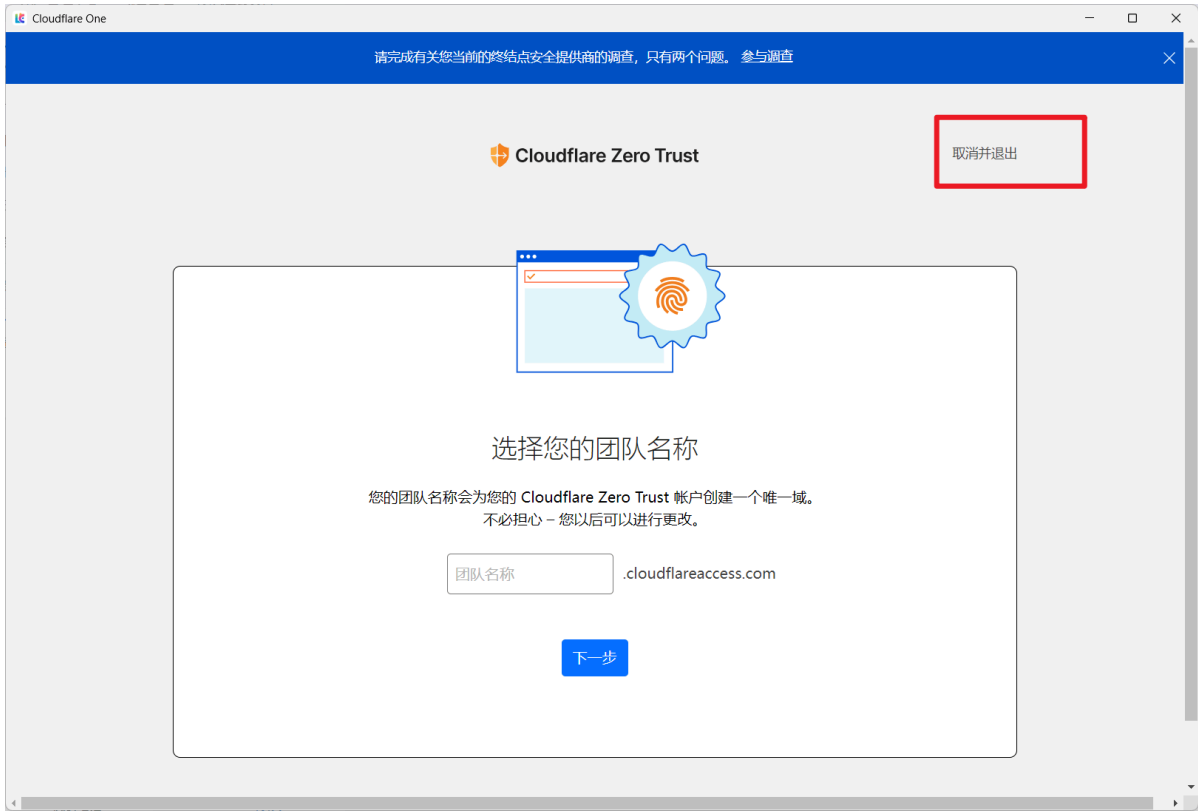
- 这里会出现CloudFlare的登录页面——>点击右上角切换语言“中文简体”——>这里有CloudFlare账号就登录，没有需要注册一个



