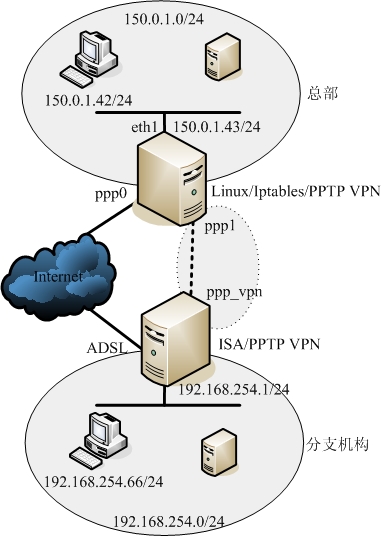
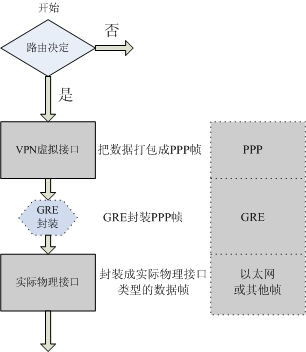
# 1．搭建VPN的目的

如下图1所示，上面部分是总部网络150.0.1.0/24，网络边界是一台RedHat 9.0，使用Netfilter/Iptables防火墙，准备在这台服务器上安装PPTP VPN服务，这样分支机构就可通过PPTP VPN与其实现站点到站点的VPN连接，这里的分支机构是下方的192.168.254.0/24网络。两方都是通过ADSL接入因特网的，我们现在要通过PPTP VPN把分支网络与总部网络连接起来，使它们就像位于同一个局域网一样，这样分支机构就可以通过VPN连接总部的服务器，反之亦然。



# 2．PPTP VPN的基本原理

PPTP VPN本质上是虚拟的点对点链路，它先把到达远方内网的数据包打包成PPP帧，然后再对这些PPP帧进行二次封装，以便于能够在其他物理链路上进行传送。 PPTP VPN有控制信道和数据信道之分，控制信道连接到VPN报务器的TCP1723端口，起着控制和管理VPN隧道的功能，数据信道是传送PPP帧的信道，关于PPTP数据帧的封装过程如图所示：  
  
在打包PPP帧的过程中，将对PPP数据包进行加密，为了取得最大的安全性，我们这里将使用MPPE加密和MSCHAPv2身份验证方法。下面我们来看看具体配置过程。

注：这里的PPTP只是一种协议，PPTP: Point to Point Tunneling Protocol -- [点到点隧道协议](http://baike.baidu.com/view/3871505.htm)。读者不要被他迷惑了。实现VPN有多种协议可以选择，本文讲述的是使用PPTP协议来实现VPN，即：PPTP VPN.

# ****3．安装和配置PPTP VPN 服务器****

## 1）下载必要的安装包

由于Linux本身并没有集成PPTP功能，所以需要安装几种程序以让我们的RedHat支持PPTP，根据内核的版本，下载相应的安装包。

1. 下载的PPTP服务器版本了，我这里选择的是pptpd-1.1.4-b4.i386.rpm，安装命令：  
   # rpm –ivh pptpd-1.1.4-b4.i386.rpm
2. PPTP需要PPP支持，虽然系统本身有PPP功能，但它并不一定支持MPPE，用以下命令检查PPP是否支持MPPE：

strings '/usr/sbin/pppd'|grep -i mppe|wc --lines

如果以上命令输出为“0”则表示不支持；输出为“30”或更大的数字就表示支持，如果不支持则需要更新系统的PPP组件，下载ppp-2.4.2\_cvs\_20030610-1.i386.rpm，安装命令如下：

# rpm -Uvh ppp-2.4.2\_cvs\_20030610-1.i386.rpm

1. 要使用MPPE加密，还需要内核支持，Linux 2.6.14起Linux核心提供完整的PPTP支援《包括自由版本的MPPE》。查看内核版本可以使用命令：uname –r，如果内核版本低于2.6.14则还要下载一个MPPE内核补丁：kernel-mppe-2.4.20-8.i686.rpm，安装命令如下：  
   # rpm –ivh kernel-mppe-2.4.20-8.i686.rpm  
   安装之后可以检查kernel MPPE是否安装成功，输入如下命令：  
   # depmod –a   
   # modprobe ppp-compress-18    
   如果安装成功，将会有下面提示信息：  
   Warning: loading /lib/modules/2.4.20-8/kernel/drivers/net/ppp\_mppe.o will taint the kernel: non-GPL license - BSD without advertisement clause  
   See http://www.sje.cn/s_UploadFile/UploadFile/2004112813143135.gifhttp://www.tux.org/lkml/#export-tainted for information about tainted modules  
   Module ppp\_mppe loaded, with warnings
2. 下载dkms-2.0.17.5-1.noarch.rpm，（个人理解这个工具不是必需的，只是可以优化某些东西），安装方法同上。

