使用HASSIO插件配置域名与证书(1)

【操作步骤】

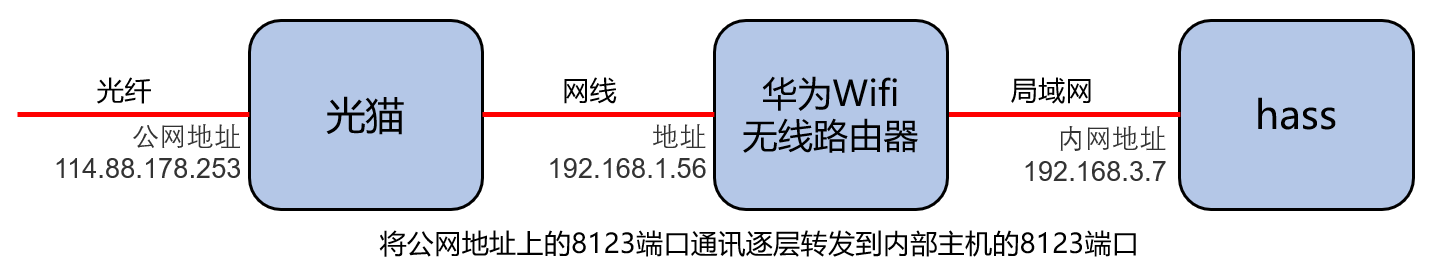
1. 公网访问三大件

* 打通公网访问链路
* 配置域名
* 申请与配置网站数字证书

1. 基于家庭宽带的公网IP+端口映射实现公网IP访问

- 申请家庭宽带公网IP

- 逐层端口转发



1. DuckDNS动态域名服务（ddns）的实现

解决公网IP地址不便于记忆，以及公网IP地址在重启上网设备时会变化的问题

使用HASSIO插件配置域名与证书(2)

【操作步骤】

1. 公网访问三大件

* 打通公网访问链路
* 配置域名
* 申请与配置网站数字证书

1. 获得网站数字证书

duckdns插件中的配置

lets\_encrypt:

accept\_terms: true

certfile: fullchain.pem

keyfile: privkey.pem

……

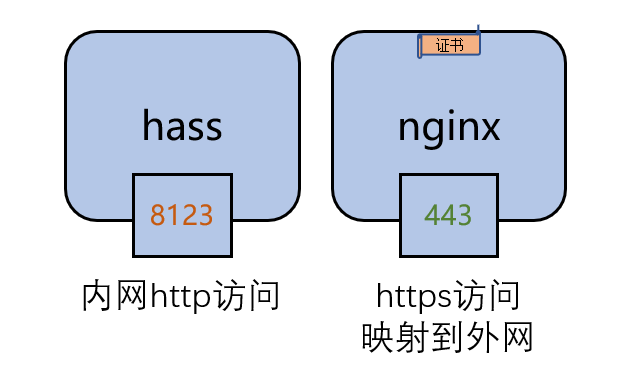
1. 将数字证书配置在HomeAssistant中

http:

ssl\_certificate: /ssl/fullchain.pem

ssl\_key: /ssl/privkey.pem

1. nginx SSL Proxy



申请免费的Amazon云服务器

【操作步骤】

1. 访问亚马逊云服务，注册账号
2. 申请ec2云服务器
3. 使用puttygen生成访问密钥
4. 使用putty访问云服务器
5. 云服务器防火墙设置

【参考】

* 亚马逊云服务网站

https://aws.amazon.com/

* puttygen下载地址

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

* SUSE下nano的安装

suse的很多操作命令与其它linux有差别（大家可以选熟悉的ubuntu系统，它的操作基本与树莓派一致）

很多同学问SUSE下nano的安装，参考：<https://software.opensuse.org/download.html?project=editors&package=nano>

