实验1 配置 GRE VPN

1.1 实验内容与目标

完成本实验，您应该能够：

- 配置 GRE VPN 隧道
- 配置 GRE VPN 与路由协议协同工作
- 使用 display 命令获取 GRE VPN 配置和运行信息

1.2 实验组网图

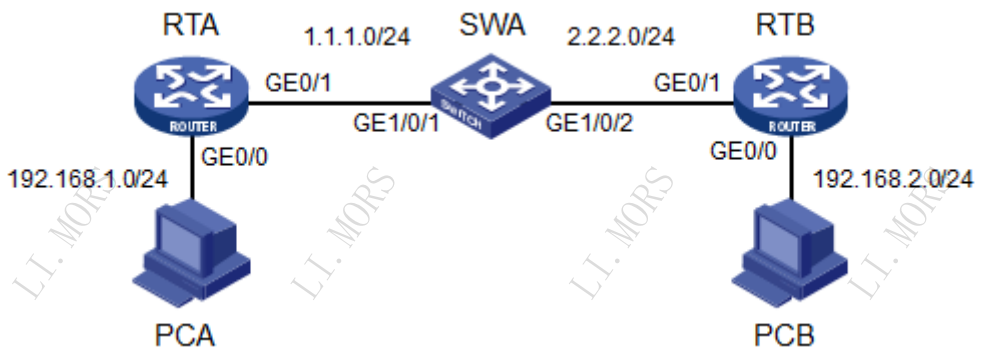


图1-1 GRE VPN 实验环境图

实验组网如图 1-1 所示。

1.3 实验设备与版本

本实验所需之主要设备器材如表 1-1 所示。

表1-1 实验设备和器材

名称和型号	版本	数量	描述
MSR36-20	Version 7.1.049, Release 0106P08	2	
S5820V2-54QS-GE	Version 7.1.045, Release 2311P03	1	
PC	Windows 7 SP1	2	或其它Windows系统主机
第5类UTP以太网连接线	--	4	其中包括交叉线2根

1.4 实验过程

实验任务一：GRE VPN 基本配置

实验前请先清空所有设备的配置，确保其使用出厂默认配置。

步骤一：搭建实验环境

连接设备。在 SWA 上配置 VLAN2，将接口 GE1/0/2 加入 VLAN2。

```
[SWA]vlan 2
[SWA-vlan2]port GigabitEthernet 1/0/2
```

根据表 1-2 配置各物理接口的地址。其中 PCA、PCB 的默认网关分别配置为 RTA 和 RTB。

表1-2 各设备接口 IP 地址

设备	接口	地址
RTA	GE0/0	192.168.1.1/24
	GE0/1	1.1.1.1/24
	Tunnel0	192.168.3.1/30
RTB	GE0/0	192.168.2.1/24
	GE0/1	2.2.2.1/24
	Tunnel0	192.168.3.2/30
SWA	VLAN1	1.1.1.2/24
	VLAN2	2.2.2.2/24
PCA	以太口	192.168.1.2/24
PCB	以太口	192.168.2.2/24

步骤二：检测公网连通性

查看 SWA 的路由表和端口状态，确认其工作正常。

```
[SWA]display ip interface brief
*down: administratively down
(s): spoofing
Interface                  Physical Protocol IP Address     Description
M-GE0/0/0                   down     down   unassigned     M-Gigabit...
Vlan-interface1             up       up       1.1.1.2      Vlan-inte...
Vlan-interface2             up       up       2.2.2.2      Vlan-inte...
[SWA]display ip routing-table
Destinations : 16       Routes : 16

Destination/Mask    Proto  Pre  Cost        NextHop         Interface
0.0.0.0/32          Direct 0    0           127.0.0.1       InLoop0
1.1.1.0/24          Direct 0    0           1.1.1.2         Vlan1
1.1.1.0/32          Direct 0    0           1.1.1.2         Vlan1
1.1.1.2/32          Direct 0    0           127.0.0.1       InLoop0
1.1.1.255/32        Direct 0    0           1.1.1.2         Vlan1
2.2.2.0/24          Direct 0    0           2.2.2.2         Vlan2
```

```
2.2.2.0/32          Direct 0    0           2.2.2.2         Vlan2
2.2.2.2/32          Direct 0    0           127.0.0.1       InLoop0
2.2.2.255/32        Direct 0    0           2.2.2.2         Vlan2
127.0.0.0/8         Direct 0    0           127.0.0.1       InLoop0
127.0.0.0/32        Direct 0    0           127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0           127.0.0.1       InLoop0
127.255.255.255/32 Direct 0    0           127.0.0.1       InLoop0
224.0.0.0/4         Direct 0    0           0.0.0.0         NULL0
224.0.0.0/24        Direct 0    0           0.0.0.0         NULL0
```

255.255.255.255/32 Direct 0 0 127.0.0.1 InLoop0 在 RTA 和 RTB 上配置公网接口互通所需的静态路由。

```
[RTA]interface GigabitEthernet0/0
[RTA-GigabitEthernet0/0]ip address 192.168.1.1 255.255.255.0
[RTA-GigabitEthernet0/0]interface GigabitEthernet0/1
[RTA-GigabitEthernet0/1]ip address 1.1.1.1 255.255.255.0
[RTA-GigabitEthernet0/1]ip route-static 2.2.2.0 255.255.255.0 1.1.1.2

[RTB]interface GigabitEthernet0/0
[RTB-GigabitEthernet0/0]ip address 192.168.2.1 255.255.255.0
[RTB-GigabitEthernet0/0]interface GigabitEthernet0/1
[RTB-GigabitEthernet0/1]ip address 2.2.2.1 255.255.255.0
[RTB-GigabitEthernet0/1]ip route-static 1.1.1.0 255.255.255.0 2.2.2.2
```

在 RTA 上检测与 RTB 的连通性。此时应该可以连通。

至此，实际上以 SWA 模拟的公网已经通信正常。

步骤三：配置 GRE 隧道接口

在 RTA 和 RTB 上建立隧道接口，配置隧道起点和终点。

```
[RTA]interface Tunnel0 mode gre
[RTA-Tunnel0]ip address 192.168.3.1 255.255.255.252
[RTA-Tunnel0]source 1.1.1.1
[RTA-Tunnel0]destination 2.2.2.1

[RTB]interface Tunnel0 mode gre
[RTB-Tunnel0]ip address 192.168.3.2 255.255.255.252
[RTB-Tunnel0]source 2.2.2.1
[RTB-Tunnel0]destination 1.1.1.1
```

步骤四：为私网配置静态路由

在 RTA 和 RTB 上为私网配置静态路由。

```
[RTA]ip route-static 192.168.2.0 255.255.255.0 Tunnel0

[RTB]ip route-static 192.168.1.0 255.255.255.0 Tunnel0
```

步骤五：检验隧道工作状况

用 ping 命令验证 PCA 与 PCB 之间的连通性，此时应该是可以连通的。

查看 RTA 与 RTB 的路由表，可见公网、私网路由均存在于路由表中：

```
[RTB]display ip routing-table
      Destinations : 22      Routes : 22

Destination/Mask    Proto  Pre  Cost        NextHop         Interface
Destination/Mask    Proto   Pre Cost       NextHop         Interface
0.0.0.0/32          Direct  0   0          127.0.0.1       InLoop0
1.1.1.0/24          Static  60  0          2.2.2.2         GE0/2
2.2.2.0/24          Direct  0   0          2.2.2.1         GE0/2
2.2.2.0/32          Direct  0   0          2.2.2.1         GE0/2
```

```
2.2.2.1/32          Direct  0   0          127.0.0.1       InLoop0
2.2.2.255/32        Direct  0   0          2.2.2.1         GE0/2
127.0.0.0/8         Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32        Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32        Direct  0   0          127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
192.168.1.0/24      Static  60  0          0.0.0.0         Tun0
192.168.2.0/24      Direct  0   0          192.168.2.1     GE0/0
192.168.2.0/32      Direct  0   0          192.168.2.1     GE0/0
192.168.2.1/32      Direct  0   0          127.0.0.1       InLoop0
192.168.2.255/32   Direct  0   0          192.168.2.1     GE0/0
192.168.3.0/30      Direct  0   0          192.168.3.2     Tun0
192.168.3.0/32      Direct  0   0          192.168.3.2     Tun0
192.168.3.2/32      Direct  0   0          127.0.0.1       InLoop0
192.168.3.3/32      Direct  0   0          192.168.3.2     Tun0
224.0.0.0/4         Direct  0   0          0.0.0.0         NULL0
224.0.0.0/24        Direct  0   0          0.0.0.0         NULL0
```

255.255.255.255/32 Direct 0 0 127.0.0.1 InLoop0 查看 RTA 和 RTB 的隧道接口状态，可见其使用 GRE 封装，状态为 UP：

```
[RTB]display interface Tunnel 0
Tunnel0
Current state: UP
Line protocol state: UP
Description: Tunnel0 Interface
Bandwidth: 64kbps
Maximum Transmit Unit: 1476
Internet Address is 192.168.3.2/30 Primary
Tunnel source 2.2.2.1, destination 1.1.1.1
Tunnel keepalive disabled
Tunnel TTL 255
Tunnel protocol/transport GRE/IP
   GRE key disabled
   Checksumming of GRE packets disabled
Output queue - Urgent queuing: Size/Length/Discards 0/100/0
Output queue - Protocol queuing: Size/Length/Discards 0/500/0
Output queue - FIFO queuing: Size/Length/Discards 0/75/0
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Input: 16 packets, 1344 bytes, 0 drops
```

Output: 10 packets, 840 bytes, 0 drops 在 RTA 上打开 GRE 协议调试开关用 debugging 命令检验路由器实际收发的报文，说明其地址已经改变。

```
<RTA>terminal monitor
<RTA>terminal debugging
<RTA>debugging gre packet
```

在 PCA 上对 RTB 运行 ping 命令，但只发送一个 ICMP 包：

```
C:\Documents and Settings\User>ping -n 1 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:

Reply from 192.168.2.1: bytes=32 time<1ms TTL=254

Ping statistics for 192.168.2.1:
   Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
   Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

观察 RTA 上的输出信息：

```
<RTA>
```

```
*Oct 15 15:57:03:126 2014 MSR36-20_4 GRE/7/packet:
 Tunnel0 packet: Before encapsulation according to adjacency table,
   192.168.1.2->192.168.2.1 (length = 84)
*Oct 15 15:57:03:126 2014 MSR36-20_4 GRE/7/packet:
 Tunnel0 packet: After encapsulation,
   1.1.1.1->2.2.2.1 (length = 108)
*Oct 15 15:57:03:126 2014 MSR36-20_4 GRE/7/packet:
 Tunnel0 packet: Before de-encapsulation,
   2.2.2.1->1.1.1.1 (length = 108)
*Oct 15 15:57:03:126 2014 MSR36-20_4 GRE/7/packet:
 Tunnel0 packet: After de-encapsulation,
```

192.168.2.1->192.168.1.2 (length = 84)可见 RTA 从 Tunnel0 接口发出了一个包，源地址为 1.1.1.1，目的地址为 2.2.2.1。这说明发送的包已经被 GRE 封装后在公网发送了。

关闭所有 debugging 开关：

```
<RTA>undo debugging all
```

步骤六：清除静态路由

用 undo ip route-static 命令在 RTA 和 RTB 上清除全部静态路由。

```
[RTA]undo ip route-static 192.168.2.0 255.255.255.0 Tunnel0
[RTA]undo ip route-static 2.2.2.0 255.255.255.0 1.1.1.2

[RTB]undo ip route-static 192.168.1.0 255.255.255.0 Tunnel0
[RTB]undo ip route-static 1.1.1.0 255.255.255.0 2.2.2.2
```

步骤七：为公网配置动态路由

在 SWA、RTA 和 RTB 上为公网配置 OSPF：

```
[RTA]ospf 1
[RTA-ospf-1]area 0.0.0.0
[RTA-ospf-1-area-0.0.0.0]network 1.1.1.0 0.0.0.255

[RTB]ospf 1
[RTB-ospf-1]area 0.0.0.0
[RTB-ospf-1-area-0.0.0.0]network 2.2.2.0 0.0.0.255

[SWA]ospf 1
[SWA-ospf-1]area 0.0.0.0
[SWA-ospf-1-area-0.0.0.0]network 1.1.1.0 0.0.0.255
[SWA-ospf-1-area-0.0.0.0]network 2.2.2.0 0.0.0.255
```

步骤八：为私网配置动态路由

在 RTA 和 RTB 上为包括 Tunnel 接口在内的私网接口配置 RIPv2：

```
[RTA]rip 1
[RTA-rip-1]version 2
[RTA-rip-1]network 192.168.1.0
[RTA-rip-1]network 192.168.3.0

[RTB]rip
[RTB-rip-1]version 2
[RTB-rip-1]network 192.168.2.0
[RTB-rip-1]network 192.168.3.0
```

步骤九：再次检验隧道工作状况

用 ping 命令验证 PCA 与 PCB 之间的连通性，此时应该是可以连通的。

查看 RTA 与 RTB 的路由表，可见公网、私网路由均存在于路由表中：

```
<RTB>display ip routing-table

       Destinations : 22      Routes : 22

Destination/Mask    Proto  Pre  Cost        NextHop         Interface
0.0.0.0/32          Direct  0   0           127.0.0.1       InLoop0
1.1.1.0/24          O_INTRA 10  2           2.2.2.2         GE0/2
2.2.2.0/24          Direct  0   0           2.2.2.1         GE0/2
2.2.2.0/32          Direct  0   0           2.2.2.1         GE0/2
2.2.2.1/32          Direct  0   0           127.0.0.1       InLoop0
2.2.2.255/32        Direct  0   0           2.2.2.1         GE0/2
127.0.0.0/8         Direct  0   0           127.0.0.1       InLoop0
127.0.0.0/32        Direct  0   0           127.0.0.1       InLoop0
127.0.0.1/32        Direct  0   0           127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0           127.0.0.1       InLoop0
192.168.1.0/24      RIP     100 1           192.168.3.1     Tun0
192.168.2.0/24      Direct  0   0           192.168.2.1     GE0/0
192.168.2.0/32      Direct  0   0           192.168.2.1     GE0/0
192.168.2.1/32      Direct  0   0           127.0.0.1       InLoop0
192.168.2.255/32    Direct  0   0           192.168.2.1     GE0/0
192.168.3.0/30      Direct  0   0           192.168.3.2     Tun0
192.168.3.0/32      Direct  0   0           192.168.3.2     Tun0
192.168.3.2/32      Direct  0   0           127.0.0.1       InLoop0
192.168.3.3/32      Direct  0   0           192.168.3.2     Tun0
224.0.0.0/4         Direct  0   0           0.0.0.0         NULL0
224.0.0.0/24        Direct  0   0           0.0.0.0         NULL0
```

255.255.255.255/32 Direct 0 0 127.0.0.1 InLoop0 转入下一实验任务。

实验任务二：GRE VPN 隧道验证

步骤一：单方配置隧道验证

首先在 RTA 上单方启动隧道验证：

```
[RTA-Tunnel0]gre key 1234
```

步骤二：检验隧道连通性

用 ping 命令验证 PCA 与 PCB 之间的连通性。由于仅单方配置了隧道验证，此时应该无法连通。

```
C:\Documents and Settings\User>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

步骤三：配置错误的隧道验证

在 RTB 上也启动隧道验证，但验证值配置与 RTA 不同：

```
[RTB-Tunnel0]gre key 12345
```

步骤四：检验隧道连通性

用 ping 命令验证 PCA 与 PCB 之间的连通性。由于配置的隧道验证值错误，此时应该无法连通。

```
C:\Documents and Settings\User>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

步骤五：正确配置隧道验证

在 RTB 上配置与 RTA 相同的验证值：

```
[RTB-Tunnel0]gre key 1234
```

步骤六：检验隧道连通性

用 ping 命令验证 PCA 与 PCB 之间的连通性。由于配置的隧道验证正确，此时应该可以连通。

```
C:\Documents and Settings\User>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:

Reply from 192.168.2.1: bytes=32 time=1ms TTL=254
Reply from 192.168.2.1: bytes=32 time<1ms TTL=254
Reply from 192.168.2.1: bytes=32 time<1ms TTL=254
Reply from 192.168.2.1: bytes=32 time<1ms TTL=254

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

注意：

由于 RTA 和 RTB 上配置了 RIP 路由，如果隧道验证值长时间不匹配，RIP 会删除来自对方的私网路由。在这种情况下，配置了正确的隧道验证值后需要等待 RIP 重新学习路由。

实验任务三：GRE VPN 隧道 Keepalive

步骤一：恢复静态路由配置

在 RTA 和 RTB 上删除 RIP 路由协议的配置，恢复静态路由配置，验证 PCA 和 PCB 可以连通。

```
[RTA]undo rip
Undo RIP process? [Y/N]:y
[RTA]undo ospf
Undo OSPF process? [Y/N]:y
[RTA]ip route-static 192.168.2.0 255.255.255.0 Tunnel0
[RTA]ip route-static 2.2.2.0 255.255.255.0 1.1.1.2

[RTB]undo rip
```

```
Undo RIP process? [Y/N]:y
[RTB]undo ospf
Undo OSPF process? [Y/N]:y
[RTB]ip route-static 192.168.1.0 255.255.255.0 Tunnel0
[RTB]ip route-static 1.1.1.0 255.255.255.0 2.2.2.2
```

步骤二：模拟网络故障

关闭 SWA 的 VLAN2 接口，模拟网络突然发生故障。

```
[SWA-Vlan-interface2]shutdown
```

步骤三：检查 RTA 上的隧道接口状态

在 RTA 上检查隧道接口状态，发现隧道接口状态仍然正常：

```
[RTA]display interface Tunnel 0
Tunnel0
Current state: UP
Line protocol state: UP
Description: Tunnel0 Interface
Bandwidth: 64kbps
Maximum Transmit Unit: 1476
Internet Address is 192.168.3.1/30 Primary
Tunnel source 1.1.1.1, destination 2.2.2.1
Tunnel keepalive disabled
Tunnel TTL 255
Tunnel protocol/transport GRE/IP
   GRE key disabled
   Checksumming of GRE packets disabled
Output queue - Urgent queuing: Size/Length/Discards 0/100/0
Output queue - Protocol queuing: Size/Length/Discards 0/500/0
Output queue - FIFO queuing: Size/Length/Discards 0/75/0
Last clearing of counters: Never
Last 300 seconds input rate: 0 bytes/sec, 0 bits/sec, 0 packets/sec
Last 300 seconds output rate: 1 bytes/sec, 8 bits/sec, 0 packets/sec
Input: 22 packets, 1528 bytes, 0 drops
```

Output: 30 packets, 2136 bytes, 0 drops 这说明其无法了解对端变化情况。这是因为在 RTA 上，隧道源地址所属接口正常，隧道目的地址所需的路由仍然存在。

步骤四：恢复网络故障

```
[SWA-Vlan-interface2]undo shutdown
```

步骤五：配置隧道 Keepalive

在 RTA 和 RTB 上配置隧道 Keepalive。

```
[RTA]interface Tunnel 0
[RTA-Tunnel0]keepalive

[RTB]interface Tunnel 0
[RTB-Tunnel0]keepalive
```

步骤六：模拟网络故障

在 RTA 上启动 debugging 开关：

```
<RTA>terminal monitor
<RTA>terminal debugging
<RTA>debugging gre all
<RTA>debugging tunnel all
```

关闭 SWA 的 VLAN2 接口，模拟公网路由突然发生故障。

```
[SWA-Vlan-interface2]shutdown
```

步骤七：观察效果，检验隧道连通性

在 RTA 上观察 debugging 信息。输出信息形如：

```
<RTA>
*Oct 15 17:05:49:654 2014 MSR36-20_4 GRE/7/packet:
 Tunnel0 packet: Before encapsulation,
   2.2.2.1->1.1.1.1 (length = 24)
*Oct 15 17:05:49:654 2014 MSR36-20_4 GRE/7/packet:
 Tunnel0 packet: After encapsulation,
   1.1.1.1->2.2.2.1 (length = 48)
*Oct 15 17:05:59:654 2014 MSR36-20_4 GRE/7/packet:
 Tunnel0 packet: Before encapsulation,
   2.2.2.1->1.1.1.1 (length = 24)
*Oct 15 17:05:59:654 2014 MSR36-20_4 GRE/7/packet:
 Tunnel0 packet: After encapsulation,
   1.1.1.1->2.2.2.1 (length = 48)
*Oct 15 17:06:09:654 2014 MSR36-20_4 GRE/7/packet:
 Tunnel0 packet: Before encapsulation,
   2.2.2.1->1.1.1.1 (length = 24)
*Oct 15 17:06:09:654 2014 MSR36-20_4 GRE/7/packet:
 Tunnel0 packet: After encapsulation,
   1.1.1.1->2.2.2.1 (length = 48)
%Oct 15 17:06:09:656 2014 MSR36-20_4 IFNET/5/LINK_UPDOWN: Line protocol state on
the interface Tunnel0 changed to down.
*Oct 15 17:06:11:654 2014 MSR36-20_4 TUNNEL/7/event:
```

Tunnel0: No keepalive packet received from the peer.可见经过一段时间后，Tunnel0 接口状态变为 DOWN，根据 debugging 信息，原因是 RTA 未能从对端收到 keepalive 消息。

关闭 debugging 开关，查看 Tunnel0 接口信息：

```
<RTA>undo debugging all
All possible debugging has been turned off
<RTA>display interface tunnel 0
Tunnel0
Current state: UP
Line protocol state: DOWN
Description: Tunnel0 Interface
Bandwidth: 64kbps
Maximum Transmit Unit: 1476
Internet Address is 192.168.3.1/30 Primary
Tunnel source 1.1.1.1, destination 2.2.2.1
Tunnel keepalive enabled, Period(10 s), Retries(3)
Tunnel TTL 255
Tunnel protocol/transport GRE/IP
    GRE key disabled
    Checksumming of GRE packets disabled
Output queue - Urgent queuing: Size/Length/Discards 0/100/0
Output queue - Protocol queuing: Size/Length/Discards 0/500/0
Output queue - FIFO queuing: Size/Length/Discards 0/75/0
Last clearing of counters: Never
Last 300 seconds input rate: 1 bytes/sec, 8 bits/sec, 0 packets/sec
Last 300 seconds output rate: 1 bytes/sec, 8 bits/sec, 0 packets/sec
Input: 32 packets, 1744 bytes, 0 drops
```

Output: 56 packets, 2760 bytes, 0 drops 可见 Tunnel0 接口状态确实已经变为 DOWN。

在 SWA 上重新打开 VLAN2 接口，过一段时间之后， Tunnel0 接口状态以及 PCA 与 PCB 之间的连通性可以恢复正常。

1.5 实验中的命令列表

表1-3 实验命令列表

命令	描述
interface tunnel *interface-number* **mode gre**	创建一个Tunnel接口，并进入该Tunnel接口视图
source { *ip-address* | *interface-type interface-number* }	设置Tunnel接口的源端地址或接口
destination *ip-address*	设置Tunnel接口的目的端地址
debugging gre packet	打开GRE包调试
gre key *key-number*	设置GRE类型隧道接口的密钥

1.6 思考题

1. 若配置路由协议，而不启用 Keepalive，能否发现对方故障？

答：路由协议通过自身的计时器或可靠传送机制可以察觉对端的设备发生故障，并更新路由表中的路由，使数据包从其它可用路径转发，但隧道接口状态并不会因此而发生变化。

实验2 配置 L2TP VPN

2.1 实验内容与目标

完成本实验，您应该能够：

- 配置独立 LAC 模式的 L2TP
- 以 iNode 为客户端配置客户 LAC 模式的 L2TP
- 使用 display 命令获取 L2TP VPN 配置和运行信息
- 使用 debugging 命令了解 L2TP VPN 运行时的重要事件和异常情况

2.2 实验组网图

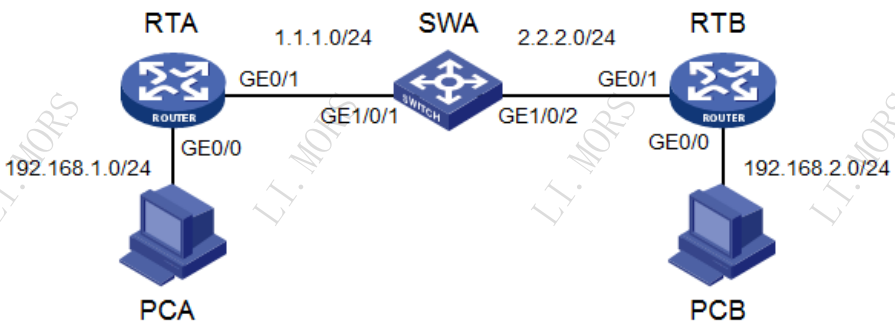


图2-1 L2TP VPN 实验环境图

实验组网如图 2-1 所示。

2.3 实验设备与版本

本实验所需之主要设备器材如表 2-1 所示。

表2-1 实验设备和器材

名称和型号	版本	数量	描述
MSR36-20	Version 7.1.049, Release 0106P08	2	
S5820V2-54QS-GE	Version 7.1.045, Release 2311P03	1	

名称和型号	版本	数量	描述
PC	Windows 7 SP1	2	或其它Windows系统主机
第5类UTP以太网连接线	--	4	其中包括交叉线2根
iNode客户端安装程序	V5.2 E0408	1	

2.4 实验过程

实验任务一：配置独立 LAC 模式

本实验任务中，以 PCA 为客户端，RTA 为 LAC，RTB 为 LNS。

步骤一：搭建实验环境

连接设备。在 SWA 上配置 VLAN2，将接口 E1/0/2 加入 VLAN2。

```
[SWA]vlan 2
[SWA-vlan2]port GigabitEthernet 1/0/2
```

根据表 2-2 配置各接口的地址。其中 PCA、PCB 的默认网关分别配置为 RTA 和 RTB。

表2-2 各设备接口 IP 地址

设备	接口	地址
RTA	GE0/0	无地址
	GE0/1	1.1.1.1/24
	Virtual-template0	无地址
RTB	GE0/0	192.168.2.1/24
	GE0/1	2.2.2.1/24
	Virtual-template1	192.168.1.1/24
SWA	VLAN1	1.1.1.2/24
	VLAN2	2.2.2.2/24
PCA	以太口	自动获取
	PPPoE连接	自动获取
PCB	以太口	192.168.2.2/24

步骤二：检测公网连通性

查看 SWA 的路由表和端口状态，确认其工作正常。

```
[SWA]display ip interface brief
*down: administratively down
(s): spoofing
Interface                   Physical Protocol IP Address     Description
M-GE0/0/0                    up       up       unassigned    M-Gigabit...
```

```
Vlan-interface1            up      up      1.1.1.2       Vlan-inte...
Vlan-interface2            up      up      2.2.2.2       Vlan-inte...
[SWA]display ip routing-table
      Destinations : 16       Routes : 16

Destination/Mask    Proto  Pre  Cost        NextHop         Interface
0.0.0.0/32          Direct 0    0           127.0.0.1       InLoop0
1.1.1.0/24          Direct 0    0           1.1.1.2         Vlan1
1.1.1.0/32          Direct 0    0           1.1.1.2         Vlan1
1.1.1.2/32          Direct 0    0           127.0.0.1       InLoop0
1.1.1.255/32        Direct 0    0           1.1.1.2         Vlan1
2.2.2.0/24          Direct 0    0           2.2.2.2         Vlan2
2.2.2.0/32          Direct 0    0           2.2.2.2         Vlan2
2.2.2.2/32          Direct 0    0           127.0.0.1       InLoop0
2.2.2.255/32        Direct 0    0           2.2.2.2         Vlan2
127.0.0.0/8         Direct 0    0           127.0.0.1       InLoop0
127.0.0.0/32        Direct 0    0           127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0           127.0.0.1       InLoop0
127.255.255.255/32 Direct 0    0           127.0.0.1       InLoop0
224.0.0.0/4         Direct 0    0           0.0.0.0         NULL0
224.0.0.0/24        Direct 0    0           0.0.0.0         NULL0
255.255.255.255/32 Direct 0    0           127.0.0.1       InLoop0
```

在 RTA 和 RTB 上配置公网接口互通所需的静态路由。

```
[RTA]interface GigabitEthernet0/1
[RTA-GigabitEthernet0/1]ip address 1.1.1.1 255.255.255.0
[RTA-GigabitEthernet0/1]ip route-static 2.2.2.0 255.255.255.0 1.1.1.2

[RTB]interface GigabitEthernet0/0
[RTB-GigabitEthernet0/0]ip address 192.168.2.1 255.255.255.0
[RTB-GigabitEthernet0/0]interface GigabitEthernet0/1
[RTB-GigabitEthernet0/1]ip address 2.2.2.1 255.255.255.0
[RTB-GigabitEthernet0/1]ip route-static 1.1.1.0 255.255.255.0 2.2.2.2
```

在 RTA 上检测与 RTB 的连通性。此时应该可以连通。

至此，实际上以 SWA 模拟的公网已经通信正常。

步骤三：配置 PPPoE

在 RTA 上配置 PPPoE Server，以便接受 PCA 发起的拨号。

首先配置验证域 abc.com，目的是为了给 PPPoE 和 L2TP 验证提供验证参数：

```
[RTA]domain abc.com
[RTA-isp-abc.com]authentication ppp local
```

然后配置 PPPoE 用户和密码。此用户名和密码也将被用于 L2TP 验证。

```
[RTA]local-user vpdnuser class network
[RTA-luser-vpdnuser]password simple Hello
[RTA-luser-vpdnuser]service-type ppp
```

