Project 2-ipesec conf 报告

2151140 王谦 信息安全

一、实验目标

> 基于 linux 搭建一个基本 ipsec-vpn 原型
>
> 1.选择一个开源程序库，总结一下 linux 下 ipsec 网关程序模块构架及其相互关系
>
> （1）总结配置 ike 的方法与参数
>
> （2）总结 l2tp-ipsec 的配置方法与参数
>
> （3）总结 ipsec 模式选择及参数配置方法
>
> 2.搭建一个 gate2gate ipsec vpn 网关对（gate 后 lan 为私有网络）

二、目标分析

总结 linux 下 ipsec 网关程序模块构架及其相互关系

总结配置 ike 的方法与参数

总结 l2tp-ipsec 的配置方法与参数

总结 ipsec 模式选择及参数配置方法

搭建一个 gate2gate ipsec vpn 网关对（gate 后 lan 为私 有网络）

三、过程展示

参照gihub上学长的相关内容，逐步学习执行。

使用了4 个阿里云 ECS 实例，CentOS 7.9， Libreswan 开源程序

实例

编号	公有IP	私有IP	性质
LG	120.26.6.168	172.24.65.94	网关
LN	-	172.24.65.95	内网主机
RG	8.149.133.151	172.24.65.1	网关
RN	-	172.24.65.92	内网主机

1.搭建 Gate to Gate 网关

将主机 RG 配置成网关服务器

```
# 开启转发
echo "net.ipv4.ip_forward = 1" >> /etc/sysctl.conf
# 生效
sysctl -p
```

```
# 配置 iptables 做 SNAT，指定具体的网段
iptables -t nat -I POSTROUTING -s 172.24.65.0/24 -j SNAT --to-source 172.24.65.1
# 查看 iptables
iptables -L -t na
```

```
[root@iZbp1fneblnwh49ypmlustZ ~]# # 开启转发
[root@iZbp1fneblnwh49ypmlustZ ~]# echo "net.ipv4.ip_forward = 1" >> /etc/sysctl.conf
[root@iZbp1fneblnwh49ypmlustZ ~]# # 生效
[root@iZbp1fneblnwh49ypmlustZ ~]# sysctl -p
vm.swappiness = 0
kernel.sysrq = 1
net.ipv4.neigh.default.gc_stale_time = 120
net.ipv4.conf.all.rp_filter = 0
net.ipv4.conf.default.rp_filter = 0
net.ipv4.conf.default.arp_announce = 2
net.ipv4.conf.lo.arp_announce = 2
net.ipv4.conf.all.arp_announce = 2
net.ipv4.tcp_max_tw_buckets = 5000
net.ipv4.tcp_syncookies = 1
net.ipv4.tcp_max_syn_backlog = 1024
net.ipv4.tcp_synack_retries = 2
net.ipv4.tcp_slow_start_after_idle = 0
net.ipv4.ip_forward = 1
[root@iZbp1fneblnwh49ypmlustZ ~]# # 配置 iptables 做 SNAT，指定具体的网段
[root@iZbp1fneblnwh49ypmlustZ ~]# iptables -t nat -I POSTROUTING -s 172.24.65.0/24 -j SNAT --to-source 172.24.65.1
[root@iZbp1fneblnwh49ypmlustZ ~]# # 查看 iptables
[root@iZbp1fneblnwh49ypmlustZ ~]# iptables -L -t nat
Chain PREROUTING (policy ACCEPT)
target     prot opt source               destination

Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain POSTROUTING (policy ACCEPT)
target     prot opt source               destination
SNAT       all  --  172.24.65.0/24       anywhere             to:172.24.65.1
[root@iZbp1fneblnwh49ypmlustZ ~]#
```

配置主机 RN

```
vim /etc/cloud/cloud.cfg
```

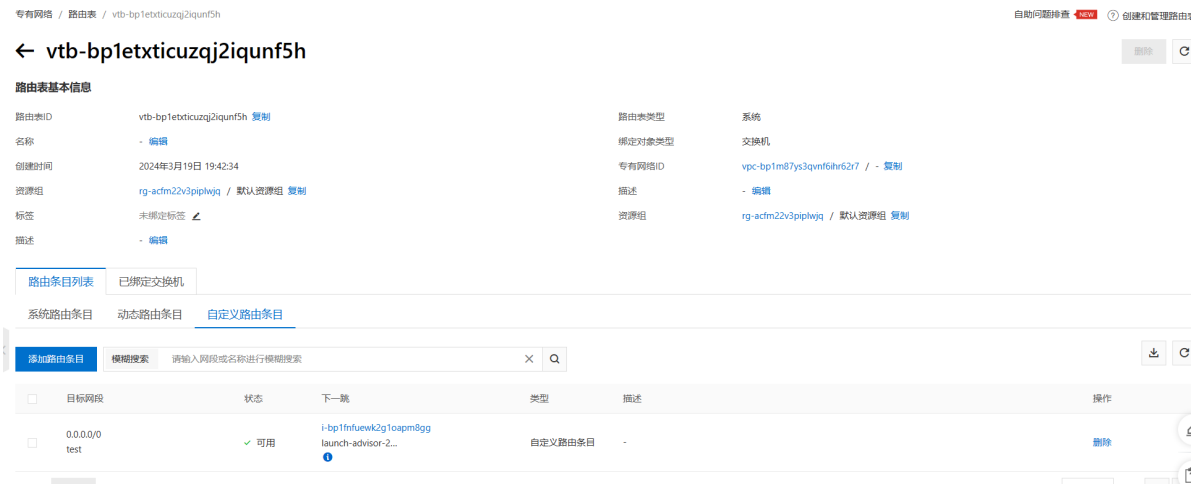
```
# This will cause the set+update hostname module to not operate (if true)
preserve_hostname: false

manage_etc_hosts: localhost

network:
  config: disabled

datasource_list: [ AliYun ]
```

打开 ifcfg-eth0，并编辑该文件

```
vim /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
IPADDR=172.24.65.92
NETMASK=255.255.255.0
GATEWAY=172.24.65.1
BROADCAST=172.24.65.255
```