- 登录后选择一个账户



- 这里CloudFlare会让您搞一个团队名称，这个根据业务需求自行选择，这里我是直接点击右上角“取消并退出”——>回到仪表盘主页



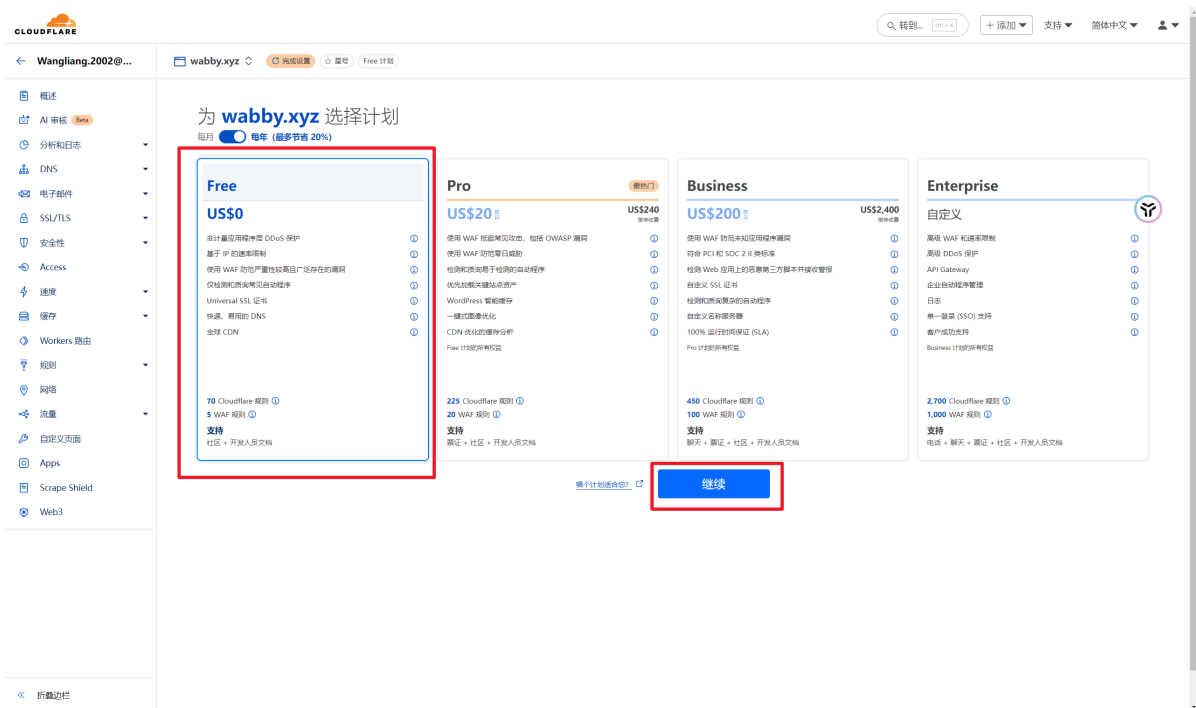
## 2、域名托管

### 1、托管域名配置

- 在仪表盘中输入您事先准备好的域名——>点击继续



- 点击第一个Free——>再点继续



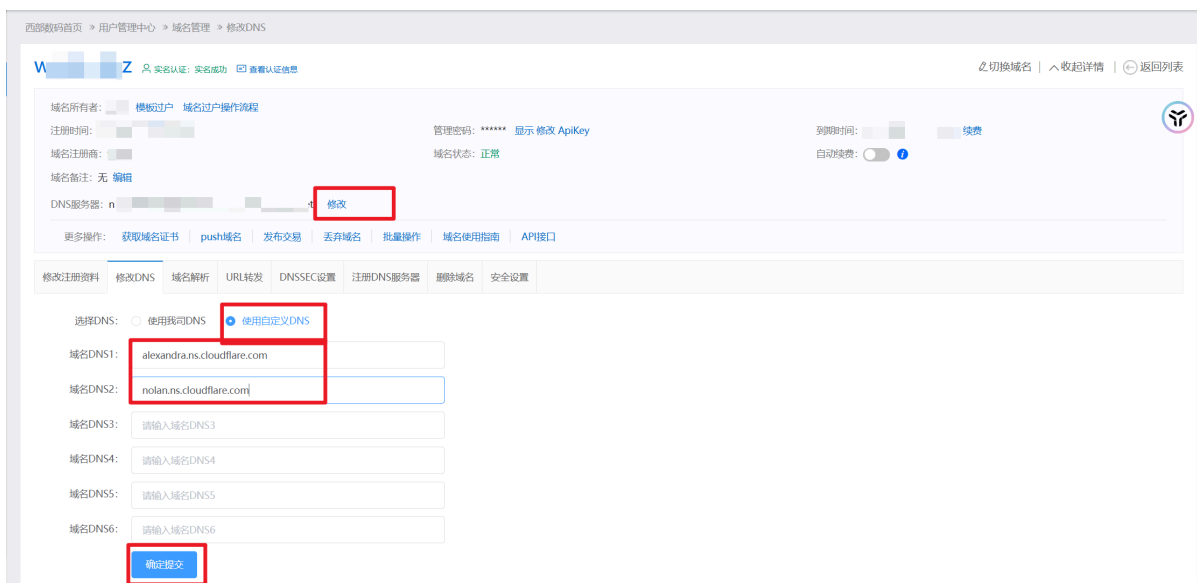
- 继续前往激活——>继续



- 在“将您当前的名称服务器替换为 Cloudflare 名称服务器”中的B项中的两个域名复制一下（备用）



- 回到您注册域名的平台，找到域名管理界面——>修改注册资料——>把DNS服务器换成上一步复制的域名——>确定提交



## 2、等待生效

- 更换之后请耐心等待生效时，各个域名平台检测时效性不同，可以用命令测试是否生效

#在Linux上  
dig NS 域名

#如果一直出不来可以更换一下主机的DNS;如: 223.5.5.5; 8.8.8.8等

#在ANSWER SECTION处产生新配的DNS就证明成功了

```
lzcbox-2b287587 ~ # dig NS wabby.xyz

; <<>> DiG 9.18.28-1~deb12u2-Debian <<>> NS wabby.xyz
;; global options: +cmd
;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 24773
