openvpn 的几种组网方式

服务器和客户端均为电脑主机，若要下挂私网，则主机需要有两块网卡，一块连外网，一块连私网。主机均安装 Ubuntu 操作系统。如果客户端/服务器本身就是路由器，可以省略 SNAT 的路由配置和路由转发功能步骤。路由模式-TUN 是 3 层路由模式，网桥模式-TAP 是 2 层连接模式。

1. 路由模式-TUN

1.1. 独立单客户端

1.1.1. 拓扑图

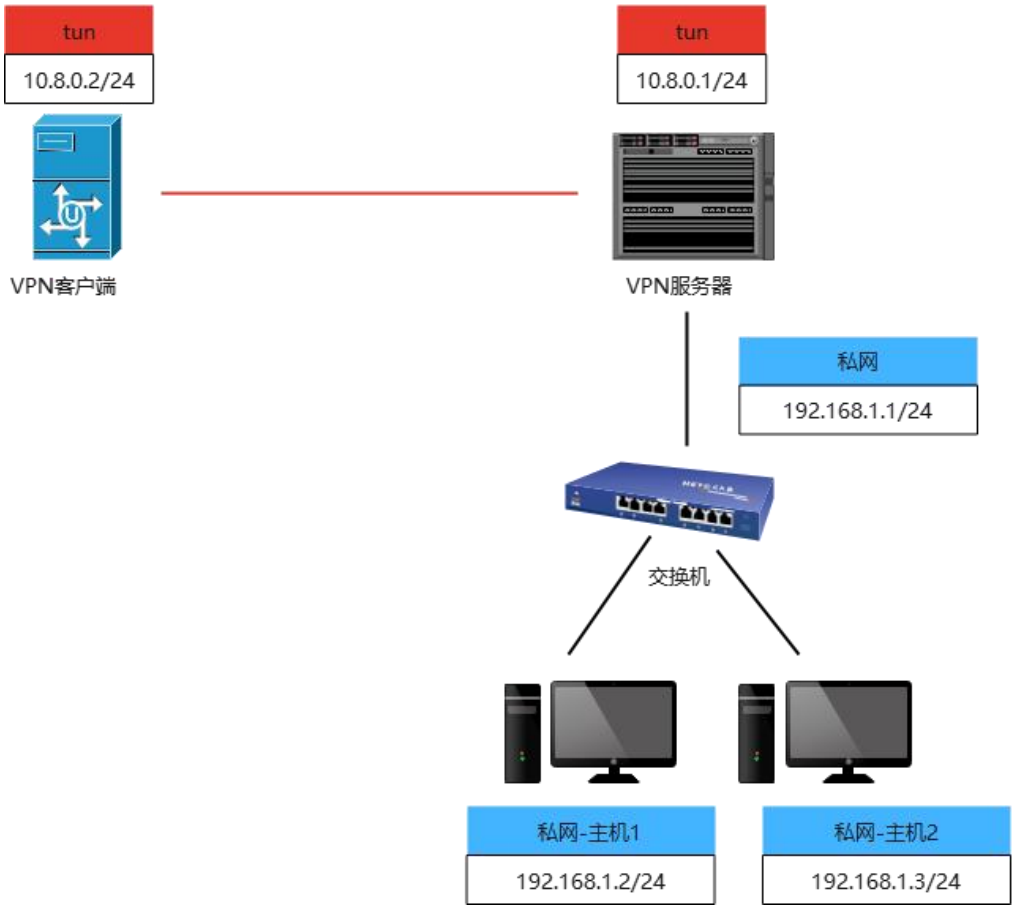

图 1 独立单客户端

1.1.2. 目标

（1）客户端需要远程访问服务器；

（2）客户端不能访问服务器后面的私网；

即客户端可以访问服务器 192.168.1.1，但是不能访问主机 1 和主机 2；

服务端的配置无需特殊配置，连接服务器后，可以直接连接成功。

1.1.3. 服务端路由推送

```
push "route 192.168.1.0 255.255.255.0"
```

1.2. 单客户端互联互通

1.2.1. 拓扑图

跟图 1 一致；

1.2.2. 目标

（1）客户端需要远程访问服务器；

（2）客户端能访问服务器后面的私网，即能访问主机 1 和主机 2；

1.2.3. 配置文件

配置在独立单客户端的基础上，完善服务端的 SANT 和路由转发功能，如下所述。

1.2.4. 服务端增加 SNAT 的路由

将来自 10.8.0.0/24 网段的数据包进行源地址封装伪装当前主机与内容的通信，使用 iptables 添加 SNAT 路由，如下图所示。

```
iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -j MASQUERADE
```

此命令的配置为一次性配置，设备重启后配置消失，如果想永久保存此配置，需要安装 iptables-persistent，每次修改完 iptables 以后用如下命令保存。

```
iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -j MASQUERADE
apt-get install iptables-persistent
netfilter-persistent save
```

1.2.5. 服务端开启路由转发功能

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

此命令的配置为一次性配置，设备重启后配置消失，如果想永久保存此配置，需要将 net.ipv4.ip_forward = 1 导入到内核里去，使系统永久生效开启路由转发功能，命令如下：

```
echo "net.ipv4.ip_forward = 1" >> /etc/sysctl.conf
```