添加路由表信息



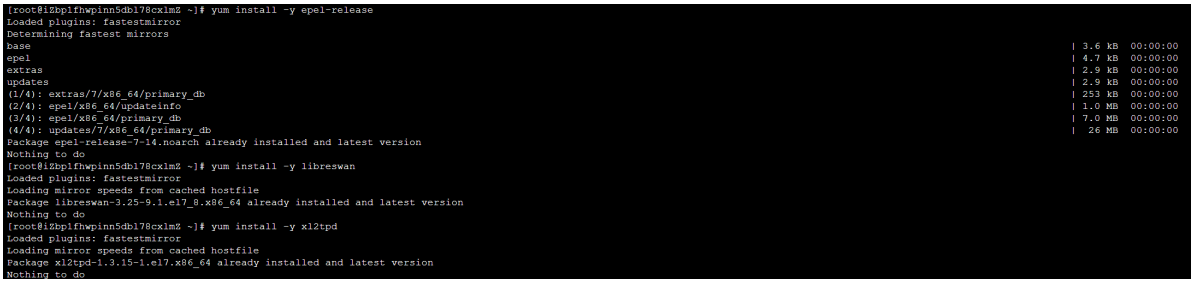

```
[root@iZbp1fneb1nwh49ypmlustZ ~]# systemctl start ipsec
[root@iZbp1fneb1nwh49ypmlustZ ~]# systemctl status ipsec
● ipsec.service - Internet Key Exchange (IKE) Protocol Daemon for IPsec
   Loaded: loaded (/usr/lib/systemd/system/ipsec.service; disabled; vendor preset: disabled)
   Active: active (running) since Fri 2024-04-05 17:07:00 CST; 2s ago
     Docs: man:ipsec(8)
           man:pluto(8)
           man:ipsec.conf(5)
  Process: 2088 ExecStartPre=/usr/sbin/ipsec --checknflog (code=exited, status=0/SUCCESS)
  Process: 2080 ExecStartPre=/usr/sbin/ipsec --checknss (code=exited, status=0/SUCCESS)
  Process: 1658 ExecStartPre=/usr/libexec/ipsec/_stackmanager start (code=exited, status=0/SUCCESS)
  Process: 1656 ExecStartPre=/usr/libexec/ipsec/addconn --config /etc/ipsec.conf --checkconfig (code=exited, status=0/SUCCESS)
 Main PID: 2102 (pluto)
   Status: "Startup completed."
   CGroup: /system.slice/ipsec.service
           └─2102 /usr/libexec/ipsec/pluto --leak-detective --config /etc/ipsec.conf --nofork

Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: adding interface lo/lo 127.0.0.1:500
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: adding interface lo/lo 127.0.0.1:4500
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: adding interface lo/lo ::1:500
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: | setup callback for interface lo:500 fd 19
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: | setup callback for interface lo:4500 fd 18
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: | setup callback for interface lo:500 fd 17
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: | setup callback for interface eth0:4500 fd 16
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: | setup callback for interface eth0:500 fd 15
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: loading secrets from "/etc/ipsec.secrets"
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: no secrets filename matched "/etc/ipsec.d/*.secrets"
[root@iZbp1fneb1nwh49ypmlustZ ~]#
```

2.在网关（LG、RG）上配置 IPsec

安装 epel-release、libreswan、xl2tpd，查看 IPsec 配置相关内容

```
yum install -y epel-release
yum install -y libreswan
yum install y xl2tpd
```

```
rpm -ql libreswan|grep -E -v "share|libe
```



修改/etc/sysctl.conf 文件

开启路由转发（ip_forward）、关闭源路由验证（rp_filter）、关闭 ICMP 重定向 （send_redirects、accept_redirects）。

```
vim /etc/sysctl.conf
```

```
# ipsec
## 开启路由转发
net.ipv4.ip_forward = 1
## 关闭源路由验证、关闭 ICMP 重定向
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.all.rp_filter = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv4.conf.default.rp_filter = 0
net.ipv4.conf.default.send_redirects = 0
net.ipv4.conf.lo.accept_redirects = 0
net.ipv4.conf.lo.rp_filter = 0
net.ipv4.conf.lo.send_redirects = 0
~
```

```
[root@iZbp1fneb1nwh49ypmlustZ ~]# sysctl -p
vm.swappiness = 0
kernel.sysrq = 1
net.ipv4.neigh.default.gc_stale_time = 120
net.ipv4.conf.all.rp_filter = 0
net.ipv4.conf.default.rp_filter = 0
net.ipv4.conf.default.arp_announce = 2
net.ipv4.conf.lo.arp_announce = 2
net.ipv4.conf.all.arp_announce = 2
net.ipv4.tcp_max_tw_buckets = 5000
net.ipv4.tcp_syncookies = 1
net.ipv4.tcp_max_syn_backlog = 1024
net.ipv4.tcp_synack_retries = 2
net.ipv4.tcp_slow_start_after_idle = 0
net.ipv4.ip_forward = 1
net.ipv4.ip_forward = 1
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.all.rp_filter = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv4.conf.default.rp_filter = 0
net.ipv4.conf.default.send_redirects = 0
net.ipv4.conf.lo.accept_redirects = 0
net.ipv4.conf.lo.rp_filter = 0
net.ipv4.conf.lo.send_redirects = 0
```