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:: udp: 65494
;; QUESTION SECTION:
;wabby.xyz.                IN      NS

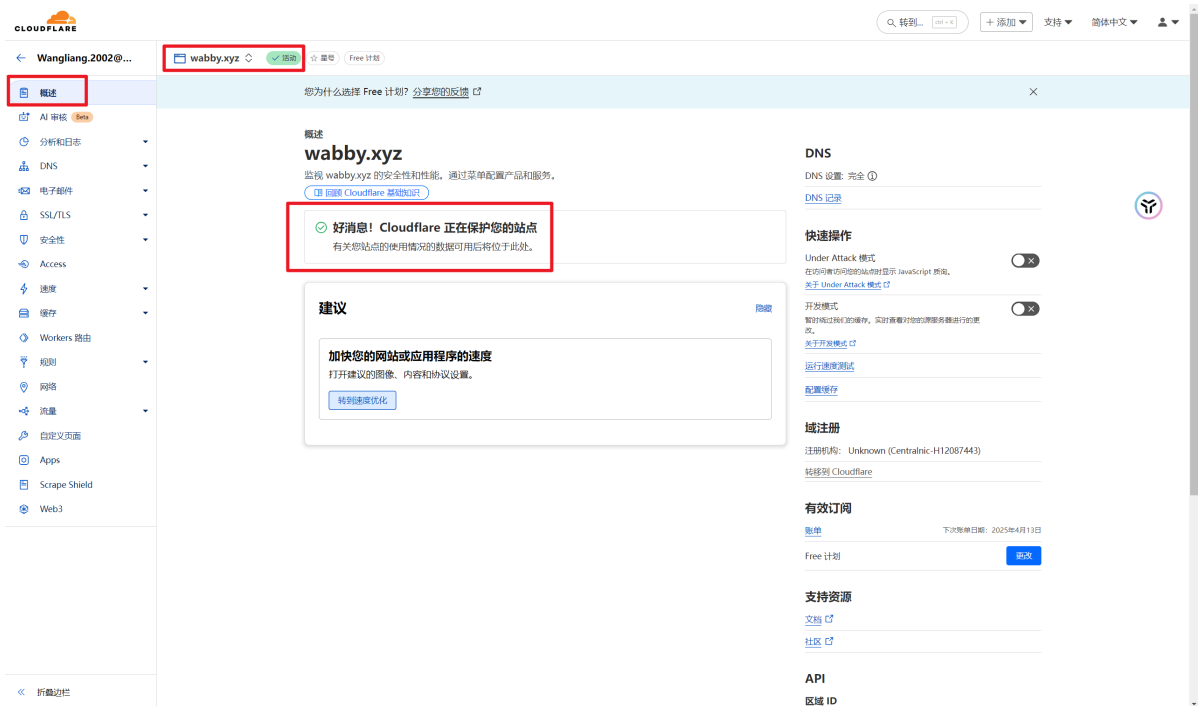
;; ANSWER SECTION:
wabby.xyz.                3600    IN      NS      alexandra.ns.cloudflare.com.
wabby.xyz.                3600    IN      NS      nolan.ns.cloudflare.com.

;; Query time: 460 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Thu Mar 13 10:53:53 CST 2025
;; MSG SIZE  rcvd: 99

lzcbox-2b287587 ~ #
```

### 3、托管完成

- 点击“概况”页面刷新后，出现“好消息！Cloudflare 正在保护您的站点”就证明托管成功了

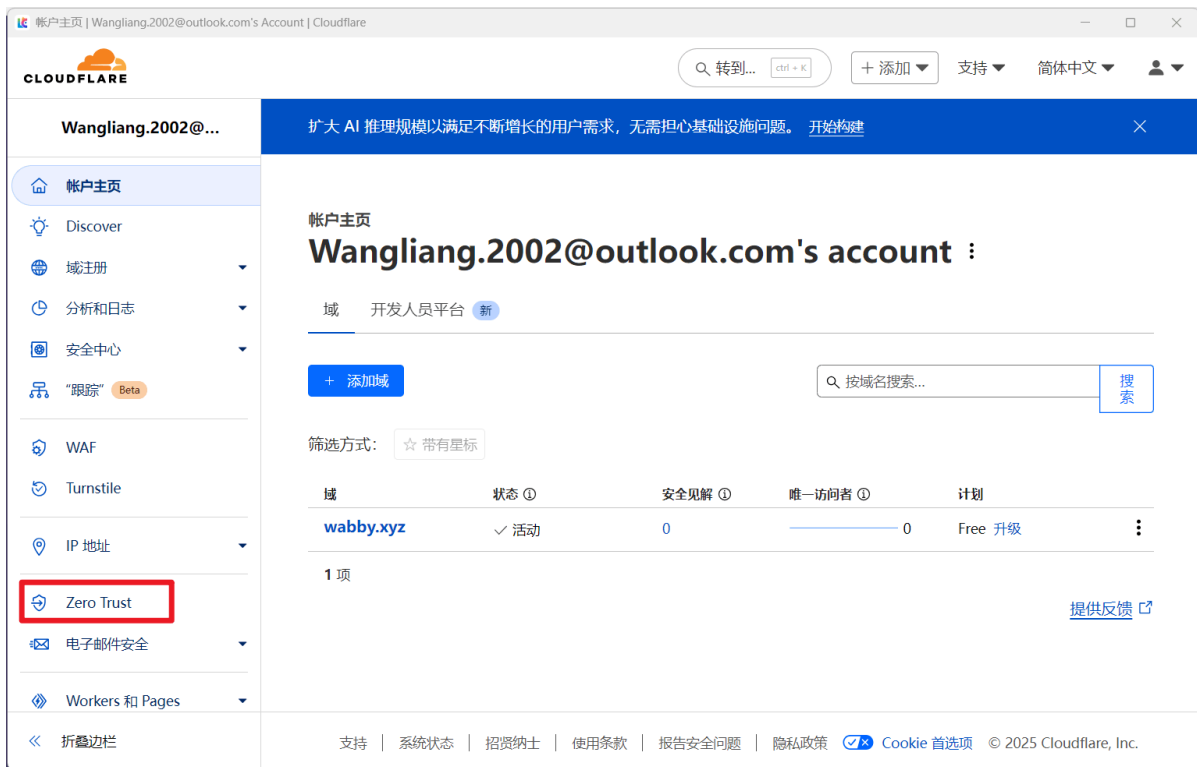


### 3、配置转发

#### 1、微服关联Cloudflare

##### 1、创建隧道

- 在Cloudflare账户主页侧边导航栏打开“Zero Trust”