配置一个虚模版接口，并为物理接口启动 PPPoE 服务，以接受 PPPoE 拨号连接并进行验证：

```
[RTA-luser-vpdnuser]interface Virtual-Template0
[RTA-Virtual-Template0]ppp authentication-mode chap domain abc.com
[RTA-Virtual-Template0]interface GigabitEthernet0/0
[RTA-GigabitEthernet0/0]pppoe-server bind Virtual-Template 0
```

步骤四：配置 LAC

在 RTA 上进行配置。首先启动 L2TP 功能：

```
[RTA]l2tp enable
```

然后配置 L2TP 组：

```
[RTA]l2tp-group 1 mode lac
[RTA-l2tp1]tunnel password simple aabbcc
[RTA-l2tp1]tunnel name LAC
[RTA-l2tp1]user domain abc.com[RTA-l2tp1]lns-ip 2.2.2.1
```

其中 lns-ip 命令指定了 LNS 的地址，user domain 命令指定 abc.com 域内的用户为 L2TP 用户。这样 abc.com 域内的用户拨入将触发 L2TP 隧道建立。

步骤五：配置 LNS

在 RTB 上进行配置。首先启动 L2TP 功能：

```
[RTB]l2tp enable
```

然后配置 abc.com 域，并配置 IP 地址池。此域用于提供对 L2TP VPN 用户进行身份验证的参数，此地址池用于对 L2TP VPN 客户端分配 IP 地址：

```
[RTB]ip pool 1 192.168.1.2 192.168.1.100
[RTB]domain abc.com
[RTB-isp-abc.com]authentication ppp local
```

随后添加一个本地用户，并配置其密码和服务类型，用于对 L2TP VPN 用户进行身份验证：

```
[RTB-isp-abc.com]local-user vpdnuser class network
[RTB-luser-vpdnuser]password simple Hello
[RTB-luser-vpdnuser]service-type ppp
```

接着配置 L2TP 组，指定其接受来自 abc.com 域且名为 LAC 的对端设备发起的控制连接，并配置了相应的隧道本端名称、隧道验证密码等：

```
[RTB-luser-vpdnuser]l2tp-group 1 mode lns
[RTB-l2tp1]allow l2tp virtual-template 1 remote LAC
[RTB-l2tp1]tunnel password simple aabbcc
[RTB-l2tp1]tunnel name LNS
```

最后还需要配置一个虚模版接口，以便对拨入的 L2TP VPN 用户进行身份验证，为其分配地址并与其进行 IP 通信：

```
[RTB-l2tp1]interface Virtual-Template1
[RTB-Virtual-Template1]ppp authentication-mode chap domain abc.com
[RTB-Virtual-Template1]remote address pool 1
[RTB-Virtual-Template1]ip address 192.168.1.1 255.255.255.0
```

步骤六：配置 PPPoE 客户端，发起 L2TP 呼叫

在 PCA 上创建 PPPoE 连接。在 Windows 7 中，在任务栏上单击【开始】->【控制面板】，打开如图 2-2 所示的【控制面板】界面。

图2-2 【控制面板】界面

单击【查看网络状态和任务】，进入图 2-3 所示的【网络和共享中心】界面。

图2-3 网络和共享中心

单击【设置新的连接或网络】，进入图 2-4 所示的窗口，选择【连接到 Internet】，并单击【下一步】，进入图 2-5 所示的窗口。

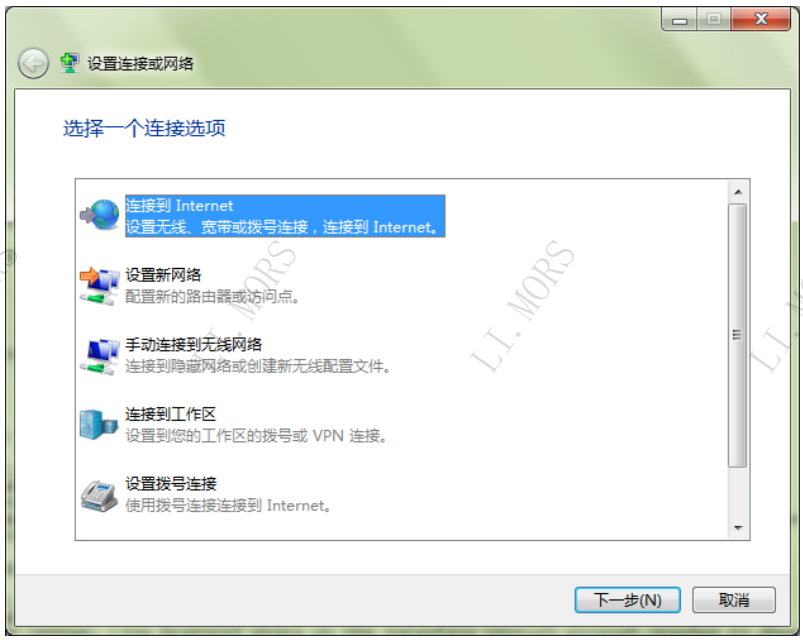


图2-4 设置连接或网络

图2-5 连接到 Internet

选择【宽带(PPPoE)(R)】，进入图 2-6 所示的窗口，在【用户名】处输入用户名 vpdnuser@abc.com，在【密码】处输入密码 Hello。

单击【连接】，即可完成连接设置并发起 L2TP 连接请求。

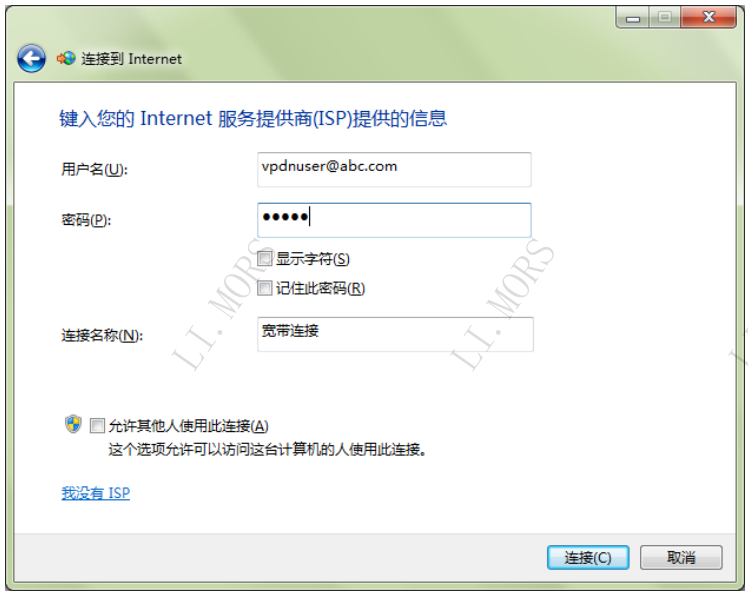


图2-6 连接到 Internet（续）

步骤七：检测私网连通性

从 PCA 上 ping PCB，检测连通性。应该可以连通。

步骤八：观察隧道建立过程

在 RTA 和 RTB 上用 display 命令查看相关信息，可见 RTA 与 RTB 之间建立了一个 L2TP 隧道，其中有一个 L2TP 会话：

```
[RTA]display l2tp tunnel
LocalTID RemoteTID State       Sessions RemoteAddress   RemotePort RemoteName
64111    22991    Established  1        2.2.2.1         1701       LNS
[RTA]display l2tp session
LocalSID    RemoteSID    LocalTID    State
1352        515          64111       Established
```

```
<RTB>display l2tp tunnel
LocalTID RemoteTID State       Sessions RemoteAddress   RemotePort RemoteName
22991    64111    Established  1        1.1.1.1         1701       LAC
<RTB>display l2tp session
LocalSID    RemoteSID    LocalTID    State
515         1352         22991       Established
```

用 reset 命令终止隧道：

```
<RTB>reset l2tp tunnel name LAC
%Oct 16 15:53:20:276 2014 RTB IFNET/3/PHY_UPDOWN: Physical state on the interface
Virtual-Access0 changed to down.
```

%Oct 16 15:53:20:277 2014 RTB IFNET/5/LINK_UPDOWN: Line protocol state on the interface Virtual-Access0 changed to down.用 display 命令查看相关信息，发现隧道和会话都消失。

在 RTA 和 RTB 上打开 debugging 开关。

```
<RTA>debugging l2tp event
<RTA>debugging l2tp control-packet
```

```
<RTB>debugging l2tp event
<RTB>debugging l2tp control-packet
```

重新发起呼叫，通过 debugging 信息观察隧道建立的过程：

```
<RTA>
*Oct 16 15:55:30:616 2014 RTA L2TPV2/7/KEVENT:
 Interface Virtual-Access0 created.
%Oct 16 15:55:30:627 2014 RTA IFNET/3/PHY_UPDOWN: Physical state on the interface
Virtual-Access0 changed to up.
%Oct 16 15:55:33:729 2014 RTA IFNET/5/LINK_UPDOWN: Line protocol state on the
interface Virtual-Access0 changed to up.
*Oct 16 15:55:33:733 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Message-Type AVP:
 80 08 00 00  00 00 00 01
*Oct 16 15:55:33:733 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Protocol-Version AVP:
 80 08 00 00  00 02 01 00
*Oct 16 15:55:33:733 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Host-Name AVP:
 80 09 00 00  00 07 4c 41  43
*Oct 16 15:55:33:733 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Vendor-Name AVP:
 00 12 00 00  00 08 48 33  43 20 4d 53  52 33 36 2d
 32 30
*Oct 16 15:55:33:733 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Framing-Capabilities AVP:
 80 0a 00 00  00 03 00 00  00 03
*Oct 16 15:55:33:733 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Assigned-Tunnel-ID AVP:
 80 08 00 00  00 09 ae ae
*Oct 16 15:55:33:733 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Bearer-Capabilities AVP:
 80 0a 00 00  00 04 00 00  00 03
*Oct 16 15:55:33:733 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Receive-Window-Size AVP:
 80 08 00 00  00 0a 04 00
*Oct 16 15:55:33:734 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Challenge AVP:
 80 16 00 00  00 0b 2b ba  79 75 04 79  d2 15 50 cb
 32 2c d7 0c  b7 fc
*Oct 16 15:55:33:735 2014 RTA L2TPV2/7/CONTROL-PKT:
 Received SCCRP packet from port 1701 (TunnelID=44718, length=117, Ns=0, Nr=1).
Packet content:
 c8 02 00 75  ae ae 00 00  00 00 00 01  80 08 00 00
 00 00 00 02  80 08 00 00  00 02 01 00  80 09 00 00
 00 07 4c 4e  53 80 0a 00  00 00 03 00  00 00 00 80
 08 00 00 00  09 fe 13 80  0a 00 00 00  04 00 00 00
 03 80 08 00  00 00 0a 04  00 80 16 00  00 00 0b 5b
 75 9e aa 7c  5c 9e 3f 03  8e e7 fc ad  d1 3f f5 80
 16 00 00 00  0d fa f5 c1  af f0 29 fc  d4 5a 10 44
 63 29 55 5f  80
*Oct 16 15:55:33:736 2014 RTA L2TPV2/7/KEVENT:
cioctl add tunnel vrf 0 lip 16843009 rip 33686017 lport 1701 rport 1701 ltid 44718
rtid 65043 type 0
*Oct 16 15:55:33:735 2014 RTA L2TPV2/7/EVENT:
 Parsed Message-Type AVP: 2.
*Oct 16 15:55:33:735 2014 RTA L2TPV2/7/EVENT:
 Parsed Protocol-Version AVP. Version=1, Revision=0.
*Oct 16 15:55:33:735 2014 RTA L2TPV2/7/EVENT:
 Parsed Host-Name AVP: LNS.
*Oct 16 15:55:33:735 2014 RTA L2TPV2/7/EVENT:
 Parsed Framing-Capabilities AVP: 0.
*Oct 16 15:55:33:735 2014 RTA L2TPV2/7/EVENT:
 Parsed Assigned-Tunnel-ID AVP: 65043.
*Oct 16 15:55:33:735 2014 RTA L2TPV2/7/EVENT:
```

```
 Parsed Bearer-Capabilities AVP: 3.
*Oct 16 15:55:33:735 2014 RTA L2TPV2/7/EVENT:
 Parsed Receive-Window-Size AVP: 1024.
*Oct 16 15:55:33:735 2014 RTA L2TPV2/7/EVENT:
 Parsed Challenge AVP: 5b 75 9e aa 7c 5c 9e 3f 03 8e e7 fc ad d1 3f f5
*Oct 16 15:55:33:735 2014 RTA L2TPV2/7/EVENT:
 Parsed Challenge-Response AVP: fa f5 c1 af f0 29 fc d4 5a 10 44 63 29 55 5f 80
*Oct 16 15:55:33:735 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Message-Type AVP:
 80 08 00 00  00 00 00 03
*Oct 16 15:55:33:735 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Challenge-Response AVP:
 80 16 00 00  00 0d e6 d9  8b 29 9a eb  2a 70 a1 fe
 dd 71 37 6a  99 b9
*Oct 16 15:55:33:736 2014 RTA L2TPV2/7/EVENT:
 TunnelID=44718: Processed SCCRP packet in Wait-reply state, sent SCCCN packet
and changed the tunnel state to Established.
*Oct 16 15:55:33:736 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Message-Type AVP:
 80 08 00 00  00 00 00 0a
*Oct 16 15:55:33:736 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Assigned-Session-ID AVP:
 80 08 00 00  00 0e 05 48
*Oct 16 15:55:33:736 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Call-Serial-Number AVP:
 80 0a 00 00  00 0f 00 00  05 48
*Oct 16 15:55:33:736 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Bearer-Type AVP:
 80 0a 00 00  00 12 00 00  00 01
*Oct 16 15:55:33:736 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Physical-Channel-ID AVP:
 00 0a 00 00  00 19 00 00  00 00
*Oct 16 15:55:33:736 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Called-Number AVP:
 80 06 00 00  00 15
*Oct 16 15:55:33:736 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Calling-Number AVP:
 80 06 00 00  00 16
*Oct 16 15:55:33:736 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Sub-Address AVP:
 80 06 00 00  00 17
*Oct 16 15:55:33:737 2014 RTA L2TPV2/7/CONTROL-PKT:
 Received ICRP packet from port 1701 (TunnelID=44718, length=28, Ns=1, Nr=3).
Packet content:
 c8 02 00 1c  ae ae 05 48  00 01 00 03  80 08 00 00
 00 00 00 0b  80 08 00 00  00 0e 02 03
*Oct 16 15:55:33:737 2014 RTA L2TPV2/7/EVENT:
 Parsed Message-Type AVP: 11.
*Oct 16 15:55:33:737 2014 RTA L2TPV2/7/EVENT:
 Parsed Assigned-Session-ID AVP: 515.
*Oct 16 15:55:33:738 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Message-Type AVP:
 80 08 00 00  00 00 00 0c
*Oct 16 15:55:33:738 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated (Tx)Connect-Speed AVP:
 80 0a 00 00  00 18 3b 9a  ca 00
*Oct 16 15:55:33:738 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Framing-Type AVP:
 80 0a 00 00  00 13 00 00  00 01
*Oct 16 15:55:33:738 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Rx-Connect-Speed AVP:
 00 0a 00 00  00 26 3b 9a  ca 00
*Oct 16 15:55:33:738 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Initial-Received-LCP-CONFREQ AVP:
 00 17 00 00  00 1a 01 04  05 c8 05 06  0c f7 23 9e
```

```
 07 02 08 02  0d 03 06
*Oct 16 15:55:33:738 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Last-Sent-LCP-CONFREQ AVP:
 00 11 00 00  00 1b 03 05  c2 23 05 05  06 ac b3 f9
 71
*Oct 16 15:55:33:738 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Last-Received-LCP-CONFREQ AVP:
 00 14 00 00  00 1c 01 04  05 c8 05 06  0c f7 23 9e
 07 02 08 02
*Oct 16 15:55:33:738 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Proxy-Authen-Type AVP:
 00 08 00 00  00 1d 00 02
*Oct 16 15:55:33:739 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Proxy-Authen-Name AVP:
 00 16 00 00  00 1e 76 70  64 6e 75 73  65 72 40 61
 62 63 2e 63  6f 6d
*Oct 16 15:55:33:739 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Proxy-Authen-Challenge AVP:
 00 16 00 00  00 1f 82 5b  0a 78 ac 4a  f0 37 d6 17
 13 fa e4 68  24 d2
*Oct 16 15:55:33:739 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Proxy-Authen-ID AVP:
 00 08 00 00  00 20 00 01
*Oct 16 15:55:33:739 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Proxy-Authen-Response AVP:
 00 16 00 00  00 21 b2 f4  33 41 e4 04  2e f5 60 a8
 22 28 de a1  63 f1
*Oct 16 15:55:33:739 2014 RTA L2TPV2/7/EVENT:
 TunnelID=44718, SessionID=1352: Processed ICRP packet in Wait-reply state, sent
ICCN packet to the peer and changed the session state to Established.
*Oct 16 15:55:33:995 2014 RTA L2TPV2/7/CONTROL-PKT:
 Received ZLB-ACK packet from port 1701 (TunnelID=44718, length=12, Ns=2, Nr=4).
Packet content:
```

c8 02 00 0c　ae ae 00 00　00 02 00 04 断开连接，观察 debugging 信息：

```
<RTA>
%Oct 16 15:56:17:688 2014 RTA IFNET/5/LINK_UPDOWN: Line protocol state on the
interface Virtual-Access0 changed to down.
%Oct 16 15:56:17:688 2014 RTA IFNET/3/PHY_UPDOWN: Physical state on the interface
Virtual-Access0 changed to down.
*Oct 16 15:56:17:689 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Message-Type AVP:
 80 08 00 00  00 00 00 0e
*Oct 16 15:56:17:689 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Assigned-Session-ID AVP:
 80 08 00 00  00 0e 05 48
*Oct 16 15:56:17:689 2014 RTA L2TPV2/7/CONTROL-PKT:
 Encapsulated Result-Code AVP:
 80 08 00 00  00 01 00 01
*Oct 16 15:56:17:691 2014 RTA L2TPV2/7/KEVENT:
 Interface Virtual-Access0 deleted.
*Oct 16 15:56:17:895 2014 RTA L2TPV2/7/CONTROL-PKT:
 Received ZLB-ACK packet from port 1701 (TunnelID=44718, length=12, Ns=2, Nr=5).
Packet content:
 c8 02 00 0c  ae ae 00 00  00 02 00 05
```

这样就可以了解呼叫中 L2TP 的主要信息交换过程。限于篇幅，此处不列出 RTB 的 debugging 输出信息，请自行观察。

实验任务二：配置客户 LAC 模式

本实验任务中，以 PCA 为客户端并安装 iNode 客户端软件，以 RTB 为 LNS，RTA 和 SWA 则模拟公网设备。

步骤一：执行基本配置

在实验任务一的连接基础上，根据表 2-3 修改 RTA 的 IP 地址为 3.3.3.1/24。为 PCA 配置 IP 地址 3.3.3.2/24，默认网关 3.3.3.1。

表2-3 各设备接口 IP 地址

设备	接口	地址
RTA	GE0/0	3.3.3.1/24
	GE0/1	1.1.1.1/24
RTB	GE0/0	192.168.2.1/24
	GE0/1	2.2.2.1/24
	Virtual-template1	192.168.1.1/24
SWA	VLAN1	1.1.1.2/24
	VLAN2	2.2.2.2/24
PCA	以太口	3.3.3.2/24
	VPN连接	自动获取
PCB	以太口	192.168.2.2/24

步骤二：配置公网路由

删除所有静态路由。在 RTA 上删除所有 PPPoE 和 L2TP 配置。

```
[RTA-GigabitEthernet0/0]undo pppoe-server bind
All PPPoE Sessions on GigabitEthernet0/0 will be deleted, continue?[Y/N]:y

[RTA]undo domain abc.com
[RTA]undo l2tp enable
[RTA]undo l2tp-group 1
[RTA]undo interface Virtual-Template 0
[RTA]undo local-user vpdnuser class network
[RTA]undo domain abc.com
```

在 RTA、RTB 和 SWA 上配置 OSPF：

```
[RTA]ospf
[RTA-ospf-1]area 0
[RTA-ospf-1-area-0.0.0.0]network 1.1.1.0 0.0.0.255
[RTA-ospf-1-area-0.0.0.0]network 3.3.3.0 0.0.0.255

[SWA]ospf
[SWA-ospf-1]area 0
[SWA-ospf-1-area-0.0.0.0]network 1.1.1.0 0.0.0.255
[SWA-ospf-1-area-0.0.0.0]network 2.2.2.0 0.0.0.255

[RTB]ospf
[RTB-ospf-1]area 0
[RTB-ospf-1-area-0.0.0.0]network 2.2.2.0 0.0.0.255
```

配置完成后，查看 RTA、RTB 和 SWA 的路由表，并使用 ping 命令，确认三台设备可以互通。

步骤三：安装 iNode 客户端

在 PCA 上安装 iNode 客户端。启动安装程序，跟随安装向导完成安装即可。

图2-7 确认安装 H3C 网络适配器

注意：

要使 iNode 客户端支持 L2TP 功能，在安装过程中必须确认安装 H3C 网络适配器，如图 2-7 所示。

步骤四：配置 iNode 客户端

启动 iNode 客户端程序，在其主界面窗口中单击菜单【文件】|【新建连接】，启动新建连接向导，如图 2-8 所示。首次运行 iNode 客户端时程序将自动弹出“新建连接向导”对话框。

单击【下一步】，进入图 2-9 所示窗口，单击选定【L2TP IPSec VPN 协议】。

图2-8 新建连接向导

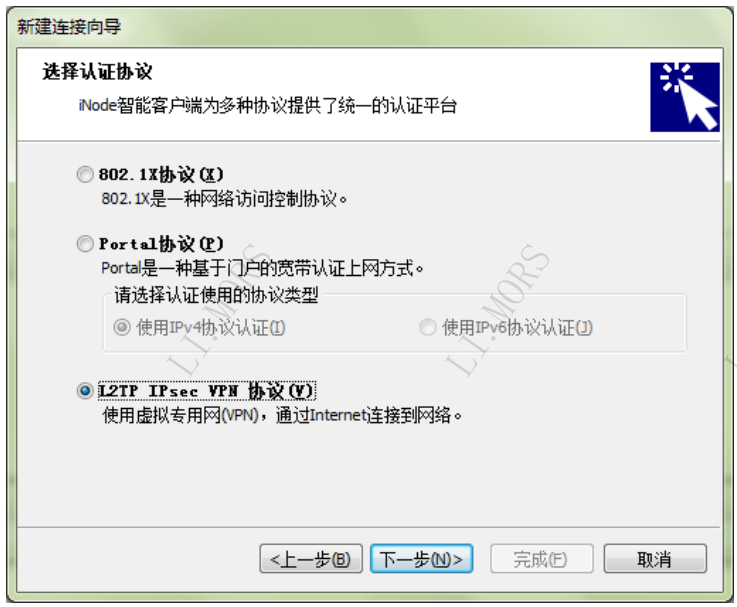


图2-9 选择认证协议

单击【下一步】，进入图 2-10 所示窗口，单击选定【普通连接】。

单击【下一步】，进入图 2-11 所示窗口，在【连接名】处输入一个连接名称，例如“我的 VPN 连接”，在【登录用户名】处输入用户名，在【登录密码】处输入密码。

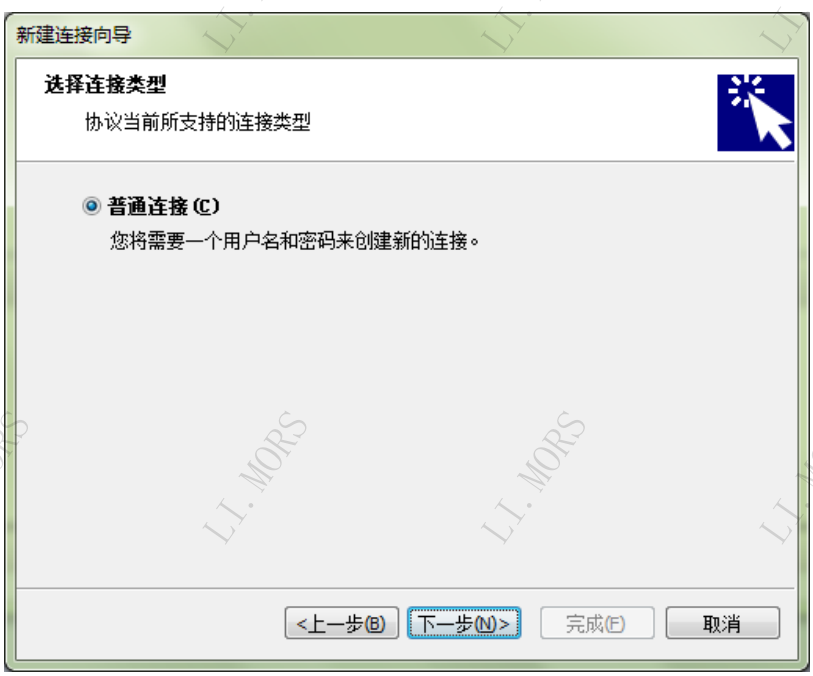


图2-10 选择连接类型

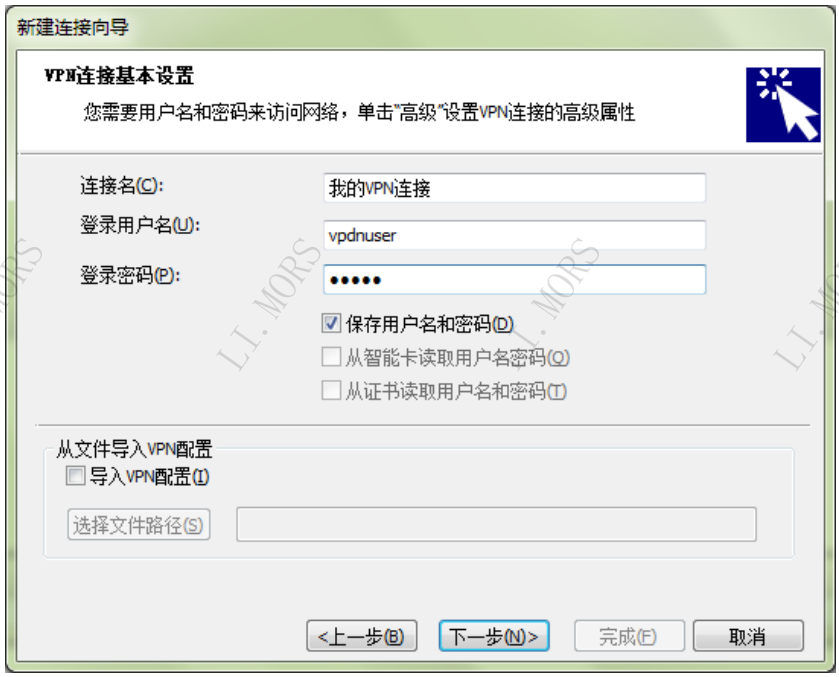


图2-11 设置用户名和密码

单击【下一步】，进入图 2-12 所示窗口，输入 LNS 服务器地址。

单击【高级】进入图 2-13 所示的窗口，进入【L2TP 设置】选项卡，输入隧道名称 LAC，选择认证模式为 CHAP，单击选定【使用隧道验证密码】并输入隧道验证密码 aabbcc。单击【确定】回到图 2-12 所示窗口。

