

## 冬笋云远程配置网关的 OPENVPN

内置了 FreeIOE 的网关都已支持 OPENVPN。基于 OPENWRT 19.07 版本的网关配置 OPENVPN 更加方便些。

1. 登录冬笋云，将需要配置 OPENVPN 的网关加入自己账户下。
2. 由于远程网关通过 4G 上网或者在企业内网，一般不能直接进入网关的系统配置界面。因此需要借助互联网上内网穿透中转服务器，或者自己搭建内网穿透中转服务器，FreeIOE 的应用 port\_tunnel 是基于内网穿透工具 FRP 封装的，因此只要是互联网上开放的 FRPS 服务器都可以使用。自己搭建也非常方便，公网上有服务器即可。冬笋云临时提供的 FRPS 服务器如下：

FRPS 服务器域名	FRPS 端口	接入 Token
zjk.proxy.thingsroot.com	1699	F^AYnHp29U=M96#o&ESqXB3pL=\$)W*qr
sz.proxy.thingsroot.com	1699	F^AYnHp29U=M96#o&ESqXB3pL=\$)W*qr

3. 给网关安装应用 port\_tunnel。



给安装的应用取运行实例名，一般就命名 port\_tunnel，如安装多个，取不同名字即可，服务器信息我使用列表中的 zjk.proxy.thingsroot.com 这个服务器的。Tcp 映射就是指将远程网关所在网络中的设备端口映射到互联网或是本地的某个端口如是网关自己的 WEB 管理端口 80，则增加一组映射：self\_webadmin | 127.0.0.1 | 80 | 0。远程开发可以去掉勾选，应用配置截图如下：

可视化编辑 文本编辑(JSON)

实例名: port\_tunnel

应用配置信息

服务器地址: zjk.proxy.thingsroot.com

服务器端口: 1699

连接授权: F^AYnHp29U=M96#o&ESqXB3pL=\$JW"qr

连接协议: tcp

远程开发:

tcp映射设备

tcp映射设备

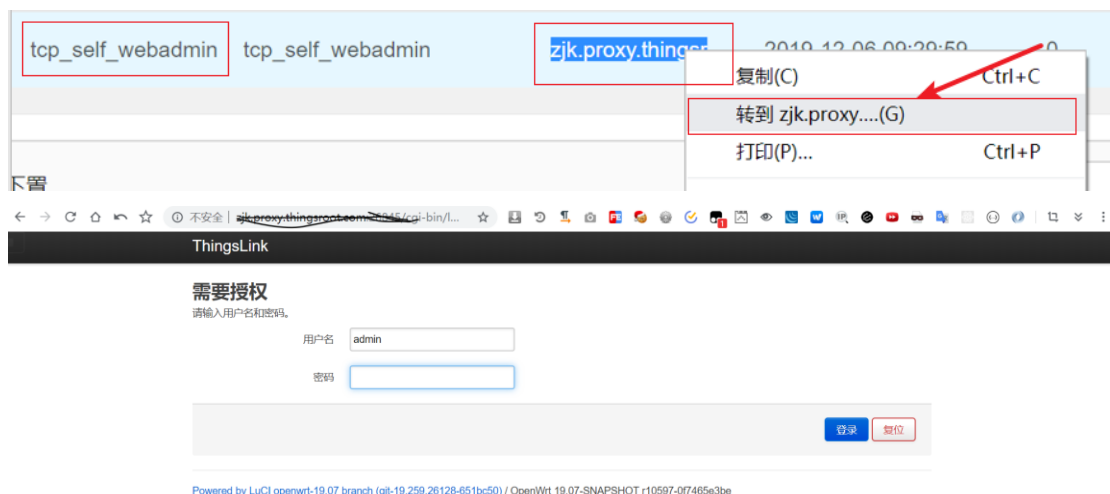
添加

设备序列号	设备IP	设备端口	映射端口	操作
self_webadmin	127.0.0.1	80	0	删除

安装 取消

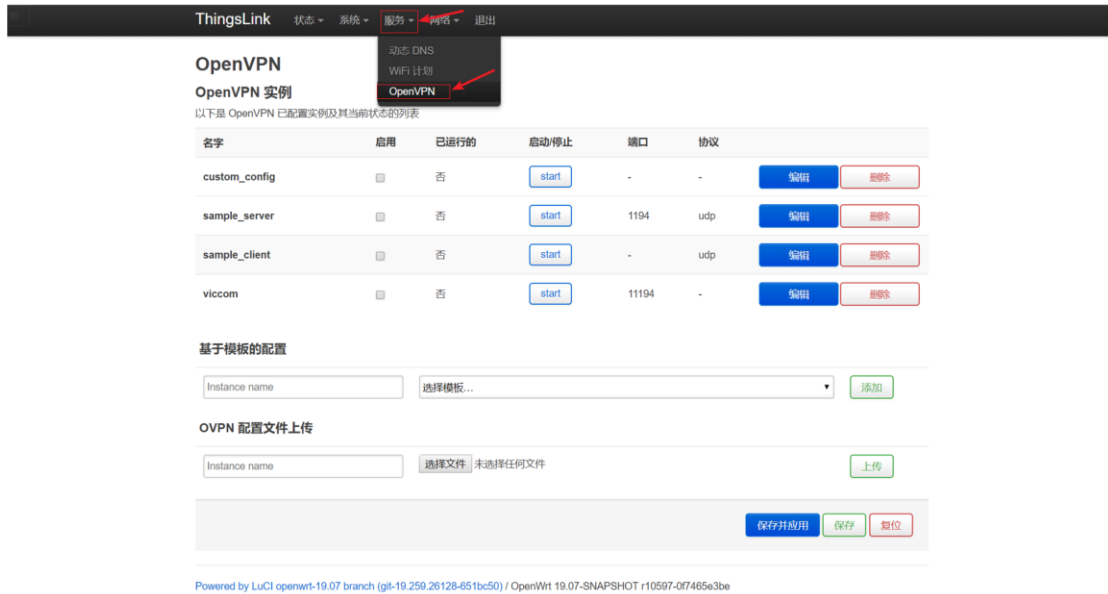
上图中映射端口填 0 是因为无法确定指定在服务器上创建的端口是否被占用，如是自己搭建的服务器，是可以指定一个服务器上未使用的端口的。

- 从设备列表在查看 port\_tunnel 设备中映射端口当前的值，上面配置了序号为 self\_webadmin 的 tcp 设备的映射，在设备 port\_tunnel 中找到名为 tcp\_self\_webadmin 变量名，看一看值是多少，将数值全选，粘贴到新的浏览器标签页中，就可以访问远程网关的系统配置页面了。



- 使用远程网关的系统管理的登录用户名/密码 (admin/admin) 登录系统管理。在顶部导

航中点击“服务”下面的“OPENVPN”，进入 OPENVPN 配置页面。



6. 使用“OVPN 配置文件上传”这个功能将已经可用的 OPENVPN 配置文件上传到远程网关中。

## OpenVPN

### OpenVPN 实例

以下是 OpenVPN 已配置实例及其当前状态的列表

名字	启用	已运行的	启动/停止	端口	协议		
custom_config	<input type="checkbox"/>	否	<input type="button" value="start"/>	-	-	<input type="button" value="编辑"/>	<input type="button" value="删除"/>
sample_server	<input type="checkbox"/>	否	<input type="button" value="start"/>	1194	udp	<input type="button" value="编辑"/>	<input type="button" value="删除"/>
sample_client	<input type="checkbox"/>	否	<input type="button" value="start"/>	-	udp	<input type="button" value="编辑"/>	<input type="button" value="删除"/>
viccom	<input type="checkbox"/>	否	<input type="button" value="start"/>	11194	-	<input type="button" value="编辑"/>	<input type="button" value="删除"/>

### 基于模板的配置

Instance name  选择模板...

**OVPN 配置文件上传**

1. 定义个名称  2. 上传OPENVPN配置文件  viccom.ovpn  3

7. 上传后的页面上出现了刚刚定义的 OPENVPN 实例名，目前的状态是未启用，不在运行中。

## OpenVPN

### OpenVPN 实例

以下是 OpenVPN 已配置实例及其当前状态的列表

名字	启用	已运行的	启动/停止	端口	协议		
custom_config	<input type="checkbox"/>	否	<a href="#">start</a>	-	-	<a href="#">编辑</a>	<a href="#">删除</a>
sample_server	<input type="checkbox"/>	否	<a href="#">start</a>	1194	udp	<a href="#">编辑</a>	<a href="#">删除</a>
sample_client	<input type="checkbox"/>	否	<a href="#">start</a>	-	udp	<a href="#">编辑</a>	<a href="#">删除</a>
viccom	<input type="checkbox"/>	否	<a href="#">start</a>	11194	-	<a href="#">编辑</a>	<a href="#">删除</a>
viccomVPN	<input checked="" type="checkbox"/>	否	<a href="#">start</a>	11194	-	<a href="#">编辑</a>	<a href="#">删除</a>

#### 基于模板的配置

Instance name  选择模板... [添加](#)

#### OVPN 配置文件上传

Instance name  [选择文件](#) 未选择任何文件 [上传](#)