注：Oikawa等人在1996年提出一种与LKM类似的动态核心模块（DKMs）技术。与LKM一样，DKMs以文件的形式存储并能在系统运行过程中动态地加载和卸载。DKMs由一个用户层的DKM服务器来管理，并非由内核来管理。当核心需要某模块时，由DKM服务器负责把相应的DKM加载；当核心的内存资源紧缺时，由DKM服务器负责卸载一个没有被使用的DKM。缺点是所有的DKM是存储在本地系统上的，占用了大量宝贵的存储空间。

## 2）修改配置文件

1. **修改modules.conf文件**   
   　　编辑/etc/modules.conf配置文件，加入如下内容：   
   　　alias net-pf-47 ip\_gre
2. 修改：**/etc/pptpd.conf ：**PPTP服务PPTPD运行时使用的配置文件

debug--------------------------------------------把所有debug信息记入系统日志/var/log/messages

option /etc/ppp/options.pptpd------------PPP组件将使用的配置文件

localip 221.6.15.150--------------------------分配给VPN服务器的地址，可以任意起

remoteip 192.168.1.1-100------------------VPN服务器分配给客户端的IP段，可以任意起

1. 修改：**/etc/ppp/options.pptpd**

就是上面/etc/pptpd.conf中的option所指定的，它是PPP功能组件pppd将使用的配置文件，由于PPTP VPN的加密和验证都与PPP相关，所以PPTP的加密和验证选项都将在这个配置文件中进行配置。

auth---------------------------------启用身份验证，为了安全，肯定需要进行身份验证  
debug------------------------------  
name cc3-------------------------自行设定的VPN服务器的名字，可以任意  
refuse-pap------------------------拒绝pap身份验证  
refuse-chap----------------------拒绝chap身份验证  
refuse-mschap------------------拒绝mschap身份验证  
refuse-eap -----------------------拒绝eap身份验证，这种方式本身不错，但这里不使用  
require-mschap-v2------------为了最高的安全性，我们使用mschap-v2身份验证方法  
require-mppe-128-------------要求128位MPPE加密，还可以是require-mppe  
nomppe-stateful----------------无状态，有状态有mppe-stateful  
ms-dns 150.0.1.88------------VPN服务器的DNS，存放在/etc/resolv.conf中  
#ms-wins 150.0.1.88---------

proxyarp-------------------------启用ARP代理，如果分配给客户端的IP与内网卡同一个子网

logfile /var/log/pptpd.log-----存放pptpd服务运行的的日志

multilink--------------------------

dump------------------------------

lock--------------------------------

1. 修改：**/etc/ppp/chap-secrets**

在这里配置能够连接到VPN服务器的用户和密码信息

# Secrets for authentication using CHAP

# client server secret IP addresses

xyx \* 123456 \*

####### redhat-config-network will overwrite this part!!! (begin) ##########

####### redhat-config-network will overwrite this part!!! (end) ############

“xyx”是Client端的VPN用户名；“server”对应的是VPN服务器的名字，该名字必须和/etc/ppp/options.pptpd文件中指明的一样，或者设置成“\*”号来表示自动识别服务器；“secret”对应的是登录密码；“IP addresses”对应的是可以拨入的客户端IP地址，如果不需要做特别限制，可以将其设置为“\*”号

1. 启动服务

到此VPN的配置文件就全部配置完成了，此时pptpd服务还没有启动，必需启动后才能被客户端连接

相关命令如下：

启动： service pptpd start

停止： service pptpd stop

重新启动： service pptpd restart

pptpd服务使用的端口是1723，这个端口是系统固定分配的，可以通过查看该端口检查pptpd服务的运行情况，命令：netstat –ntpl

如果服务已经启动，会看到下面的内容

tcp 0 0 0.0.0.0:1723 0.0.0.0:\* LISTEN 24981/pptpd

如果希望pptpd服务在每次服务器重启后自动开启，可以做下面的设置

键入：ntsysv 出现如下内容，把pptpd服务选择后OK即可



# 4．设置VPN服务器的防火墙

通过第三步我们已经把VPN服务器部署好了，这一步的目的是制定一些约束规则，以便我们可以控制哪些请求可以被VPN服务器处理，哪些请求需要被拒绝。即：设定防火墙的参数。Linux下设定防火墙使用iptables命令，详细用法请参考