命令：

sudo zypper addrepo https://download.opensuse.org/repositories/editors/SLE\_15/editors.repo

sudo zypper refresh

sudo zypper install nano

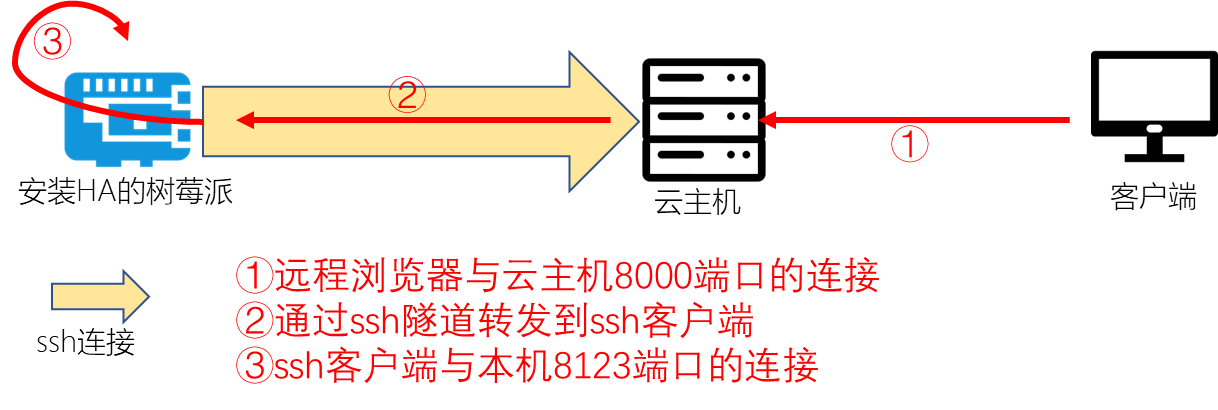
ssh隧道构建

【操作步骤】

1. 在树莓派上使用ssh登录云主机
2. 构建ssh隧道
3. 安装与使用autossh
4. 启动时自动执行autossh命令

【参考】

* 连接示意图



* 自启动/etc/rc.local添加内容

sudo -u pi /usr/bin/autossh -i "/home/pi/etc/amazon\_xinjiapo.pem" -R 0.0.0.0:8000:127.0.0.1:8123 ec2-user@ec2-54-251-155-96.ap-southeast-1.compute.amazonaws.com -N -f

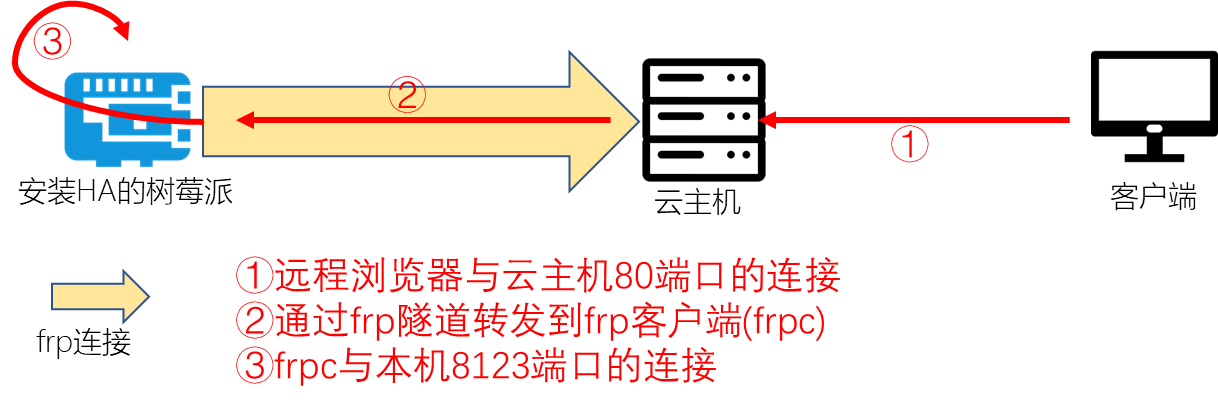
frp隧道构建

【操作步骤】

1. 打开防火墙规则
2. 在云主机上下载、配置、运行frp服务器端
3. 在树莓派上下载、配置、运行frp客户端
4. 将frps和frpc命令分别加入云主机与树莓派的启动执行中

【参考】

* 连接示意图



* frp软件github地址

<https://github.com/fatedier/frp/>

* frp软件下载地址

<https://github.com/fatedier/frp/releases>

* frp服务器端配置

[common]

bind\_port = 7000

token = a1234

* frp客户端配置

[common]

server\_addr = ec2-54-251-155-96.ap-southeast-1.compute.amazonaws.com

server\_port = 7000

token = a1234

[HA]

type = tcp

local\_ip = 127.0.0.1

local\_port = 8123

remote\_port = 80

* frp客户端自启动添加内容

注：前半部分在等待网络启动后才运行对应命令，其中网关地址需要修改为你实际环境中的地址

(

until ping -nq -c3 192.168.31.1; do

# Waiting for network

sleep 5

done

/home/pi/frp\_0.21.0\_linux\_arm/frpc -c /home/pi/frp\_0.21.0\_linux\_arm/frpc.ini