[保存并应用](#) [保存](#) [复位](#)

8. 勾选启用后，可让操作系统开机就运行此 OPENVPN 实例，在实例名右边勾选启用。并点击“保存并应用”按钮。稍等几秒后，页面中状态就变化了，勾选的实例已经启动。

## OpenVPN

### OpenVPN 实例

以下是 OpenVPN 已配置实例及其当前状态的列表

名字	启用	已运行的	启动/停止	端口	协议		
custom_config	<input type="checkbox"/>	否	<a href="#">start</a>	-	-	<a href="#">编辑</a>	<a href="#">删除</a>
sample_server	<input type="checkbox"/>	否	<a href="#">start</a>	1194	udp	<a href="#">编辑</a>	<a href="#">删除</a>
sample_client	<input type="checkbox"/>	否	<a href="#">start</a>	-	udp	<a href="#">编辑</a>	<a href="#">删除</a>
viccom	<input type="checkbox"/>	否	<a href="#">start</a>	11194	-	<a href="#">编辑</a>	<a href="#">删除</a>
viccomVPN	<input checked="" type="checkbox"/>	是 (574)	<a href="#">stop</a>	11194	-	<a href="#">编辑</a>	<a href="#">删除</a>

#### 基于模板的配置

Instance name  选择模板... [添加](#)

#### OVPN 配置文件上传

Instance name  [选择文件](#) 未选择任何文件 [上传](#)

[保存并应用](#) [保存](#) [复位](#)

9. 如需要修改此实例的配置文件，点击实例名后面的编辑按钮。

10. OPENVPN 启动的日志可通过点击导航菜单“状态”下的“系统日志”进入查看。



11. OPENVPN 的实例如果运行正常，可通过点击导航菜单“状态”下的“系统日志”进入查看路由表，检查是否出现名称包含 tun 的网卡设备，同时查看路由表中是否出现应该出现的路由。

ThingsLink 状态 - 系统 - 服务 - 网络 - 退出

**路由表**  
以下规则当前在系统

**ARP**

IPv4 地址	MAC 地址	接口
192.168.1.114	30:B4:9E:5C:87:CA	lan

概览  
防火墙  
**路由表**  
系统日志  
内核日志  
系统进程  
实时信息  
WireGuard 状态

**活动的 IPv4 路由**

网络	对象	IPv4 网关	跃点数	表
wwan0	0.0.0.0/0	10.100.7.105	10	main
tun0	10.0.0.0/16	172.30.252.1	0	main
wwan0	10.100.7.96/28	-	10	main
tun0	172.18.18.0/24	172.30.252.1	0	main
tun0	172.30.0.0/16	172.30.252.1	0	main
tun0	172.30.252.0/22	172.30.252.1	0	main
tun0	172.30.252.1	-	0	main
lan	192.168.1.0/24	-	0	main
tun0	192.168.199.0/24	172.30.252.1	0	main
tun0	192.168.255.0/24	172.30.252.1	0	main

**活动的 IPv6 路由**

网络	对象	源地址	跃点数	表
lan	fd69:ed0b:62c::/64		1024	main
lan	:::8		256	local
(wwan0)	:::8		256	local
(tun0)	:::8		256	local

**IPv6 网上邻居**

IPv6 地址	MAC 地址	接口
---------	--------	----

zjk.proxy.thingsroot.com:26845/cgi-bin/luci/admin/status/routes

12. 可通过点击导航菜单“网络”下的“网络诊断”进入网络诊断页面，通过页面提供的 ping 等工具检测远程网关是否能 ping 通目标 IP 地址。

ThingsLink 状态 - 系统 - 服务 - 网络 - 退出

**网络诊断**

网络工具

172.30.1.1

IPv4

```

PING 172.30.1.1 (172.30.1.1): 56 data bytes
64 bytes from 172.30.1.1: seq=0 ttl=64 time=90.370 ms
64 bytes from 172.30.1.1: seq=1 ttl=64 time=136.827 ms
64 bytes from 172.30.1.1: seq=2 ttl=64 time=89.841 ms
64 bytes from 172.30.1.1: seq=3 ttl=64 time=134.645 ms
64 bytes from 172.30.1.1: seq=4 ttl=64 time=137.538 ms

--- 172.30.1.1 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 89.841/117.844/137.538 ms
  
```

Powered by LuCI openwrt-19.07 branch (git-19.259.26128-651bc50) / OpenWrt 19.07-SNAPSHOT r10597-07465e3be

zjk.proxy.thingsroot.com:26845/cgi-bin/luci/admin/network/diagnostics