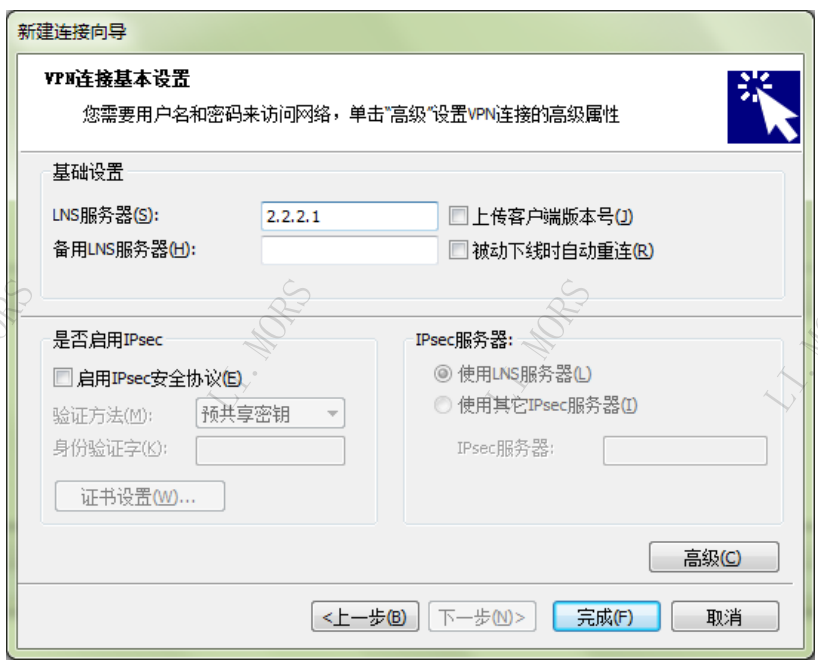


图2-12 VPN 连接基本设置

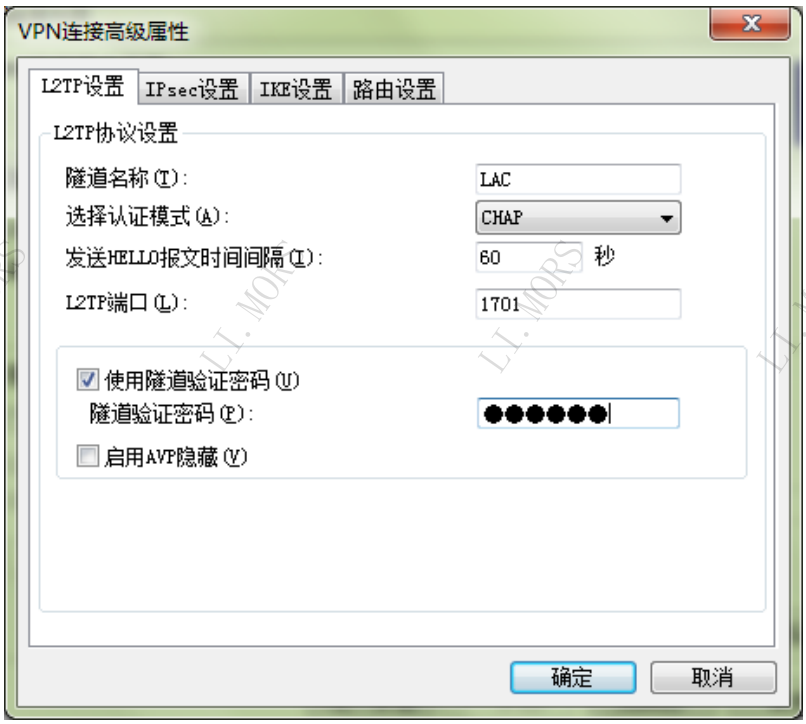


图2-13 VPN 连接高级属性

单击【下一步】进入图 2-14 所示的窗口，单击【创建】，即可创建新建连接。

图2-14 完成新建连接向导

步骤五：配置 LNS

在 RTB 上保留上一实验任务中的 LNS 配置即可。为方便起见，此处列出相关的配置作为参考，如果已经修改了 RTB 的相关配置，可清除原 L2TP 相关配置并执行下列配置：

```
[RTB]l2tp enable
[RTB]ip pool 1 192.168.1.2 192.168.1.100
[RTB]domain abc.com
[RTB-isp-abc.com]authentication ppp local

[RTB-isp-abc.com]local-user vpdnuser class network
[RTB-luser-vpdnuser]password simple Hello
[RTB-luser-vpdnuser]service-type ppp
[RTB-luser-vpdnuser]l2tp-group 1 mode lns
[RTB-l2tp1]allow l2tp virtual-template 1 remote LAC
[RTB-l2tp1]tunnel password simple aabbcc
[RTB-l2tp1]tunnel name LNS
[RTB-l2tp1]interface Virtual-Template1
[RTB-Virtual-Template1]ppp authentication-mode chap domain abc.com
[RTB-Virtual-Template1]remote address pool 1
```

步骤六：发起 L2TP 呼叫，建立 L2TP 隧道

由客户端发起呼叫。此时应可以呼叫成功。

在 PCA 上用查看连接，可见除了物理的以太网连接之外，还出现了一个新的连接，其地址处于 192.168.1.0/24 网段，是从 RTB 的接口 Virtual-template 1 上动态获得的。

```
C:\Users\j06566>ipconfig

Windows IP 配置

以太网适配器 本地连接:

   连接特定的 DNS 后缀 . . . . . :
   本地链接 IPv6 地址. . . . . : fe80::fda0:29b8:84f0:d02d%29
   IPv4 地址 . . . . . : 192.168.1.2
   子网掩码  . . . . . : 255.255.255.255
   默认网关. . . . . : 0.0.0.0
                               192.168.1.1

以太网适配器 Intel(R) 82579LM Gigabit Network Connection:

   连接特定的 DNS 后缀 . . . . . :
   IPv4 地址 . . . . . : 3.3.3.2
   子网掩码  . . . . . : 255.255.255.0
   默认网关. . . . . : 3.3.3.1
```

在 PCA 上检测对 PCB 的连通性，此时应可以连通：

```
C:\Users\j06566>ping 192.168.2.2

正在 Ping 192.168.2.2 具有 32 字节的数据:
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=254
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=254
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=254
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=254

192.168.2.2 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位):
```

最短 = 0ms，最长 = 0ms，平均 = 0ms 在 RTB 上用 display 命令查看 L2TP 隧道和会话信息，可见隧道已建立，其中包含一个会话：

```
<RTB>display l2tp tunnel
LocalTID RemoteTID State       Sessions RemoteAddress   RemotePort RemoteName
5692     1         Established     1       3.3.3.2        64643     LAC
<RTB>display l2tp session
LocalSID    RemoteSID    LocalTID    State
```

1542 8322 5692 Established

2.5 实验中的命令列表

表2-4 实验命令列表

命令	描述
l2tp enable	启用L2TP功能
ip pool *pool-name start-ip-address* [*end-ip-address*] [**group** *group-name*]	配置地址池
l2tp-group *group-number* **mode** { **lac** | **lns** }	创建L2TP组，并进入L2TP组视图
tunnel name *name*	配置隧道本端的名称
tunnel password { **cipher** | **simple** } *password*	配置隧道验证密码
user { **domain** *domain-name* | **fullusername** *user-name* }	配置本端作为L2TP LAC端时发起隧道建立请求的触发条件
lns-ip { *ip-address* }&<1-5>	配置LNS的IP地址
domain *isp-name*	创建一个ISP域，并进入ISP域视图
authentication ppp local	配置PPP域用户的AAA本地验证方案
interface virtual-template *virtual-template-number*	创建虚接口模板，进入虚接口模板视图
ppp authentication-mode { **chap | ms-chap** | **ms-chap-v2** | **pap** } * [[**call-in**] | **domain** *isp-name*]	配置本端对PPP用户进行验证
remote address { **pool** [*pool-number*] | *ip-address* }	指定给对端分配地址所用的地址池或直接给对端分配IP地址
allow l2tp virtual-template *virtual-template-number* **remote** *remote-name* [**domain** *domain-name*]	指定接收隧道连接请求的虚拟接口模板、隧道对端名称和域名
display l2tp tunnel	显示当前L2TP隧道的信息
display l2tp session	显示当前L2TP会话的信息

2.6 思考题

1. 在实验任务一中，如果两个用户使用同一域名发起连接，在 RTB 上可以查看到几个隧道几个会话？

   答：一个隧道两个会话。RTA 会对两个用户使用同一隧道，而不会新建一个隧道。

2. 在实验任务二中，如果两个用户使用同一域名发起连接，在 RTB 上可以查看到几个隧道几个会话？

   答：两个隧道，每个隧道一个会话。每个 PC 会独立对 RTB 建立隧道和会话。

实验3 IPsec VPN 基本配置

3.1 实验内容与目标

完成本实验，您应该能够：

- 配置 IPsec+预共享密钥的 IKE 主模式
- 配置 IPsec+预共享密钥的 IKE 野蛮模式

3.2 实验组网图

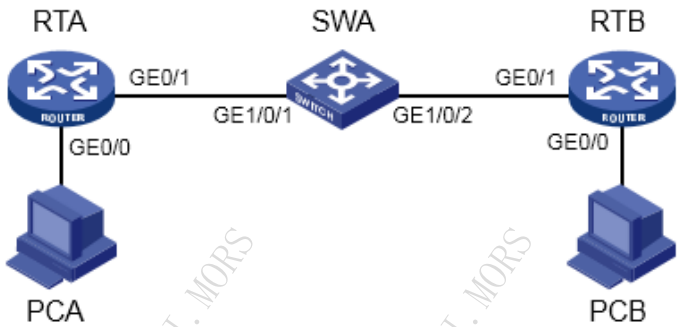


图3-1 IPsec VPN 实验环境图

实验组网如图 3-1 所示。

3.3 实验设备与版本

本实验所需之主要设备器材如表 3-1 所示。

表3-1 实验设备和器材

名称和型号	版本	数量	描述
MSR36-20	CMW 7.1.049-R0106P08	2	
S5820V2-54QS-GE	CMW 7.1.045-R2311P03	1	
PC	Windows 7 SP1	2	
第5类UTP以太网连接线	--	4	

3.4 实验过程

实验任务一：配置 IPsec+IKE 主模式

本实验任务要求在 RTA 和 RTB 之间建立隧道。使用 IKE 预共享密钥验证方式。

步骤一：搭建实验环境

连接设备。在 SWA 上配置 VLAN2，将接口 GE1/0/2 加入 VLAN2。

```
[SWA]vlan 2
[SWA-vlan2]port GigabitEthernet 1/0/2
```

根据表 3-2 配置各接口的地址。其中 PCA、PCB 的默认网关分别配置为 RTA 和 RTB。

表3-2 各设备接口 IP 地址

设备	接口	地址
RTA	GE0/0	192.168.1.1/24
	GE0/1	1.1.1.1/24
RTB	GE0/0	192.168.2.1/24
	GE0/1	2.2.2.1/24
SWA	VLAN1	1.1.1.2/24
	VLAN2	2.2.2.2/24
PCA	以太口	192.168.1.2/24
PCB	以太口	192.168.2.2/24

步骤二：配置路由协议

在 RTA、SWA 和 RTB 之间配置 OSPF：

```
[RTA]ospf 1
[RTA-ospf-1]area 0.0.0.0
[RTA-ospf-1-area-0.0.0.0]network 1.1.1.0 0.0.0.255
[RTA-ospf-1-area-0.0.0.0]quit
[RTA-ospf-1]quit

[SWA]ospf 1
[SWA-ospf-1]area 0.0.0.0
[SWA-ospf-1-area-0.0.0.0]network 1.1.1.0 0.0.0.255
[SWA-ospf-1-area-0.0.0.0]network 2.2.2.0 0.0.0.255
[SWA-ospf-1-area-0.0.0.0]quit
[SWA-ospf-1]quit

[RTB]ospf 1
[RTB-ospf-1]area 0.0.0.0
[RTB-ospf-1-area-0.0.0.0]network 2.2.2.0 0.0.0.255
[RTB-ospf-1-area-0.0.0.0]quit
[RTB-ospf-1]quit
```

OSPF 自治系统不包括 RTA、RTB 与 PCA、PCB 互连的接口，因此，作为模拟公网设备的 SWA 上不具备 192.168.1.0 和 192.168.2.0 网段的路由，只有公网路由。

在 RTA 和 RTB 上为私网配置静态路由：

```
[RTA]ip route-static 192.168.2.0 255.255.255.0 1.1.1.2

[RTB]ip route-static 192.168.1.0 255.255.255.0 2.2.2.2
```

配置后查看 RTA、RTB 和 SWA 的路由表，可见 SWA 上没有私网路由：

```
<RTA>display ip routing-table

Destinations : 18      Routes : 18

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.0/24         Direct  0   0          1.1.1.1         GE0/1
1.1.1.0/32         Direct  0   0          1.1.1.1         GE0/1
1.1.1.1/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.255/32       Direct  0   0          1.1.1.1         GE0/1
2.2.2.0/24         O_INTRA 10  2          1.1.1.2         GE0/1
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
192.168.1.0/24     Direct  0   0          192.168.1.1     GE0/0
192.168.1.0/32     Direct  0   0          192.168.1.1     GE0/0
192.168.1.1/32     Direct  0   0          127.0.0.1       InLoop0
192.168.1.255/32   Direct  0   0          192.168.1.1     GE0/0
192.168.2.0/24     Static  60  0          1.1.1.2         GE0/1
224.0.0.0/4        Direct  0   0          0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0          0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0


<SWA>display ip routing-table

Destinations : 16      Routes : 16

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.0/24         Direct  0   0          1.1.1.2         Vlan1
1.1.1.0/32         Direct  0   0          1.1.1.2         Vlan1
1.1.1.2/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.255/32       Direct  0   0          1.1.1.2         Vlan1
2.2.2.0/24         Direct  0   0          2.2.2.2         Vlan2
2.2.2.0/32         Direct  0   0          2.2.2.2         Vlan2
2.2.2.2/32         Direct  0   0          127.0.0.1       InLoop0
2.2.2.255/32       Direct  0   0          2.2.2.2         Vlan2
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
224.0.0.0/4        Direct  0   0          0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0          0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0


<RTB>display ip routing-table

Destinations : 18      Routes : 18

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.0/24         O_INTRA 10  2          2.2.2.2         GE0/1
2.2.2.0/24         Direct  0   0          2.2.2.1         GE0/1
2.2.2.0/32         Direct  0   0          2.2.2.1         GE0/1
2.2.2.1/32         Direct  0   0          127.0.0.1       InLoop0
2.2.2.255/32       Direct  0   0          2.2.2.1         GE0/1
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
```

```
192.168.1.0/24    Static  60  0          2.2.2.2        GE0/1
192.168.2.0/24    Direct  0   0          192.168.2.1    GE0/0
192.168.2.0/32    Direct  0   0          192.168.2.1    GE0/0
192.168.2.1/32    Direct  0   0          127.0.0.1      InLoop0
192.168.2.255/32  Direct  0   0          192.168.2.1    GE0/0
224.0.0.0/4       Direct  0   0          0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0          0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0          127.0.0.1      InLoop0
```

验证 PCA 与 PCB 之间的连通性：

```
C:\Users\administrator>ping 192.168.2.2

正在 Ping 192.168.2.2 具有 32 字节的数据:
请求超时。
请求超时。
请求超时。
请求超时。

192.168.2.2 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 0，丢失 = 4 (100% 丢失)，
```

可见由于此时 SWA 没有私网的路由，PCA 是无法 ping 通 PCB 的。

步骤三：配置 IKE proposal

```
[RTA]ike proposal 1
[RTA-ike-proposal-1]authentication-method pre-share
[RTA-ike-proposal-1]authentication-algorithm md5
[RTA-ike-proposal-1]encryption-algorithm 3des-cbc
[RTA-ike-proposal-1]quit

[RTB]ike proposal 1
[RTB-ike-proposal-1]authentication-method pre-share
[RTB-ike-proposal-1]authentication-algorithm md5
[RTB-ike-proposal-1]encryption-algorithm 3des-cbc
[RTB-ike-proposal-1]quit
```

步骤四：配置 IKE keychain

```
[RTA]ike keychain keychain1
[RTA-ike-keychain-keychain1]pre-shared-key address 2.2.2.1 255.255.255.0 key
simple h3c
[RTA-ike-keychain-keychain1]quit

[RTB]ike keychain keychain1
[RTB-ike-keychain-keychain1]pre-shared-key address 1.1.1.1 255.255.255.0 key
simple h3c
[RTB-ike-keychain-keychain1]quit
```

步骤五：配置 IKE profile

采用用预共享密钥方式：

```
[RTA]ike profile profile1
[RTA-ike-profile-profile1]local-identity address 1.1.1.1
[RTA-ike-profile-profile1]match remote identity address 2.2.2.1 255.255.255.0
[RTA-ike-profile-profile1]keychain keychain1
[RTA-ike-profile-profile1]proposal 1
[RTA-ike-profile-profile1]quit

[RTB]ike profile profile1
[RTB-ike-profile-profile1]local-identity address 2.2.2.1
[RTB-ike-profile-profile1]match remote identity address 1.1.1.1 255.255.255.0
```

```
[RTB-ike-profile-profile1]keychain keychain1
[RTB-ike-profile-profile1]proposal 1
[RTB-ike-profile-profile1]quit
```

步骤六：配置安全 ACL

由于 IPsec 隧道需要保护的是私网数据，因此安全 ACL 应匹配 192.168.1.0/24 网段与 192.168.2.0/24 网段之间的数据流。

```
[RTA]acl advanced 3000
[RTA-acl-ipv4-adv-3000]rule 0 permit ip source 192.168.1.0 0.0.0.255 destination
192.168.2.0 0.0.0.255
[RTA-acl-ipv4-adv-3000]quit

[RTB]acl advanced 3000
[RTB-acl-ipv4-adv-3000] rule 0 permit ip source 192.168.2.0 0.0.0.255 destination
192.168.1.0 0.0.0.255
[RTB-acl-ipv4-adv-3000]quit
```

步骤七：配置 IPsec 安全提议

```
[RTA]ipsec transform-set tran1
[RTA-ipsec-proposal-prop1]esp authentication-algorithm sha1
[RTA-ipsec-proposal-prop1]esp encryption-algorithm aes-cbc-128
[RTA-ipsec-transform-set-tran1]quit

[RTB]ipsec transform-set tran1
[RTB-ipsec-transform-set-tran1]esp authentication-algorithm sha1
[RTB-ipsec-transform-set-tran1]esp encryption-algorithm aes-cbc-128
[RTB-ipsec-transform-set-tran1]quit
```

步骤八：配置并应用 IPsec 安全策略

配置 IPsec 安全策略，并将其应用于通往对方的物理接口上：

```
[RTA]ipsec policy policy1 1 isakmp
[RTA-ipsec-policy-isakmp-policy1-1]remote-address 2.2.2.1
[RTA-ipsec-policy-isakmp-policy1-1]security acl 3000
[RTA-ipsec-policy-isakmp-policy1-1]transform-set tran1
[RTA-ipsec-policy-isakmp-policy1-1]ike-profile profile1
[RTA-ipsec-policy-isakmp-policy1-1]quit
[RTA]interface GigabitEthernet 0/1
[RTA-GigabitEthernet0/1]ipsec apply policy policy1
[RTA-GigabitEthernet0/1]quit

[RTB]ipsec policy policy1 1 isakmp
[RTB-ipsec-policy-isakmp-policy1-1]remote-address 1.1.1.1
[RTB-ipsec-policy-isakmp-policy1-1]security acl 3000
[RTB-ipsec-policy-isakmp-policy1-1]transform-set tran1
[RTB-ipsec-policy-isakmp-policy1-1]ike-profile profile1
[RTB-ipsec-policy-isakmp-policy1-1]quit
[RTB]interface GigabitEthernet 0/1
[RTB-GigabitEthernet0/1]ipsec apply policy policy1
[RTB-GigabitEthernet0/1]quit
```

步骤九：检验配置

在 RTA 和 RTB 上用 display 命令检查配置参数：

```
[RTA]display ike proposal
 Priority Authentication Authentication Encryption  Diffie-Hellman Duration
            method      algorithm    algorithm      group     (seconds)
-------------------------------------------------------------------------------
```

```
 1       PRE-SHARED-KEY    MD5       3DES-CBC    Group 1       86400
 default  PRE-SHARED-KEY    SHA1      DES-CBC     Group 1       86400

[RTA]display ipsec transform-set
IPsec transform set: tran1
  State: complete
  Encapsulation mode: tunnel
  Transform: ESP
  ESP protocol:
    Integrity: SHA1
    Encryption: AES-CBC-128

[RTA]display ipsec policy
-----------------------------------------
IPsec Policy: policy1
Interface: GigabitEthernet0/1
-----------------------------------------

  ----------------------------
  Sequence number: 1
  Mode: ISAKMP
  ----------------------------
  Security data flow: 3000
  Selector mode: standard
  Local address:
  Remote address: 2.2.2.1
  Transform set:  tran1
  IKE profile: profile1
  SA duration(time based): 3600 seconds
  SA duration(traffic based): 1843200 kilobytes
  SA idle time:

[RTB]display ike proposal
 Priority Authentication Authentication Encryption  Diffie-Hellman Duration
            method      algorithm    algorithm      group     (seconds)
------------------------------------------------------------------------------
 1       PRE-SHARED-KEY    MD5       3DES-CBC    Group 1       86400
 default  PRE-SHARED-KEY    SHA1      DES-CBC     Group 1       86400

[RTA]display ipsec transform-set
IPsec transform set: tran1
  State: complete
  Encapsulation mode: tunnel
  Transform: ESP
  ESP protocol:
    Integrity: SHA1
    Encryption: AES-CBC-128

[RTA]display ipsec policy
-----------------------------------------
IPsec Policy: policy1
Interface: GigabitEthernet0/1
-----------------------------------------

  ----------------------------
  Sequence number: 1
  Mode: ISAKMP
  ----------------------------
  Security data flow: 3000
  Selector mode: standard
  Local address:
  Remote address: 1.1.1.1
  Transform set:  tran1
  IKE profile: profile1
  SA duration(time based): 3600 seconds
```

```
  SA duration(traffic based): 1843200 kilobytes
  SA idle time:
```

由这些命令输出可以看到当前配置所设定的 IPsec/IKE 参数。

步骤十：检验隧道工作状况

从 PCA 检测与 PCB 的连通性：

```
C:\Users\administrator>ping 192.168.2.2

正在 Ping 192.168.2.2 具有 32 字节的数据:
请求超时。
来自 192.168.2.2 的回复: 字节=32 时间=1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126

192.168.2.2 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 3，丢失 = 1 (25% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms，最长 = 1ms，平均 = 0ms
```

可见除第一个 ICMP Echo Request 包被报告超时之外，其他的都成功收到 Echo Reply 包。这是因为第一个包触发了 IKE 协商，在 IPsec SA 成功建立之前，这个包无法获得 IPsec 服务，只能被丢弃。而 IPsec SA 很快就成功建立了，后续的包也就可以顺利到达目的。

在 RTA 与 RTB 上查看 IPsec/IKE 相关信息：

```
<RTA>display ike sa
   Connection-ID   Remote              Flag       DOI
-----------------------------------------------------------------------------
   9             2.2.2.1             RD         IPSEC
Flags:
RD--READY RL--REPLACED FD-FADING

<RTA>display ike sa verbose
  ----------------------------------------------
  Connection ID: 9
  Outside VPN:
  Inside VPN:
  Profile: profile1
  Transmitting entity: Initiator
  ----------------------------------------------
  Local IP: 1.1.1.1
  Local ID type: IPV4_ADDR
  Local ID: 1.1.1.1

  Remote IP: 2.2.2.1
  Remote ID type: IPV4_ADDR
  Remote ID: 2.2.2.1

  Authentication-method: PRE-SHARED-KEY
  Authentication-algorithm: MD5
  Encryption-algorithm: 3DES-CBC

  Life duration(sec): 86400
  Remaining key duration(sec): 86348
  Exchange-mode: Main
  Diffie-Hellman group: Group 1
  NAT traversal: Not detected

<RTA>display ipsec sa
------------------------------
```

```
Interface: GigabitEthernet0/1
------------------------------

  ----------------------------
  IPsec policy: policy1
  Sequence number: 1
  Mode: isakmp
  ----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect forward secrecy:
    Path MTU: 1427
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.1
    Flow:
    sour addr: 192.168.1.0/255.255.255.0  port: 0  protocol: ip
    dest addr: 192.168.2.0/255.255.255.0  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 4206531119 (0xfaba922f)
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843199/3545
      Max received sequence-number: 3
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: active

    [Outbound ESP SAs]
      SPI: 3011773065 (0xb3840289)
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843199/3545
      Max sent sequence-number: 3
      UDP encapsulation used for NAT traversal: N
      Status: active
<RTB>dis ike sa
    Connection-ID   Remote              Flag        DOI
-----------------------------------------------------------------------
    9             1.1.1.1             RD          IPSEC
Flags:
RD--READY RL--REPLACED FD-FADING

<RTB>display ike sa verbose
   ------------------------------------------------
   Connection ID: 9
   Outside VPN:
   Inside VPN:
   Profile: profile1
   Transmitting entity: Responder
   ------------------------------------------------
   Local IP: 2.2.2.1
   Local ID type: IPV4_ADDR
   Local ID: 2.2.2.1

   Remote IP: 1.1.1.1
   Remote ID type: IPV4_ADDR
   Remote ID: 1.1.1.1

   Authentication-method: PRE-SHARED-KEY
   Authentication-algorithm: MD5
   Encryption-algorithm: 3DES-CBC
```

```
  Life duration(sec): 86400
  Remaining key duration(sec): 86324
  Exchange-mode: Main
  Diffie-Hellman group: Group 1
  NAT traversal: Not detected

<RTB>display ips sa
------------------------------
Interface: GigabitEthernet0/1
------------------------------

 ----------------------------
 IPsec policy: policy1
 Sequence number: 1
 Mode: isakmp
 ----------------------------
   Tunnel id: 0
   Encapsulation mode: tunnel
   Perfect forward secrecy:
   Path MTU: 1427
   Tunnel:
       local  address: 2.2.2.1
       remote address: 1.1.1.1
   Flow:
   sour addr: 192.168.2.0/255.255.255.0  port: 0  protocol: ip
   dest addr: 192.168.1.0/255.255.255.0  port: 0  protocol: ip

   [Inbound ESP SAs]
     SPI: 3011773065 (0xb3840289)
     Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
     SA duration (kilobytes/sec): 1843200/3600
     SA remaining duration (kilobytes/sec): 1843199/3521
     Max received sequence-number: 3
     Anti-replay check enable: Y
     Anti-replay window size: 64
     UDP encapsulation used for NAT traversal: N
     Status: active

   [Outbound ESP SAs]
     SPI: 4206531119 (0xfaba922f)
     Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
     SA duration (kilobytes/sec): 1843200/3600
     SA remaining duration (kilobytes/sec): 1843199/3521
     Max sent sequence-number: 3
     UDP encapsulation used for NAT traversal: N
     Status: active
```

可见 ISAKMP SA 和 IPsec SA 都已经正常生成。观察 IPsec SA 中 IP 地址、SPI 等参数的对应关系。其中可以观察到 RTA 和 RTB 的对应方向的 SPI 值是相同的，采用的验证算法和加密算法也相同。

步骤十一：观察 IPsec 工作过程

为了了解 IKE 和 IPsec 协商和加密操作过程，首先清除 IPsec SA 和 ISAKMP SA，中断 IPsec 隧道，以便重新观察整个过程：

```
<RTA>reset ike sa
<RTA>reset ipsec sa

<RTB>reset ike sa
<RTB>reset ipsec sa
```

打开 debugging 开关：

```
<RTA>terminal monitor
% Current terminal monitor is on
<RTA>terminal debugging
% Current terminal debugging is on
<RTA>debugging ike packet
<RTA>debugging ipsec packet
```

在 PCA 上 ping PCB，重新触发 IPsec 隧道建立：

```
C:\Users\administrator>ping 192.168.2.2

正在 Ping 192.168.2.2 具有 32 字节的数据:
请求超时。
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126

192.168.2.2 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 3，丢失 = 1 (25% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms，最长 = 0ms，平均 = 0ms
```