- 点击“网络”——>Tunnel——>创建隧道



- 隧道类型选择“Cloudflared”

[← 返回到“隧道”](#)

## 创建隧道

创建一个隧道，将 HTTP Web 服务器、SSH 服务器、远程桌面和其他协议安全地连接到 Cloudflare。

[隧道文档](#)

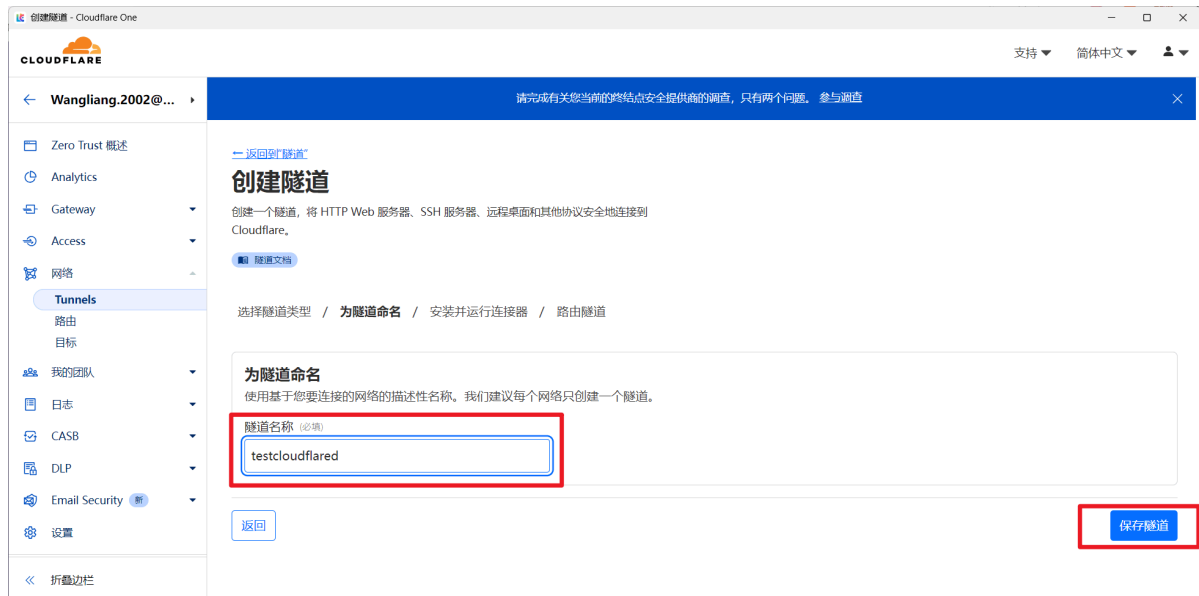
### 选择隧道类型

选择用于将您的资源连接到 Cloudflare 的全球网络的方法。

**Cloudflared**  
推荐  
为用户到网络连接建立安全的仅出站 Cloudflare 连接。  
[选择 Cloudflared](#)

**WARP Connector** Beta  
仅限 Linux 发行版  
支持站点到站点、双向和网状网络连接的进场和离场流量。  
[选择 WARP Connector](#)

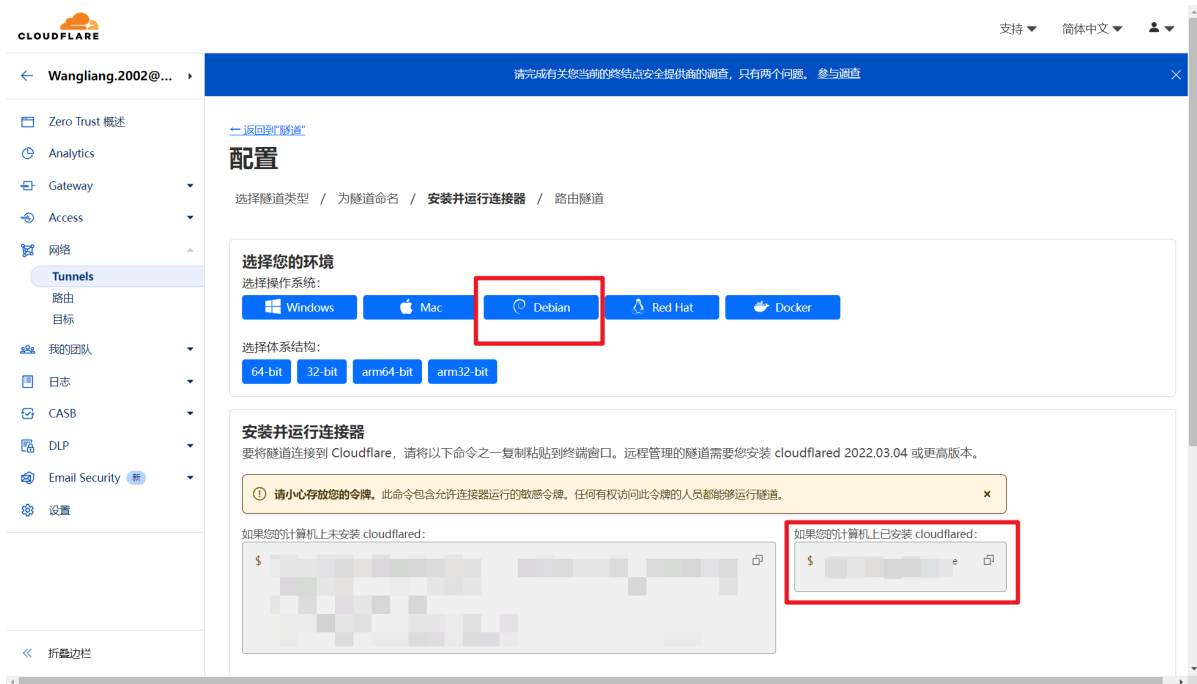
- 为隧道命名——>保存隧道



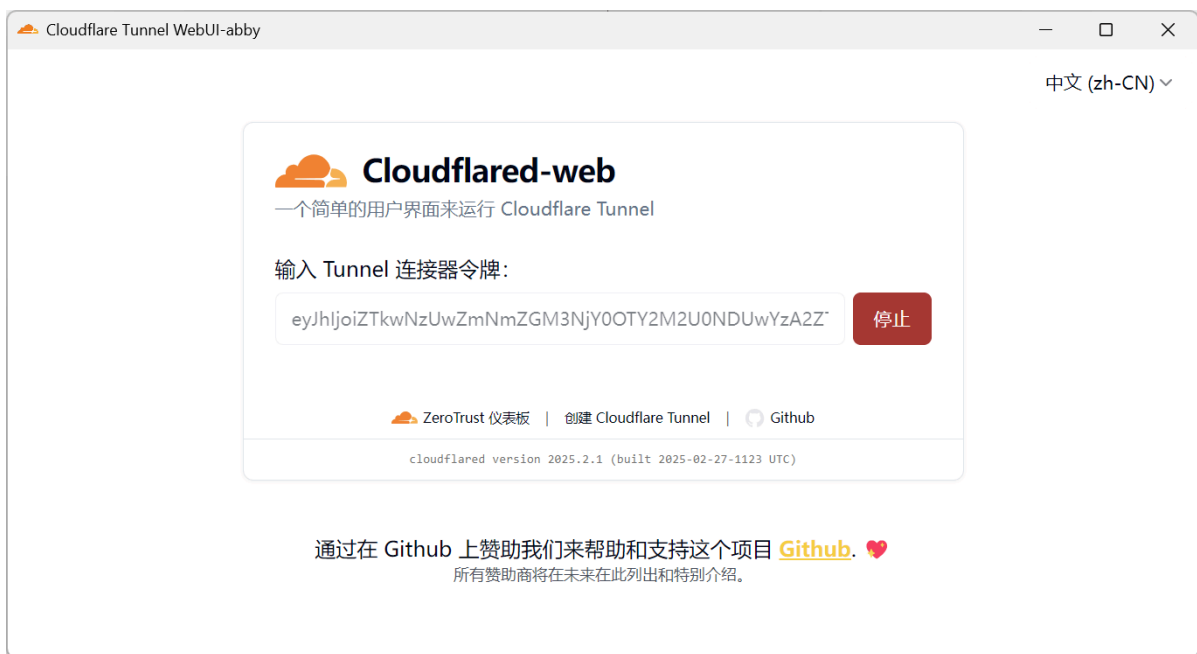
## 2、连接隧道

- 选择Debin——>复制左边“如果您的计算机上已安装 cloudflared：”内容

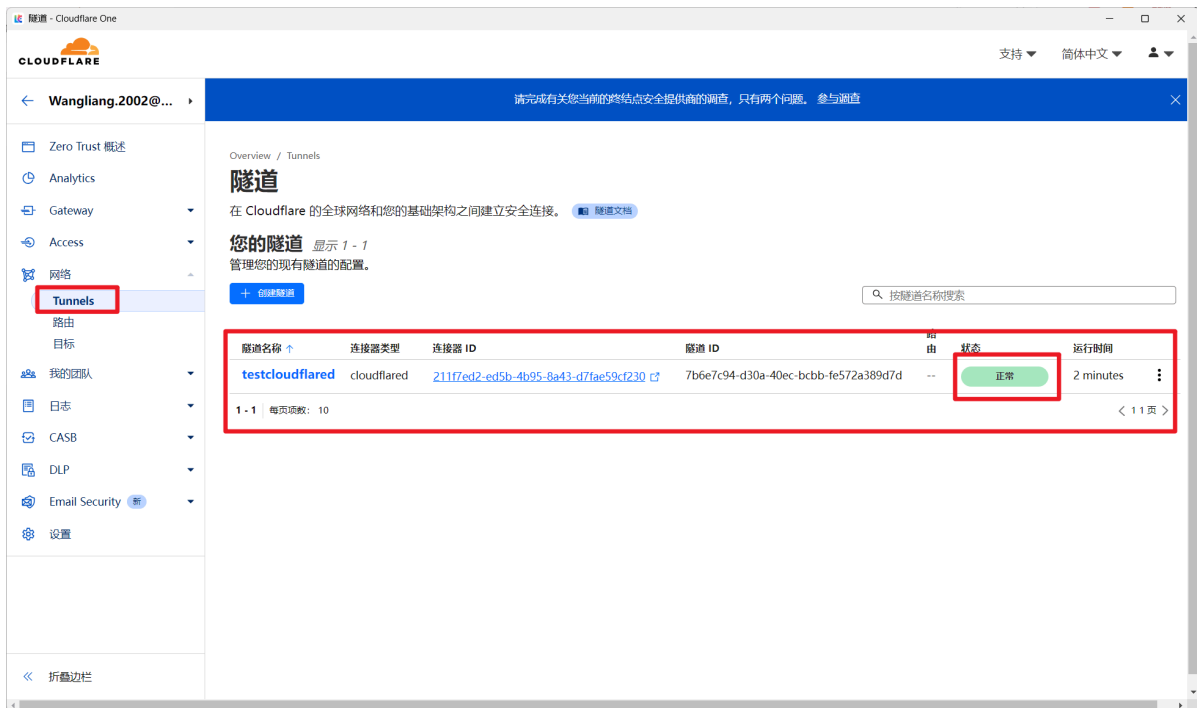




- 粘贴到微服的这个窗口，点击保存——>启动



- 再次点击侧边导航栏“Tunnel”可以看出隧道连接状态正常



## 2、配置应用转发

### 1、获取应用端口

- 进入微服os界面

```
cd /data/system/pkgm/lpks
```

#这个是所安装软件的软件包存放目录

- 找到需要转发到应用，查看是否运行

```
lpk-manager status |grep alist
```

#lpk-manager status |grep 服务名

#这里以alist懒散为例

```