启动 ipsec 服务，验证内核配置

```
systemctl start ipsec
systemctl status ipsec
```

```
[root@iZbp1fneb1nwh49ypmlustZ ~]# systemctl start ipsec
[root@iZbp1fneb1nwh49ypmlustZ ~]# systemctl status ipsec
● ipsec.service - Internet Key Exchange (IKE) Protocol Daemon for IPsec
   Loaded: loaded (/usr/lib/systemd/system/ipsec.service; disabled; vendor preset: disabled)
   Active: active (running) since Fri 2024-04-05 17:07:00 CST; 2s ago
     Docs: man:ipsec(8)
           man:pluto(8)
           man:ipsec.conf(5)
  Process: 2088 ExecStartPre=/usr/sbin/ipsec --checknflog (code=exited, status=0/SUCCESS)
  Process: 2080 ExecStartPre=/usr/sbin/ipsec --checknss (code=exited, status=0/SUCCESS)
  Process: 1658 ExecStartPre=/usr/libexec/ipsec/_stackmanager start (code=exited, status=0/SUCCESS)
  Process: 1656 ExecStartPre=/usr/libexec/ipsec/addconn --config /etc/ipsec.conf --checkconfig (code=exited, status=0/SUCCESS)
 Main PID: 2102 (pluto)
   Status: "Startup completed."
   CGroup: /system.slice/ipsec.service
           └─2102 /usr/libexec/ipsec/pluto --leak-detective --config /etc/ipsec.conf --nofork

Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: adding interface lo/lo 127.0.0.1:500
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: adding interface lo/lo 127.0.0.1:4500
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: adding interface lo/lo ::1:500
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: | setup callback for interface lo:500 fd 19
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: | setup callback for interface lo:4500 fd 18
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: | setup callback for interface lo:500 fd 17
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: | setup callback for interface eth0:4500 fd 16
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: | setup callback for interface eth0:500 fd 15
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: loading secrets from "/etc/ipsec.secrets"
Apr 05 17:07:00 iZbp1fneb1nwh49ypmlustZ pluto[2102]: no secrets filename matched "/etc/ipsec.d/*.secrets"
[root@iZbp1fneb1nwh49ypmlustZ ~]#
```

通过 ipsec verify 查看 ipsec 状态

```
ipsec verify
```

```
[root@iZbp1fneb1nwh49ypmlustZ ~]# ipsec verify
Verifying installed system and configuration files

Version check and ipsec on-path                         [OK]
Libreswan 3.25 (netkey) on 3.10.0-1160.108.1.el7.x86_64
Checking for IPsec support in kernel                    [OK]
 NETKEY: Testing XFRM related proc values
         ICMP default/send_redirects                    [OK]
         ICMP default/accept_redirects                  [OK]
         XFRM larval drop                               [OK]
Pluto ipsec.conf syntax                                 [OK]
Two or more interfaces found, checking IP forwarding    [OK]
Checking rp_filter                                      [OK]
Checking that pluto is running                          [OK]
 Pluto listening for IKE on udp 500                     [OK]
 Pluto listening for IKE/NAT-T on udp 4500              [OK]
 Pluto ipsec.secret syntax                              [OK]
Checking 'ip' command                                   [OK]
Checking 'iptables' command                             [OK]
Checking 'prelink' command does not interfere with FIPS [OK]
Checking for obsolete ipsec.conf options                [OBSOLETE KEYWORD]
warning: could not open include filename: '/etc/ipsec.d/*.conf'
[root@iZbp1fneb1nwh49ypmlustZ ~]#
```

查看端口开放状态，IPsec 需要开放的端口：500、4500、50、51，发现 4500、500 已经开放

```
[root@iZbp1fneb1nwh49ypmlustZ ~]# netstat -unlp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
udp        0      0 0.0.0.0:68              0.0.0.0:*                           802/dhclient
udp        0      0 0.0.0.0:111             0.0.0.0:*                           552/rpcbind
udp        0      0 127.0.0.1:323           0.0.0.0:*                           555/chronyd
udp        0      0 127.0.0.1:4500          0.0.0.0:*                           2102/pluto
udp        0      0 172.24.65.1:4500        0.0.0.0:*                           2102/pluto
udp        0      0 127.0.0.1:500           0.0.0.0:*                           2102/pluto
udp        0      0 172.24.65.1:500         0.0.0.0:*                           2102/pluto
udp        0      0 0.0.0.0:714             0.0.0.0:*                           552/rpcbind
udp6       0      0 :::111                  :::*                                552/rpcbind
udp6       0      0 ::1:323                 :::*                                555/chronyd
udp6       0      0 ::1:500                 :::*                                2102/pluto
udp6       0      0 :::714                  :::*                                552/rpcbind
```

关闭防火墙

```
systemctl disable firewalld
```

安装 nmap，LG、RG 互相扫描对方的公网 ip，确认并验证 500、4500 端口的开放状态

```
yum install nmap
```

```
nmap -sU 120.26.6.168 -p 500,4500 -Pn
```

```
[root@iZbp1fneb1nwh49ypmlustZ ~]# nmap -sU 120.26.6.168 -p 500,4500 -Pn

Starting Nmap 6.40 ( http://nmap.org ) at 2024-04-05 17:10 CST
Nmap scan report for 120.26.6.168
Host is up.
PORT     STATE         SERVICE
500/udp  open|filtered isakmp
4500/udp open|filtered nat-t-ike

Nmap done: 1 IP address (1 host up) scanned in 3.05 seconds
```