观察 debugging 输出信息，分析其过程：

```
<RTA>*Jan  2 18:39:46:323 2015 RTA IPSEC/7/packet:
Failed to find SA by SP.
*Jan  2 18:39:46:323 2015 RTA IPSEC/7/packet:
The reason of dropping packet is no available IPsec tunnel.
*Jan  2 18:39:46:323 2015 RTA IKE/7/Packet:  Encryption algorithm is 3DES-CBC.
*Jan  2 18:39:46:323 2015 RTA IKE/7/Packet:  Hash algorithm is HMAC-MD5.
*Jan  2 18:39:46:323 2015 RTA IKE/7/Packet:  DH group 1.
*Jan  2 18:39:46:323 2015 RTA IKE/7/Packet:  Authentication method is Pre-shared
key.
*Jan  2 18:39:46:323 2015 RTA IKE/7/Packet:  Lifetime type is in seconds.
*Jan  2 18:39:46:323 2015 RTA IKE/7/Packet:  Life duration is 86400.
*Jan  2 18:39:46:324 2015 RTA IKE/7/Packet: Construct transfrom payload for
transform 1.
*Jan  2 18:39:46:324 2015 RTA IKE/7/Packet: Construct SA payload.
*Jan  2 18:39:46:324 2015 RTA IKE/7/Packet: Construct NAT-T rfc3947 vendor ID
payload.
*Jan  2 18:39:46:324 2015 RTA IKE/7/Packet: Construct NAT-T draft3 vendor ID
payload.
*Jan  2 18:39:46:324 2015 RTA IKE/7/Packet: Construct NAT-T draft2 vendor ID
payload.
*Jan  2 18:39:46:324 2015 RTA IKE/7/Packet: Construct NAT-T draft1 vendor ID
payload.
*Jan  2 18:39:46:324 2015 RTA IKE/7/Packet: Sending packet to 2.2.2.1 remote port
500, local port 500.
*Jan  2 18:39:46:324 2015 RTA IKE/7/Packet:
  I-Cookie: 2f3a7dd5255e195f
  R-Cookie: 0000000000000000
  next payload: SA
  version: ISAKMP Version 1.0
  exchange mode: Main
  flags:
  message ID: 0
  length: 164
*Jan  2 18:39:46:324 2015 RTA IKE/7/Packet: Sending an IPv4 packet.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet: Received packet from 2.2.2.1 source
port 500 destination port 500.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet:
  I-Cookie: 2f3a7dd5255e195f
  R-Cookie: 7135150829533058
```

```
  next payload: SA
  version: ISAKMP Version 1.0
  exchange mode: Main
  flags:
  message ID: 0
  length: 104
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet: Received ISAKMP Security Association
Payload.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet: Received ISAKMP Vendor ID Payload.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet: Process SA payload.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet: Check ISAKMP transform 1.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet:   Encryption algorithm is 3DES-CBC.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet:   HASH algorithm is HMAC-MD5.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet:   DH group is 1.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet:   Authentication method is Pre-shared
key.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet:   Lifetime type is 1.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet:   Life duration is 86400.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet: Attributes is acceptable.
*Jan  2 18:39:46:325 2015 RTA IKE/7/Packet: Process vendor ID payload.
*Jan  2 18:39:46:327 2015 RTA IKE/7/Packet: Construct KE payload.
*Jan  2 18:39:46:327 2015 RTA IKE/7/Packet: Construct NONCE payload.
*Jan  2 18:39:46:327 2015 RTA IKE/7/Packet: Construct NAT-D payload.
*Jan  2 18:39:46:327 2015 RTA IKE/7/Packet: Construct DPD vendor ID payload.
*Jan  2 18:39:46:328 2015 RTA IKE/7/Packet: Sending packet to 2.2.2.1 remote port
500, local port 500.
*Jan  2 18:39:46:328 2015 RTA IKE/7/Packet:
  I-Cookie: 2f3a7dd5255e195f
  R-Cookie: 7135150829533058
  next payload: KE
  version: ISAKMP Version 1.0
  exchange mode: Main
  flags:
  message ID: 0
  length: 208
*Jan  2 18:39:46:328 2015 RTA IKE/7/Packet: Sending an IPv4 packet.
*Jan  2 18:39:46:332 2015 RTA IKE/7/Packet: Received packet from 2.2.2.1 source
port 500 destination port 500.
*Jan  2 18:39:46:332 2015 RTA IKE/7/Packet:
  I-Cookie: 2f3a7dd5255e195f
  R-Cookie: 7135150829533058
  next payload: KE
  version: ISAKMP Version 1.0
  exchange mode: Main
  flags:
  message ID: 0
  length: 208
*Jan  2 18:39:46:332 2015 RTA IKE/7/Packet: Received ISAKMP Key Exchange Payload.
*Jan  2 18:39:46:333 2015 RTA IKE/7/Packet: Received ISAKMP Nonce Payload.
*Jan  2 18:39:46:333 2015 RTA IKE/7/Packet: Received ISAKMP NAT-D Payload.
*Jan  2 18:39:46:333 2015 RTA IKE/7/Packet: Received ISAKMP NAT-D Payload.
*Jan  2 18:39:46:333 2015 RTA IKE/7/Packet: Received ISAKMP Vendor ID Payload.
*Jan  2 18:39:46:333 2015 RTA IKE/7/Packet: Process KE payload.
*Jan  2 18:39:46:333 2015 RTA IKE/7/Packet: Process NONCE payload.
*Jan  2 18:39:46:335 2015 RTA IKE/7/Packet: Received 2 NAT-D payload.
*Jan  2 18:39:46:335 2015 RTA IKE/7/Packet: Local ID type: IPV4_ADDR (1).
*Jan  2 18:39:46:335 2015 RTA IKE/7/Packet: Local ID value: 1.1.1.1.
*Jan  2 18:39:46:335 2015 RTA IKE/7/Packet: Construct ID payload.
*Jan  2 18:39:46:335 2015 RTA IKE/7/Packet: HASH:
 f6f2e6f2 67bbab45 5a5011d5 dbda7548
*Jan  2 18:39:46:335 2015 RTA IKE/7/Packet: Construct authentication by
pre-shared-key.
*Jan  2 18:39:46:336 2015 RTA IKE/7/Packet: Construct INITIAL-CONTACT payload.
*Jan  2 18:39:46:336 2015 RTA IKE/7/Packet: Encrypt the packet.
*Jan  2 18:39:46:336 2015 RTA IKE/7/Packet: Process vendor ID payload.
```

```
*Jan  2 18:39:46:336 2015 RTA IKE/7/Packet: Sending packet to 2.2.2.1 remote port
500, local port 500.
*Jan  2 18:39:46:336 2015 RTA IKE/7/Packet:
  I-Cookie: 2f3a7dd5255e195f
  R-Cookie: 7135150829533058
  next payload: ID
  version: ISAKMP Version 1.0
  exchange mode: Main
  flags: ENCRYPT
  message ID: 0
  length: 92
*Jan  2 18:39:46:336 2015 RTA IKE/7/Packet: Sending an IPv4 packet.
*Jan  2 18:39:46:337 2015 RTA IKE/7/Packet: Received packet from 2.2.2.1 source
port 500 destination port 500.
*Jan  2 18:39:46:337 2015 RTA IKE/7/Packet:
  I-Cookie: 2f3a7dd5255e195f
  R-Cookie: 7135150829533058
  next payload: ID
  version: ISAKMP Version 1.0
  exchange mode: Main
  flags: ENCRYPT
  message ID: 0
  length: 60
*Jan  2 18:39:46:337 2015 RTA IKE/7/Packet: Decrypt the packet.
*Jan  2 18:39:46:338 2015 RTA IKE/7/Packet: Received ISAKMP Identification
Payload.
*Jan  2 18:39:46:338 2015 RTA IKE/7/Packet: Received ISAKMP Hash Payload.
*Jan  2 18:39:46:338 2015 RTA IKE/7/Packet: Process ID payload.
*Jan  2 18:39:46:338 2015 RTA IKE/7/Packet: Peer ID type: IPV4_ADDR (1).
*Jan  2 18:39:46:338 2015 RTA IKE/7/Packet: Peer ID value: address 2.2.2.1.
*Jan  2 18:39:46:338 2015 RTA IKE/7/Packet: Verify HASH payload.
*Jan  2 18:39:46:338 2015 RTA IKE/7/Packet: HASH:
 8afbb1e7 34cebdfa 6c9dce12 f2c598c5
*Jan  2 18:39:46:339 2015 RTA IKE/7/Packet: Set attributes according to phase 2
transform.
*Jan  2 18:39:46:339 2015 RTA IKE/7/Packet:   Encapsulation mode is Tunnel.
*Jan  2 18:39:46:340 2015 RTA IKE/7/Packet:   in seconds
*Jan  2 18:39:46:340 2015 RTA IKE/7/Packet:   Life duration is 3600.
*Jan  2 18:39:46:340 2015 RTA IKE/7/Packet:   in kilobytes
*Jan  2 18:39:46:340 2015 RTA IKE/7/Packet:   Life duration is 1843200.
*Jan  2 18:39:46:340 2015 RTA IKE/7/Packet:   Authentication algorithm is
HMAC-SHA1.
*Jan  2 18:39:46:340 2015 RTA IKE/7/Packet:   Key length is 128 bytes.
*Jan  2 18:39:46:340 2015 RTA IKE/7/Packet:   Transform ID is AES-CBC.
*Jan  2 18:39:46:340 2015 RTA IKE/7/Packet: Construct transform 1.
*Jan  2 18:39:46:340 2015 RTA IKE/7/Packet: Construct IPsec proposal 1.
*Jan  2 18:39:46:340 2015 RTA IKE/7/Packet: Construct IPsec SA payload.
*Jan  2 18:39:46:341 2015 RTA IKE/7/Packet: Construct NONCE payload.
*Jan  2 18:39:46:341 2015 RTA IKE/7/Packet: Construct IPsec ID payload.
*Jan  2 18:39:46:341 2015 RTA IKE/7/Packet: Construct IPsec ID payload.
*Jan  2 18:39:46:341 2015 RTA IKE/7/Packet: Construct HASH(1) payload.
*Jan  2 18:39:46:341 2015 RTA IKE/7/Packet: Encrypt the packet.
*Jan  2 18:39:46:341 2015 RTA IKE/7/Packet: Sending packet to 2.2.2.1 remote port
500, local port 500.
*Jan  2 18:39:46:341 2015 RTA IKE/7/Packet:
  I-Cookie: 2f3a7dd5255e195f
  R-Cookie: 7135150829533058
  next payload: HASH
  version: ISAKMP Version 1.0
  exchange mode: Quick
  flags: ENCRYPT
  message ID: 36a08563
  length: 164
*Jan  2 18:39:46:341 2015 RTA IKE/7/Packet: Sending an IPv4 packet.
*Jan  2 18:39:46:346 2015 RTA IKE/7/Packet: Received packet from 2.2.2.1 source
```

```
port 500 destination port 500.
*Jan  2 18:39:46:346 2015 RTA IKE/7/Packet:
  I-Cookie: 2f3a7dd5255e195f
  R-Cookie: 7135150829533058
  next payload: HASH
  version: ISAKMP Version 1.0
  exchange mode: Quick
  flags: ENCRYPT
  message ID: 36a08563
  length: 164
*Jan  2 18:39:46:347 2015 RTA IKE/7/Packet: Decrypt the packet.
*Jan  2 18:39:46:347 2015 RTA IKE/7/Packet: Received ISAKMP Hash Payload.
*Jan  2 18:39:46:347 2015 RTA IKE/7/Packet: Received ISAKMP Security Association
Payload.
*Jan  2 18:39:46:347 2015 RTA IKE/7/Packet: Received ISAKMP Nonce Payload.
*Jan  2 18:39:46:347 2015 RTA IKE/7/Packet: Received ISAKMP Identification Payload
(IPsec DOI).
*Jan  2 18:39:46:347 2015 RTA IKE/7/Packet: Received ISAKMP Identification Payload
(IPsec DOI).
*Jan  2 18:39:46:347 2015 RTA IKE/7/Packet: Process HASH payload.
*Jan  2 18:39:46:347 2015 RTA IKE/7/Packet: Process IPsec SA payload.
*Jan  2 18:39:46:348 2015 RTA IKE/7/Packet: Check IPsec proposal 1.
*Jan  2 18:39:46:348 2015 RTA IKE/7/Packet: Parse transform 1.
*Jan  2 18:39:46:348 2015 RTA IKE/7/Packet:   Encapsulation mode is Tunnel.
*Jan  2 18:39:46:348 2015 RTA IKE/7/Packet:   Lifetime type is in seconds.
*Jan  2 18:39:46:348 2015 RTA IKE/7/Packet:   Life duration is 3600.
*Jan  2 18:39:46:348 2015 RTA IKE/7/Packet:   Lifetime type is in kilobytes.
*Jan  2 18:39:46:348 2015 RTA IKE/7/Packet:   Life duration is 1843200.
*Jan  2 18:39:46:348 2015 RTA IKE/7/Packet:   Authentication algorithm is
HMAC-SHA1.
*Jan  2 18:39:46:348 2015 RTA IKE/7/Packet:   Key length is 128 bytes.
*Jan  2 18:39:46:349 2015 RTA IKE/7/Packet:   Transform ID is AES-CBC.
*Jan  2 18:39:46:349 2015 RTA IKE/7/Packet: The proposal is acceptable.
*Jan  2 18:39:46:349 2015 RTA IKE/7/Packet: Process IPsec ID payload.
*Jan  2 18:39:46:349 2015 RTA IKE/7/Packet: Process IPsec ID payload.
*Jan  2 18:39:46:352 2015 RTA IKE/7/Packet: Construct HASH(3) payload.
*Jan  2 18:39:46:352 2015 RTA IKE/7/Packet: Encrypt the packet.
*Jan  2 18:39:46:353 2015 RTA IKE/7/Packet: Sending packet to 2.2.2.1 remote port
500, local port 500.
*Jan  2 18:39:46:353 2015 RTA IKE/7/Packet:
  I-Cookie: 2f3a7dd5255e195f
  R-Cookie: 7135150829533058
  next payload: HASH
  version: ISAKMP Version 1.0
  exchange mode: Quick
  flags: ENCRYPT
  message ID: 36a08563
  length: 52
*Jan  2 18:39:46:353 2015 RTA IKE/7/Packet: Sending an IPv4 packet.
*Jan  2 18:39:51:083 2015 RTA IPSEC/7/packet:
--- Sent IPsec packet ---
*Jan  2 18:39:51:083 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Src : 192.168.1.2 Dst : 192.168.2.2 SPI : 3011773065
*Jan  2 18:39:51:083 2015 RTA IPSEC/7/packet:
Added IP fast forwarding entry.
*Jan  2 18:39:51:083 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  2 18:39:51:083 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-encryption.
*Jan  2 18:39:51:083 2015 RTA IPSEC/7/packet:
Outbound IPsec ESP processing: Encryption succeeded, anti-replay SN is 1.
*Jan  2 18:39:51:083 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Sent packet back to IP forwarding.
*Jan  2 18:39:51:084 2015 RTA IPSEC/7/packet:
```

```
--- Received IPsec(ESP) packet, Data length : 120 ---
*Jan  2 18:39:51:084 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Src : 2.2.2.1 Dst : 1.1.1.1 SPI : 4206531119
*Jan  2 18:39:51:084 2015 RTA IPSEC/7/packet:
Added IP fast forwarding entry.
*Jan  2 18:39:51:084 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  2 18:39:51:084 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-decryption.
*Jan  2 18:39:51:084 2015 RTA IPSEC/7/packet:
Inbound IPsec ESP processing: Authentication succeeded.
*Jan  2 18:39:51:084 2015 RTA IPSEC/7/packet:
Inbound IPsec ESP processing: Decryption succeeded.
*Jan  2 18:39:51:084 2015 RTA IPSEC/7/packet:
IPsec processing: Tunnel mode.
*Jan  2 18:39:51:084 2015 RTA IPSEC/7/packet:
Inbound ESP IPsec processing: Sent packet back to IP forwarding. Pkt len is 60.
*Jan  2 18:39:51:084 2015 RTA IPSEC/7/packet:
IPsec has already pre-process this packet
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
--- Sent packet by IPsec fast forwarding ---
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Src : 192.168.1.2 Dst : 192.168.2.2 SPI : 3011773065
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-encryption.
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
Outbound IPsec ESP processing: Encryption succeeded, anti-replay SN is 2.
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Sent packet back to IP forwarding.
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
--- Received IPsec packet from fast forwarding, Protocl : 50---
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Src : 2.2.2.1 Dst : 1.1.1.1 SPI : 4206531119
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-decryption.
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
Inbound fast IPsec ESP processing: Authentication succeeded.
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
Inbound IPsec ESP processing: Decryption succeeded.
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
IPsec processing: Tunnel mode.
*Jan  2 18:39:52:097 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Sent packet back to IP forwarding.
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
--- Sent packet by IPsec fast forwarding ---
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Src : 192.168.1.2 Dst : 192.168.2.2 SPI : 3011773065
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-encryption.
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
Outbound IPsec ESP processing: Encryption succeeded, anti-replay SN is 3.
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Sent packet back to IP forwarding.
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
--- Received IPsec packet from fast forwarding, Protocl : 50---
```

```
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Src : 2.2.2.1 Dst : 1.1.1.1 SPI : 4206531119
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-decryption.
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
Inbound fast IPsec ESP processing: Authentication succeeded.
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
Inbound IPsec ESP processing: Decryption succeeded.
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
IPsec processing: Tunnel mode.
*Jan  2 18:39:53:111 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Sent packet back to IP forwarding.
```

关闭 debugging 开关：

```
<RTA>undo debugging all
All possible debugging has been turned off.
```

这样就可以看到 IKE 的交换过程，以及 IPsec 对数据包的加密处理过程。

实验任务二：配置 IPsec+IKE 野蛮模式

本实验任务要求在 RTA 和 RTB 之间建立隧道。SWA 作为 DHCP Server，为 RTA 分配地址。

步骤一：配置 IP 地址

根据表 3-3 配置各接口的地址。其中 PCA、PCB 的默认网关分别配置为 RTA 和 RTB。

表3-3 各设备接口 IP 地址

设备	接口	地址
RTA	GE0/0	192.168.1.1/24
	GE0/1	自动获取
RTB	GE0/0	192.168.2.1/24
	GE0/1	2.2.2.1/24
SWA	VLAN1	1.1.1.2/24
	VLAN2	2.2.2.2/24
PCA	以太口	192.168.1.2/24
PCB	以太口	192.168.2.2/24

步骤二：清除所有 IPsec 和 IKE 配置

在 RTA 和 RTB 上清楚所有 IPsec 和 IKE 配置，使用命令如：

```
[RTA]interface GigabitEthernet 0/1
```

```
[RTA-GigabitEthernet0/1]undo ipsec apply policy
[RTA]undo ipsec policy policy1
[RTA]undo ipsec transform-set tran1
[RTA]undo ike profile profile1
[RTA]undo ike keychain keychain1
[RTA]undo ike proposal 1
[RTA]undo acl advanced 3000

[RTB]interface GigabitEthernet 0/1
[RTB-GigabitEthernet0/1]undo ipsec apply policy
[RTB]undo ipsec policy policy1
[RTB]undo ipsec transform-set tran1
[RTB]undo ike profile profile1
[RTB]undo ike keychain keychain1
[RTB]undo ike proposal 1
[RTB]undo acl advanced 3000
```

步骤三：配置公网连接

在 SWA 上配置 DHCP Server。设置 RTA 从 SWA 动态获得 IP 地址和默认路由。

```
[SWA]dhcp enable
[SWA]dhcp server ip-pool 1
[SWA-dhcp-pool-1]network 1.1.1.0 mask 255.255.255.0
[SWA-dhcp-pool-1]gateway-list 1.1.1.2
[SWA-dhcp-pool-1]quit


[RTA]undo ospf 1
Warning : Undo OSPF process? [Y/N]:y
[RTA]undo ip route-static 192.168.2.0 255.255.255.0
[RTA]interface GigabitEthernet0/1
[RTA-GigabitEthernet0/1] ip address dhcp-alloc
```

在 RTA 上查看路由，可见已经从 SWA 获得地址和默认路由：

```
[RTA]display ip routing-table

Destinations : 19      Routes : 19

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/0          Static  70  0          1.1.1.2         GE0/1
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.0/24         Direct  0   0          1.1.1.1         GE0/1
1.1.1.0/32         Direct  0   0          1.1.1.1         GE0/1
1.1.1.1/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.255/32       Direct  0   0          1.1.1.1         GE0/1
2.2.2.0/24         O_INTRA 10  2          1.1.1.2         GE0/1
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
192.168.1.0/24     Direct  0   0          192.168.1.1     GE0/0
192.168.1.0/32     Direct  0   0          192.168.1.1     GE0/0
192.168.1.1/32     Direct  0   0          127.0.0.1       InLoop0
192.168.1.255/32   Direct  0   0          192.168.1.1     GE0/0
192.168.2.0/24     Static  60  0          1.1.1.2         GE0/1
224.0.0.0/4        Direct  0   0          0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0          0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
```

在 PCA 上验证 PCA 与 PCB 之间的连通性。由于 SWA 没有私网路由，应该是无法连通的：

```
C:\Users\administrator>ping 192.168.2.2
```

```
正在 Ping 192.168.2.2 具有 32 字节的数据:
请求超时。
请求超时。
请求超时。
请求超时。

192.168.2.2 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 0，丢失 = 4 (100% 丢失)，
```

步骤四：配置 IKE proposal

```
[RTA]ike proposal 1
[RTA-ike-proposal-1]authentication-method pre-share
[RTA-ike-proposal-1]authentication-algorithm md5
[RTA-ike-proposal-1]encryption-algorithm 3des-cbc
[RTA-ike-proposal-1]quit

[RTB]ike proposal 1
[RTB-ike-proposal-1]authentication-method pre-share
[RTB-ike-proposal-1]authentication-algorithm md5
[RTB-ike-proposal-1]encryption-algorithm 3des-cbc
[RTB-ike-proposal-1]quit
```

步骤五：配置 IKE 身份信息

```
[RTA]ike identity  fqdn rta

[RTB]ike identity  fqdn rtb
```

步骤六：配置 IKE keychain

```
[RTA]ike keychain keychain1
[RTA-ike-keychain-keychain1]pre-shared-key address 2.2.2.1 255.255.255.0 key
simple h3c
[RTA-ike-keychain-keychain1]quit

[RTB]ike keychain keychain1
[RTB-ike-keychain-keychain1]pre-shared-key hostname rta 255.255.255.0 key simple
h3c
[RTB-ike-keychain-keychain1]quit
```

步骤七：配置 IKE profile

配置 IKE profile，并设置 IKE 第一阶段协商模式为野蛮模式：

```
[RTA]ike profile profile1
[RTA-ike-profile-profile1]exchange-mode aggressive
[RTA-ike-profile-profile1]match remote identity fqdn rtb
[RTA-ike-profile-profile1]keychain keychain1
[RTA-ike-profile-profile1]proposal 1
[RTA-ike-profile-profile1]quit

[RTB]ike profile profile1
[RTB-ike-profile-profile1]exchange-mode aggressive
[RTB-ike-profile-profile1]match remote identity fqdn rta
[RTB-ike-profile-profile1]keychain keychain1
[RTB-ike-profile-profile1]proposal 1
[RTB-ike-profile-profile1]quit
```

步骤八：配置安全 ACL

安全 ACL 应匹配 192.168.1.0/24 网段与 192.168.2.0/24 网段之间的数据流。

```
[RTA]acl advanced 3000
[RTA-acl-ipv4-adv-3000] rule 0 permit ip source 192.168.1.0 0.0.0.255 destination
192.168.2.0 0.0.0.255
```

```
[RTA-acl-ipv4-adv-3001]quit

[RTB]acl advanced 3000
[RTB-acl-ipv4-adv-3000] rule 0 permit ip source 192.168.2.0 0.0.0.255 destination
192.168.1.0 0.0.0.255
[RTB-acl-ipv4-adv-3000]quit
```

步骤九：配置 IPsec 安全提议

```
[RTA]ipsec transform-set tran1
[RTA-ipsec-proposal-prop1]esp authentication-algorithm sha1
[RTA-ipsec-proposal-prop1]esp encryption-algorithm aes-cbc-128
[RTA-ipsec-transform-set-tran1]quit

[RTB]ipsec transform-set tran1
[RTB-ipsec-transform-set-tran1]esp authentication-algorithm sha1
[RTB-ipsec-transform-set-tran1]esp encryption-algorithm aes-cbc-128
[RTB-ipsec-transform-set-tran1]quit
```

步骤十：配置并应用 IPsec 安全策略

配置 IPsec 安全策略，并将其应用于通往对方的物理接口上：

```
[RTA]ipsec policy policy1 1 isakmp
[RTA-ipsec-policy-isakmp-policy1-1]remote-address 2.2.2.1
[RTA-ipsec-policy-isakmp-policy1-1]security acl 3000
[RTA-ipsec-policy-isakmp-policy1-1]transform-set tran1
[RTA-ipsec-policy-isakmp-policy1-1]ike-profile profile1
[RTA-ipsec-policy-isakmp-policy1-1]quit
[RTA]interface GigabitEthernet 0/1
[RTA-GigabitEthernet0/1]ipsec apply policy policy1
[RTA-GigabitEthernet0/1]quit
```

RTB 作为响应方，无法获取对端的 IP 地址，需要配置成模板形式。

```
[RTB]ipsec policy-template templete1 1
[RTB-ipsec-policy-template-templete1-1]security acl 3000
[RTB-ipsec-policy-template-templete1-1]transform-set tran1
[RTB-ipsec-policy-template-templete1-1]ike-profile proflie1
[RTB]ipsec policy policy1 1 isakmp template templete1
[RTB]interface GigabitEthernet 0/1
[RTB-GigabitEthernet0/1]ipsec apply policy policy1
[RTB-GigabitEthernet0/1]quit
```

步骤十一：检验配置

在 RTA 和 RTB 上用 display 命令检查配置参数：

```
<RTA>display ike proposal
 Priority Authentication Authentication Encryption  Diffie-Hellman Duration
          method      algorithm    algorithm      group     (seconds)
-------------------------------------------------------------------------
 1       PRE-SHARED-KEY    MD5       3DES-CBC    Group 1       86400
 default  PRE-SHARED-KEY    SHA1      DES-CBC     Group 1       86400

<RTA>dis ipsec transform-set
IPsec transform set: tran1
  State: complete
  Encapsulation mode: tunnel
  Transform: ESP
  ESP protocol:
    Integrity: SHA1
    Encryption: AES-CBC-128

    <RTA>display ipsec policy
```

```
    ------------------------------------------
    IPsec Policy: policy1
    Interface: GigabitEthernet0/1
    ------------------------------------------

      ----------------------------
      Sequence number: 1
      Mode: ISAKMP
      ----------------------------
      Security data flow: 3000
      Selector mode: standard
      Local address:
      Remote address: 2.2.2.1
      Transform set:  tran1
      IKE profile: profile1
      SA duration(time based): 3600 seconds
      SA duration(traffic based): 1843200 kilobytes
  SA idle time:

<RTB>display ike proposal
 Priority Authentication Authentication Encryption  Diffie-Hellman Duration
            method      algorithm    algorithm      group     (seconds)
------------------------------------------------------------------------------
 1       PRE-SHARED-KEY    MD5       3DES-CBC    Group 1       86400
 default  PRE-SHARED-KEY    SHA1      DES-CBC    Group 1       86400

<RTB>display ipsec transform-set
IPsec transform set: tran1
  State: complete
  Encapsulation mode: tunnel
  Transform: ESP
  ESP protocol:
    Integrity: SHA1
    Encryption: AES-CBC-128

<RTB>display ipsec policy-template
------------------------------------------
IPsec Policy Template: templete1
------------------------------------------

  ----------------------------
  Sequence number: 1
  ----------------------------
  Security data flow : 3000
  IKE profile: proflie1
  Remote address:
  Transform set: tran1
  IPsec SA local duration(time based):
  IPsec SA local duration(traffic based):

<RTB>display ipsec policy
------------------------------------------
IPsec Policy: policy1
Interface: GigabitEthernet0/1
------------------------------------------

  ----------------------------
  Sequence number: 1
  Mode: Template
  ----------------------------
  Policy template name: templete1
--------------------------
```

步骤十二：检验隧道工作状况

从 PCA 检测与 PCB 的连通性：

```
C:\Users\administrator>ping 192.168.2.2

正在 Ping 192.168.2.2 具有 32 字节的数据:
请求超时。
来自 192.168.2.2 的回复: 字节=32 时间=1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126

192.168.2.2 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 3，丢失 = 1 (25% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms，最长 = 1ms，平均 = 0ms
```

可见除第一个 ICMP Echo Request 包被报告超时之外，其他的都成功收到 Echo Reply 包。这是因为第一个包触发了 IKE 协商，在 IPsec SA 成功建立之前，这个包无法得到 IPsec 服务，只能被丢弃。而 IPsec SA 很快就成功建立了，后续的包也就可以顺利到达目的。

在 RTA 与 RTB 上查看 IPsec/IKE 相关信息：

```
<RTA>display ike sa
   Connection-ID   Remote              Flag        DOI
-----------------------------------------------------------------------------
   10              2.2.2.1             RD          IPSEC
Flags:
RD--READY RL--REPLACED FD-FADING

<RTA>display ike sa verbose
  ----------------------------------------------
  Connection ID: 10
  Outside VPN:
  Inside VPN:
  Profile: profile1
  Transmitting entity: Initiator
  ----------------------------------------------
  Local IP: 1.1.1.1
  Local ID type: FQDN
  Local ID: rta

  Remote IP: 2.2.2.1
  Remote ID type: FQDN
  Remote ID: rtb

  Authentication-method: PRE-SHARED-KEY
  Authentication-algorithm: MD5
  Encryption-algorithm: 3DES-CBC

  Life duration(sec): 86400
  Remaining key duration(sec): 86366
  Exchange-mode: Aggressive
  Diffie-Hellman group: Group 1
  NAT traversal: Not detected
```

看到 IKE 的协商模式是野蛮模式（Aggressive）

```
<RTA>display ipsec sa
------------------------------
Interface: GigabitEthernet0/1
------------------------------

  ----------------------------
```

```
  IPsec policy: policy1
  Sequence number: 1
  Mode: isakmp
  ----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect forward secrecy:
    Path MTU: 1427
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.1
    Flow:
    sour addr: 192.168.1.0/255.255.255.0  port: 0  protocol: ip
    dest addr: 192.168.2.0/255.255.255.0  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 3767621453 (0xe091574d)
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843199/3561
      Max received sequence-number: 3
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: active

    [Outbound ESP SAs]
      SPI: 2842447130 (0xa96c4d1a)
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843199/3561
      Max sent sequence-number: 3
      UDP encapsulation used for NAT traversal: N
      Status: active

<RTB>display ike sa
    Connection-ID   Remote              Flag        DOI
-----------------------------------------------------------------------------
    10            1.1.1.1             RD          IPSEC
Flags:
RD--READY RL--REPLACED FD-FADING

<RTB>display ike sa verbose
   ------------------------------------------------
   Connection ID: 10
   Outside VPN:
   Inside VPN:
   Profile: profile1
   Transmitting entity: Responder
   ------------------------------------------------
   Local IP: 2.2.2.1
   Local ID type: FQDN
   Local ID: rtb

   Remote IP: 1.1.1.1
   Remote ID type: FQDN
   Remote ID: rta

   Authentication-method: PRE-SHARED-KEY
   Authentication-algorithm: MD5
   Encryption-algorithm: 3DES-CBC

   Life duration(sec): 86400
   Remaining key duration(sec): 86342
   Exchange-mode: Aggressive
```

```
  Diffie-Hellman group: Group 1
  NAT traversal: Not detected
```

可见 ISAKMP SA 是通过 IKE 野蛮模式协商生成的。

```
<RTB>display ipsec sa
-------------------------------
Interface: GigabitEthernet0/1
-------------------------------

  ----------------------------
  IPsec policy: policy1
  Sequence number: 1
  Mode: template
  ----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect forward secrecy:
    Path MTU: 1427
    Tunnel:
        local  address: 2.2.2.1
        remote address: 1.1.1.1
    Flow:
    sour addr: 192.168.2.0/255.255.255.0  port: 0  protocol: ip
    dest addr: 192.168.1.0/255.255.255.0  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 2842447130 (0xa96c4d1a)
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843199/3536
      Max received sequence-number: 3
      Anti-replay check enable: Y
      Anti-replay window size: 64
      UDP encapsulation used for NAT traversal: N
      Status: active

    [Outbound ESP SAs]
      SPI: 3767621453 (0xe091574d)
      Transform set: ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843199/3536
      Max sent sequence-number: 3
      UDP encapsulation used for NAT traversal: N
      Status: active
```

步骤十三：观察 IPsec 工作过程

为了了解 IKE 和 IPsec 协商和加密操作过程，首先清除 IPsec SA 和 ISAKMP SA，中断 IPsec 隧道，以便重新观察整个过程：

```
<RTA>reset ike sa
<RTA>reset ipsec sa

<RTB>reset ike sa
<RTB>reset ipsec sa
```