lzcbox-2b287587 /data/system/pkgm/lpks # lpk-manager status |grep alist
cloud.lazycat.app.alist(wlabby) | running
```

- 解压lpk包到外部目录（不要把包解压到当前目录！！！！）

```
ls |grep alsit
```

#查看当前目录包含“alsit”的文件或目录

```
unzip cloud.lazycat.app.alist.lpk -d /root
```


#把cloud.lazycat.app.alist.lpk解压到root目录下（这里解压的目录可以自定义的，切记不要是当前目录！！！！）

```
lzcbox-2b287587 /data/system/pkgm/lpks # ls |grep alist
cloud.lazycat.app.alist.lpk
lzcbox-2b287587 /data/system/pkgm/lpks # ^C
lzcbox-2b287587 /data/system/pkgm/lpks # unzip cloud.lazycat.app.alist.lpk -d /root
Archive: cloud.lazycat.app.alist.lpk
  inflating: /root/content.tar
  inflating: /root/icon.png
  inflating: /root/manifest.yml
lzcbox-2b287587 /data/system/pkgm/lpks #
```

- 回到root目录查看服务端口
- （需注意！！！一个应用包含前端后端等其他组件，用户需要选择需要映射出去的服务端口）

```
cd /root
#回到root目录
cat manifest.yml
#查看manifest.yml，在routes这会有端口，例如现在这个5244
```

```
lzcbox-2b287587 /data/system/pkgm/lpks # cd