1.3. 单客户端全联通

1.3.1. 拓扑图

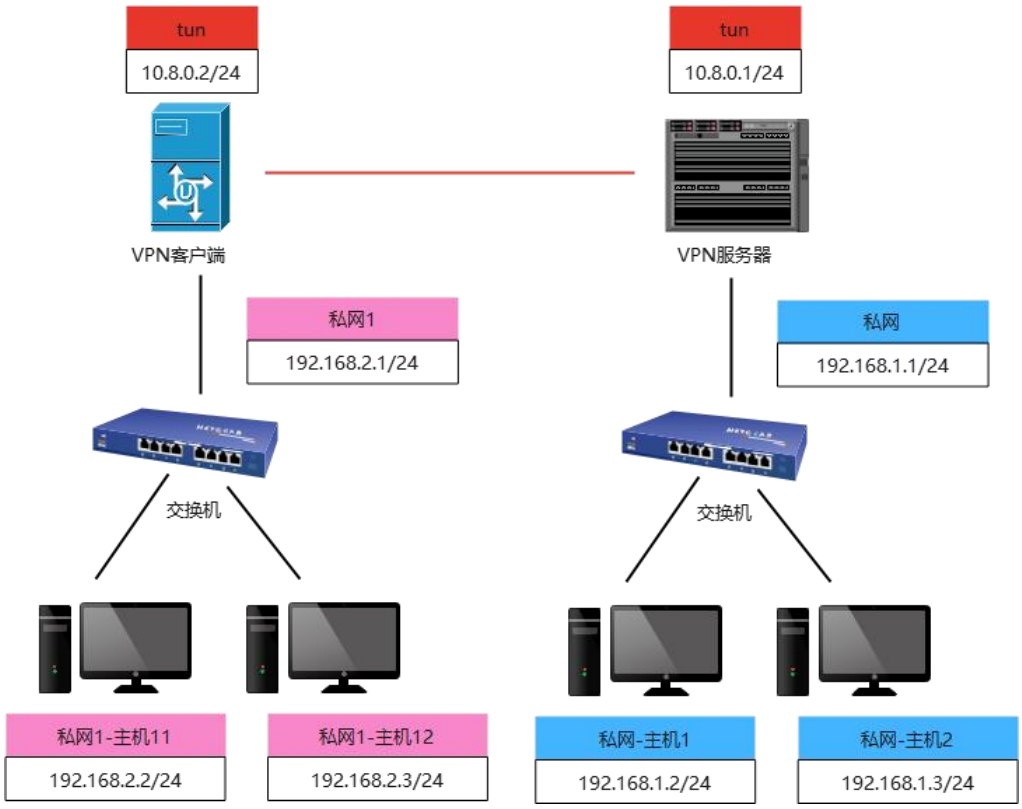

图 2 单客户端全联通

1.3.2. 目标

（1）客户端需要访问服务器及私网；

（2）服务器需要访问客户端及私网 1；

1.3.3. 配置文件

在单客户端互联互通配置的基础上，增加如下配置。

1.3.4. 服务端路由配置

```
route 192.168.2.0 255.255.255.0
```

1.3.5. 服务端 ccd 配置

ccd 文件夹中存放客户端的配置，比如客户端名称为 client1，对应 192.168.158.0/24 网段。则在 ccd 需要创建文件：client1，然后存放以下内容：

```
iroute 192.168.2.0 255.255.255.0
```

此配置有 2 个作用：

(1)为服务端添加路由表，在 VPN 服务端进行数据转发时，将 192.168.2.0 255.255.255.0 转发给 client1