打开 debugging 开关：

```
<RTA>terminal monitor
% Current terminal monitor is on
<RTA>terminal debugging
% Current terminal debugging is on
<RTA>debugging ike exchange
<RTA>debugging ipsec packet
```

在 PCA 上 ping PCB，重新触发 IPsec 隧道建立：

```
C:\Users\administrator>ping 192.168.2.2

正在 Ping 192.168.2.2 具有 32 字节的数据:
请求超时。
来自 192.168.2.2 的回复: 字节=32 时间=1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126

192.168.2.2 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 3，丢失 = 1 (25% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms，最长 = 1ms，平均 = 0ms
```

观察 debugging 输出信息，分析其过程：

```
<RTA>*Jan  2 19:05:04:444 2015 RTA IPSEC/7/packet:
Failed to find SA by SP.
*Jan  2 19:05:04:444 2015 RTA IPSEC/7/packet:
The reason of dropping packet is no available IPsec tunnel.
*Jan  2 19:05:04:445 2015 RTA IKE/7/Packet:   Encryption algorithm is 3DES-CBC.
*Jan  2 19:05:04:445 2015 RTA IKE/7/Packet:   Hash algorithm is HMAC-MD5.
*Jan  2 19:05:04:445 2015 RTA IKE/7/Packet:   DH group 1.
*Jan  2 19:05:04:445 2015 RTA IKE/7/Packet:   Authentication method is Pre-shared
key.
*Jan  2 19:05:04:445 2015 RTA IKE/7/Packet:   Lifetime type is in seconds.
*Jan  2 19:05:04:445 2015 RTA IKE/7/Packet:   Life duration is 86400.
*Jan  2 19:05:04:445 2015 RTA IKE/7/Packet: Construct transfrom payload for
transform 1.
*Jan  2 19:05:04:445 2015 RTA IKE/7/Packet: Construct SA payload.
*Jan  2 19:05:04:447 2015 RTA IKE/7/Packet: Construct KE payload.
*Jan  2 19:05:04:447 2015 RTA IKE/7/Packet: Construct NONCE payload.
*Jan  2 19:05:04:447 2015 RTA IKE/7/Packet: Local ID type: FQDN (2).
*Jan  2 19:05:04:447 2015 RTA IKE/7/Packet: Local ID value: rta.
*Jan  2 19:05:04:447 2015 RTA IKE/7/Packet: Construct ID payload.
*Jan  2 19:05:04:447 2015 RTA IKE/7/Packet: Construct DPD vendor ID payload.
*Jan  2 19:05:04:447 2015 RTA IKE/7/Packet: Construct NAT-T rfc3947 vendor ID
payload.
*Jan  2 19:05:04:448 2015 RTA IKE/7/Packet: Construct NAT-T draft3 vendor ID
payload.
*Jan  2 19:05:04:448 2015 RTA IKE/7/Packet: Construct NAT-T draft2 vendor ID
payload.
*Jan  2 19:05:04:448 2015 RTA IKE/7/Packet: Construct NAT-T draft1 vendor ID
payload.
*Jan  2 19:05:04:448 2015 RTA IKE/7/Packet: Sending packet to 2.2.2.1 remote port
500, local port 500.
*Jan  2 19:05:04:448 2015 RTA IKE/7/Packet:
  I-Cookie: aba6e824c77740da
  R-Cookie: 0000000000000000
  next payload: SA
  version: ISAKMP Version 1.0
  exchange mode: Aggressive
  flags:
  message ID: 0
  length: 316
*Jan  2 19:05:04:448 2015 RTA IKE/7/Packet: Sending an IPv4 packet.
*Jan  2 19:05:04:452 2015 RTA IKE/7/Packet: Received packet from 2.2.2.1 source
port 500 destination port 500.
*Jan  2 19:05:04:452 2015 RTA IKE/7/Packet:
  I-Cookie: aba6e824c77740da
  R-Cookie: bdbebcfab080d21b
  next payload: SA
  version: ISAKMP Version 1.0
```

```
  exchange mode: Aggressive
  flags:
  message ID: 0
  length: 316
*Jan  2 19:05:04:452 2015 RTA IKE/7/Packet: Received ISAKMP Security Association
Payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Received ISAKMP Key Exchange Payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Received ISAKMP Nonce Payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Received ISAKMP Identification
Payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Received ISAKMP Vendor ID Payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Received ISAKMP Vendor ID Payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Received ISAKMP NAT-D Payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Received ISAKMP NAT-D Payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Received ISAKMP Hash Payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Process NONCE payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Process KE payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Process ID payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Peer ID type: FQDN (2).
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Peer ID value: FQDN rtb.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Process SA payload.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet: Check ISAKMP transform 1.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet:   Encryption algorithm is 3DES-CBC.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet:   HASH algorithm is HMAC-MD5.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet:   DH group is 1.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet:   Authentication method is Pre-shared
key.
*Jan  2 19:05:04:453 2015 RTA IKE/7/Packet:   Lifetime type is 1.
*Jan  2 19:05:04:454 2015 RTA IKE/7/Packet:   Life duration is 86400.
*Jan  2 19:05:04:454 2015 RTA IKE/7/Packet: Attributes is acceptable.
*Jan  2 19:05:04:454 2015 RTA IKE/7/Packet: Process vendor ID payload.
*Jan  2 19:05:04:454 2015 RTA IKE/7/Packet: Received 2 NAT-D payload.
*Jan  2 19:05:04:455 2015 RTA IKE/7/Packet: Verify HASH payload.
*Jan  2 19:05:04:456 2015 RTA IKE/7/Packet: HASH:
 91cce48a ce1f09d0 71a80614 a3ca1f12
*Jan  2 19:05:04:456 2015 RTA IKE/7/Packet: Construct NAT-D payload.
*Jan  2 19:05:04:456 2015 RTA IKE/7/Packet: HASH:
 576339a6 ff00e278 d62fe6bb 711fec02
*Jan  2 19:05:04:456 2015 RTA IKE/7/Packet: Construct authentication by
pre-shared-key.
*Jan  2 19:05:04:456 2015 RTA IKE/7/Packet: Construct INITIAL-CONTACT payload.
*Jan  2 19:05:04:456 2015 RTA IKE/7/Packet: Encrypt the packet.
*Jan  2 19:05:04:457 2015 RTA IKE/7/Packet: Sending packet to 2.2.2.1 remote port
500, local port 500.
*Jan  2 19:05:04:457 2015 RTA IKE/7/Packet:
  I-Cookie: aba6e824c77740da
  R-Cookie: bdbebcfab080d21b
  next payload: NAT-D
  version: ISAKMP Version 1.0
  exchange mode: Aggressive
  flags: ENCRYPT
  message ID: 0
  length: 116
*Jan  2 19:05:04:457 2015 RTA IKE/7/Packet: Sending an IPv4 packet.
*Jan  2 19:05:04:458 2015 RTA IKE/7/Packet: Set attributes according to phase 2
transform.
*Jan  2 19:05:04:458 2015 RTA IKE/7/Packet:   Encapsulation mode is Tunnel.
*Jan  2 19:05:04:458 2015 RTA IKE/7/Packet:   in seconds
*Jan  2 19:05:04:459 2015 RTA IKE/7/Packet:   Life duration is 3600.
*Jan  2 19:05:04:459 2015 RTA IKE/7/Packet:   in kilobytes
*Jan  2 19:05:04:459 2015 RTA IKE/7/Packet:   Life duration is 1843200.
*Jan  2 19:05:04:459 2015 RTA IKE/7/Packet:   Authentication algorithm is
HMAC-SHA1.
*Jan  2 19:05:04:459 2015 RTA IKE/7/Packet:   Key length is 128 bytes.
*Jan  2 19:05:04:459 2015 RTA IKE/7/Packet:   Transform ID is AES-CBC.
```

```
*Jan  2 19:05:04:459 2015 RTA IKE/7/Packet: Construct transform 1.
*Jan  2 19:05:04:459 2015 RTA IKE/7/Packet: Construct IPsec proposal 1.
*Jan  2 19:05:04:459 2015 RTA IKE/7/Packet: Construct IPsec SA payload.
*Jan  2 19:05:04:460 2015 RTA IKE/7/Packet: Construct NONCE payload.
*Jan  2 19:05:04:460 2015 RTA IKE/7/Packet: Construct IPsec ID payload.
*Jan  2 19:05:04:460 2015 RTA IKE/7/Packet: Construct IPsec ID payload.
*Jan  2 19:05:04:460 2015 RTA IKE/7/Packet: Construct HASH(1) payload.
*Jan  2 19:05:04:460 2015 RTA IKE/7/Packet: Encrypt the packet.
*Jan  2 19:05:04:460 2015 RTA IKE/7/Packet: Sending packet to 2.2.2.1 remote port
500, local port 500.
*Jan  2 19:05:04:460 2015 RTA IKE/7/Packet:
  I-Cookie: aba6e824c77740da
  R-Cookie: bdbebcfab080d21b
  next payload: HASH
  version: ISAKMP Version 1.0
  exchange mode: Quick
  flags: ENCRYPT
  message ID: c62f5417
  length: 164
*Jan  2 19:05:04:460 2015 RTA IKE/7/Packet: Sending an IPv4 packet.
*Jan  2 19:05:04:465 2015 RTA IKE/7/Packet: Received packet from 2.2.2.1 source
port 500 destination port 500.
*Jan  2 19:05:04:465 2015 RTA IKE/7/Packet:
  I-Cookie: aba6e824c77740da
  R-Cookie: bdbebcfab080d21b
  next payload: HASH
  version: ISAKMP Version 1.0
  exchange mode: Quick
  flags: ENCRYPT
  message ID: c62f5417
  length: 164
*Jan  2 19:05:04:465 2015 RTA IKE/7/Packet: Decrypt the packet.
*Jan  2 19:05:04:466 2015 RTA IKE/7/Packet: Received ISAKMP Hash Payload.
*Jan  2 19:05:04:466 2015 RTA IKE/7/Packet: Received ISAKMP Security Association
Payload.
*Jan  2 19:05:04:466 2015 RTA IKE/7/Packet: Received ISAKMP Nonce Payload.
*Jan  2 19:05:04:466 2015 RTA IKE/7/Packet: Received ISAKMP Identification Payload
(IPsec DOI).
*Jan  2 19:05:04:466 2015 RTA IKE/7/Packet: Received ISAKMP Identification Payload
(IPsec DOI).
*Jan  2 19:05:04:466 2015 RTA IKE/7/Packet: Process HASH payload.
*Jan  2 19:05:04:466 2015 RTA IKE/7/Packet: Process IPsec SA payload.
*Jan  2 19:05:04:467 2015 RTA IKE/7/Packet: Check IPsec proposal 1.
*Jan  2 19:05:04:467 2015 RTA IKE/7/Packet: Parse transform 1.
*Jan  2 19:05:04:467 2015 RTA IKE/7/Packet:   Encapsulation mode is Tunnel.
*Jan  2 19:05:04:467 2015 RTA IKE/7/Packet:   Lifetime type is in seconds.
*Jan  2 19:05:04:467 2015 RTA IKE/7/Packet:   Life duration is 3600.
*Jan  2 19:05:04:467 2015 RTA IKE/7/Packet:   Lifetime type is in kilobytes.
*Jan  2 19:05:04:467 2015 RTA IKE/7/Packet:   Life duration is 1843200.
*Jan  2 19:05:04:467 2015 RTA IKE/7/Packet:   Authentication algorithm is
HMAC-SHA1.
*Jan  2 19:05:04:467 2015 RTA IKE/7/Packet:   Key length is 128 bytes.
*Jan  2 19:05:04:468 2015 RTA IKE/7/Packet:   Transform ID is AES-CBC.
*Jan  2 19:05:04:468 2015 RTA IKE/7/Packet: The proposal is acceptable.
*Jan  2 19:05:04:468 2015 RTA IKE/7/Packet: Process IPsec ID payload.
*Jan  2 19:05:04:468 2015 RTA IKE/7/Packet: Process IPsec ID payload.
*Jan  2 19:05:04:471 2015 RTA IKE/7/Packet: Construct HASH(3) payload.
*Jan  2 19:05:04:471 2015 RTA IKE/7/Packet: Encrypt the packet.
*Jan  2 19:05:04:472 2015 RTA IKE/7/Packet: Sending packet to 2.2.2.1 remote port
500, local port 500.
*Jan  2 19:05:04:472 2015 RTA IKE/7/Packet:
  I-Cookie: aba6e824c77740da
  R-Cookie: bdbebcfab080d21b
  next payload: HASH
  version: ISAKMP Version 1.0
```

```
  exchange mode: Quick
  flags: ENCRYPT
  message ID: c62f5417
  length: 52
*Jan  2 19:05:04:472 2015 RTA IKE/7/Packet: Sending an IPv4 packet.
*Jan  2 19:05:09:103 2015 RTA IPSEC/7/packet:
--- Sent IPsec packet ---
*Jan  2 19:05:09:103 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Src : 192.168.1.2 Dst : 192.168.2.2 SPI : 2842447130
*Jan  2 19:05:09:103 2015 RTA IPSEC/7/packet:
Added IP fast forwarding entry.
*Jan  2 19:05:09:103 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  2 19:05:09:103 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-encryption.
*Jan  2 19:05:09:103 2015 RTA IPSEC/7/packet:
Outbound IPsec ESP processing: Encryption succeeded, anti-replay SN is 1.
*Jan  2 19:05:09:103 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Sent packet back to IP forwarding.
*Jan  2 19:05:09:104 2015 RTA IPSEC/7/packet:
--- Received IPsec(ESP) packet, Data length : 120 ---
*Jan  2 19:05:09:104 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Src : 2.2.2.1 Dst : 1.1.1.1 SPI : 3767621453
*Jan  2 19:05:09:104 2015 RTA IPSEC/7/packet:
Added IP fast forwarding entry.
*Jan  2 19:05:09:104 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  2 19:05:09:104 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-decryption.
*Jan  2 19:05:09:104 2015 RTA IPSEC/7/packet:
Inbound IPsec ESP processing: Authentication succeeded.
*Jan  2 19:05:09:104 2015 RTA IPSEC/7/packet:
Inbound IPsec ESP processing: Decryption succeeded.
*Jan  2 19:05:09:104 2015 RTA IPSEC/7/packet:
IPsec processing: Tunnel mode.
*Jan  2 19:05:09:104 2015 RTA IPSEC/7/packet:
Inbound ESP IPsec processing: Sent packet back to IP forwarding. Pkt len is 60.
*Jan  2 19:05:09:104 2015 RTA IPSEC/7/packet:
IPsec has already pre-process this packet
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
--- Sent packet by IPsec fast forwarding ---
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Src : 192.168.1.2 Dst : 192.168.2.2 SPI : 2842447130
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-encryption.
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
Outbound IPsec ESP processing: Encryption succeeded, anti-replay SN is 2.
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Sent packet back to IP forwarding.
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
--- Received IPsec packet from fast forwarding, Protocl : 50---
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Src : 2.2.2.1 Dst : 1.1.1.1 SPI : 3767621453
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-decryption.
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
Inbound fast IPsec ESP processing: Authentication succeeded.
```

```
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
Inbound IPsec ESP processing: Decryption succeeded.
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
IPsec processing: Tunnel mode.
*Jan  2 19:05:10:117 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Sent packet back to IP forwarding.
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
--- Sent packet by IPsec fast forwarding ---
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Src : 192.168.1.2 Dst : 192.168.2.2 SPI : 2842447130
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-encryption.
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
Outbound IPsec ESP processing: Encryption succeeded, anti-replay SN is 3.
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Sent packet back to IP forwarding.
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
--- Received IPsec packet from fast forwarding, Protocl : 50---
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Src : 2.2.2.1 Dst : 1.1.1.1 SPI : 3767621453
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-decryption.
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
Inbound fast IPsec ESP processing: Authentication succeeded.
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
Inbound IPsec ESP processing: Decryption succeeded.
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
IPsec processing: Tunnel mode.
*Jan  2 19:05:11:131 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Sent packet back to IP forwarding.
```

关闭 debugging 开关：

```
<RTA>undo debugging all
All possible debugging has been turned off.
```

这样就可以看到 IKE 的交换过程，以及 IPsec 对数据包的加密处理过程。

3.5 实验中的命令列表

表3-4 实验命令列表

命令	描述
ike identity { address *address* **| dn | fqdn [** *fqdn-name* **] | user-fqdn [** *user-fqdn-name* **]**	配置本端身份信息
ike proposal *proposal-number*	创建IKE安全提议，并进入安全提议视图
encryption-algorithm { 3des-cbc | aes-cbc-128 | aes-cbc-192 | aes-cbc-256 | des-cbc	配置IKE安全协议采用的加密算法

命令	描述
authentication-method { dsa-signature | pre-share | rsa-signature	配置IKE安全协议采用的认证方法
authentication-algorithm { md5 | sha }	配置IKE安全协议采用的认证算法
ike keychain *keychain-name*	创建并进入一个IKE keychain视图
pre-shared-key { address *address* **[** *mask* | *mask-length* **] | hostname** *host-name* **} key { cipher** *cipher-key* **| simple** *simple-key* **}**	配置预共享密钥
match local address { *interface-type interface-number* **|** *address* **[vpn-instance** *vpn-name* **] }**	（keychain视图）限制IKE keychain的使用范围
ike profile *profile-name*	创建IKE profile，并进入IKE profile视图
exchange-mode { aggressive | main }	配置IKE第一阶段的协商模式
keychain *keychain-name*	指定采用预共享密钥认证时使用的IKE keychain
certificate domain *domain-name*	指定IKE协商采用数字签名认证时使用的PKI域
local-identity { address *address* **| dn | fqdn [** *fqdn-name* **] | user-fqdn [** *user-fqdn-name* **] }**	配置本端身份信息，用于在IKE认证协商阶段向对端标识自己的身份
proposal *proposal-number*	配置IKE profile引用的IKE提议
match remote { certificate *policy-name* **| identity { address** *address* **[** *mask* | *mask-length* **] | range** *low-address high-address* **[vpn-instance** *vpn-name* **] | fqdn** *fqdn-name* **| user-fqdn** *user-fqdn-name* **} }**	配置一条用于匹配对端身份的规则
match local address { *interface-type interface-number* | *address* **[vpn-instance** *vpn-name* **] }**	（ike profile视图）来限制IKE profile的使用范围
display ike proposal	显示每个IKE提议配置的参数
display ike sa [verbose [connection-id *connection-id* **| remote-address** *remote-address* **[vpn-instance** *vpn-name* **]]]**	显示当前IKE SA的信息
reset ike sa [connection-id *connection-id* **]**	清除IKE建立的安全隧道
debugging ike { all | error | event | packet }	调试IKE信息

命令	描述
ipsec transform-set *transform-set-name*	创建安全提议，并进入安全提议视图
protocol { **ah** | **ah-esp** | **esp** }	配置安全提议采用的安全协议
esp encryption-algorithm {3des-cbc | aes-cbc-128 | aes-cbc-192 | aes-cbc-256 | des-cbc | null }	配置ESP协议采用的加密算法
esp authentication-algorithm { **md5** | **sha1** }	配置ESP协议采用的认证算法
ah authentication-algorithm { **md5** | **sha1** }	配置AH协议采用的认证算法
encapsulation-mode { **transport** | **tunnel** }	配置安全协议对IP报文的封装形式
ipsec policy *policy-name seq-number* **isakmp**	创建一条安全策略，并进入安全策略视图
security acl *acl-number*	指定IPsec安全策略/IPsec安全策略模板引用的ACL
transform-set *transform-set-name*	指定IPsec安全策略/IPsec安全策略模板/IPsec安全框架所引用的IPsec安全提议
ike-profile *profile-name*	指定IPsec安全策略/IPsec安全策略模板引用的IKE profile
local-address *ip-address*	配置IPsec隧道的本端IP地址
remote-address *ip-address*	指定IPsec隧道的对端IP地址
sa duration { time-based *seconds* **| traffic-based** *kilobytes* **}**	配置IPsec SA的生存时间
ipsec apply policy *policy-name*	应用指定的安全策略组
display ipsec policy [*policy-name* **[** *seq-number* **]]**	显示安全策略的信息
display ipsec transform-set [*transform-set-name* **]**	显示安全提议的信息
display ipsec sa [brief | count | policy *policy-name* **[** *seq-number* **] | interface** *interface-type interface-number* **| remote** *ip-address* **]**	显示安全联盟的相关信息
display ipsec statistics [tunnel-id *tunnel-id* **]**	显示IPsec处理报文的统计信息
display ipsec tunnel	显示IPsec隧道的信息
reset ipsec sa [spi *spi-number* **| policy** *policy-name* **[** *seq-number* **] | remote** *ip-address* **]**	清除已经建立的安全联盟
debugging ipsec { all | error | packet [policy *policy-name* **[** *seq-number* **] | remote** *ip-address* **I spi *spi-number*] }**	调试IPsec信息

3.6 思考题

1. 如果在 RTA 和 RTB 上添加下列 IKE 提议配置并在 IKE profile 中引用，则协商出的 ISAKMP SA 将采用的验证算法是什么？

```
[RTA]ike proposal 10
[RTA-ike-proposal-10]encryption-algorithm aes-cbc 128
[RTA-ike-proposal-10]dh group2
[RTA-ike-proposal-10]authentication-algorithm md5
[RTA]ike profile profile1
[RTA-ike-profile-profile1]proposal 10

[RTB]ike proposal 20
[RTB-ike-proposal-20]encryption-algorithm aes-cbc 128
[RTB-ike-proposal-20]dh group2
[RTB-ike-proposal-20]authentication-algorithm md5
[RTB]ike profile profile1
[RTB-ike-profile-profile1]proposal 20
```

答：协商出的 ISAKMP SA 将采用的验证算法是 MD5，因为上述 IKE 提议配置将优先于默认 IKE 提议生效，因此默认的 SHA 算法没有被选中。

实验4 配置 IPsec 保护传统 VPN 数据

4.1 实验内容与目标

完成本实验，您应该能够：

- 配置 IPsec 保护 GRE 隧道
- 以 iNode 和路由器配合，用 IPsec 保护 L2TP 隧道

4.2 实验组网图

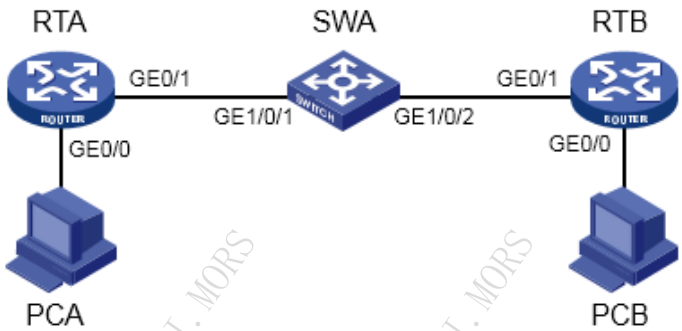


图4-1 配置 IPsec 保护传统 VPN 实验图

实验组网如图 4-1 所示。

4.3 实验设备与版本

本实验所需之主要设备器材如表 4-1 所示。

表4-1 实验设备和器材

名称和型号	版本	数量	描述
MSR36-20	CMW 7.1.049-R0106P08	2	
S5820V2-54QS-GE	CMW 7.1.045-R2311P03	1	
PC	Windows 7 SP1	2	
第5类UTP以太网连接线	--	4	
iNode客户端安装程序	V7.0 E0113	1	

4.4 实验过程

实验任务一：配置 GRE over IPsec

本实验任务要求在 RTA 和 RTB 之间建立 GRE 隧道，并在 RTA 和 RTB 直接建立 IPsec 隧道保护 GRE 隧道。

步骤一：搭建实验环境

连接设备。在 SWA 上配置 VLAN2，将接口 E1/0/2 加入 VLAN2。

```
[SWA]vlan 2
[SWA-vlan2]port GigabitEthernet 1/0/2
```

根据表 4-2 配置各物理接口的地址。其中 PCA、PCB 的默认网关分别配置为 RTA 和 RTB。

表4-2 各设备接口 IP 地址

设备	接口	地址
RTA	GE0/0	192.168.1.1/24
	GE0/1	1.1.1.1/24
	Tunnel0	192.168.3.1/30
RTB	GE0/0	192.168.2.1/24
	GE0/1	2.2.2.1/24
	Tunnel0	192.168.3.2/30
SWA	VLAN1	1.1.1.2/24
	VLAN2	2.2.2.2/24
PCA	以太口	192.168.1.2/24
PCB	以太口	192.168.2.2/24

步骤二：配置公网路由，检测公网连通性

查看 SWA 的路由表和端口状态，确认其工作正常：

```
<SWA>display ip interface brief
*down: administratively down
(s): spoofing  (l): loopback
Interface              Physical Protocol IP Address     Description
MGE0/0/0               down     down     --             --
Vlan1                  up       up       1.1.1.2        --
Vlan2                  up       up       2.2.2.2        --
```

在 RTA 和 RTB 上配置公网接口互通所需的 OSPF 协议：

```
[RTA] ospf 1
[RTA-ospf-1] area 0.0.0.0
[RTA-ospf-1-area-0.0.0.0]network 1.1.1.0 0.0.0.255
[RTA-ospf-1-area-0.0.0.0]quit
[RTA-ospf-1]quit
```

```
[SWA] ospf 1
[SWA-ospf-1] area 0.0.0.0
[SWA-ospf-1-area-0.0.0.0]network 1.1.1.0 0.0.0.255
[SWA-ospf-1-area-0.0.0.0]network 2.2.2.0 0.0.0.255
[SWA-ospf-1-area-0.0.0.0]quit
[SWA-ospf-1]quit

[RTB]ospf 1
[RTB-ospf-1] area 0.0.0.0
[RTB-ospf-1-area-0.0.0.0]network 2.2.2.0 0.0.0.255
[RTB-ospf-1-area-0.0.0.0]quit
[RTB-ospf-1]quit
```

在 RTA 上查看路由表，确认 OSPF 路由已正确学习：

```
[RTA]display ip routing-table

Destinations : 17      Routes : 17

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.0/24         Direct  0   0          1.1.1.1         GE0/1
1.1.1.0/32         Direct  0   0          1.1.1.1         GE0/1
1.1.1.1/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.255/32       Direct  0   0          1.1.1.1         GE0/1
2.2.2.0/24         O_INTRA 10  2          1.1.1.2         GE0/1
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
192.168.1.0/24     Direct  0   0          192.168.1.1     GE0/0
192.168.1.0/32     Direct  0   0          192.168.1.1     GE0/0
192.168.1.1/32     Direct  0   0          127.0.0.1       InLoop0
192.168.1.255/32   Direct  0   0          192.168.1.1     GE0/0
224.0.0.0/4        Direct  0   0          0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0          0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0

[RTB]display ip routing-table

Destinations : 17      Routes : 17

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.0/24         O_INTRA 10  2          2.2.2.2         GE0/1
2.2.2.0/24         Direct  0   0          2.2.2.1         GE0/1
2.2.2.0/32         Direct  0   0          2.2.2.1         GE0/1
2.2.2.1/32         Direct  0   0          127.0.0.1       InLoop0
2.2.2.255/32       Direct  0   0          2.2.2.1         GE0/1
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
192.168.2.0/24     Direct  0   0          192.168.2.1     GE0/0
192.168.2.0/32     Direct  0   0          192.168.2.1     GE0/0
192.168.2.1/32     Direct  0   0          127.0.0.1       InLoop0
192.168.2.255/32   Direct  0   0          192.168.2.1     GE0/0
224.0.0.0/4        Direct  0   0          0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0          0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
```

检测 RTA 与 RTB 的连通性。此时应该可以连通：

```
[RTA]ping 2.2.2.1
Ping 2.2.2.1 (2.2.2.1): 56 data bytes, press CTRL_C to break
56 bytes from 2.2.2.1: icmp_seq=0 ttl=254 time=4.121 ms
```

```
56 bytes from 2.2.2.1: icmp_seq=1 ttl=254 time=2.155 ms
56 bytes from 2.2.2.1: icmp_seq=2 ttl=254 time=1.996 ms
56 bytes from 2.2.2.1: icmp_seq=3 ttl=254 time=2.412 ms
56 bytes from 2.2.2.1: icmp_seq=4 ttl=254 time=2.062 ms

--- Ping statistics for 2.2.2.1 ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/std-dev = 1.996/2.549/4.121/0.799 ms
[RTA]%Dec 28 16:36:52:831 2014 RTA PING/6/PING_STATISTICS: Ping statistics for
2.2.2.1: 5 packets transmitted, 5 packets received, 0.0% packet loss, round-trip
min/avg/max/std-dev = 1.996/2.549/4.121/0.799 ms.
```

步骤三：配置 GRE 隧道接口

在 RTA 和 RTB 上建立隧道接口，配置隧道起点和终点。

```
[RTA]interface tunnel 0 mode gre
[RTA-Tunnel0]ip address 192.168.3.1 255.255.255.252
[RTA-Tunnel0]source 1.1.1.1
[RTA-Tunnel0]destination 2.2.2.1
[RTA-Tunnel0]quit

[RTB] interface tunnel 0 mode gre
[RTB-Tunnel0]ip address 192.168.3.2 255.255.255.252
[RTB-Tunnel0]source 2.2.2.1
[RTB-Tunnel0]destination 1.1.1.1
[RTB-Tunnel0]quit
```