lzcbox-2b287587 ~ # cat manifest.yml
lzc-sdk-version: 0.1
package: cloud.lazycat.app.alist
version: 3.34.0-patch3
name: Alist
description: 一个支持多存储的文件列表程序，使用 Gin 和 Solidjs
license: https://choosealicense.com/licenses/mit/
homepage:
author:
usage: 如果想使用 WebDAV 访问 AList，请先去 AList 的管理界面更改密码。
application:
  subdomain: alist
  user_app: true
  routes:
    - /=http://server.cloud.lazycat.app.alist.lzcapp:5244
    - /selfhost/=file:///lzcapp/pkg/content/selfhost
    - /static/manifest.json=file:///lzcapp/pkg/content/manifest.json
services:
  server:
    image: registry.lazycat.cloud/xhofe/alist:v3.34.0-patch11
    environment:
      - ALIST_ADMIN_PASSWORD=admin
    entrypoint: alist server --data /lzcapp/var
lzcbox-2b287587 ~ #
```



## 2、配置转发

- 打开微服——>下载“局域网端口转发工具”应用——>打开——>添加映射规则——>开始填写
- （需注意！！！一个应用包含前端后端等其他组件，用户需要选择需要映射出去的服务端口）

注意事项：

左侧：

局域网出口类型：选择“微服虚拟网卡（仅微服应用容器内可访问）”