(2)在进行路由推送时，忽略 192.168.2.0 255.255.255.0 对 client1 的推送。

1.3.6. 客户端 1 增加 SNAT 的路由

跟服务器添加方法一致，如 1.2.4。

1.3.7. 客户端 1 开启路由转发功能

跟服务器添加方法一致，如 1.2.5。

1.4. 双独立客户端

1.4.1. 拓扑图

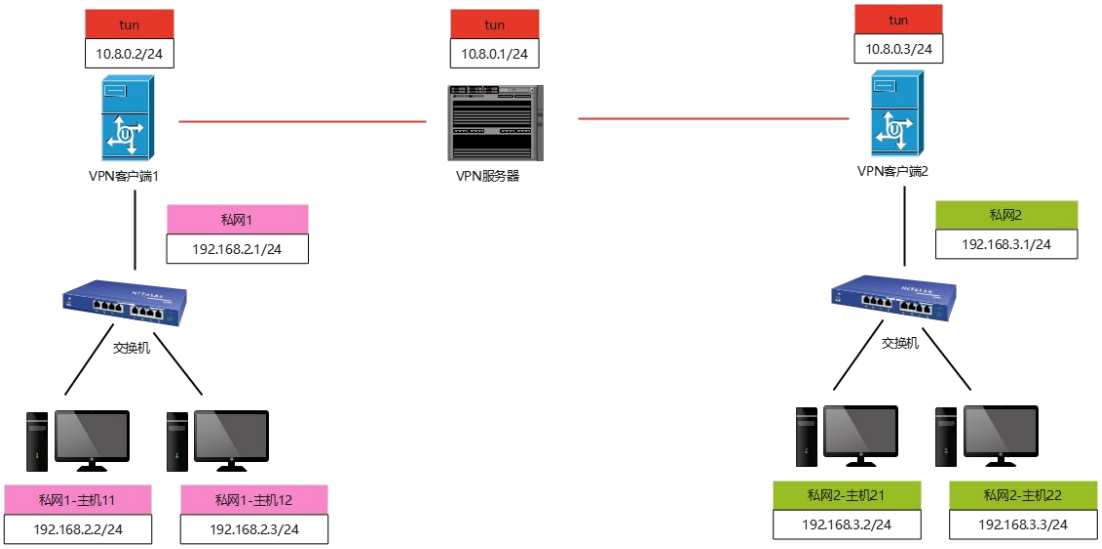

图 3 双独立客户端

1.4.2. 目标

（1）VPN 服务端能够与两个路由器均连通；

（2）VPN 服务端能够与主机 11，主机 12 连通；

（3）VPN 服务端能够与主机 21，主机 22 连通；

（4）主机 11 与主机 21，VPN 客户端 1 与 VPN 客户端 2，均不连通。

1.4.3. 服务端路由配置

```
route 192.168.2.0 255.255.255.0
route 192.168.3.0 255.255.255.0
```

