  随着chatGPT的发布，深刻地影响了人们的学习习惯，由凡事问google变成了凡事问chatGPT。但苦于chatGPT服务还未对国内用户开放，要想访问chatGPT 只能通过科学上网来实现。本文主要介绍了通过v2ray搭建代理服务器，通过科学上网实现对chatGPT的访问。阅读本文所需的知识点主要包括以下几点：

基本知识：

1) 懂得云平台的基本操作，比如创建虚拟机，设置入站端口规则等。

2) 了解Linux操作系统，会简单的命令。

进阶知识：

1) 知道nginx是什么，知道基本的配置， 知道什么是反向代理，懂得如何用nginx实现反向代理。

2) 了解https基本原理，知道openssl是什么，懂得基本的证书相关的配置命令。

掌握上述知识后，即可开始配置代理服务器，实现科学上网之旅。

# 申请虚拟机

推荐在微软云上(azure.microsoft.com)申请一台虚拟机，微软和中国Z.F的关系处的比较好，不容易被封。假设下面是申请好的一台虚拟机:

位 置：Japan East

大 小：Standard B2s (2 vcpu，4 GiB 内存)

操作系统：Linux (ubuntu 20.04)

公网IP 地址：20.210.100.4

# **v2ray服务器端安装配置**

1. 安装v2ray服务端

# bash < (curl -L https://raw.githubusercontent.com/v2fly/fhs-install-v2ray/master/install-release.sh)

*注意：需要以su权限运行*

1. 修改配置文件

配置文件默认路径：/usr/local/etc/v2ray/config.json，初始值为空{}

|  |
| --- |
| {  "inbounds": [  {  "port": 10086, // 服务器监听端口  "protocol": "vmess",  "settings": {  "clients": [  {  "id": "e3e717d8-8e80-4356-8e02-f83d7456d354" //需要与客户端保持一致，可以在客户端生成后拷贝到此处  }  ]  }  }  ],  "outbounds": [  {  "protocol": "freedom"  }  ]  } |

1. 重启服务并检查服务状态

# systemctl start v2ray //启动服务

# systemctl status v2ray //确定服务状态正常

# netstat -an | grep 10086 //确定服务端口已开启

*注意：需通过微软云的控制台配置网络入站端口规则，允许访问10086端口的流量进入*

# **v2ray客户端安装和配置**

1. 下载安装客户端程序

这里下载的是5.39版本，5.x的最后一个版本：

https://github.com/2dust/v2rayN/releases/download/5.39/v2rayN-Core.zip

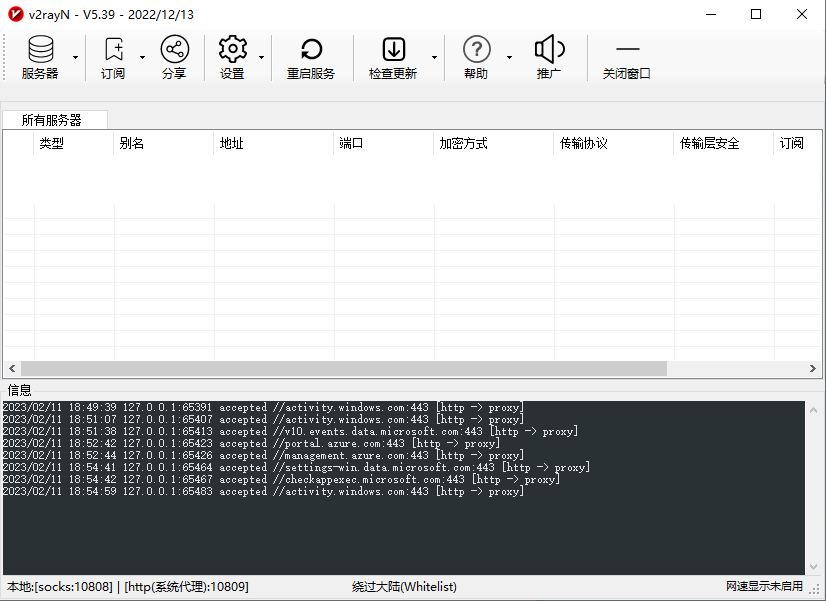
下载后拷贝到一个合适的目录（比如d:\）,直接解压就行。

1. 启动客户端程序

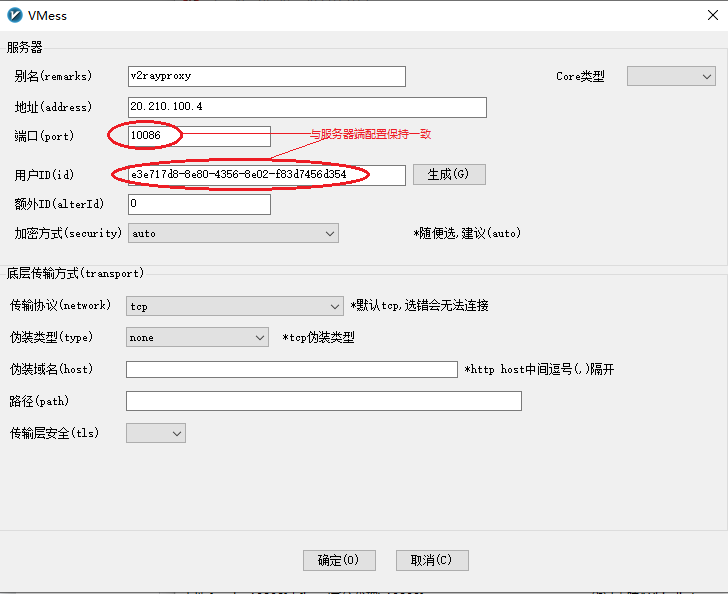
进入d:\v2rayN-Core目录，双击v2rayN.exe，在右下角会出现一个托盘：