微服虚拟网卡：“host.lzcapp” 端口：尽量与需要映射的端口一致

右侧：

转发目标类型：微服应用

微服应用：选择需要映射的应用（这一有个注意点：这个下拉应用选项中需要选Running的）server tcp 端口（这三个根据自身服务来）

添加完成之后点击“测速连接”确保能正常——>创建



### 3、测试Cloudflare应用与转发应用是否正常访问

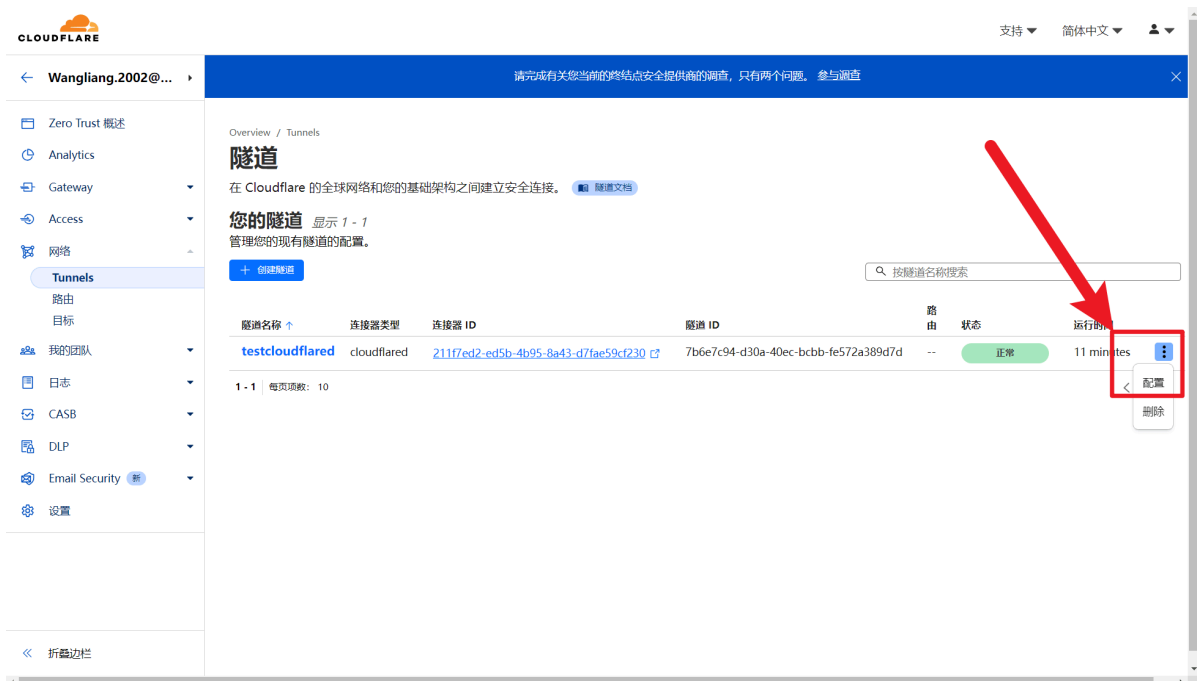
```
lzc-docker ps -a |grep cloudfalre
#找出cloudfalre应用运行的容器
lzc-docker exec -it cloudlazycatappcloudfalredweb-cloudflaredweb-1 bash
#以交互式的 Bash shell, 进入cloudfalred的容器
curl -I host.lzcapp:5244
#访问微服的5244端口, -I只显示请求头信息, 返回状态码200, 证明访问成功
```

```
lzcbox-2b287587 ~ # lzc-docker ps -a |grep cloudfalre
3cfd46466db registry.lazycat.cloud/lzc/lzcapp:3.28.3 ccloudlazycatappcloudfalredweb-app-1 /lzcinit/cloud.lazycat 7 hours ago
Up 7 hours (healthy)
ffa54212cd91 docker.hlmirror.com/wisdomsky/cloudflared-web:2025.2.1 ccloudlazycatappcloudfalredweb-cloudflaredweb-1 /bin/sh -c 'node /v...' 7 hours ago
Up 7 hours
lzcbox-2b287587 ~ # lzc-docker exec -it ccloudlazycatappcloudfalredweb-cloudflaredweb-1 bash
root@lzcbox-2b287587:/var/app# curl -I host.lzcapp:5244
HTTP/1.1 200 OK
Content-Type: text/html
Date: Thu, 13 Mar 2025 08:29:08 GMT
root@lzcbox-2b287587:/var/app#
```

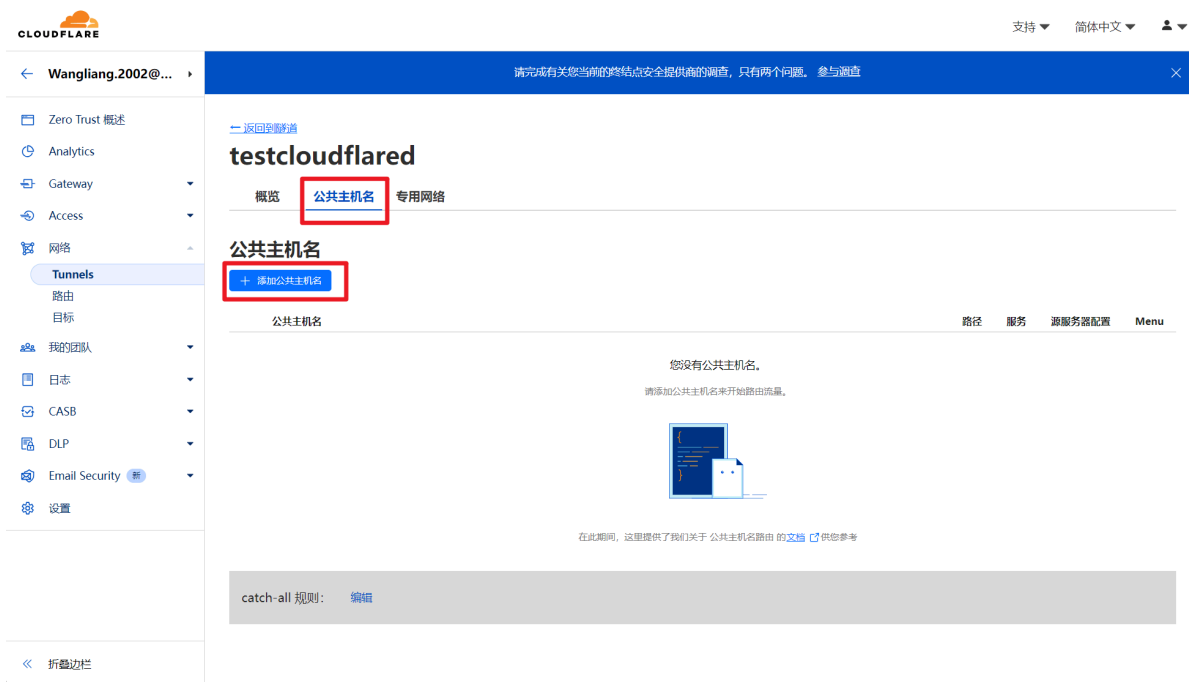
## 4、配置转发域名

### 1、添加公共主机名

- 回到cloudfalre的web页面——>点击隧道后面的“配置”