步骤四：配置私网路由

在 RTA 和 RTB 上为私网配置 RIP 协议：

```
[RTA]rip 1
[RTA-rip-1]version 2
[RTA-rip-1]network 192.168.1.0
[RTA-rip-1]network 192.168.3.0
[RTA-rip-1]quit

[RTB]rip 1
[RTB-rip-1]version 2
[RTB-rip-1]network 192.168.2.0
[RTB-rip-1]network 192.168.3.0
[RTA-rip-1]quit
```

在 RTA 和 RTB 上查看路由表，此时私网路由应已经正确学习：

```
[RTA]display ip routing-table

Destinations : 22      Routes : 22

Destination/Mask   Proto   Pre Cost      NextHop         Interface
0.0.0.0/32         Direct  0   0         127.0.0.1       InLoop0
1.1.1.0/24         Direct  0   0         1.1.1.1         GE0/1
1.1.1.0/32         Direct  0   0         1.1.1.1         GE0/1
1.1.1.1/32         Direct  0   0         127.0.0.1       InLoop0
1.1.1.255/32       Direct  0   0         1.1.1.1         GE0/1
2.2.2.0/24         O_INTRA 10  2         1.1.1.2         GE0/1
127.0.0.0/8        Direct  0   0         127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0         127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0         127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1       InLoop0
192.168.1.0/24     Direct  0   0         192.168.1.1     GE0/0
192.168.1.0/32     Direct  0   0         192.168.1.1     GE0/0
192.168.1.1/32     Direct  0   0         127.0.0.1       InLoop0
192.168.1.255/32   Direct  0   0         192.168.1.1     GE0/0
```

```
192.168.2.0/24    RIP    100 1         192.168.3.2    Tun0
192.168.3.0/30    Direct  0   0         192.168.3.1    Tun0
192.168.3.0/32    Direct  0   0         192.168.3.1    Tun0
192.168.3.1/32    Direct  0   0         127.0.0.1      InLoop0
192.168.3.3/32    Direct  0   0         192.168.3.1    Tun0
224.0.0.0/4       Direct  0   0         0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0         0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0

[RTB]display ip routing-table

Destinations : 22      Routes : 22

Destination/Mask   Proto   Pre Cost      NextHop         Interface
0.0.0.0/32        Direct  0   0         127.0.0.1      InLoop0
1.1.1.0/24        O_INTRA 10  2         2.2.2.2        GE0/1
2.2.2.0/24        Direct  0   0         2.2.2.1        GE0/1
2.2.2.0/32        Direct  0   0         2.2.2.1        GE0/1
2.2.2.1/32        Direct  0   0         127.0.0.1      InLoop0
2.2.2.255/32      Direct  0   0         2.2.2.1        GE0/1
127.0.0.0/8       Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/32      Direct  0   0         127.0.0.1      InLoop0
127.0.0.1/32      Direct  0   0         127.0.0.1      InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
192.168.1.0/24    RIP    100 1         192.168.3.1    Tun0
192.168.2.0/24    Direct  0   0         192.168.2.1    GE0/0
192.168.2.0/32    Direct  0   0         192.168.2.1    GE0/0
192.168.2.1/32    Direct  0   0         127.0.0.1      InLoop0
192.168.2.255/32   Direct  0   0         192.168.2.1    GE0/0
192.168.3.0/24    Direct  0   0         192.168.3.2    Tun0
192.168.3.0/32    Direct  0   0         192.168.3.2    Tun0
192.168.3.2/32    Direct  0   0         127.0.0.1      InLoop0
192.168.3.255/32   Direct  0   0         192.168.3.2    Tun0
224.0.0.0/4       Direct  0   0         0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0         0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
```

在 PCA 上检测与 PCB 的连通性，此时应该可以连通：

```
C:\Users\administrator>ping 192.168.2.2

正在 Ping 192.168.2.2 具有 32 字节的数据:
来自 192.168.2.2 的回复: 字节=32 时间=8ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126

192.168.2.2 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms，最长 = 8ms，平均 = 2ms
```

步骤五：配置 IPsec 保护 GRE 隧道

配置 IPsec+IKE 主模式，使用预共享密钥方式，对 GRE 隧道封装数据进行保护。

```
[RTA]acl advanced 3000
[RTA-acl-ipv4-adv-3000] rule 0 permit ip source 1.1.1.1 0 destination 2.2.2.1 0
[RTA]ike proposal 1
[RTA-ike-proposal-1]authentication-method pre-share
[RTA-ike-proposal-1]authentication-algorithm md5
[RTA-ike-proposal-1]encryption-algorithm 3des-cbc
[RTA-ike-proposal-1]quit
[RTA]ike keychain keychain1
[RTA-ike-keychain-keychain1]pre-shared-key address 2.2.2.1 key simple h3c
```

```
[RTA-ike-keychain-keychain1]quit
[RTA]ike profile profile1
[RTA-ike-profile-profile1]local-identity address 1.1.1.1
[RTA-ike-profile-profile1]match remote identity address 2.2.2.1 255.255.255.0
[RTA-ike-profile-profile1]keychain keychain1
[RTA-ike-profile-profile1]proposal 1
[RTA-ike-profile-profile1]quit
[RTA]ipsec transform-set tran1
[RTA-ipsec-proposal-prop1]esp authentication-algorithm sha1
[RTA-ipsec-proposal-prop1]esp encryption-algorithm aes-cbc-128
[RTA-ipsec-transform-set-tran1]quit
[RTA]ipsec policy policy1 1 isakmp
[RTA-ipsec-policy-isakmp-policy1-1]remote-address 2.2.2.1
[RTA-ipsec-policy-isakmp-policy1-1]security acl 3000
[RTA-ipsec-policy-isakmp-policy1-1]transform-set tran1
[RTA-ipsec-policy-isakmp-policy1-1]ike-profile profile1
[RTA-ipsec-policy-isakmp-policy1-1]quit
[RTA]interface GigabitEthernet 0/1
[RTA-GigabitEthernet0/1]ipsec apply policy policy1
[RTA-GigabitEthernet0/1]quit


[RTB]acl advanced 3000
[RTB-acl-ipv4-adv-3000] rule 0 permit ip source 2.2.2.1 0 destination 1.1.1.1 0
[RTB]ike proposal 1
[RTB-ike-proposal-1]authentication-method pre-share
[RTB-ike-proposal-1]authentication-algorithm md5
[RTB-ike-proposal-1]encryption-algorithm 3des-cbc
[RTB-ike-proposal-1]quit
[RTB]ike keychain keychain1
[RTB-ike-keychain-keychain1]pre-shared-key address 1.1.1.1 255.255.255.0 key sim
ple h3c
[RTB-ike-keychain-keychain1]quit
[RTB]ike profile profile1
[RTB-ike-profile-profile1]local-identity address 2.2.2.1
[RTB-ike-profile-profile1]match remote identity address 1.1.1.1 255.255.255.0
[RTB-ike-profile-profile1]keychain keychain1
[RTB-ike-profile-profile1]proposal 1
[RTB-ike-profile-profile1]quit
[RTB]ipsec transform-set tran1
[RTB-ipsec-transform-set-tran1]esp authentication-algorithm sha1
[RTB-ipsec-transform-set-tran1]esp encryption-algorithm aes-cbc-128
[RTB-ipsec-transform-set-tran1]quit
[RTB]ipsec policy policy1 1 isakmp
[RTB-ipsec-policy-isakmp-policy1-1]remote-address 1.1.1.1
[RTB-ipsec-policy-isakmp-policy1-1]security acl 3000
[RTB-ipsec-policy-isakmp-policy1-1]transform-set tran1
[RTB-ipsec-policy-isakmp-policy1-1]ike-profile profile1
[RTB-ipsec-policy-isakmp-policy1-1]quit
[RTB]interface GigabitEthernet 0/1
[RTB-GigabitEthernet0/1]ipsec apply policy policy1
[RTB-GigabitEthernet0/1]quit
```

注意安全 ACL 匹配的是隧道源、目的 IP 地址之间的数据流。

步骤六：检验隧道工作状况

稍候一会儿，检查 RTA 上的路由表，应该仍然具有来自 RTB 的 RIP 路由：

```
<RTA>display ip routing-table

Destinations : 22      Routes : 22

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
```

```
1.1.1.0/24        Direct  0   0         1.1.1.1        GE0/1
1.1.1.0/32        Direct  0   0         1.1.1.1        GE0/1
1.1.1.1/32        Direct  0   0         127.0.0.1      InLoop0
1.1.1.255/32      Direct  0   0         1.1.1.1        GE0/1
2.2.2.0/24        O_INTRA 10  2         1.1.1.2        GE0/1
127.0.0.0/8       Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/32      Direct  0   0         127.0.0.1      InLoop0
127.0.0.1/32      Direct  0   0         127.0.0.1      InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
192.168.1.0/24    Direct  0   0         192.168.1.1    GE0/0
192.168.1.0/32    Direct  0   0         192.168.1.1    GE0/0
192.168.1.1/32    Direct  0   0         127.0.0.1      InLoop0
192.168.1.255/32  Direct  0   0         192.168.1.1    GE0/0
192.168.2.0/24    RIP     100 1         192.168.3.2    Tun0
192.168.3.0/30    Direct  0   0         192.168.3.1    Tun0
192.168.3.0/32    Direct  0   0         192.168.3.1    Tun0
192.168.3.1/32    Direct  0   0         127.0.0.1      InLoop0
192.168.3.3/32    Direct  0   0         192.168.3.1    Tun0
224.0.0.0/4       Direct  0   0         0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0         0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
```

同样检查 RTB 上的路由表，应该也具有来自 RTA 的 RIP 路由：

```
[RTB]display ip routing-table

Destinations : 22      Routes : 22

Destination/Mask   Proto   Pre Cost      NextHop         Interface
0.0.0.0/32        Direct  0   0         127.0.0.1      InLoop0
1.1.1.0/24        O_INTRA 10  2         2.2.2.2        GE0/1
2.2.2.0/24        Direct  0   0         2.2.2.1        GE0/1
2.2.2.0/32        Direct  0   0         2.2.2.1        GE0/1
2.2.2.1/32        Direct  0   0         127.0.0.1      InLoop0
2.2.2.255/32      Direct  0   0         2.2.2.1        GE0/1
127.0.0.0/8       Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/32      Direct  0   0         127.0.0.1      InLoop0
127.0.0.1/32      Direct  0   0         127.0.0.1      InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
192.168.1.0/24    RIP     100 1         192.168.3.1    Tun0
192.168.2.0/24    Direct  0   0         192.168.2.1    GE0/0
192.168.2.0/32    Direct  0   0         192.168.2.1    GE0/0
192.168.2.1/32    Direct  0   0         127.0.0.1      InLoop0
192.168.2.255/32  Direct  0   0         192.168.2.1    GE0/0
192.168.3.0/24    Direct  0   0         192.168.3.2    Tun0
192.168.3.0/32    Direct  0   0         192.168.3.2    Tun0
192.168.3.2/32    Direct  0   0         127.0.0.1      InLoop0
192.168.3.255/32  Direct  0   0         192.168.3.2    Tun0
224.0.0.0/4       Direct  0   0         0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0         0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
```

验证 PCA 与 PCB 之间的连通性，此时应该是可以连通的：

```
C:\Users\administrator>ping 192.168.2.2

正在 Ping 192.168.2.2 具有 32 字节的数据:
来自 192.168.2.2 的回复: 字节=32 时间=9ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间=1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间=1ms TTL=126

192.168.2.2 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位):
```

```
    最短 = 0ms，最长 = 9ms，平均 = 2ms
```

查看 RTA 与 RTB 的 IPsec/IKE 相关信息，可见：

```
<RTA>display ike sa
   Connection-ID   Remote              Flag        DOI
-----------------------------------------------------------------------
   18            2.2.2.1             RD          IPsec
Flags:
RD--READY RL--REPLACED FD-FADING

<RTA>display ike sa verbose
  ----------------------------------------------
  Connection ID: 18
  Outside VPN:
  Inside VPN:
  Profile: profile1
  Transmitting entity: Responder
  ----------------------------------------------
  Local IP: 1.1.1.1
  Local ID type: IPV4_ADDR
  Local ID: 1.1.1.1

  Remote IP: 2.2.2.1
  Remote ID type: IPV4_ADDR
  Remote ID: 2.2.2.1

  Authentication-method: PRE-SHARED-KEY
  Authentication-algorithm: MD5
  Encryption-algorithm: 3DES-CBC

  Life duration(sec): 86400
  Remaining key duration(sec): 86003
  Exchange-mode: Main
  Diffie-Hellman group: Group 1
  NAT traversal: Not detected

<RTA>display ipsec sa
------------------------------
Interface: GigabitEthernet0/1
------------------------------

  ----------------------------
  IPsec policy: policy1
  Sequence number: 1
  Mode: ISAKMP
  ----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect forward secrecy:
    Path MTU: 1427
    Tunnel:
        local  address: 1.1.1.1
        remote address: 2.2.2.1
    Flow:
        sour addr: 1.1.1.1/255.255.255.255  port: 0  protocol: ip
        dest addr: 2.2.2.1/255.255.255.255  port: 0  protocol: ip

    [Inbound ESP SAs]
      SPI: 3455755562 (0xcdfaa52a)
      Connection ID: 365072220161
      Transform set:  ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
      SA duration (kilobytes/sec): 1843200/3600
      SA remaining duration (kilobytes/sec): 1843198/3165
      Max received sequence-number: 19
```

```
    Anti-replay check enable: Y
    Anti-replay window size: 64
    UDP encapsulation used for NAT traversal: N
    Status: Active

  [Outbound ESP SAs]
    SPI: 2954508091 (0xb01a373b)
    Connection ID: 201863462912
    Transform set:  ESP-ENCRYPT-AES-CBC-128 ESP-AUTH-SHA1
    SA duration (kilobytes/sec): 1843200/3600
    SA remaining duration (kilobytes/sec): 1843198/3165
    Max sent sequence-number: 19
    UDP encapsulation used for NAT traversal: N
    Status: Active
```

在 RTA 上打开 debugging 开关：

```
<RTA>terminal monitor
% Current terminal monitor is on
<RTA>terminal debugging
% Current terminal debugging is on
<RTA>debug ike packet
<RTA>debug ipsec packet
```

在 RTA 上 ping RTB，同时观察 debugging 信息输出，检验路由器实际收发的报文：

```
C:\Users\administrator>ping 192.168.2.2

正在 Ping 192.168.2.2 具有 32 字节的数据:
来自 192.168.2.2 的回复: 字节=32 时间=9ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间=1ms TTL=126
来自 192.168.2.2 的回复: 字节=32 时间=1ms TTL=126

192.168.2.2 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms，最长 = 9ms，平均 = 2ms
```

RTA 上的输出信息如下：

```
<RTA>*Jan  3 09:10:38:430 2015 RTA IPSEC/7/packet:
--- Sent packet by IPsec fast forwarding ---
*Jan  3 09:10:38:430 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Src : 1.1.1.1 Dst : 2.2.2.1 SPI : 863282178
*Jan  3 09:10:38:430 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  3 09:10:38:430 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-encryption.
*Jan  3 09:10:38:430 2015 RTA IPSEC/7/packet:
Outbound IPsec ESP processing: Encryption succeeded, anti-replay SN is 11.
*Jan  3 09:10:38:430 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Sent packet back to IP forwarding.
*Jan  3 09:10:38:431 2015 RTA IPSEC/7/packet:
--- Received IPsec packet from fast forwarding, Protocl : 50---
*Jan  3 09:10:38:431 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Src : 2.2.2.1 Dst : 1.1.1.1 SPI : 881901192
*Jan  3 09:10:38:431 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  3 09:10:38:431 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-decryption.
```

```
*Jan  3 09:10:38:431 2015 RTA IPSEC/7/packet:
Inbound fast IPsec ESP processing: Authentication succeeded.
*Jan  3 09:10:38:431 2015 RTA IPSEC/7/packet:
Inbound IPsec ESP processing: Decryption succeeded.
*Jan  3 09:10:38:431 2015 RTA IPSEC/7/packet:
IPsec processing: Tunnel mode.
*Jan  3 09:10:38:431 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Sent packet back to IP forwarding.
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
--- Sent packet by IPsec fast forwarding ---
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Src : 1.1.1.1 Dst : 2.2.2.1 SPI : 863282178
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-encryption.
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
Outbound IPsec ESP processing: Encryption succeeded, anti-replay SN is 12.
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Sent packet back to IP forwarding.
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
--- Received IPsec packet from fast forwarding, Protocl : 50---
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Src : 2.2.2.1 Dst : 1.1.1.1 SPI : 881901192
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-decryption.
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
Inbound fast IPsec ESP processing: Authentication succeeded.
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
Inbound IPsec ESP processing: Decryption succeeded.
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
IPsec processing: Tunnel mode.
*Jan  3 09:10:39:427 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Sent packet back to IP forwarding.
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
--- Sent packet by IPsec fast forwarding ---
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Src : 1.1.1.1 Dst : 2.2.2.1 SPI : 863282178
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-encryption.
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
Outbound IPsec ESP processing: Encryption succeeded, anti-replay SN is 13.
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Sent packet back to IP forwarding.
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
--- Received IPsec packet from fast forwarding, Protocl : 50---
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Src : 2.2.2.1 Dst : 1.1.1.1 SPI : 881901192
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-decryption.
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
Inbound fast IPsec ESP processing: Authentication succeeded.
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
Inbound IPsec ESP processing: Decryption succeeded.
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
```

```
IPsec processing: Tunnel mode.
*Jan  3 09:10:40:441 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Sent packet back to IP forwarding.
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
--- Sent packet by IPsec fast forwarding ---
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Src : 1.1.1.1 Dst : 2.2.2.1 SPI : 863282178
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-encryption.
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
Outbound IPsec ESP processing: Encryption succeeded, anti-replay SN is 14.
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
Outbound IPsec processing: Sent packet back to IP forwarding.
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
--- Received IPsec packet from fast forwarding, Protocl : 50---
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Src : 2.2.2.1 Dst : 1.1.1.1 SPI : 881901192
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: ESP auth algorithm: SHA1, ESP encp algorithm:
AES-CBC-128.
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
Packet will be sent to CCF for sync-decryption.
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
Inbound fast IPsec ESP processing: Authentication succeeded.
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
Inbound IPsec ESP processing: Decryption succeeded.
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
IPsec processing: Tunnel mode.
*Jan  3 09:10:41:455 2015 RTA IPSEC/7/packet:
Inbound IPsec processing: Sent packet back to IP forwarding.
```

关闭 debugging 开关：

```
<RTA>undo debugging all
All possible debugging has been turned off.
```

可见路由器通过隧道发送了一些数据包。这些包的地址均不是 PC 的地址，这是由于所有包都被首先封装在 GRE 隧道中，再被封装在 IPsec 隧道中发送。

实验任务二：配置 L2TP over IPsec

本实验以 iNode 客户端与 RTB 建立 L2TP 隧道，并同时在 iNode 客户端和 RTB 上配置 IPsec 保护 L2TP 隧道。

执行本实验任务前，请清除上一实验任务的所有配置。

步骤一：搭建实验环境

连接设备。在 SWA 上配置 VLAN2，将接口 E1/0/2 加入 VLAN2。

```
[SWA]vlan 2
[SWA-vlan2]port GigabitEthernet 1/0/2
```

根据表 4-3 配置各接口的地址。其中 PCA、PCB 的默认网关分别配置为 RTA 和 RTB。

表4-3 各设备接口 IP 地址

设备	接口	地址
RTA	GE0/0	3.3.3.1/24
	GE0/1	1.1.1.1/24
RTB	GE0/0	192.168.2.1/24
	GE0/1	2.2.2.1/24
	Virtual-Template1	192.168.1.1/24
SWA	VLAN1	1.1.1.2/24
	VLAN2	2.2.2.2/24
PCA	以太口	3.3.3.2/24
PCB	以太口	192.168.2.1/24

步骤二：配置公网路由

在 RTA、SWA 和 RTB 上配置公网接口互通所需的 OSPF 协议：

```
[RTA]ospf 1
[RTA-ospf-1]area 0.0.0.0
[RTA-ospf-1-area-0.0.0.0]network 1.1.1.0 0.0.0.255
[RTA-ospf-1-area-0.0.0.0]network 3.3.3.0 0.0.0.255
[RTA-ospf-1-area-0.0.0.0]quit

[SWA]ospf 1
[SWA-ospf-1]area 0.0.0.0
[SWA-ospf-1-area-0.0.0.0]network 1.1.1.0 0.0.0.255
[SWA-ospf-1-area-0.0.0.0]network 2.2.2.0 0.0.0.255
[SWA-ospf-1-area-0.0.0.0]quit

[RTB]ospf 1
[RTB-ospf-1]area 0.0.0.0
[RTB-ospf-1-area-0.0.0.0]network 2.2.2.0 0.0.0.255
[RTB-ospf-1-area-0.0.0.0]quit
```

在各设备上查看路由表，确认 OSPF 路由已正确学习：

```
[RTA]display ip routing-table

Destinations : 17      Routes : 17

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.0/24         Direct  0   0          1.1.1.1         GE0/1
1.1.1.0/32         Direct  0   0          1.1.1.1         GE0/1
1.1.1.1/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.255/32       Direct  0   0          1.1.1.1         GE0/1
2.2.2.0/24         O_INTRA 10  2          1.1.1.2         GE0/1
3.3.3.0/24         Direct  0   0          3.3.3.1         GE0/0
3.3.3.0/32         Direct  0   0          3.3.3.1         GE0/0
3.3.3.1/32         Direct  0   0          127.0.0.1       InLoop0
3.3.3.255/32       Direct  0   0          3.3.3.1         GE0/0
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
```

```
127.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
224.0.0.0/4       Direct  0   0         0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0         0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0

[RTB]display ip routing-table

Destinations : 14      Routes : 14

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32        Direct  0   0         127.0.0.1      InLoop0
1.1.1.0/24        O_INTRA 10  2         2.2.2.2        GE0/1
2.2.2.0/24        Direct  0   0         2.2.2.1        GE0/1
2.2.2.0/32        Direct  0   0         2.2.2.1        GE0/1
2.2.2.1/32        Direct  0   0         127.0.0.1      InLoop0
2.2.2.255/32      Direct  0   0         2.2.2.1        GE0/1
3.3.3.0/24        O_INTRA 10  3         2.2.2.2        GE0/1
127.0.0.0/8       Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/32      Direct  0   0         127.0.0.1      InLoop0
127.0.0.1/32      Direct  0   0         127.0.0.1      InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
224.0.0.0/4       Direct  0   0         0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0         0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
```

步骤三：配置 LNS

在 RTB 上执行下列配置：

```
[RTB]ip pool 1 192.168.1.2 192.168.1.100
[RTB]domain abc.com
[RTB-isp-abc.com]authentication ppp local
[RTB-isp-abc.com]quit
[RTB]local-user vpdnuser class network
[RTB-luser-network-vpdnuser]password simple h3c
[RTB-luser-network-vpdnuser]service-type ppp
[RTB-luser-network-vpdnuser]quit
[RTB]interface Virtual-Template 1
[RTB-Virtual-Template1]ip address 192.168.1.1 255.255.255.0
[RTB-Virtual-Template1]ppp authentication-mode chap domain abc.com
[RTB-Virtual-Template1]remote address pool 1
[RTB-Virtual-Template1]quit
[RTB]l2tp enable
[RTB]l2tp-group 1 mode lns
[RTB-l2tp1]tunnel name LNS
[RTB-l2tp1]tunnel password simple aabbcc
[RTB-l2tp1]allow l2tp virtual-template 1 remote LAC
[RTB-l2tp1]quit
```

步骤四：安装 iNode 客户端

在 PCA 上安装 iNode 客户端。启动安装程序，跟随安装向导完成安装即可。

注意：

要使 iNode 客户端支持 L2TP 功能，在安装过程中必须确认安装虚拟网卡（Virtual NIC）。

当使用 iNode 客户端建立 L2TP 连接时，如果系统提示【Windows IPSEC Services(IPSEC Services or IPsec Policy Agent) is running, please stop it and try again.】，则说明系统内的 IPsec 服务

已经启动，需要关闭之。在【控制面板】->【管理工具】->【服务】中找到【IPSEC services】或者【IPsec Policy Agent】服务，将其禁用即可。

步骤五：在 iNode 客户端上配置 L2TP

启动 iNode 客户端程序，在其主界面窗口中单击菜单【文件】|【新建连接】，启动新建连接向导，如图 4-2 所示。

图4-2 进入新建连接向导

单击【下一步】，进入图 4-3 所示窗口，单击选定【L2TP IPsec VPN 协议】。

图4-3 选择认证协议

单击【下一步】，进入图 4-4 所示窗口，单击选定【普通连接】。

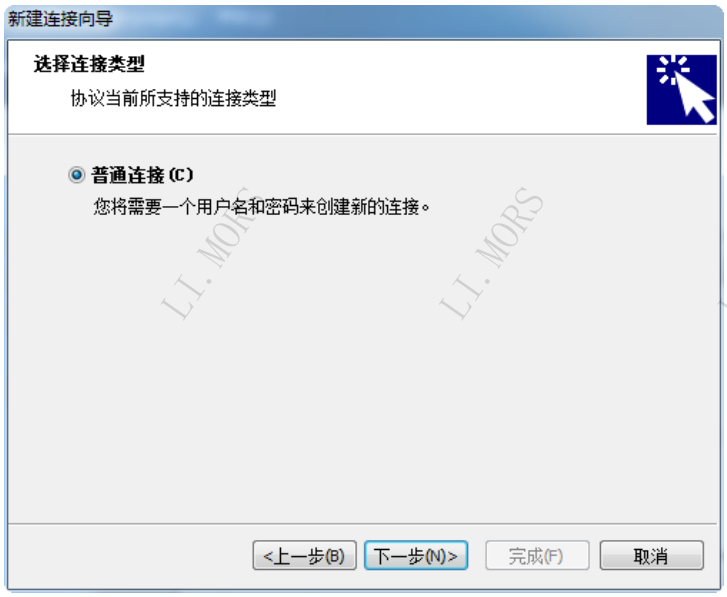


图4-4 选择连接类型

单击【下一步】，进入图 4-5 所示窗口，在【连接名】处输入一个连接名称，例如“我的 VPN 连接”，在【登录用户名】处输入用户名，在【登录密码】处输入密码。

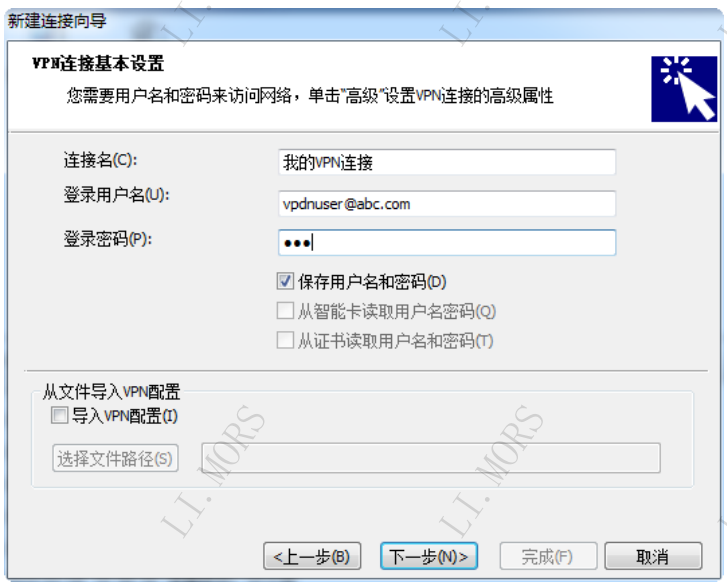


图4-5 设置用户名和密码

单击【下一步】，进入图 4-6 所示窗口，输入 LNS 服务器地址。

图4-6 VPN 连接基本设置

单击【高级】进入图 4-7 所示的窗口，进入【L2TP 设置】选项卡，输入隧道名称 LAC，选择认证模式为 CHAP，单击选定【使用隧道验证密码】并输入隧道验证密码 aabbcc。单击【确定】回到图 4-6 所示窗口。