设置防火墙的net表，目的是为了实现数据包的伪装及转发，从内网发出的包通过VPN SERVER伪装后变成由外网发出。

iptables –t nat –F-------清除nat表中所有规则

防火墙设置数据包转发伪装

iptables -t nat -A POSTROUTING -s 192.168.0.0/24 -o eth1 -j SNAT –to-source (外网IP)---这句话的意思是：对即将发送出去的数据包进行修改，对来自设备eth1且源地址是192.168.0.0/24的数据包，把源地址修改为(外网IP)

只有设置了IP伪装转发，通过VPN连接上来的远程[计算机](http://product.enet.com.cn/price/plist3.shtml)才能互相ping通，实现像局域网那样的共享。用下面的命令进行设置：

echo 1> /proc/sys/net/ipv4/ip\_forward

将Linux服务器的1723端口和47端口打开，并打开GRE协议，以便允许 VPN连接

iptables –I INPUT –p tcp –dport 1723 –j ACCESS------让防火墙方通目的端口是1723的tcp数据包

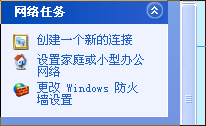
iptables –I INPUT –p tcp –dport 47 –j ACCESS------让防火墙方通目的端口是47的tcp数据包

iptables –I INPUT –p gre –j ACCESS--------------------让防火墙方通gre协议的数据包

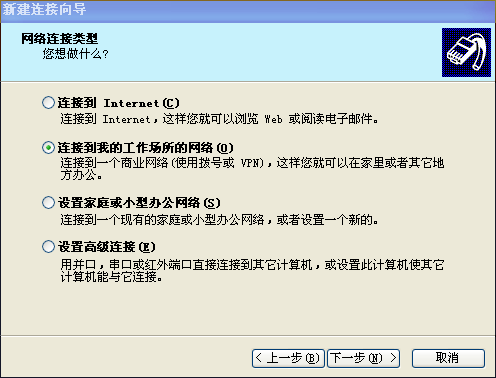
# 5. 设置客户端

客户端的设置很简单，这里就以XP为例简单说明一下。

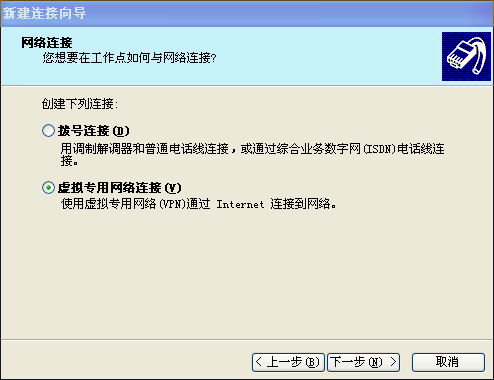
1. 在“网上邻居”点右键“属性”



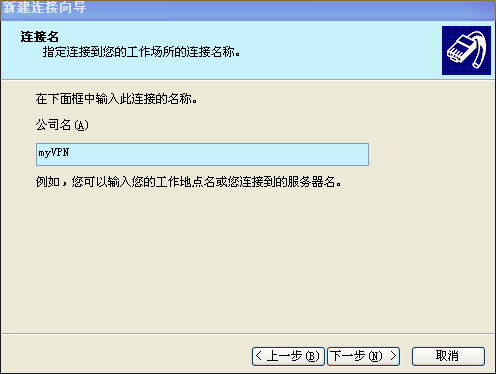
1. 选择“创建一个新的连接”，“下一步”后选择“连接到我的工作场所的网络”



1. “下一步”选择“虚拟专用网络连接”