1. 新建与服务器的连接

点击托盘会弹出以下主界面：



选择左上角的“服务器”图标，在弹出菜单中选择“添加[VMess]服务器”，弹出配置界面，按下图提示进行配置：



1. 设置代理

右键托盘图标，在弹出菜单中选择：系统代理->自动设置系统代理

1. 访问外网

通过上述步骤，即可开始访问外网，比如：facexxxx.com, gxxgle.com,以及当前流行的openai.com.

# **添加https伪装**

采用上述简易配置已可实现对外网的访问，但也容易被封堵（国家FHQ还是很厉害的）。要降低被封堵的风险，可以将流量伪装成https的流量。

## 创建自签名证书

1. 创建配置文件

/etc/nginx/certs/v2ray.conf

|  |
| --- |
| [req]  distinguished\_name = req\_distinguished\_name  x509\_extensions = v3\_req  prompt = no  encrypt\_key = no  [req\_distinguished\_name]  C = CN  L = Beijing  O = MyCompany  CN = 20.210.100.4  [v3\_req]  keyUsage = critical, digitalSignature, keyAgreement  extendedKeyUsage = serverAuth |

1. 创建证书私钥和证书请求

# cd /etc/nginx/certs

# openssl req -new -config v2ray.conf -keyout v2ray.key -out v2ray.csr

1. 生成自签名证书

# openssl x509 -req -days 365 -in v2ray.csr -signkey v2ray.key -out v2ray.crt

## **安装配置nginx https反向代理**

1. 安装nginx

# apt install nginx

# systemctl start nginx //启动服务

通过浏览器访问http://20.210.100.4，能正常显示Nginx初始页面

*注意：需通过微软云的控制台配置网络入站端口规则，允许访问80端口的流量进入*

1. 创建反向代理配置文件

/etc/nginx/sites-available/v2ray

|  |
| --- |
| server {  listen 443 ssl;  listen [::]:443 ssl;    ssl\_certificate /etc/nginx/certs/v2ray.crt; #对应上面创建的证书  ssl\_certificate\_key /etc/nginx/certs/v2ray.key; #对应上面创建的密钥  ssl\_session\_timeout 1d;  ssl\_session\_cache shared:MozSSL:10m;  ssl\_session\_tickets off;    ssl\_protocols TLSv1.2 TLSv1.3;  ssl\_ciphers ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384;  ssl\_prefer\_server\_ciphers off;    server\_name 20.210.100.4;  location /ray { # 与 v2ray（包括客户端和服务端）配置中的 path 保持一致  if ($http\_upgrade != "websocket") { # WebSocket协商失败时返回404  return 404;  }  proxy\_redirect off;  proxy\_pass http://127.0.0.1:10000; # 与v2ray服务端的配置保持一致  proxy\_http\_version 1.1;  proxy\_set\_header Upgrade $http\_upgrade;  proxy\_set\_header Connection "upgrade";  proxy\_set\_header Host $host;  # Show real IP in v2ray access.log  proxy\_set\_header X-Real-IP $remote\_addr;  proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;  }  } |

1. 激活配置文件

# cd /etc/nginx/sites-enabled

# ln -s /etc/nginx/sites-available/v2ray v2ray

1. 重启nginx服务

# systemctl restart nginx //重新启动服务

# systemctl status nginx //确定服务状态正常

# netstat -an | grep 443 //确定服务端口已开启

*注意：需通过微软云的控制台配置网络入站端口规则，允许访问443端口的流量进入*

## **修改v2ray服务器端配置**

1. 修改配置文件

/usr/local/etc/v2ray/config.json

|  |
| --- |
| {  "inbounds": [  {  "port": 10000,  "listen":"127.0.0.1", //只监听 127.0.0.1，避免除本机外的机器探测到开放了 10000 端口  "protocol": "vmess",  "settings": {  "clients": [  {  "id": "e3e717d8-8e80-4356-8e02-f83d7456d354", //保持和客户端相同  "alterId": 0  }  ]  },  "streamSettings": {  "network": "ws",  "wsSettings": {  "path": "/ray" //需要和客户端的配置保持一致  }  }  }  ],  "outbounds": [  {  "protocol": "freedom",  "settings": {}  }  ]  } |

1. 重启服务并检查服务状态

# systemctl restart v2ray //重新启动服务

# systemctl status v2ray //确定服务状态正常

# netstat -an | grep 10000 //确定服务端口已开启

## **修改v2ray客户端配置**

按照下图所示修改客户端配置：