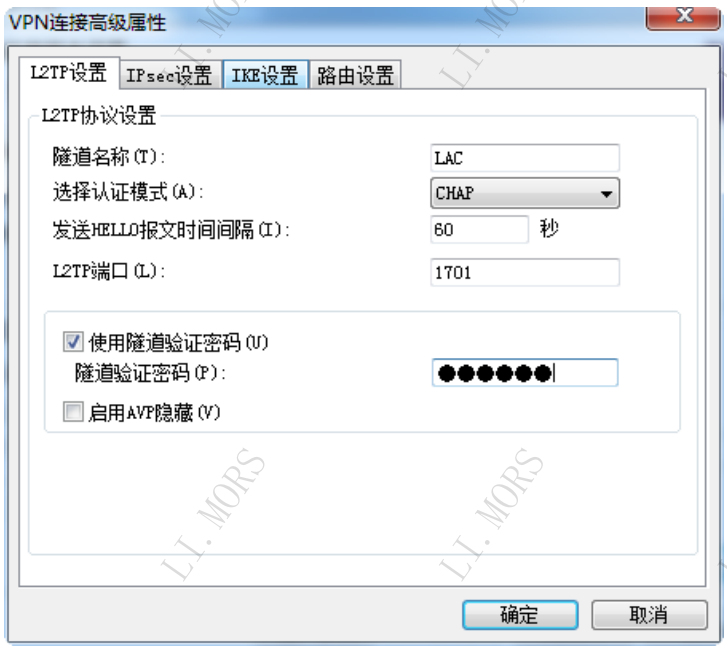


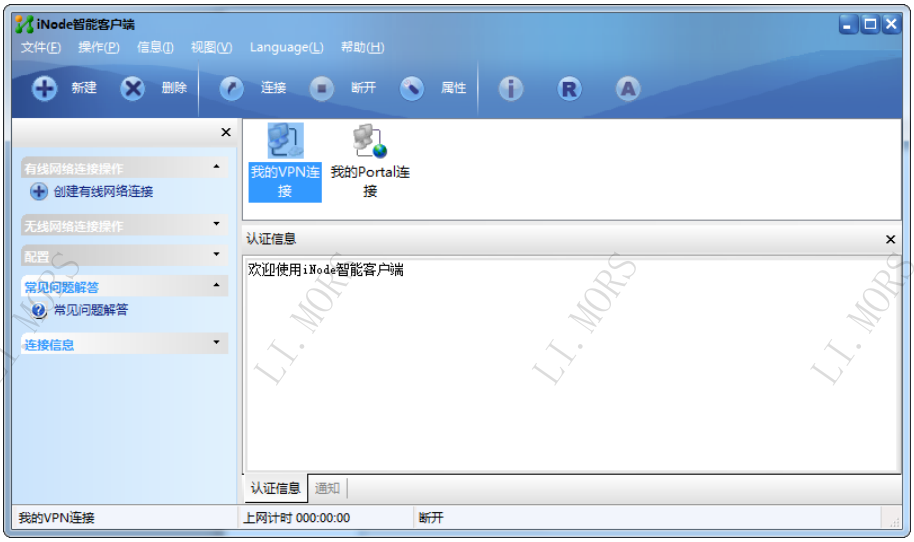
图4-7 VPN 连接高级属性

单击【下一步】进入图 4-8 所示的窗口，单击【创建】，即可创建新建连接。

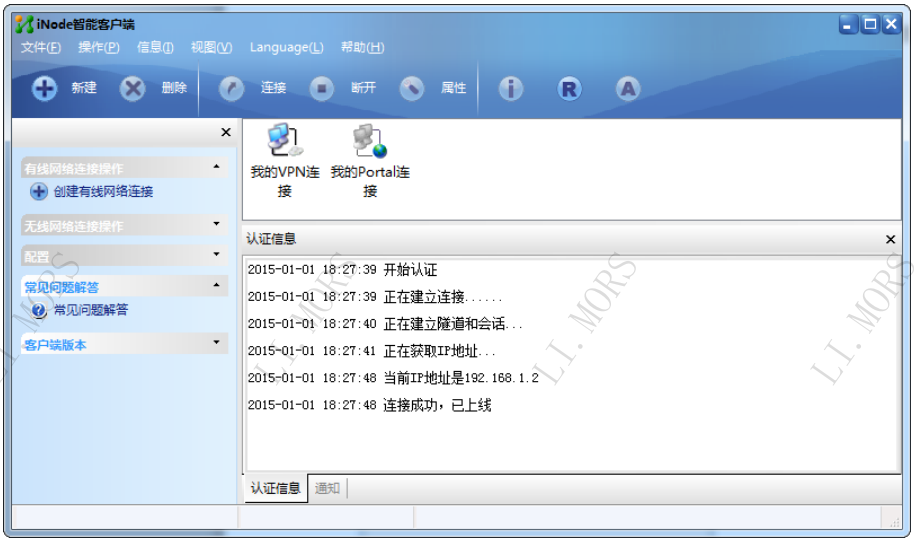
图4-8 完成新建连接向导

步骤六：测试 L2TP 连通性

从 PCA 上发起 L2TP 连接。此时 L2TP 连接应可以正常工作。


iNode智能客户端
文件(F) 操作(P) 信息(I) 视图(V) Language(L) 帮助(H)
新建 删除 连接 断开 属性
有线网络连接操作
创建有线网络连接
无线网络连接操作
配置
常见问题解答
常见问题解答
连接信息
我的VPN连接
我的Portal连接
认证信息
欢迎使用iNode智能客户端
认证信息 通知
我的VPN连接
上网计时 000:00:00
断开


我的VPN连接
用户名(U):
vpdnuser@abc.com
密码(P):
保存用户名和密码(R)
从智能卡读取用户名密码(A)
从证书读取用户名和密码(T)
连接(C)
取消
属性...(Y)




```
[RTB]display l2tp tunnel
LocalTID RemoteTID State       Sessions RemoteAddress   RemotePort RemoteName
4514     1        Established  1        3.3.3.2         51687      LAC
[RTB]display l2tp session
LocalSID    RemoteSID    LocalTID    State
1542        16982        4514        Established
```

确保 L2TP 工作正常后，进入下一步骤。

步骤七：在 LNS 上配置 IPsec/IKE

在 RTB 上配置 IPsec/IKE 参数：

```
[RTB]ike identity fqdn rtb
[RTB]ike proposal 1
[RTB-ike-proposal-1]encryption-algorithm des-cbc
[RTB-ike-proposal-1]authentication-method pre-share
[RTB-ike-proposal-1]authentication-algorithm sha
[RTB-ike-proposal-1]quit
[RTB]ike keychain keychain1
[RTB-ike-keychain-keychain1]pre-shared-key hostname users key simple h3c
[RTB-ike-keychain-keychain1]quit
[RTB]ike profile profile1
[RTB-ike-profile-profile1]proposal 1
[RTB-ike-profile-profile1]keychain keychain1
[RTB-ike-profile-profile1]exchange-mode aggressive
[RTB-ike-profile-profile1]match remote identity fqdn users
[RTB-ike-profile-profile1]quit
[RTB]ipsec transform-set tran1
[RTB-ipsec-transform-set-tran1]protocol esp
[RTB-ipsec-transform-set-tran1]esp authentication-algorithm md5
[RTB-ipsec-transform-set-tran1]esp encryption-algorithm des-cbc
[RTB-ipsec-transform-set-tran1]quit
[RTB]ipsec policy-template templete1 1
[RTB-ipsec-policy-template-templete1-1]transform-set tran1
[RTB-ipsec-policy-template-templete1-1]ike-profile profile1
[RTB-ipsec-policy-template-templete1-1]quit
[RTB]ipsec policy policy1 1 isakmp template templete1
[RTB]interface GigabitEthernet 0/1
[RTB-GigabitEthernet0/1]ipsec apply policy policy1
[RTB-GigabitEthernet0/1]quit
```

步骤八：在 iNode 客户端配置 IPsec/IKE

在 iNode 客户端界面上右击“我的 VPN 连接”图标，在弹出的快捷菜单中单击【属性】，进入图 4-9 所示的窗口。选中【启用 IPsec 安全协议】，并将【验证方法】选择为【预共享密钥】，将【身份验证字】设置为 h3c。选中【使用 LNS 服务器】。

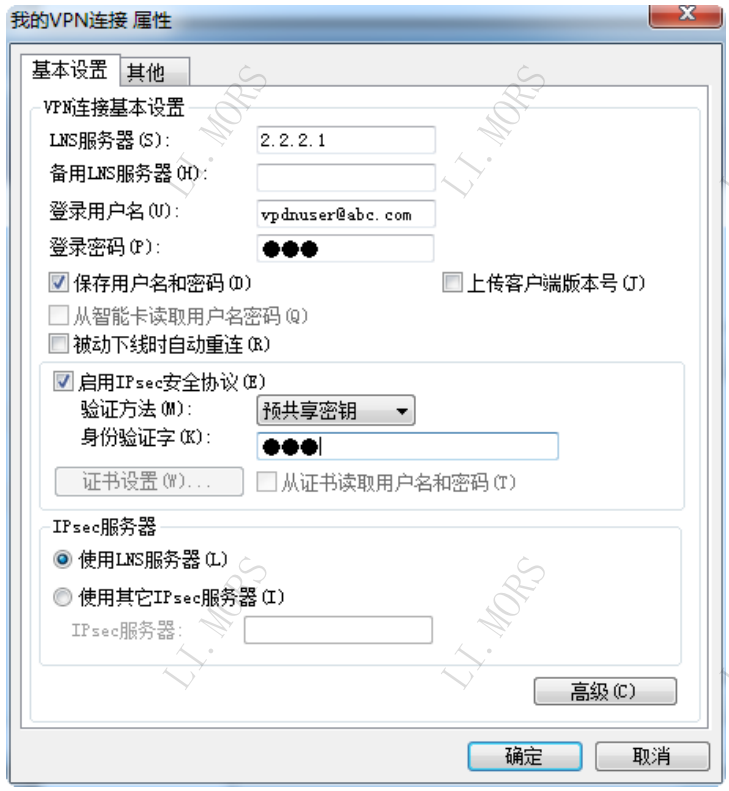


图4-9 VPN 连接属性设置

单击【高级】，进入图 4-10 所示窗口。单击进入【IPsec 设置】选项卡。将【封装模式】设置为【Tunnel】，【采用的安全协议】设置为 ESP，【ESP 协议验证算法】设置为【MD5】，【ESP 协议加密算法】设置为【DES】。

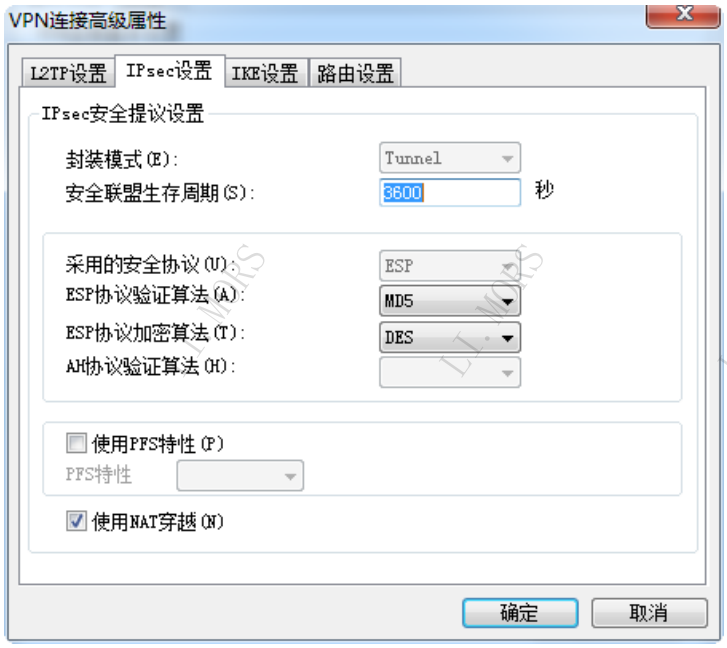


图4-10 IPsec 设置

单击进入【IKE 设置】选项卡，如图 4-11。将【协商模式】设置为【Aggressive】，【ID 的类型】设置为【name】，【验证算法】设置为【SHA】，【加密算法】设置为【DES-CBC】，【Diffie-Hellman 组标识】设置为【Group1】。将【本端安全网关名字】设置为 users，【对端安全网关设备名字】设置为 rtb。

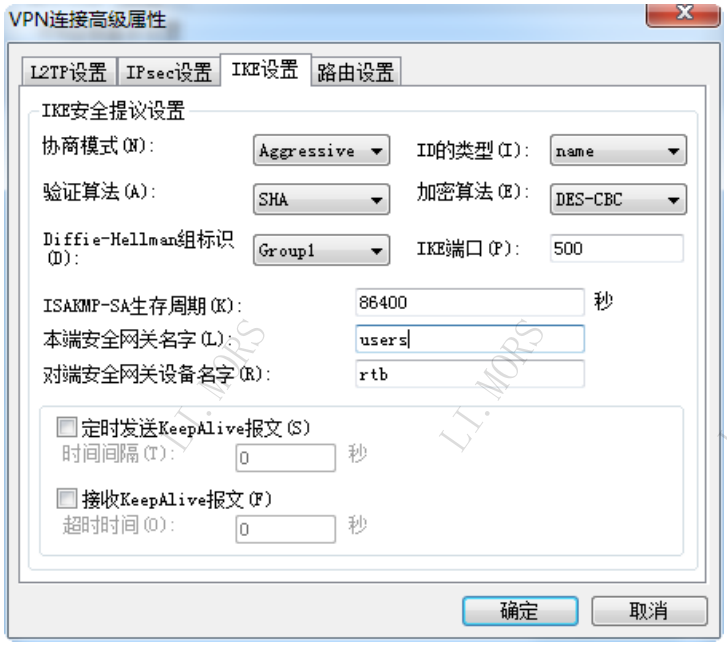
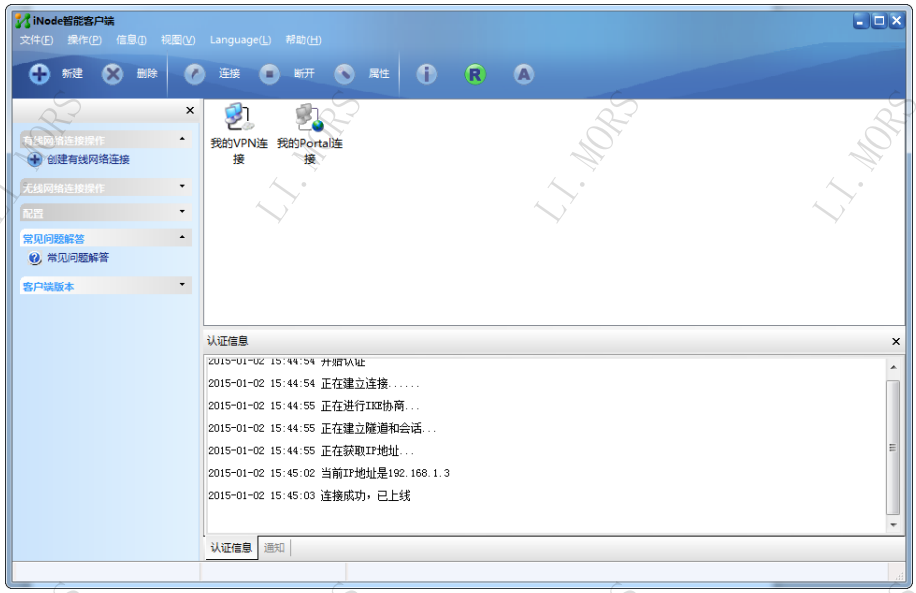


图4-11 IKE 设置

单击【确定】，回到图 4-9 所示窗口，再单击【确定】完成属性设置。

步骤九：检验隧道工作状况

在 PCA 上用 iNode 客户端发起呼叫，检验是否成功通信。此时应可以正常建立 L2TP+IPsec 隧道，从客户端反馈信息应可以看到呼叫成功，并获得 IP 地址。



在 PCA 上查看连接：

```
C:\Users\administrator>ipconfig

Windows IP 配置


以太网适配器 本地连接 2:

   连接特定的 DNS 后缀 . . . . . :
   本地链接 IPv6 地址. . . . . : fe80::9416:507e:5424:1e28%13
   IPv4 地址 . . . . . : 192.168.1.3
   子网掩码 . . . . . : 255.255.255.255
   默认网关. . . . . : 192.168.1.1


以太网适配器 本地连接:

   连接特定的 DNS 后缀 . . . . . :
   本地链接 IPv6 地址. . . . . : fe80::3c2e:9652:cb3a:c0b%11
   IPv4 地址 . . . . . : 3.3.3.2
   子网掩码 . . . . . : 255.255.255.0
   默认网关. . . . . : 3.3.3.1
```

可见连接已经建立，因此除原有的以太网连接之外，还出现一个 L2TP 连接。在 PCA 上检测与 PCB 的连通性，此时应可以连通：

```
C:\Users\administrator>ping 192.168.2.2

正在 Ping 192.168.2.2 具有 32 字节的数据:
来自 192.168.2.2 的回复: 字节=32 时间<1ms TTL=63
```

```
来自 192.168.2.2 的回复: 字节=32 时间=1ms TTL=63
来自 192.168.2.2 的回复: 字节=32 时间=1ms TTL=63
来自 192.168.2.2 的回复: 字节=32 时间=1ms TTL=63

192.168.2.2 的 Ping 统计信息:
    数据包: 已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 0ms，最长 = 1ms，平均 = 0ms
```

在 RTB 上查看 IPsec 和 IKE 信息：

```
<RTB>display ike sa
   Connection-ID   Remote              Flag       DOI
-----------------------------------------------------------------------------
   32              3.3.3.2             RD         IPSEC
<RTB>display ike sa verbose
  ----------------------------------------------
  Connection ID: 32
  Outside VPN:
  Inside VPN:
  Profile: profile1
  Transmitting entity: Responder
  ----------------------------------------------
  Local IP: 2.2.2.1
  Local ID type: FQDN
  Local ID: rtb

  Remote IP: 3.3.3.2
  Remote ID type: FQDN
  Remote ID: users

  Authentication-method: PRE-SHARED-KEY
  Authentication-algorithm: SHA1
  Encryption-algorithm: DES-CBC

  Life duration(sec): 86400
  Remaining key duration(sec): 86196
  Exchange-mode: Aggressive
  Diffie-Hellman group: Group 1
  NAT traversal: Detected
```

可见 ISAKMP SA 是通过 IKE 野蛮模式协商生成的。

```
<RTB>display ips sa
-------------------------------
Interface: GigabitEthernet0/1
-------------------------------

  ----------------------------
  IPsec policy: policy1
  Sequence number: 1
  Mode: template
  ----------------------------
    Tunnel id: 0
    Encapsulation mode: tunnel
    Perfect forward secrecy:
    Path MTU: 1427
    Tunnel:
        local  address: 2.2.2.1
        remote address: 3.3.3.2
    Flow:
    sour addr: 2.2.2.1/255.255.255.255  port: 1701  protocol: udp
    dest addr: 3.3.3.2/255.255.255.255  port: 0  protocol: udp

    [Inbound ESP SAs]
```

```
  SPI: 3961109332 (0xec19bb54)
  Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-MD5
  SA duration (kilobytes/sec): 1843200/3600
  SA remaining duration (kilobytes/sec): 1843179/3392
  Max received sequence-number: 209
  Anti-replay check enable: Y
  Anti-replay window size: 64
  UDP encapsulation used for NAT traversal: Y
  Status: active

[Outbound ESP SAs]
  SPI: 213081373 (0x0cb35d1d)
  Transform set: ESP-ENCRYPT-DES-CBC ESP-AUTH-MD5
  SA duration (kilobytes/sec): 1843200/3600
  SA remaining duration (kilobytes/sec): 1843192/3392
  Max sent sequence-number: 95
  UDP encapsulation used for NAT traversal: Y
  Status: active
```

4.5 实验中的命令列表

表4-4 实验命令列表

命令	描述
ike identity { address *address* **| dn | fqdn [** *fqdn-name* **] | user-fqdn [** *user-fqdn-name* **]**	全局配置本端身份信息
ike proposal *proposal-number*	创建IKE安全提议，并进入安全提议视图
encryption-algorithm { 3des-cbc | aes-cbc-128 | aes-cbc-192 | aes-cbc-256 | des-cbc	配置IKE安全协议采用的加密算法
authentication-method { dsa-signature | pre-share | rsa-signature	配置IKE安全协议采用的认证方法
authentication-algorithm { md5 | sha }	配置IKE安全协议采用的认证算法
ike keychain *keychain-name*	创建并进入一个IKE keychain视图
pre-shared-key { address *address* **[** *mask* | *mask-length* **] | hostname** *host-name* **} key { cipher** *cipher-key* **| simple** *simple-key* **}**	配置预共享密钥
match local address { *interface-type interface-number* **|** *address* **[vpn-instance** *vpn-name* **] }**	（keychain视图）限制IKE keychain的使用范围
ike profile *profile-name*	创建IKE profile，并进入IKE profile视图
exchange-mode { aggressive | main }	配置IKE第一阶段的协商模式
keychain *keychain-name*	指定采用预共享密钥认证时使用的IKE keychain

命令	描述
certificate domain *domain-name*	指定IKE协商采用数字签名认证时使用的PKI域
local-identity { address *address* **| dn | fqdn [** *fqdn-name* **] | user-fqdn [** *user-fqdn-name* **] }**	配置本端身份信息，用于在IKE认证协商阶段向对端标识自己的身份
proposal *proposal-number*	配置IKE profile引用的IKE提议
match remote { certificate *policy-name* **| identity { address** *address* **[** *mask* | *mask-length* **] | range** *low-address high-address* **[vpn-instance** *vpn-name* **] | fqdn** *fqdn-name* **| user-fqdn** *user-fqdn-name* **} }**	配置一条用于匹配对端身份的规则
match local address { *interface-type interface-number* | *address* **[vpn-instance** *vpn-name* **] }**	（ike profile视图）来限制IKE profile的使用范围
display ike proposal	显示每个IKE提议配置的参数
display ike sa [verbose [connection-id *connection-id* **| remote-address** *remote-address* **[vpn-instance** *vpn-name* **]]]**	显示当前IKE SA的信息
reset ike sa [connection-id *connection-id* **]**	清除IKE建立的安全隧道
debugging ike { all | error | event | packet }	调试IKE信息
ipsec transform-set *transform-set-name*	创建安全提议，并进入安全提议视图
protocol { **ah** | **ah-esp** | **esp** }	配置安全提议采用的安全协议
esp encryption-algorithm {3des-cbc | aes-cbc-128 | aes-cbc-192 | aes-cbc-256 | des-cbc | null }	配置ESP协议采用的加密算法
esp authentication-algorithm { **md5** | **sha1** }	配置ESP协议采用的认证算法
ah authentication-algorithm { **md5** | **sha1** }	配置AH协议采用的认证算法
encapsulation-mode { **transport** | **tunnel** }	配置安全协议对IP报文的封装形式
ipsec policy *policy-name seq-number* **isakmp**	创建一条安全策略，并进入安全策略视图
security acl *acl-number*	指定IPsec安全策略/IPsec安全策略模板引用的ACL
transform-set *transform-set-name*	指定IPsec安全策略/IPsec安全策略模板/IPsec安全框架所引用的IPsec安全提议

命令	描述
ike-profile *profile-name*	指定IPsec安全策略/IPsec安全策略模板引用的IKE profile
local-address *ip-address*	配置IPsec隧道的本端IP地址
remote-address *ip-address*	指定IPsec隧道的对端IP地址
sa duration { time-based *seconds* **| traffic-based** *kilobytes* **}**	配置IPsec SA的生存时间
ipsec apply policy *policy-name*	应用指定的安全策略组
display ipsec policy [*policy-name* **[** *seq-number* **]]**	显示安全策略的信息
display ipsec transform-set [*transform-set-name* **]**	显示安全提议的信息
display ipsec sa [brief | count | policy *policy-name* **[** *seq-number* **] | interface** *interface-type interface-number* **| remote** *ip-address* **]**	显示安全联盟的相关信息
display ipsec statistics [tunnel-id *tunnel-id* **]**	显示IPsec处理报文的统计信息
display ipsec tunnel	显示IPsec隧道的信息
reset ipsec sa [spi *spi-number* **| policy** *policy-name* **[** *seq-number* **] | remote** *ip-address* **]**	清除已经建立的安全联盟
debugging ipsec { all | error | packet [policy *policy-name* **[** *seq-number* **] | remote** *ip-address* **l spi *spi-number*] }**	调试IPsec信息

4.6 思考题

1. 在试验任务一中，配置 IPsec 后没有用 ping 来触发 IPsec 隧道的建立，为什么隧道会自动建立？

   答：因为 GRE 隧道中有一些固有流量。例如定时发送的 RIP 协议包。

实验5 BGP MPLS VPN 基础

5.1 实验内容与目标

完成本实验，您应该能够：

- 深入理解 BGP MPLS VPN 的实现原理
- 掌握 BGP MPLS VPN 的配置方法
- 掌握 BGP MPLS VPN 的基本故障排查手段

5.2 实验组网图

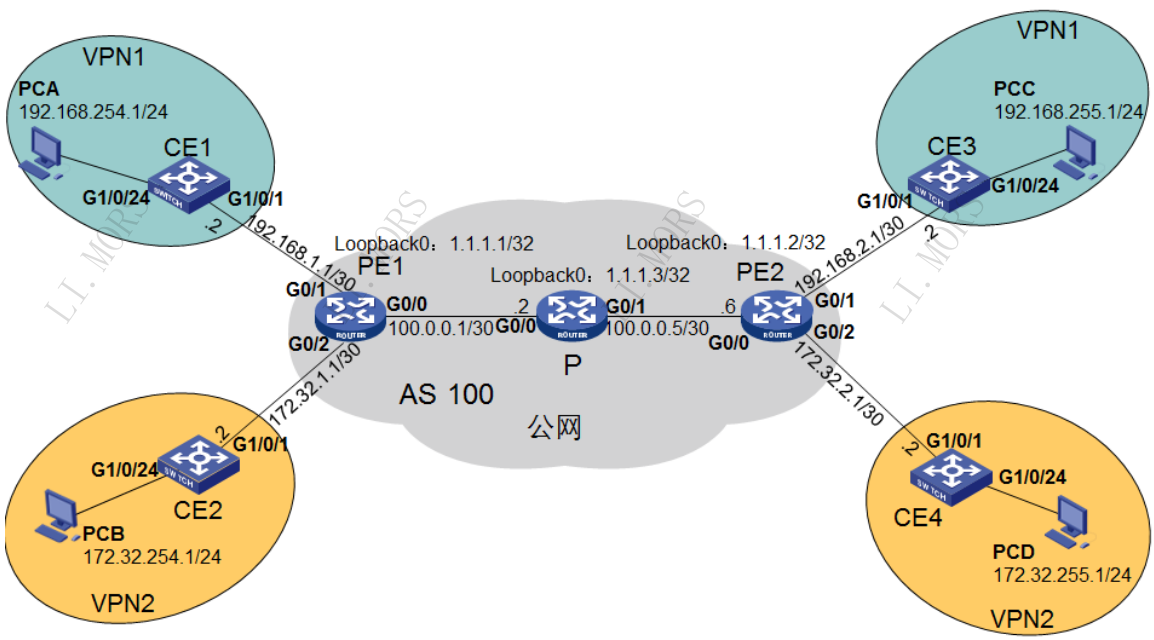


图5-1 BGP MPLS VPN 实验环境图

5.3 实验设备与版本

本实验所需之主要设备器材如表 5-1 所示。

表5-1 实验设备和器材

名称和型号	版本	数量	描述
MSR36-20	Version 7.1.059	3	HCL实验平台
S5820V2-54QS-GE	Version 7.1.059	4	HCL实验平台

名称和型号	版本	数量	描述
PC	Windows XP SP2	4	
第5类UTP以太网连接线	--	10	

如表 5-1 所示，PE1、P、PE2 采用 MSR36-20 路由器，CE1、CE2、CE3 和 CE4 采用 S5820V2-54QS-GE 交换机，PCA、PCB、PCC 和 PCD 采用四台 PC 来模拟，各台设备之间采用以太网线相连。

5.4 实验过程

实验任务一：BGP MPLS VPN 基本配置

图 5-1 所示为 BGP MPLS VPN 应用的一个典型组网，在该 MPLS 网络中承载了两个 VPN——VPN1 和 VPN2。要求 VPN1 的用户即 PCA 与 PCC 之间可以互通，VPN2 的用户即 PCB 与 PCD 之间也可以互通，相反 VPN1 的用户和 VPN2 的用户之间不能互通。

该网络地址规划如图 5-1 标注。

步骤一：搭建环境，执行基本配置

按照组网图连接各设备，并完成接口地址配置。

步骤二：配置公网 IGP 路由协议

配置 IGP 的目的是为了让公网的 PE 设备之间路由可达。实际应用中可以根据情况选择任何一种 IGP 路由协议。本实验选用 OSPF。

在 PE1、P、PE2 设备上配置 OSPF Router ID，并发布各公网接口地址网段路由，包括 PE 设备的 loopback 接口。

PE1 设备上配置：

```
[PE1]ospf 1 router-id 1.1.1.1
[PE1-ospf-1]area 0
[PE1-ospf-1-area-0.0.0.0]network 1.1.1.1 0.0.0.0
[PE1-ospf-1-area-0.0.0.0]network 100.0.0.1 0.0.0.3
```

P 设备上配置：

```
[P]ospf 1 router-id 1.1.1.3
[P-ospf-1]area 0
[P-ospf-1-area-0.0.0.0]network 1.1.1.3 0.0.0.0
[P-ospf-1-area-0.0.0.0]network 100.0.0.2 0.0.0.3
[P-ospf-1-area-0.0.0.0]network 100.0.0.5 0.0.0.3
```

PE2 设备上配置：

```
[PE2]ospf 1 router-id 1.1.1.2
[PE2-ospf-1]area 0
[PE2-ospf-1-area-0.0.0.0]network 1.1.1.2 0.0.0.0
[PE2-ospf-1-area-0.0.0.0]network 100.0.0.6 0.0.0.3
```

检查各公网设备间 OSPF 邻居状况，在 PE 设备上检查是否学习到对端 PE 的路由，并检查是否可达。

检查 OSPF 邻居状况：

```
[PE1]dis ospf peer

       OSPF Process 1 with Router ID 1.1.1.1
            Neighbor Brief Information

 Area: 0.0.0.0
 Router ID      Address         Pri Dead-Time  State           Interface
 1.1.1.3        100.0.0.2       1   40         Full/BDR        GE0/0

[P]dis ospf peer

       OSPF Process 1 with Router ID 1.1.1.3
            Neighbor Brief Information

 Area: 0.0.0.0
 Router ID      Address         Pri Dead-Time  State           Interface
 1.1.1.1        100.0.0.1       1   40         Full/DR         GE0/0
 1.1.1.2        100.0.0.6       1   33         Full/DR         GE0/1


[PE2]dis ospf peer

       OSPF Process 1 with Router ID 1.1.1.2
            Neighbor Brief Information

 Area: 0.0.0.0
 Router ID      Address         Pri Dead-Time  State           Interface
 1.1.1.3        100.0.0.5       1   38         Full/BDR        GE0/0
```