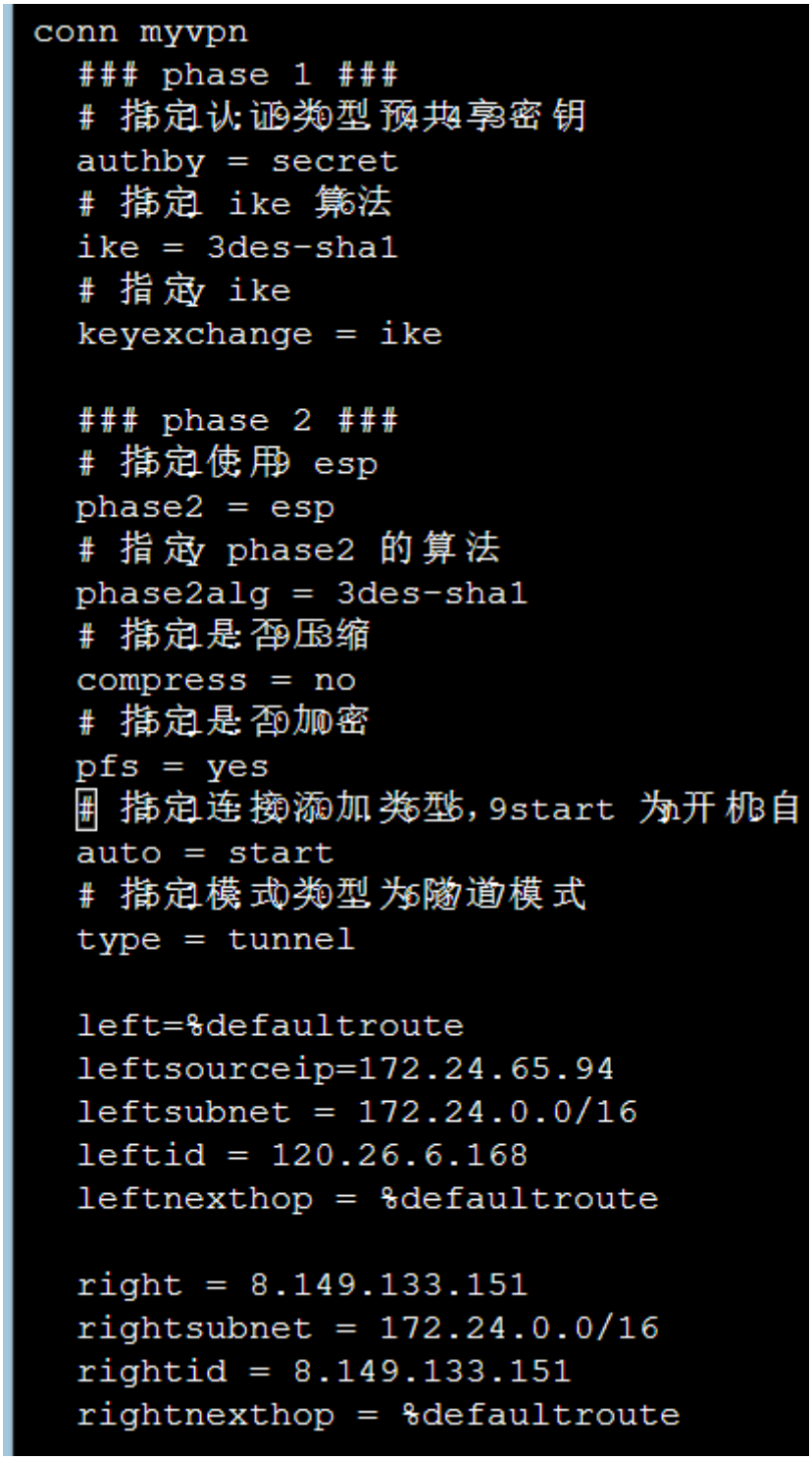
创建/etc/ipsec.d/myvpn.conf 文件，配置 ipsec。

```
conn myvpn
  ### phase 1 ###
  # 指定认证类型预共享密钥
  authby = secret
  # 指定 ike 算法
  ike = 3des-sha1
  # 指定 ike
  keyexchange = ike

  ### phase 2 ###
  # 指定使用 esp
  phase2 = esp

  # 指定 phase2 的算法
  phase2alg = 3des-sha1
  # 指定是否压缩
  compress = no
  # 指定是否加密
  pfs = yes
  # 指定连接添加类型，start 为开机自启动
  auto = start
  # 指定模式类型为隧道模式
  type = tunnel
  left=%defaultroute

  leftsourceip=172.24.65.1
  leftsubnet = 172.24.0.0/16
  leftid = 8.149.133.151
  leftnexthop = %defaultroute

  right = 120.26.6.168
  rightsubnet = 172.24.0.0/16
  rightid = 120.26.6.168
  rightnexthop = %defaultroute
```

创建/etc/ipsec.d/myvpn.conf 配置 ipsec

```
vim /etc/ipsec.d/myvpn.conf
```

```
conn myvpn
  ### phase 1 ###
  # 指定认证类型预共享密钥
  authby = secret
  # 指定 ike 算法
  ike = 3des-sha1
  # 指定 ike
  keyexchange = ike

  ### phase 2 ###
  # 指定使用 esp
  phase2 = esp

  # 指定 phase2 的算法
  phase2alg = 3des-sha1
  # 指定是否压缩
  compress = no
  # 指定是否加密
  pfs = yes
  # 指定连接添加类型，start 为开机自启动
  auto = start
  # 指定模式类型为隧道模式
  type = tunnel
  left=%defaultroute

  leftsourceip=172.24.65.1
  leftsubnet = 172.24.0.0/16
  leftid = 8.149.133.151
  leftnexthop = %defaultroute

  right = 120.26.6.168
  rightsubnet = 172.24.0.0/16
  rightid = 120.26.6.168
  rightnexthop = %defaultroute
```




```
[root@iZbp1fneb1nwh49ypmlustZ ~]# vim /etc/ipsec.d/myvpn.secrets
[root@iZbp1fneb1nwh49ypmlustZ ~]# cat /etc/ipsec.d/myvpn.secrets
PSK "2151140"
```

在 LG、RG 创建文件/etc/ipsec.d/myvpn.secrets，存放预共享密钥

```
[root@iZbp1fneb1nwh49ypmlustZ ~]# vim /etc/ipsec.d/myvpn.secrets
[root@iZbp1fneb1nwh49ypmlustZ ~]# cat /etc/ipsec.d/myvpn.secrets
PSK "2151140"
```

后续发现异常，改成：

```
0.0.0.0 0.0.0.0 : PSK "2151140"
```

```
0.0.0.0 0.0.0.0 : PSK "2151140"
~
~
```

```
vim /etc/ipsec.conf
```