1.4.4. 客户端 1 与客户端 2 增加 SNAT 的路由

跟服务器添加方法一致，如 1.2.4。

1.4.5. 客户端 1 与客户端 2 开启路由转发功能

跟服务器添加方法一致，如 1.2.5。

1.5. 双客户端互联互通

1.5.1. 拓扑图

跟双独立客户端一致，如图 3 所示。

1.5.2. 目标

（5）VPN 服务端能够与两个客户端均连通；

（6）VPN 服务端能够与主机 11，主机 12 连通；

（7）VPN 服务端能够与主机 21，主机 22 连通；

（8）主机 11 与主机 21，VPN 客户端 1 与 VPN 客户端 2，均连通。

1.5.3. 配置文件

在双独立客户端配置的基础上增加下述配置。

1.5.4. 开启 client-to-client

默认情况下 client-to-client 配置项并未启用，所以客户端间是不能够进行连接的，要开启该功能，需要先在配置文件中加入：

```
client-to-client
```

1.5.5. 服务端路由推送

两个网段互通的前提是：当向对向网段发起请求时，将数据包转发给 tun。所以

要在服务端配置下发给客户端的路由：

```
push "route 192.168.2.0 255.255.255.0"
push "route 192.168.3.0 255.255.255.0"
```

此时客户端在连接服务端后，则会在本地添加两条路由：

```
192.168.0.0/24 -> tun
192.168.1.0/24 -> tun
```

该路中的下发会引发客户端的路由地址冲突，因为客户端本身就存在 192.168.0.0/24 或 192.168.1.0/24 的路由段，该段的路由转发目的地为网关。会导致局域网访问不通的问题。