)&

为HA配上域名与数字证书

【操作步骤】

1. 申请一个免费的duckdns.org子域名
2. 下载certbot-auto
3. 申请数字证书
4. 在homeassistant中配置https访问
5. 设置frp隧道中443远程端口映射
6. 更新数字证书

【参考】

* duckdns官网

<https://www.duckdns.org/>

* Let’s Encrypt官网

<https://letsencrypt.org>

* certbot官网

<https://certbot.eff.org/>

* certbot-auto下载地址

https://dl.eff.org/[certbot](https://dl.eff.org/certbot-auto)-auto或<https://github.com/certbot/certbot/raw/master/certbot-auto>

* 申请证书命令

sudo certbot-auto certonly --standalone --preferred-challenges http-01 --http-01-port 8123 --email your@email.address -d examplehome.duckdns.org

* 增加证书文件访问权限命令

sudo chmod 755 /etc/letsencrypt/live

sudo chmod 755 /etc/letsencrypt/archive

sudo chmod +r /etc/letsencrypt/archive -R

* HomeAssistant中证书的配置样例

http:

ssl\_certificate: /etc/letsencrypt/live/examplehome.duckdns.org/fullchain.pem

ssl\_key: /etc/letsencrypt/live/examplehome.duckdns.org/privkey.pem

base\_url: examplehome.duckdns.org

* 更新证书命令

sudo certbot-auto renew --standalone --preferred-challenges http-01 --http-01-port 8123

sudo certbot-auto renew --standalone --preferred-challenges tls-sni-01 --tls-sni-01-port 8123（已过期）

注：

以上命令执行前需要先停止hass以释放8123端口。

其中的http-01用于远程80端口映射到本地的8123端口的情况，tls-sni-01用于远程的443端口映射到本地8123端口的情况（tls-sni-01更新方式已经不再被支持）。你也可以使用非8123端口，这样就不用事先停止home assistant了，但需要额外构建通讯隧道。

命令中可以中加入--pre-hook和--post-hook参数用于指定更新前自动执行的停止hass和更新后启动hass的命令，如--pre-hook “sudo systemctl stop home-assistant@pi”

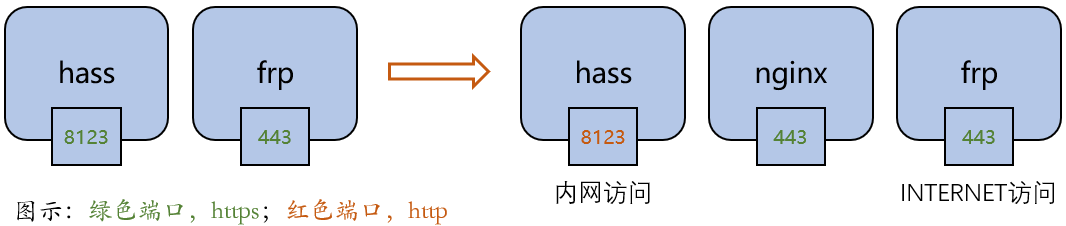
nginx代理

【操作步骤】

1. 问题与解决方案（为什么要用nginx代理）
2. 安装nginx
3. 修改nginx的配置
4. 修改HA配置
5. 修改frp配置
6. 演示

【参考】

* 示意图



* nginx官网

<http://nginx.org/>

* 安装nginx

sudo apt-get install nginx

* nginx配置内容

server {

listen 443;

server\_name zjs.duckdns.org;

ssl on;

ssl\_certificate /etc/letsencrypt/live/zjs.duckdns.org/fullchain.pem;

ssl\_certificate\_key /etc/letsencrypt/live/zjs.duckdns.org/privkey.pem;

ssl\_prefer\_server\_ciphers on;

location / {

proxy\_pass http://127.0.0.1:8123;

proxy\_set\_header Host $host;

proxy\_http\_version 1.1;

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection "upgrade";

}

}

* 修改nginx配置命令

#编辑配置文件ha\_ssl

sudo vi /etc/nginx/sites-available/ha\_ssl

#在sites-enabled目录下建立配置文件链接

sudo ln -sf /etc/nginx/sites-available/ha\_ssl /etc/nginx/sites-enabled/default

#重新加载nginx配置

sudo nginx -s reload