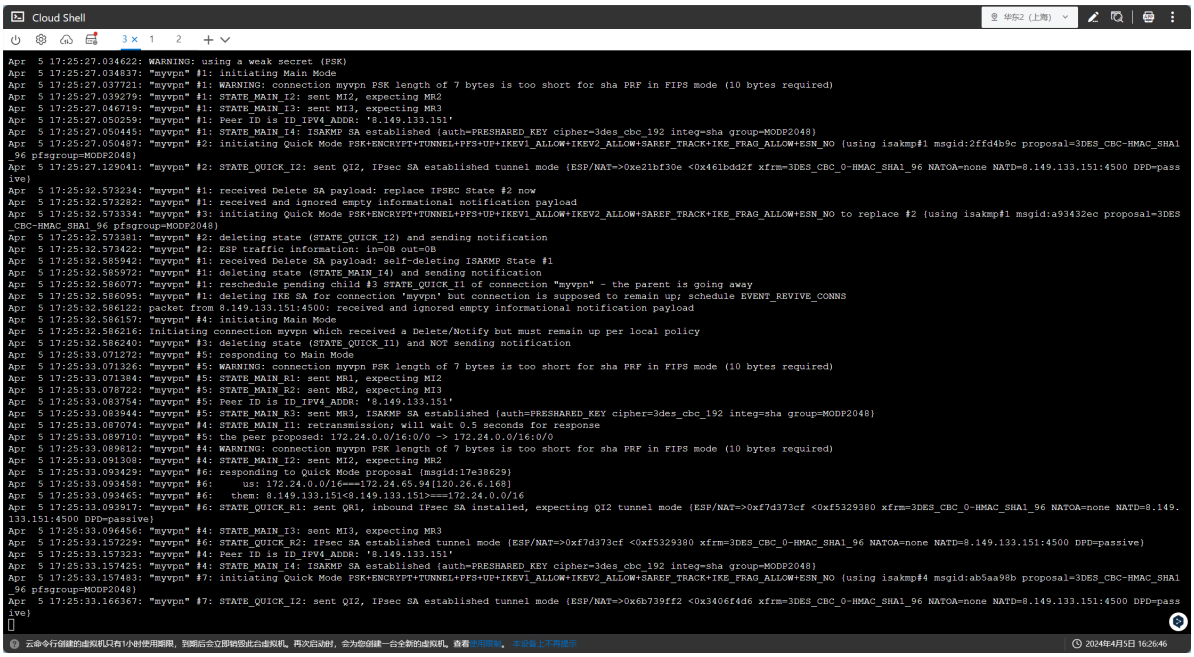
取消注释 logfile=/var/log/pluto.log

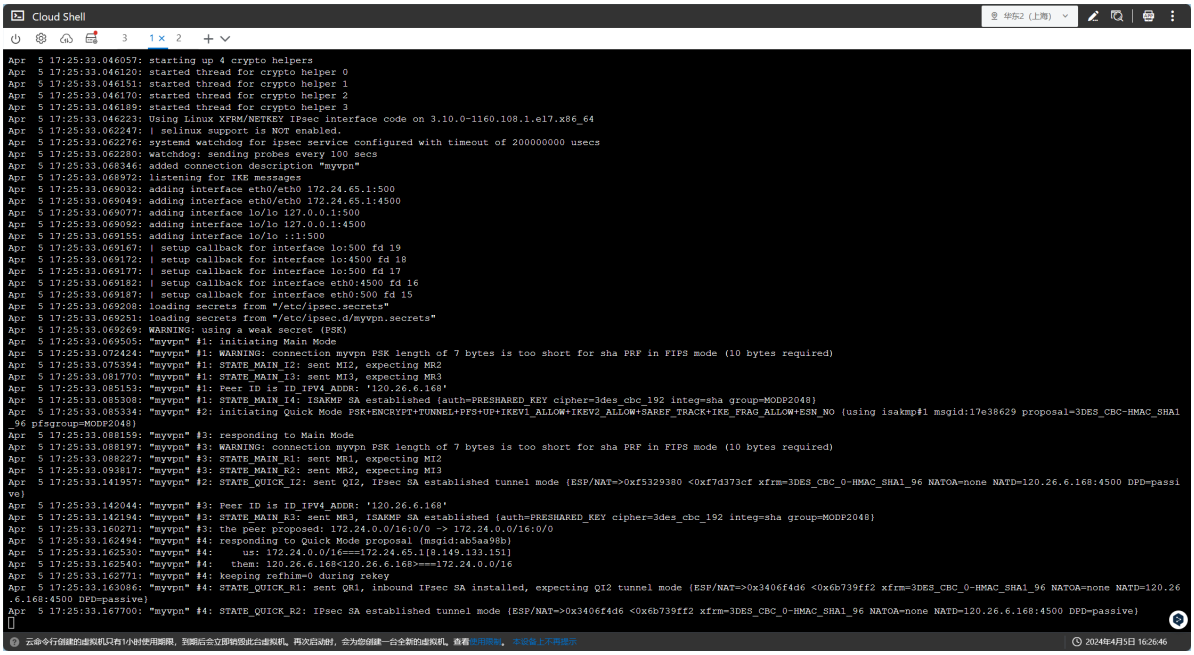
```
# /etc/ipsec.conf - Libreswan IPsec configuration file
#
# see 'man ipsec.conf' and 'man pluto' for more information
#
# For example configurations and documentation, see https://libreswan.org/wiki/

config setup
        # Normally, pluto logs via syslog.
        logfile=/var/log/pluto.log
        #
        # Do not enable debug options to debug configuration issues!
        #
        # plutodebug="control parsing"
        # plutodebug="all crypt"
        plutodebug=none
        #
        # NAT-TRAVERSAL support
        # exclude networks used on server side by adding %v4:!a.b.c.0/24
        # It seems that T-Mobile in the US and Rogers/Fido in Canada are
        # using 25/8 as "private" address space on their wireless networks.
        # This range has never been announced via BGP (at least up to 2015)
        virtual_private=%v4:10.0.0.0/8,%v4:192.168.0.0/16,%v4:172.16.0.0/12,%v4:25.0.0.0/8,%v4:100.64.0.0/10,%v6:fd00::/8
,%v6:fe80::/10

# if it exists, include system wide crypto-policy defaults
# include /etc/crypto-policies/back-ends/libreswan.config

# It is best to add your IPsec connections as separate files in /etc/ipsec.d/
include /etc/ipsec.d/*.conf
~
~
~
~
```

3.连接与测试

在 LG、RG 重启网络服务，建立 IPsec vpn，同时打印 vpn 连接过程。 IPsec SA established tunnel mode # 看到日志为建立隧道成功