所以在配置推送（下发）的路由时，必须与 ccd 配置相结合。

1.5.6. 服务端 ccd 配置

ccd 文件夹中存放客户端的配置，比如两个客户端名称分别为 client1, client2，对应 2.0 以 3.0 网段。

则在 ccd 需要创建文件：client1，然后存放以下内容：

```
iroute 192.168.2.0 255.255.255.0
```

此配置有 2 个作用：

（1）为服务端添加路由表，在 VPN 服务端进行数据转发时，将 192.168.2.0 255.255.255.0 转发给 client1；

（2）所以此时 client1 对应的路由器将仅仅得到下发的路由：192.168.2.0 255.255.255.0

同时，还需要在 ccd 中创建 client2：

```
iroute 192.168.3.0 255.255.255.0
```

最后：

客户端 1 得到的路由(示意，实际路由条件会多，但效果相同）如下：

```
192.168.2.0/24 -> tun10.8.0.1 -> tun
```

客户端 2 得到的路由(示意，实际路由条件会多，但效果相同）如下：

```
192.168.3.0/24 -> tun10.8.0.1 -> tun
```

服务端路由：

```
192.168.2.0/24 -> client1
```

```
192.168.3.0/24 -> client2
```

1.6. 全联通

我们还需要连通服务端所在的局域网。

1.6.1. 拓扑图

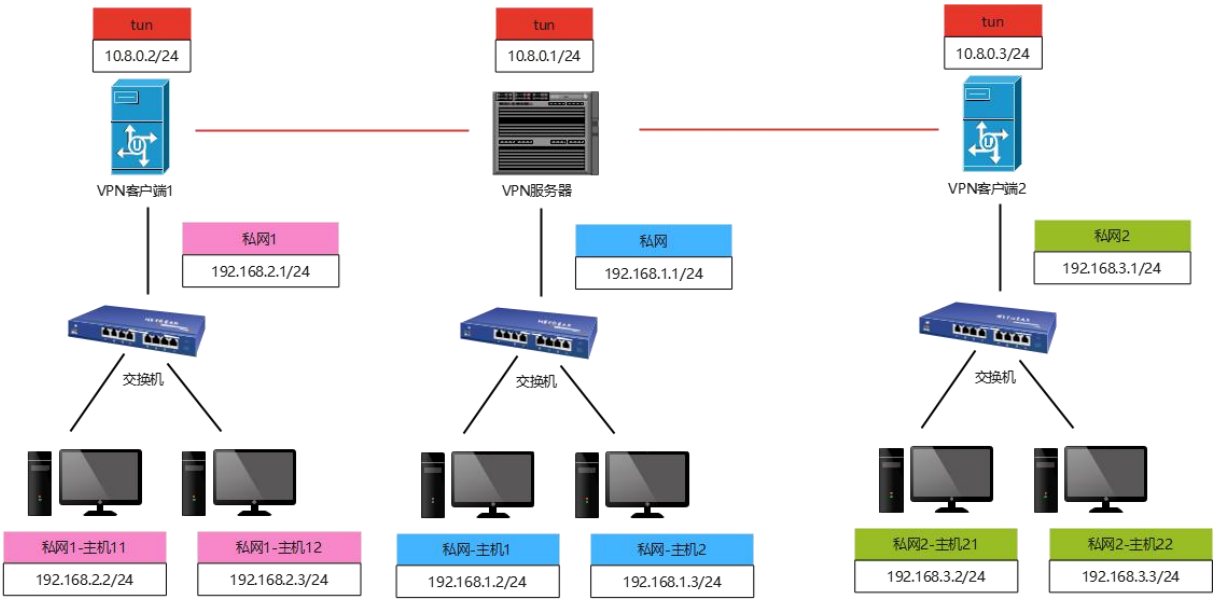

图 4 全联通

1.6.2. 目标

（1）VPN 服务端能够与两个客户端均连通；

（2）VPN 服务端能够与主机 11，主机 12 连通；

（3）VPN 服务端能够与主机 21，主机 22 连通；

（4）机 11 与主机 21，VPN 客户端 1 与 VPN 客户端 2，均连通

（5）主机 1，主机 11 与主机 21 均连通，即私网、私网 1 和私网 2 全联通。

1.6.3. 配置文件

在双客户端互联互通的基础上，完善服务器端的路由推送：

```
push "route 192.168.1.0 255.255.255.0"
push "route 192.168.2.0 255.255.255.0"
push "route 192.168.3.0 255.255.255.0"
```

2. 网桥模式-TAP

服务器和客户端均以网桥的模式连接在一起，组成一个超大的虚拟局域网，客户端和服务器的子网地址不能冲突；客户端若要扩展子网就增加一张网卡，并将网卡加入网桥；若不想暴露子网，则不将网卡加入网桥。服务器端设有网桥，连接在服务器后面内网上的子网主机，只要 IP 在同一个局域网内，可以自动组成局域网。客户端连上服务器以后，服务器会自动分配一个 IP 给客户端的 tap。