- 点击“公共主机名”——>添加公共主机名



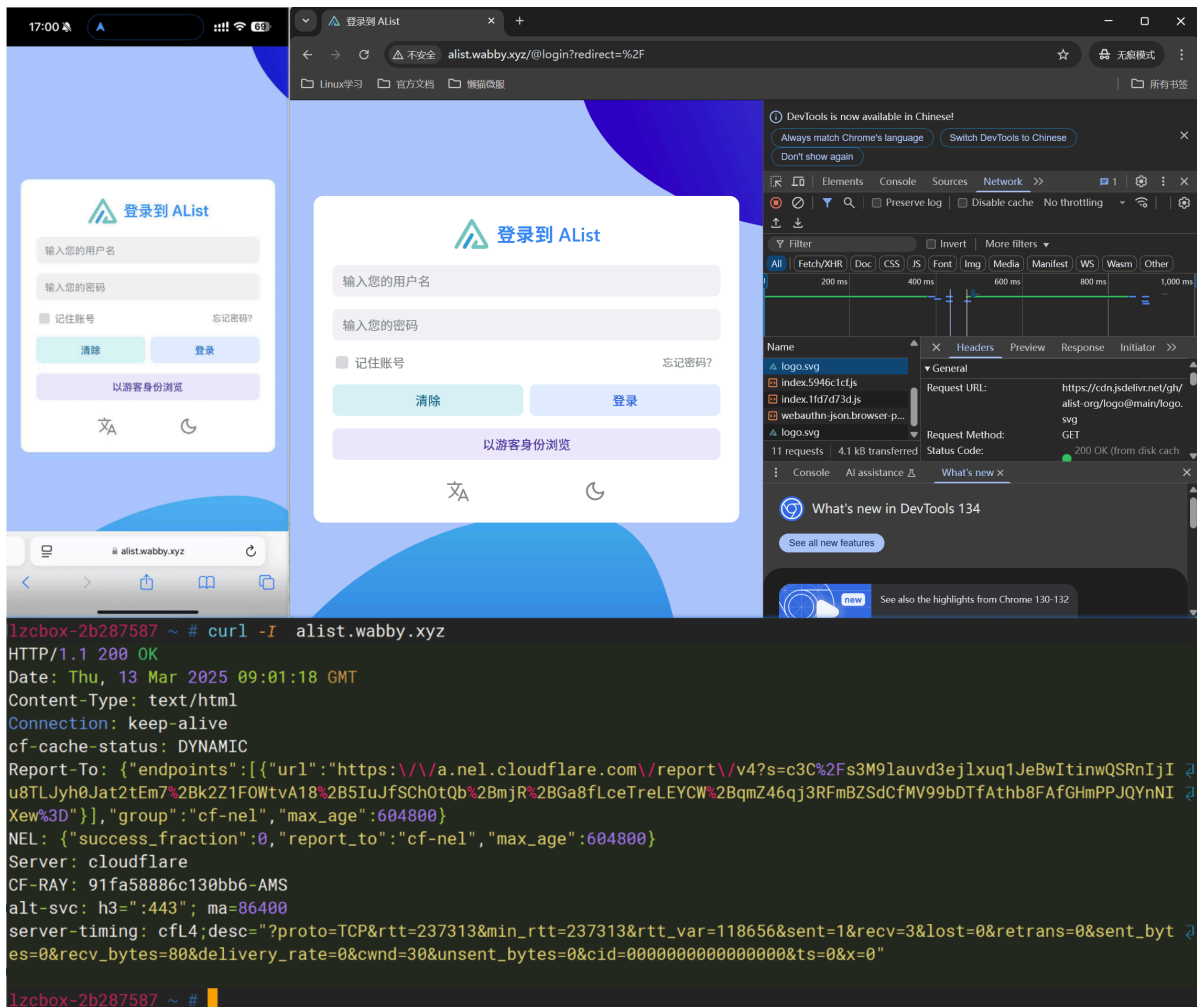
## 2、公共主机名配置

子域：可以自定义（一般写转发的应用名称）  
域：下拉复选需要的域名  
类型：HTTP  
URL: host.lzcapp: 需要转发的应用服务端口  
最后保存主机名



## 4、测试访问

- 分别在手机端、电脑端、命令行端输入alist.wabby.xyz，都能正常访问了



## 5、其他注意事项

为确保系统安全，避免暴露高风险的直连后端端口，应严格控制端口映射和转发策略。对于长期未使用的端口，应及时取消其映射或转发配置，以降低潜在的安全风险，防止因端口暴露而导致的未经授权访问或网络攻击，从而保障系统的稳定性和数据安全。