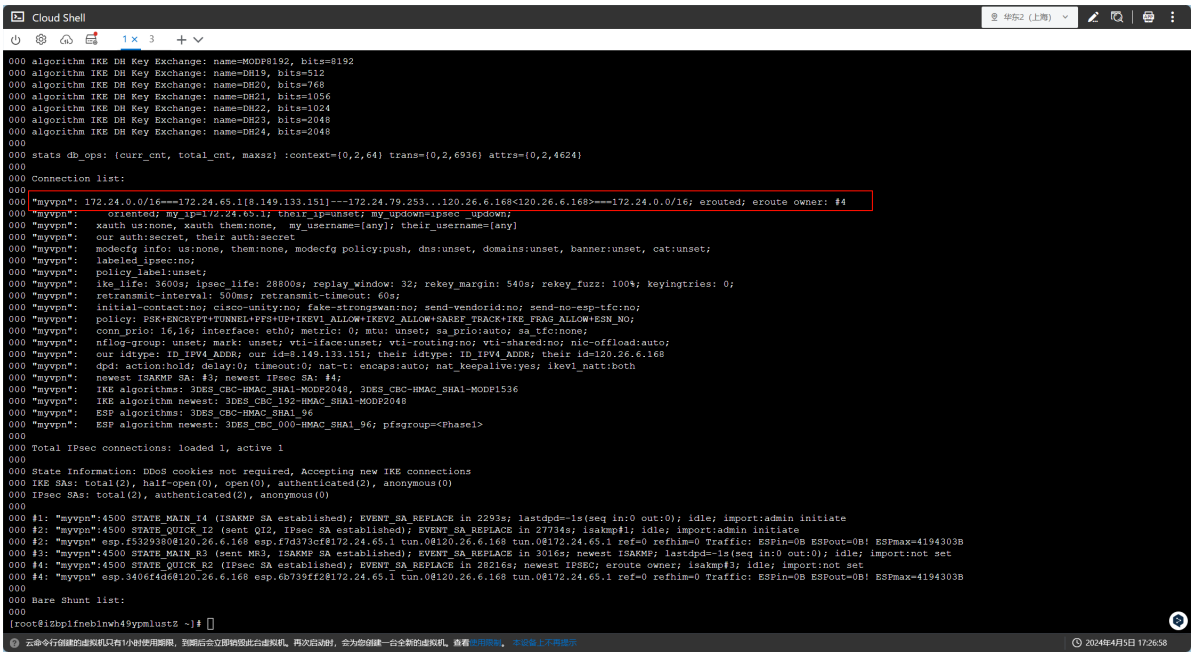
```
systemctl restart ipsec && tailf /var/log/pluto.log
```





查看 IPsec 连接状态

```
ipsec auto --status
```




```
"myvpn": 172.24.0.0/16===172.24.65.94[120.26.6.168]---172.24.79.253...8.149.133.151<8.149.133.151>===172.24.0.0/16; erouted; eroute owner: #7
```

```
000 Total IPsec connections: loaded 1, active 1
000
000 State Information: DDoS cookies not required, Accepting new IKE connections
000 IKE SAs: total(2), half-open(0), open(0), authenticated(2), anonymous(0)
000 IPsec SAs: total(2), authenticated(2), anonymous(0)
000
000 #4: "myvpn":4500 STATE_MAIN_I4 (ISAKMP SA established); EVENT_SA_REPLACE in 2243s; newest ISAKMP; lastdpd=-1s(seq in:0 out:0); idle; import:admin initiate
000 #5: "myvpn":4500 STATE_MAIN_R3 (sent MR3, ISAKMP SA established); EVENT_SA_REPLACE in 2935s; lastdpd=-1s(seq in:0 out:0); idle; import:not set
000 #6: "myvpn":4500 STATE_QUICK_R2 (IPsec SA established); EVENT_SA_REPLACE in 28135s; isakmp#5; idle; import:not set
000 #6: "myvpn" esp.f7d373cf@8.149.133.151 esp.f5329380@172.24.65.94 tun.0@8.149.133.151 tun.0@172.24.65.94 ref=0 refhim=0 Traffic: ESPin=0B ESPout=0B! ESPmax=4194303B
000 #7: "myvpn":4500 STATE_QUICK_I2 (sent QI2, IPsec SA established); EVENT_SA_REPLACE in 27434s; newest IPSEC; eroute owner; isakmp#4; idle; import:admin initiate
000 #7: "myvpn" esp.6b739ff2@8.149.133.151 esp.3406f4d6@172.24.65.94 tun.0@8.149.133.151 tun.0@172.24.65.94 ref=0 refhim=0 Traffic: ESPin=0B ESPout=0B! ESPmax=4194303B
000
000 Bare Shunt list:
```