2.1. 独立客户端

2.1.1. 拓扑图

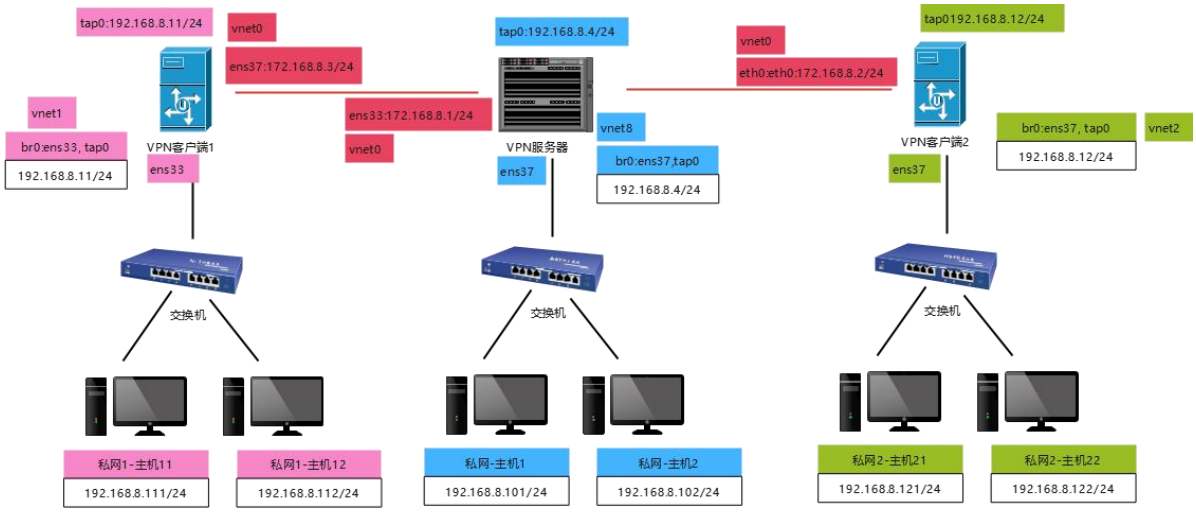

图 5 独立客户端

2.1.2. 目标

（1）VPN 服务器可以访问 VPN 客户端 1 和 VPN 客户端 2，以及下挂的子网；

（2）VPN 客户端 1 和 VPN 客户端 2 可以访问服务器以及子网；

（3）VPN 客户端 1 与 VPN 客户端 2 之间互不相能访问。

2.1.3. 创建或停止网桥

在服务端创建网桥，脚本位置一般在/usr/share/doc/openvpn-xxx/sample-scripts 目录下，xxx 是版本号，根据实际业务需求设置参数。

创建网桥的脚本，sample-scripts/bridge-start：

```
#!/bin/bash
```

```
#####################################
# Set up Ethernet bridge on Linux
# Requires: bridge-utils
#####################################

# Define Bridge Interface
br="br0"

# Define list of TAP interfaces to be bridged,
# for example tap="tap0 tap1 tap2".
tap="tap0"

# Define physical ethernet interface to be bridged
# with TAP interface(s) above.
eth="eth0"
eth_ip="192.168.8.4"
eth_netmask="255.255.255.0"
eth_broadcast="192.168.8.255"

for t in $tap; do
    openvpn --mktun --dev $t
done

brctl addbr $br
brctl addif $br $eth

for t in $tap; do
    brctl addif $br $t
done
```

```
for t in $tap; do
	ifconfig $t 0.0.0.0 promisc up
done

ifconfig $eth 0.0.0.0 promisc up

ifconfig $br $eth_ip netmask $eth_netmask broadcast $eth_broadcast
```

停止网桥的脚本 sample-scripts/bridge-start

```
#!/bin/bash

########################################
# Tear Down Ethernet bridge on Linux
########################################

# Define Bridge Interface
br="br0"

# Define list of TAP interfaces to be bridged together
tap="tap0"

ifconfig $br down
brctl delbr $br

for t in $tap; do
	openvpn --rmtun --dev $t
done
```

脚本执行顺序：

- run bridge-start

- run openvpn
- stop openvpn
- run bridge-stop

2.1.4. 修改服务器端配置

```
dev tap0
server-bridge 192.168.8.4 255.255.255.0 192.168.8.128 192.168.8.254
```

2.1.5. 修改客户端配置

```
dev tap
```

2.2. 全联通

2.2.1. 拓扑图

跟独立客户端一致，如图 5 所示。

2.2.2. 目标

（1）VPN 服务器可以访问 VPN 客户端 1 和 VPN 客户端 2，以及下挂的子网；

（2）VPN 客户端 1 和 VPN 客户端 2 可以访问服务器以及子网；

（3）VPN 客户端 1 与 VPN 客户端 2 之间也可以相互访问。

2.2.3. 配置文件

服务端配置在独立客户端的配置的基础上，增加如下配置。

2.2.4. 开启 client-to-client

默认情况下 client-to-client 配置项并未启用，所以客户端间是不能够进行连接的，要开启该功能，需要先在配置文件中加入：

```
client-to-client
```

2.3. 客户端互联互通

2.3.1. 拓扑图

跟独立客户端一致，如图 5 所示。

2.3.2. 目标

（1）VPN 服务器不能访问 VPN 客户端 1 和 VPN 客户端 2，以及下挂的子网；

（2）VPN 客户端 1 和 VPN 客户端 2 可以访问服务器以及子网；

（3）VPN 客户端 1 与 VPN 客户端 2 之间也可以相互访问。

2.3.3. 配置文件

有两种实现方案：

服务端配置在全联通配置的基础上，将服务器的配置文件改为

```
dev tap
```