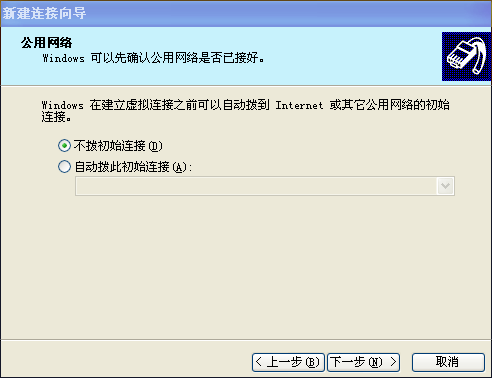


1. “下一步”输入你希望的本链接的名字，可以任意

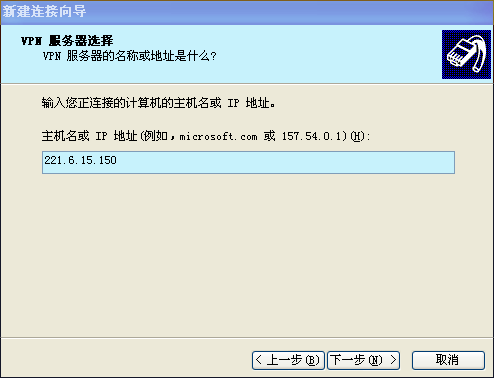


1. “下一步”后，如果你已经有一个VPN得连接了，会出现下面的提示框，选择“不拨初始连接”。

如果你没有建立过VPN连接则不会有该界面，直接跳到第六步



1. 在下面的界面里面输入VPN服务器的地址

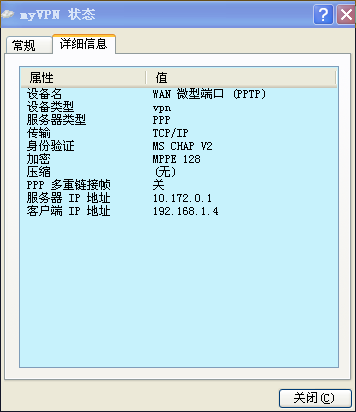


1. 点“下一步”、“完成”后会出现连接提示，输入我们在/etc/ppp/chap-secrets中设置的用户名和密码后点“连接”即向VPN服务器发起了登陆请求





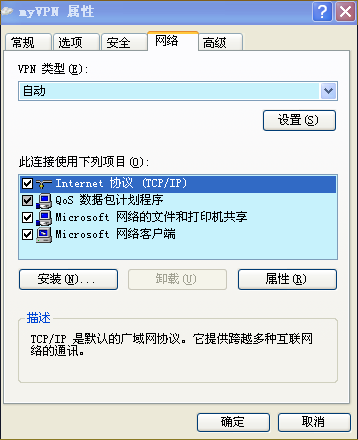
1. 连接成功后，点开右下角的本地连接，在详细信息里面可以看到

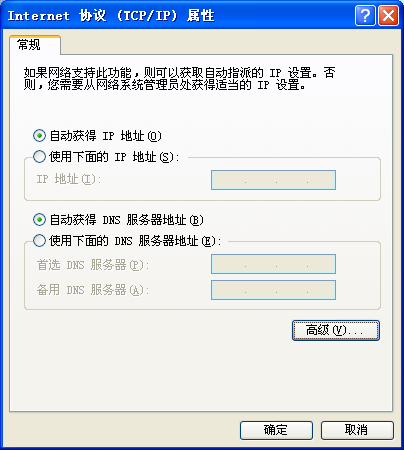


# 6． FAQ

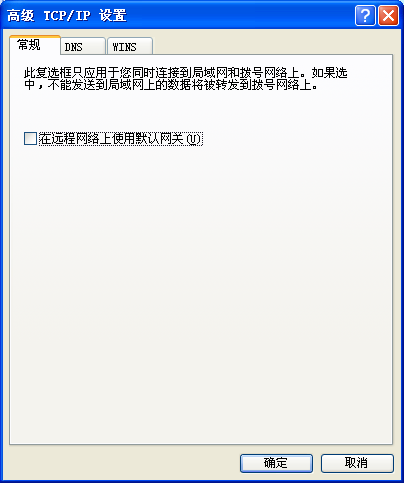
1. VPN登陆成功了，但是不能访问外网

尝试做如下修改：在建立的VPN链接上右键属性，选择“网络”并选择“TCP/IP协议”，点属性



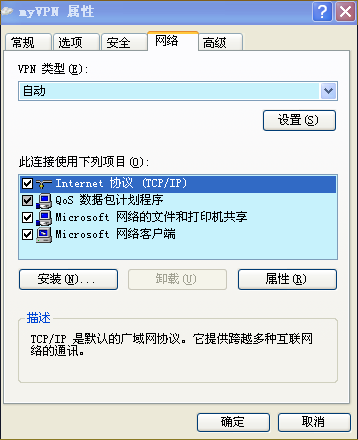


把下图的“在远程网络上使用默认网关”勾选去掉



1. 登陆VPN的时候报“IPX/SPX不兼容”（具体忘记了，没有截图）

尝试问题1相同的步骤到出现下面的界面



如果已经安装的协议里面有：



请把它卸载掉，我是这样解决的，具体为什么不清楚。