在 LG、RG 查看连接状态，出现对应连接

```
[root@iZbp150kdj3mq63thifordZ ~]# ifconfig -a
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.24.65.94  netmask 255.255.240.0  broadcast 172.24.79.255
        inet6 fe80::216:3eff:fe0d:bc  prefixlen 64  scopeid 0x20<link>
        ether 00:16:3e:0d:00:bc  txqueuelen 1000  (Ethernet)
        RX packets 365200  bytes 174803112 (166.7 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 269162  bytes 36480480 (34.7 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

ip_vti0: flags=128<NOARP>  mtu 1480
        tunnel   txqueuelen 1000  (IPIP Tunnel)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
[root@iZbp1fneb1nwh49ypmlustZ ~]# ifconfig -a
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.24.65.1  netmask 255.255.240.0  broadcast 172.24.79.255
        inet6 fe80::216:3eff:fe2f:3d1f  prefixlen 64  scopeid 0x20<link>
        ether 00:16:3e:2f:3d:1f  txqueuelen 1000  (Ethernet)
        RX packets 101338  bytes 144974144 (138.2 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 16449  bytes 2039378 (1.9 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

ip_vti0: flags=128<NOARP>  mtu 1480
        tunnel   txqueuelen 1000  (IPIP Tunnel)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

最后检验可以ping通，成功

四、内容分析

IPsec VPN 基础参数如图：

IPSec SA 生成方式	手动指定生成	IKE 协商生成	IKE SA 协商模式	主模式、野蛮模式
			IKE SA 加密方式	DES、3DES、AES
			IKE SA 验证方式	MD5-HMAC、SHA-HMAC
			IKE SA 密钥生成方式	DH1、DH2、DH5
			IKE SA 认证方式	预共享密钥认证、数字证书认证
			IKE SA 身份标识	IP、FQDN、USER-FQDN、证书 DN
			IKE SA 生命周期	60 秒到 86400 秒（缺省 86400 秒）
IPSec SA 安全协议	AH、ESP			
IPSec SA 封装模式	传输模式、隧道模式			
IPSec SA 加密方式	DES、3DES、AES			
IPSec SA 验证方式	MD5-HMAC、SHA-HMAC			
IPSec SA 生命周期	0 或者 120 秒到 86400 秒（缺省 3600 秒）、0 或 2560KB 到 536870912KB（缺省 4608000KB）			

1.ike

1.ike简介

IKE（Internet Key Exchange）是用于实现IPSec通信安全性的协议，它包括许多参数。为了确保在两个站点之间安全传输IP数据流，这些参数需要进行协商。这个协商过程是通过IKE完成的，IKE协商分为两个阶段：

1. 第一阶段（Phase 1）：建立ISAKMP SA，协商以下信息：
    - 对等体之间采用的认证方式，如预共享密钥或数字证书。
    - 使用的加密算法。
    - 使用的HMAC（Hash-based Message Authentication Code）方式，如MD5或SHA。
    - 使用的Diffie-Hellman密钥组。
    - 协商模式，可以是主模式或者主动模式。
    - SA的生存期。
2. 第二阶段（Phase 2）：建立IPsec SA，协商以下信息：
    - 使用的封装技术，可以是AH（Authentication Header）或ESP（Encapsulating Security Payload）。
    - 使用的加密算法。
    - 使用的HMAC方式，如MD5或SHA。
    - 使用的传输模式，可以是隧道模式或传输模式。
    - SA的生存期。

通过这两个阶段的协商，双方可以确保建立安全的IPSec连接，并进行加密的数据传输。

2.ike参数总结

1. **IKE SA 协商模式**：
    - 主模式（Main Mode）和野蛮模式（Aggressive Mode）。
    - 主模式分离身份信息和密钥交换信息，增加了消息的开销。
    - 野蛮模式将SA、密钥交换和认证相关的载荷组合到一条消息中，减少消息的往返次数。
    - 参数设置方法：aggressive = yes | no，默认为主模式。
2. **IKE SA 加密方式**：
    - 使用对称加密算法对数据进行加密和解密，保证数据的安全性。
    - 常用的对称加密算法有DES、3DES、AES，安全性由高到低依次是：AES、3DES、DES。
    - 参数设置方法：ike = 。
3. **IKE SA 验证方式**：

- 使用验证算法对报文完整性及来源合法性进行验证。
- 常用的验证方式有MD5-HMAC、SHA1-HMAC等，是HASH算法和HMAC两种技术的结合。
- 参数设置方法：根据需求选择合适的认证方式。

4. **IKE SA 密钥生成方式**：
   - 使用Diffie-Hellman（DH）算法进行密钥生成，确保双方密钥的安全性。
   - 常用的DH组长度有768bit（DH1）、1024bit（DH2）、1536bit（DH5）等。
   - 参数设置方法：根据需求选择合适的DH组长度。
5. **IKE SA 认证方式**：
   - 支持预共享密钥认证和数字证书认证两种方式确认对方身份的合法性。
   - 预共享密钥认证简单，而数字证书认证安全性较高。
   - 参数设置方法：authby = pubkey | rsasig | ecdsasig | psk | secret | never | xauthpsk | xauthrsasig。
6. **IKE SA 身份标识**：
   - IKE SA协商中双方需要使用相同类型的身份标识，包括IP地址、FQDN、USER-FQDN、证书DN等。
7. **IKE SA 生命周期**：
   - IKE SA的生命周期通常比IPSec SA生命周期长，设置在60秒到86400秒之间。
   - 参数设置方法：ikelifetime = 3h | ，以小时或其他时间单位表示。

3.ike配置方法与验证举例

以修改IKE SA 生命周期为例：

1. **查看当前参数**：
   - 使用命令`vim /etc/ipsec.d/myvpn.conf`打开配置文件，确认当前的`ikelifetime`参数值为3小时（3h）。
2. **修改IKE SA 生命周期**：
   - 使用文本编辑器（如vim）打开配置文件`/etc/ipsec.d/myvpn.conf`。
   - 找到`ikelifetime`参数所在的行，将其修改为所需的生命周期，比如`ikelifetime = 12h`表示将IKE SA的生命周期设置为12小时。
   - 保存并退出编辑器。
3. **验证参数配置的正确性**：
   - 使用命令`ipsec auto --status`查看IPsec VPN的状态，确认IKE SA的生命周期已经成功修改。
   - 检查输出信息中关于IKE SA的生命周期，确保其值与修改后的设置一致。

在验证过程中，会确认是否正确修改了IKE SA的生命周期，例如，如果`3h`确实被正确转换为`10800s`，则表示修改生命周期的配置方法正确。

2.l2tp-ipsec

1.AH和ESP简介

AH（Authentication Header）和ESP（Encapsulating Security Payload）是IPSec中两种主要的安全协议，用于实现IPSec的身份认证和数据加密的安全机制。

1. **AH协议**：
   - AH协议提供数据完整性确认、数据来源确认、防重放等安全特性。

- 通常使用MD5-HMAC、SHA-HMAC等验证算法实现数据完整性。
- AH协议在IPSec中的工作方式是在原始IP数据包的头部添加一个认证头部，从而保证了数据的完整性和来源确认。
- 由于AH不提供加密功能，因此数据在传输过程中不会被加密。
- 协议号为51。

2. **ESP协议**：
   - ESP协议不仅提供了数据完整性确认、数据来源确认、防重放等安全特性，还提供了数据加密功能。
   - 通常使用DES、3DES、AES等加密算法实现数据加密，使用MD5-HMAC、SHA-HMAC等验证算法实现数据完整性。
   - 在IPSec中，ESP协议在原始IP数据包的头部添加了一个安全负载，从而实现了数据的加密和完整性保护。
   - 与AH相比，ESP协议具有支持数据加密和支持NAT穿越（NAT-T）等优势，因此在IPSec VPN中较为常用。
   - 协议号为50。

总的来说，AH和ESP协议都是IPSec中用于提供安全性的重要组成部分，AH主要用于提供数据完整性和来源认证，而ESP除了提供这些功能外，还额外提供了数据加密的功能，使得ESP在实际的VPN部署中更为常见。

2.配置方法举例

以修改ESP加密算法为例，要修改ESP（Encapsulating Security Payload）的加密算法，可以按照以下步骤进行：

1. **查看当前参数**：
   - 使用命令`vim /etc/ipsec.d/myvpn.conf`打开配置文件，确认当前的ESP加密算法参数。例如，如果发现参数为`3ecs-sha1`，表示当前使用的是3DES加密算法和SHA1哈希算法。
2. **修改ESP加密算法**：
   - 使用文本编辑器（如vim）打开配置文件`/etc/ipsec.d/myvpn.conf`。
   - 找到ESP加密算法参数所在的行，将其修改为所需的算法。例如，将其改为`null-sha1`表示不使用加密，仅使用SHA1哈希算法。
   - 保存并退出编辑器。
3. **重启IPSec服务并查看日志**：
   - 使用命令`systemctl restart ipsec`重启IPSec服务。
   - 使用命令`tailf /var/log/pluto.log`查看IPSec日志，确认服务是否成功重启，并检查是否有相关的修改记录。
4. **验证修改后的ESP加密算法**：
   - 使用Wireshark等网络抓包工具，监听指定的IP地址（例如`172.28.129.83`）。
   - 进行一些数据传输操作，以产生加密的ESP流量。
   - 使用命令`tshark -i any -w /test/test2.pcap host 172.28.129.83 && esp -c 100`捕获数据包，并分析其中是否包含加密的ESP数据。
   - 如果修改后的ESP加密算法生效，应该在捕获的数据包中看不到加密的ESP流量，表示修改成功。

通过以上步骤，可以修改ESP的加密算法，并通过Wireshark验证修改的正确性。

3.总结 ipsec 模式选择及参数配置方法

1.ipsec 封装模式简述

IPSec封装模式是指定安全协议在数据传输中的封装位置，通常有传输模式和隧道模式两种：

1. **传输模式**：
    - 在传输模式下，AH头或ESP头被插入IP头和传输层协议（如TCP或UDP）之间，而不改变原始报文头部。因此，IPSec隧道的源和目的地址与最终通信双方的源和目的地址相同。传输模式通常用于保护两个IPSec对等体之间的相互通信。
    - 传输模式常用于一些特定场景，比如使用GRE over IPSec或L2TP over IPSec协议时，可以使用IPSec隧道来保护GRE或L2TP对等体之间的通信。
2. **隧道模式**：
    - 在隧道模式下，AH头或ESP头被插入原始IP头之前，并且在ESP头或AH头之前生成一个新的IP头。这样可以保护两个IPSec对等体背后的两个网络之间的通信。
    - 隧道模式通常用于站点间网络互通的场景，因为它允许通过网络隧道将数据安全地传输到不同的网络之间。

在AH和ESP协议下，不同的封装模式会影响报文的封装方式和传输路径。参数设置方法可以通过配置文件中的type字段来指定封装模式，通常可选的值有tunnel（隧道模式）和transport（传输模式），以及其他一些可能的选项，具体取决于所使用的IPSec实现。

2.ipsec 模式选择及参数配置方法

下面是进行IPsec模式选择及参数配置的方法：

查看当前参数：

- 首先，你需要查看当前的IPsec配置文件，通常是位于`/etc/ipsec.d/myvpn.conf`，并确认当前的模式为隧道模式（Tunnel）。

```
vim /etc/ipsec.d/myvpn.conf
```

重新启动IPsec服务：

- 在修改配置后，需要重新启动IPsec服务，以使更改生效。可以使用以下命令：

```
systemctl restart ipsec
```

验证配置正确性：

- 修改`/etc/ipsec.d/myvpn.conf`文件，将其改为传输模式（Transform），然后再次检查IPsec服务的日志以确保更改成功。

```
vim /etc/ipsec.d/myvpn.conf
systemctl restart ipsec
tailf /var/log/pluto.log
```

验证结果：

- 在检查IPsec服务的日志后，确认是否成功地将IPsec模式从隧道模式改为传输模式。如果没有出现错误信息，而且日志中显示成功改为传输模式，则说明配置正确。

通过上述步骤，可以成功地选择IPsec的模式并配置相应的参数，确保IPsec的安全通信能够按照你的需求进行。

4.ipsec 网关程序模块构架及关系

在Linux下，IPsec网关程序模块构架及其相互关系如下：

1. IKEv1、IKEv2 **协议处理模块**

- **功能**：负责协商安全会话的密钥，管理加密算法、身份验证和密钥交换协议，处理网络地址转换（NAT）等问题。
- **重要性**：IKE协议是建立和管理IPsec安全会话所必需的，处理IKE协议是IPsec网关程序的核心之一。

2. **加密和认证模块**

- **功能**：包括对称密钥加密算法、公钥加密算法、哈希算法和数字签名算法等。该模块管理证书和密钥，以确保安全通信的机密性和完整性。
- **重要性**：负责IPsec数据的加密和认证，确保数据在传输过程中的保密性和完整性。

3. **安全策略管理模块**

- **功能**：管理安全策略，确定是否、如何保护流量以及如何管理访问控制。
- **重要性**：定义了哪些流量需要受到IPsec保护，以及如何保护这些流量。

4. IPsec **处理模块**

- **功能**：实现IPsec协议的主要功能，包括加密、解密、认证和数据完整性保护，处理IPsec安全关联（SA），以及管理和流量选择。
- **重要性**：负责执行IPsec协议的核心功能，确保IPsec安全通信的实现。

5. **网络接口处理模块**

- **功能**：与操作系统交互，处理网络配置信息，负责数据包转发和重定向，确保流量通过正确的通道传输。
- **重要性**：作为IPsec网关程序与操作系统的接口，负责网络配置和数据包转发，确保IPsec功能与网络通信的无缝集成。

这些模块之间相互配合，共同实现了IPsec网关程序的各项功能，确保安全通信的实现和网络流量的正常传输。

五、参考内容

[1]《CentOS 7 配置成网关服务器》https://www.cnblogs.com/EasonJim/p/10206728.html

[2] ChestnutSilver. 同济大学信息安全原理课程作业[EB/OL]. 2023[2024.3.20]. https://github.com/ChestnutSilver/l2tp-analysis

[3] Linux Centos7网络配置无法ping通外网、内网以及网关_centos7 配置网卡无法到网关-CSDN博客(https://blog.csdn.net/delight_sl/article/details/91358832)

[4] Libreswan Git 存储库 https://github.com/libreswan/libreswan/

[5] chatgpt https://chat.openai.com/