检查 PE 路由：

```
[PE1]dis ip routing-table

Destinations : 16      Routes : 16

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.1/32         Direct  0   0          127.0.0.1       InLoop0
1.1.1.2/32         O_INTRA 10  2           100.0.0.2       GE0/0
1.1.1.3/32         O_INTRA 10  1           100.0.0.2       GE0/0
100.0.0.0/30       Direct  0   0          100.0.0.1       GE0/0
100.0.0.0/32       Direct  0   0          100.0.0.1       GE0/0
100.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
100.0.0.3/32       Direct  0   0          100.0.0.1       GE0/0
100.0.0.4/30       O_INTRA 10  2           100.0.0.2       GE0/0
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0           127.0.0.1       InLoop0
224.0.0.0/4        Direct  0   0          0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0          0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0           127.0.0.1       InLoop0

[PE2]dis ip routing-table

Destinations : 16      Routes : 16

Destination/Mask   Proto   Pre Cost       NextHop         Interface
```

```
0.0.0.0/32        Direct  0   0         127.0.0.1      InLoop0
1.1.1.1/32        O_INTRA 10  2          100.0.0.5      GE0/0
1.1.1.2/32        Direct  0   0         127.0.0.1      InLoop0
1.1.1.3/32        O_INTRA 10  1          100.0.0.5      GE0/0
100.0.0.0/30      O_INTRA 10  2          100.0.0.5      GE0/0
100.0.0.4/30      Direct  0   0         100.0.0.6      GE0/0
100.0.0.4/32      Direct  0   0         100.0.0.6      GE0/0
100.0.0.6/32      Direct  0   0         127.0.0.1      InLoop0
100.0.0.7/32      Direct  0   0         100.0.0.6      GE0/0
127.0.0.0/8       Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/32      Direct  0   0         127.0.0.1      InLoop0
127.0.0.1/32      Direct  0   0         127.0.0.1      InLoop0
127.255.255.255/32 Direct  0   0          127.0.0.1      InLoop0
224.0.0.0/4       Direct  0   0         0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0         0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0          127.0.0.1      InLoop0
```

检查 PE 互通性：

```
[PE1]ping -a 1.1.1.1 1.1.1.2
Ping 1.1.1.2 (1.1.1.2) from 1.1.1.1: 56 data bytes, press CTRL_C to break
56 bytes from 1.1.1.2: icmp_seq=0 ttl=254 time=3.184 ms
56 bytes from 1.1.1.2: icmp_seq=1 ttl=254 time=2.841 ms
56 bytes from 1.1.1.2: icmp_seq=2 ttl=254 time=2.329 ms
56 bytes from 1.1.1.2: icmp_seq=3 ttl=254 time=1.932 ms
56 bytes from 1.1.1.2: icmp_seq=4 ttl=254 time=2.562 ms

--- Ping statistics for 1.1.1.2 ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/std-dev = 1.932/2.570/3.184/0.428 ms

[PE2]ping -a 1.1.1.2 1.1.1.1
Ping 1.1.1.1 (1.1.1.1) from 1.1.1.2: 56 data bytes, press CTRL_C to break
56 bytes from 1.1.1.1: icmp_seq=0 ttl=254 time=3.000 ms
56 bytes from 1.1.1.1: icmp_seq=1 ttl=254 time=2.000 ms
56 bytes from 1.1.1.1: icmp_seq=2 ttl=254 time=1.000 ms
56 bytes from 1.1.1.1: icmp_seq=3 ttl=254 time=2.000 ms
56 bytes from 1.1.1.1: icmp_seq=4 ttl=254 time=3.000 ms

--- Ping statistics for 1.1.1.1 ---
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/std-dev = 1.000/2.200/3.000/0.748 ms
```

步骤三：配置 MPLS 和 MPLS LDP

该部分配置包括在公网设备的系统和接口模式下使能 MPLS 和 MPLS LDP，目的是在 PE 之间建立起 MPLS LSP，后续作为私网数据的隧道。

在系统视图设置 LSR ID 并使能 MPLS LDP：

PE1 设备配置：

```
[PE1]mpls lsr-id 1.1.1.1
[PE1]mpls ldp
[PE1-mpls-ldp]
```

P 设备配置：

```
[P]mpls lsr-id 1.1.1.3
[P]mpls ldp
[P-mpls-ldp]
```

PE2 设备配置：

```
[PE2]mpls lsr-id 1.1.1.2
[PE2]mpls ldp
[PE2-mpls-ldp]
```

在接口视图使能 MPLS 及 MPLS LDP，需要在 PE 和 P 设备的所有公网接口使能 MPLS 和 MPLS LDP。

PE1 设备配置：

```
[PE1]int g0/0
[PE1-GigabitEthernet0/0]mpls enable
[PE1-GigabitEthernet0/0]mpls ldp enable
```

P 设备配置：

```
[P]int g0/0
[P-GigabitEthernet0/0]mpls enable
[P-GigabitEthernet0/0]mpls ldp enable

[P]int g0/1
[P-GigabitEthernet0/1]mpls enable
[P-GigabitEthernet0/1]mpls ldp enable
```

PE2 设备配置：

```
[PE2]int g0/0
[PE2-GigabitEthernet0/0]mpls enable
[PE2-GigabitEthernet0/0]mpls ldp enable
```

配置完成后，检查 MPLS LDP 邻居建立状况。

在 PE1 设备上检查：

```
[PE1]dis mpls ldp peer
Total number of peers: 1
Peer LDP ID           State        Role    GR   MD5  KA Sent/Rcvd
1.1.1.3:0             Operational   Passive  Off  Off  4/4
```

在 P 设备上检查：

```
[P]dis mpls ldp peer
Total number of peers: 2
Peer LDP ID           State        Role    GR   MD5  KA Sent/Rcvd
1.1.1.1:0             Operational   Active   Off  Off  8/8
1.1.1.2:0             Operational   Active   Off  Off  4/4
```

在 PE2 设备上检查：

```
[PE2]dis mpls ldp peer
Total number of peers: 1
Peer LDP ID           State        Role    GR   MD5  KA Sent/Rcvd
1.1.1.3:0             Operational   Passive  Off  Off  6/6
```

检查 PE 之间的 LSP 是否建成。

在 PE1 设备上检查：

```
[PE1]dis mpls ldp lsp
Status Flags: * - stale, L - liberal, B - backup
FECs: 3         Ingress: 2         Transit: 2         Egress: 1

FEC             In/Out Label       Nexthop        OutInterface
1.1.1.1/32        3/-
                 -/1151(L)
1.1.1.2/32        -/1150            100.0.0.2      GE0/0
                 1150/1150         100.0.0.2      GE0/0
1.1.1.3/32        -/3               100.0.0.2      GE0/0
                 1151/3            100.0.0.2      GE0/0
```

在 PE2 设备上检查：

```
[PE2]dis mpls ldp lsp
Status Flags: * - stale, L - liberal, B - backup
FECs: 3         Ingress: 2         Transit: 2         Egress: 1

FEC             In/Out Label       Nexthop        OutInterface
1.1.1.1/32        -/1151            100.0.0.5      GE0/0
                 1151/1151         100.0.0.5      GE0/0
1.1.1.2/32        3/-
                 -/1150(L)
1.1.1.3/32        -/3               100.0.0.5      GE0/0
                 1150/3            100.0.0.5      GE0/0
```

步骤四：配置 VPN 及其 RD 和 RT

在 PE1 和 PE2 上都需要创建两个 VPN，即 VPN1 和 VPN2。其中 PE1 上的 VPN1 需要和 PE2 上的 VPN1 互通，于是可以设计 PE1 和 PE2 上的 VPN1 的 RT 参数 import target 和 export target 值均为 100:1；同时 PE1 上的 VPN2 需要和 PE2 上的 VPN2 互通，于是可以设计 PE1 和 PE2 上的 VPN2 的 RT 参数 import target 和 export target 值均为 200:1。在这样的设计下，VPN1 和 VPN2 的路由将不能互相学习，也就达到了不能互通的要求。

关于 RD 值的设计，只要同一台 PE 上不同的 VPN 的 RD 值不相同即可。这里为了更加清晰，在 PE1 和 PE2 上 VPN1 的 RD 值均设计为 100:1，而 VPN2 的 RD 值均设计为 200:1。

确定 RD 和 RT 的设计后，即可按照下面的方法在 PE1 和 PE2 上分别配置两个 VPN。

PE1 设备配置：

```
[PE1]ip vpn-instance vpn1
[PE1-vpn-instance-vpn1]route-distinguisher 100:1
[PE1-vpn-instance-vpn1]vpn-target 100:1 both

[PE1]ip vpn-instance vpn2
[PE1-vpn-instance-vpn2]route-distinguisher 200:1
[PE1-vpn-instance-vpn2]vpn-target 200:1 both
```

PE2 设备配置：

```
[PE2]ip vpn-instance vpn1
[PE2-vpn-instance-vpn1]route-distinguisher 100:1
[PE2-vpn-instance-vpn1]vpn-target 100:1 both

[PE2]ip vpn-instance vpn2
[PE2-vpn-instance-vpn2]route-distinguisher 200:1
```

```
[PE2-vpn-instance-vpn2]vpn-target 200:1 both
```

步骤五：配置私网接口与 VPN 绑定

将用户接入的接口与对用的 VPN 进行绑定。PE1 上需要将 GigabitEthernet0/1 接口与 VPN1 进行绑定，GigabitEthernet0/2 接口与 VPN2 进行绑定。

PE1 设备配置：

```
[PE1]int GigabitEthernet0/1
[PE1-GigabitEthernet0/1]ip binding vpn-instance vpn1
Some configurations on the interface are removed.
[PE1]int GigabitEthernet0/2
[PE1- GigabitEthernet0/2]ip binding vpn-instance vpn2
Some configurations on the interface are removed.
```

PE2 设备配置：

```
[PE2]int GigabitEthernet0/1
[PE2- GigabitEthernet0/1]ip binding vpn-instance vpn1
Some configurations on the interface are removed.
[PE2]int GigabitEthernet0/2
[PE2- GigabitEthernet0/2]ip binding vpn-instance vpn2
Some configurations on the interface are removed.
```

步骤六：配置 PE 和 CE 之间的路由协议

PE 和 CE 之间的路由协议有多种选择，其中在 PE 设备上需要运行对应路由协议的多实例。本实验采用应用最为广泛的 OSPF 路由协议。

PE1 设备配置：

```
[PE1]ospf 10 vpn-instance vpn1
[PE1-ospf-10]area 0
[PE1-ospf-10-area-0.0.0.0]network 192.168.1.1 0.0.0.3

[PE1]ospf 20 vpn-instance vpn2
[PE1-ospf-20]area 0
[PE1-ospf-20-area-0.0.0.0]network 172.32.1.1 0.0.0.3
```

CE1 设备配置：

```
[CE1]ospf
[CE1-ospf-1]area 0
[CE1-ospf-1-area-0.0.0.0]network 192.168.1.2 0.0.0.3
[CE1-ospf-1-area-0.0.0.0]network 192.168.254.0 0.0.0.255
```

CE2 设备配置：

```
[CE2]ospf
[CE2-ospf-1]area 0
[CE2-ospf-1-area-0.0.0.0]network 172.32.1.2 0.0.0.3
[CE2-ospf-1-area-0.0.0.0]network 172.32.254.0 0.0.0.255
```

PE2 设备配置：

```
[PE2]ospf 10 vpn-instance vpn1
[PE2-ospf-10]area 0
[PE2-ospf-10-area-0.0.0.0]network 192.168.2.1 0.0.0.3

[PE2]ospf 20 vpn-instance vpn2
```

```
[PE2-ospf-20]area 0
[PE2-ospf-20-area-0.0.0.0]network 172.32.2.1 0.0.0.3
```

CE3 设备配置：

```
[CE3]ospf
[CE3-ospf-1]area 0
[CE3-ospf-1-area-0.0.0.0]network 192.168.2.2 0.0.0.3
[CE3-ospf-1-area-0.0.0.0]network 192.168.255.0 0.0.0.255
```

CE4 设备配置：

```
[CE4]ospf
[CE4-ospf-1]area 0
[CE4-ospf-1-area-0.0.0.0]network 172.32.2.2 0.0.0.3
[CE4-ospf-1-area-0.0.0.0]network 172.32.255.0 0.0.0.255
```

检查 PE 和 CE 之间的 OSPF 邻居状况。

在 PE1 设备上检查，PE1 和 CE1 及 CE2 建立起 OSPF 邻居：

```
[PE1]dis ospf 10 peer

       OSPF Process 10 with Router ID 192.168.1.1
            Neighbor Brief Information

 Area: 0.0.0.0
 Router ID      Address        Pri Dead-Time  State           Interface
 192.168.1.2    192.168.1.2     1   31        Full/BDR        GE0/1


[PE1]dis ospf 20 peer

       OSPF Process 20 with Router ID 172.32.1.1
            Neighbor Brief Information

 Area: 0.0.0.0
 Router ID      Address        Pri Dead-Time  State           Interface
 172.32.1.2     172.32.1.2      1   35        Full/BDR        GE0/2
```

在 PE2 设备上检查，PE2 和 CE3 及 CE4 建立起 OSPF 邻居：

```
[PE2]dis ospf 10 peer

       OSPF Process 10 with Router ID 192.168.2.1
            Neighbor Brief Information

 Area: 0.0.0.0
 Router ID      Address        Pri Dead-Time  State           Interface
 192.168.2.2    192.168.2.2     1   30        Full/BDR        GE0/1

[PE2]dis ospf 20 peer

       OSPF Process 20 with Router ID 172.32.2.1
            Neighbor Brief Information

 Area: 0.0.0.0
 Router ID      Address        Pri Dead-Time  State           Interface
 172.32.2.2     172.32.2.2      1   38        Full/BDR        GE0/2
```

检查 PE 学习到了本段 CE 设备的私网路由。

在 PE1 设备上检查，VPN1 学习到了 PCA 的路由，VPN2 学习到了 PCB 的路由：

```
[PE1]dis ip routing-table vpn-instance vpn1

Destinations : 13      Routes : 13

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
192.168.1.0/30     Direct  0   0          192.168.1.1     GE0/1
192.168.1.0/32     Direct  0   0          192.168.1.1     GE0/1
192.168.1.1/32     Direct  0   0          127.0.0.1       InLoop0
192.168.1.3/32     Direct  0   0          192.168.1.1     GE0/1
192.168.254.254/24 O_INTRA 10  1          192.168.1.2     GE0/1
224.0.0.0/4        Direct  0   0          0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0          0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0


[PE1]dis ip routing-table vpn-instance vpn2

Destinations : 13      Routes : 13

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
172.32.1.0/30      Direct  0   0          172.32.1.1      GE0/2
172.32.1.0/32      Direct  0   0          172.32.1.1      GE0/2
172.32.1.1/32      Direct  0   0          127.0.0.1       InLoop0
172.32.1.3/32      Direct  0   0          172.32.1.1      GE0/2
172.32.254.254/24  O_INTRA 10  1          172.32.1.2      GE0/2
224.0.0.0/4        Direct  0   0          0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0          0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
```

在 PE2 设备上检查，VPN1 学习到了 PCC 的路由，VPN2 学习到了 PCD 的路由：

```
[PE2]dis ip routing-table vpn-instance vpn1

Destinations : 13      Routes : 13

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
192.168.2.0/30     Direct  0   0          192.168.2.1     GE0/1
192.168.2.0/32     Direct  0   0          192.168.2.1     GE0/1
192.168.2.1/32     Direct  0   0          127.0.0.1       InLoop0
192.168.2.3/32     Direct  0   0          192.168.2.1     GE0/1
192.168.255.254/24 O_INTRA 10  1          192.168.2.2     GE0/1
224.0.0.0/4        Direct  0   0          0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0          0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0

[PE2]dis ip routing-table vpn-instance vpn2
```

```
Destinations : 13      Routes : 13

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0          127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0          127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
172.32.2.0/30      Direct  0   0          172.32.2.1      GE0/2
172.32.2.0/32      Direct  0   0          172.32.2.1      GE0/2
172.32.2.1/32      Direct  0   0          127.0.0.1       InLoop0
172.32.2.3/32      Direct  0   0          172.32.2.1      GE0/2
172.32.255.254/24  O_INTRA 10  1          172.32.2.2      GE0/2
224.0.0.0/4        Direct  0   0          0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0          0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0          127.0.0.1       InLoop0
```

步骤七：配置 PE 之间 MP-BGP 邻居

注意采用 PE 设备的 loopback 接口作为建立 MP-BGP 邻居的地址,在 BGP VPNv4 视图下使能 BGP 邻居。

PE1 设备配置：

```
[PE1]bgp 100
[PE1-bgp-default]peer 1.1.1.2 as-number 100
[PE1-bgp-default]peer 1.1.1.2 connect-interface LoopBack 0
[PE1-bgp-default]address-family vpnv4
[PE1-bgp-default-vpnv4]peer 1.1.1.2 enable
```

PE2 设备配置：

```
[PE2]bgp 100
[PE2-bgp-default]peer 1.1.1.1 as-number 100
[PE2-bgp-default]peer 1.1.1.1 connect-interface LoopBack 0
[PE2-bgp-default]address-family vpnv4
[PE2-bgp-default-vpnv4]peer 1.1.1.1 enable
```

检查 MP-BGP 邻居建立状况。

在 PE1 设备上检查：

```
[PE1]dis bgp peer vpnv4

 BGP local router ID: 1.1.1.1
 Local AS number: 100
 Total number of peers: 1                 Peers in established state: 1

 * - Dynamically created peer
  Peer                AS  MsgRcvd  MsgSent OutQ PrefRcv Up/Down  State

  1.1.1.2             100       6       7    0      0 00:03:27 Established
```

在 PE2 设备上检查：

```
[PE2]dis bgp peer vpnv4

 BGP local router ID: 1.1.1.2
 Local AS number: 100
 Total number of peers: 1                 Peers in established state: 1
```

```
* - Dynamically created peer
Peer                AS  MsgRcvd  MsgSent OutQ PrefRcv Up/Down  State

1.1.1.1             100       7       7    0      0 00:03:45 Established
```

步骤八：配置本地 VPN 路由与 MP-BGP 之间的路由引入引出

首先将本地 VPN 的路由引入到 MP-BGP，以传递给远端 PE。

PE1 设备配置：

```
[PE1-bgp-default]ip vpn-instance vpn1
[PE1-bgp-default-vpn1]address-family ipv4 unicast
[PE1-bgp-default-ipv4-vpn1]import-route ospf 10
[PE1-bgp-default-ipv4-vpn1]import-route direct

[PE1-bgp-default] ip vpn-instance vpn2
[PE1-bgp-default-vpn2]address-family ipv4 unicast
[PE1-bgp-default-ipv4-vpn2]import-route ospf 20
[PE1-bgp-default-ipv4-vpn2]import-route direct
```

PE2 设备配置：

```
[PE2-bgp-default]ip vpn-instance vpn1
[PE2-bgp-default-vpn1]address-family ipv4 unicast
[PE2-bgp-default-ipv4-vpn1]import-route ospf 10
[PE2-bgp-default-ipv4-vpn1]import-route direct

[PE2-bgp-default] ip vpn-instance vpn2
[PE2-bgp-default-vpn2]address-family ipv4 unicast
[PE2-bgp-default-ipv4-vpn2]import-route ospf 20
[PE2-bgp-default-ipv4-vpn2]import-route direct
```

将通过MP-BGP路由协议从远端PE学习到的私网路由引入到PE和CE之间的路由协议，以设法将这部分路由传给对应 VPN 的 CE 设备。

PE1 设备配置：

```
[PE1]ospf 10
[PE1-ospf-10]import-route bgp

[PE1]ospf 20
[PE1-ospf-20]import-route bgp
```

PE2 设备配置：

```
[PE2]ospf 10
[PE2-ospf-10]import-route bgp

[PE2]ospf 20
[PE2-ospf-20]import-route bgp
```

检查 PE 设备是否学习到远端 VPN 的私网路由。在 PE1 设备上检查：

```
[PE1]dis ip routing-table vpn-instance vpn1

Destinations : 15      Routes : 15

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32         Direct  0   0          127.0.0.1       InLoop0
127.0.0.0/8        Direct  0   0          127.0.0.1       InLoop0
```

```
127.0.0.0/32      Direct  0   0         127.0.0.1      InLoop0
127.0.0.1/32      Direct  0   0         127.0.0.1      InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
192.168.1.0/30    Direct  0   0         192.168.1.1    GE0/1
192.168.1.0/32    Direct  0   0         192.168.1.1    GE0/1
192.168.1.1/32    Direct  0   0         127.0.0.1      InLoop0
192.168.1.3/32    Direct  0   0         192.168.1.1    GE0/1
192.168.2.0/30    BGP    255 0         1.1.1.2        GE0/0
192.168.254.254/24 O_INTRA 10  1         192.168.1.2    GE0/1
192.168.255.254/24 BGP    255 2         1.1.1.2        GE0/0
224.0.0.0/4       Direct  0   0         0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0         0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0

[PE1]dis ip routing-table vpn-instance vpn2

Destinations : 15      Routes : 15

Destination/Mask   Proto   Pre Cost      NextHop         Interface
0.0.0.0/32        Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/8       Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/32      Direct  0   0         127.0.0.1      InLoop0
127.0.0.1/32      Direct  0   0         127.0.0.1      InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
172.32.1.0/30     Direct  0   0         172.32.1.1     GE0/2
172.32.1.0/32     Direct  0   0         172.32.1.1     GE0/2
172.32.1.1/32     Direct  0   0         127.0.0.1      InLoop0
172.32.1.3/32     Direct  0   0         172.32.1.1     GE0/2
172.32.2.0/30     BGP    255 0         1.1.1.2        GE0/0
172.32.254.254/24  O_INTRA 10  1         172.32.1.2     GE0/2
172.32.255.254/24  BGP    255 2         1.1.1.2        GE0/0
224.0.0.0/4       Direct  0   0         0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0         0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
```

在 PE2 设备上检查：

```
[PE2]dis ip routing-table vpn-instance vpn1

Destinations : 15      Routes : 15

Destination/Mask   Proto   Pre Cost      NextHop         Interface
0.0.0.0/32        Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/8       Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/32      Direct  0   0         127.0.0.1      InLoop0
127.0.0.1/32      Direct  0   0         127.0.0.1      InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
192.168.1.0/30    BGP    255 0         1.1.1.1        GE0/0
192.168.2.0/30    Direct  0   0         192.168.2.1    GE0/1
192.168.2.0/32    Direct  0   0         192.168.2.1    GE0/1
192.168.2.1/32    Direct  0   0         127.0.0.1      InLoop0
192.168.2.3/32    Direct  0   0         192.168.2.1    GE0/1
192.168.254.254/24 BGP    255 2         1.1.1.1        GE0/0
192.168.255.254/24 O_INTRA 10  1         192.168.2.2    GE0/1
224.0.0.0/4       Direct  0   0         0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0         0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0

[PE2]dis ip routing-table vpn-instance vpn2

Destinations : 15      Routes : 15

Destination/Mask   Proto   Pre Cost      NextHop         Interface
0.0.0.0/32        Direct  0   0         127.0.0.1      InLoop0
```

```
127.0.0.0/8       Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/32      Direct  0   0         127.0.0.1      InLoop0
127.0.0.1/32      Direct  0   0         127.0.0.1      InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
172.32.1.0/30     BGP    255 0          1.1.1.1        GE0/0
172.32.2.0/30     Direct  0   0         172.32.2.1     GE0/2
172.32.2.0/32     Direct  0   0         172.32.2.1     GE0/2
172.32.2.1/32     Direct  0   0         127.0.0.1      InLoop0
172.32.2.3/32     Direct  0   0         172.32.2.1     GE0/2
172.32.254.254/24  BGP    255 2          1.1.1.1        GE0/0
172.32.255.254/24  O_INTRA 10  1         172.32.2.2     GE0/2
224.0.0.0/4       Direct  0   0         0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0         0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
```

检查 CE 设备是否学习到远端 VPN 的私网路由。

在 CE1 设备上检查：

```
<CE1>dis ip routing-table

Destinations : 18      Routes : 18

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32        Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/8       Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/32      Direct  0   0         127.0.0.1      InLoop0
127.0.0.1/32      Direct  0   0         127.0.0.1      InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
192.168.1.0/30    Direct  0   0         192.168.1.2    Vlan100
192.168.1.0/32    Direct  0   0         192.168.1.2    Vlan100
192.168.1.2/32    Direct  0   0         127.0.0.1      InLoop0
192.168.1.3/32    Direct  0   0         192.168.1.2    Vlan100
192.168.2.0/30    O_ASE2  150 1         192.168.1.1    Vlan100
192.168.254.0/24   Direct  0   0         192.168.254.254 Vlan200
192.168.254.0/32   Direct  0   0         192.168.254.254 Vlan200
192.168.254.254/32 Direct  0   0         127.0.0.1      InLoop0
192.168.254.255/32 Direct  0   0         192.168.254.254 Vlan200
192.168.255.254/32 O_INTER 10  3         192.168.1.1    Vlan100
224.0.0.0/4       Direct  0   0         0.0.0.0        NULL0
224.0.0.0/24      Direct  0   0         0.0.0.0        NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
```

在 CE2 设备上检查：

```
<CE2>dis ip routing-table

Destinations : 18      Routes : 18

Destination/Mask   Proto   Pre Cost       NextHop         Interface
0.0.0.0/32        Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/8       Direct  0   0         127.0.0.1      InLoop0
127.0.0.0/32      Direct  0   0         127.0.0.1      InLoop0
127.0.0.1/32      Direct  0   0         127.0.0.1      InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1      InLoop0
172.32.1.0/30     Direct  0   0         172.32.1.2     Vlan100
172.32.1.0/32     Direct  0   0         172.32.1.2     Vlan100
172.32.1.2/32     Direct  0   0         127.0.0.1      InLoop0
172.32.1.3/32     Direct  0   0         172.32.1.2     Vlan100
172.32.2.0/30     O_ASE2  150 1         172.32.1.1     Vlan100
172.32.254.0/24    Direct  0   0         172.32.254.254  Vlan200
```

```
172.32.254.0/32    Direct  0   0         172.32.254.254  Vlan200
172.32.254.254/32  Direct  0   0         127.0.0.1       InLoop0
172.32.254.255/32  Direct  0   0         172.32.254.254  Vlan200
172.32.255.254/32  O_INTER 10  3         172.32.1.1      Vlan100
224.0.0.0/4        Direct  0   0         0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0         0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1       InLoop0
```

在 CE3 设备上检查：

```
<CE3>dis ip routing-table

Destinations : 18      Routes : 18

Destination/Mask   Proto   Pre Cost      NextHop         Interface
0.0.0.0/32         Direct  0   0         127.0.0.1       InLoop0
127.0.0.0/8        Direct  0   0         127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0         127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0         127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1       InLoop0
192.168.1.0/30     O_ASE2  150 1         192.168.2.1     Vlan100
192.168.2.0/30     Direct  0   0         192.168.2.2     Vlan100
192.168.2.0/32     Direct  0   0         192.168.2.2     Vlan100
192.168.2.2/32     Direct  0   0         127.0.0.1       InLoop0
192.168.2.3/32     Direct  0   0         192.168.2.2     Vlan100
192.168.254.254/32 O_INTER 10  3         192.168.2.1     Vlan100
192.168.255.0/24   Direct  0   0         192.168.255.254 Vlan200
192.168.255.0/32   Direct  0   0         192.168.255.254 Vlan200
192.168.255.254/32 Direct  0   0         127.0.0.1       InLoop0
192.168.255.255/32 Direct  0   0         192.168.255.254 Vlan200
224.0.0.0/4        Direct  0   0         0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0         0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1       InLoop0
```

在 CE4 设备上检查：

```
<CE4>dis ip routing-table

Destinations : 18      Routes : 18

Destination/Mask   Proto   Pre Cost      NextHop         Interface
0.0.0.0/32         Direct  0   0         127.0.0.1       InLoop0
127.0.0.0/8        Direct  0   0         127.0.0.1       InLoop0
127.0.0.0/32       Direct  0   0         127.0.0.1       InLoop0
127.0.0.1/32       Direct  0   0         127.0.0.1       InLoop0
127.255.255.255/32 Direct  0   0         127.0.0.1       InLoop0
172.32.1.0/30      O_ASE2  150 1         172.32.2.1      Vlan100
172.32.2.0/30      Direct  0   0         172.32.2.2      Vlan100
172.32.2.0/32      Direct  0   0         172.32.2.2      Vlan100
172.32.2.2/32      Direct  0   0         127.0.0.1       InLoop0
172.32.2.3/32      Direct  0   0         172.32.2.2      Vlan100
172.32.254.254/32  O_INTER 10  3         172.32.2.1      Vlan100
172.32.255.0/24    Direct  0   0         172.32.255.254  Vlan200
172.32.255.0/32    Direct  0   0         172.32.255.254  Vlan200
172.32.255.254/32  Direct  0   0         127.0.0.1       InLoop0
172.32.255.255/32  Direct  0   0         172.32.255.254  Vlan200
224.0.0.0/4        Direct  0   0         0.0.0.0         NULL0
224.0.0.0/24       Direct  0   0         0.0.0.0         NULL0
255.255.255.255/32 Direct  0   0         127.0.0.1       InLoop0
```

检查用户业务之间的互通性。

PCA 访问 PCC，可以互通：

```
C:\Documents and Settings\user >ping 192.168.255.1

Pinging 192.168.255.1 with 32 bytes of data:

Reply from 192.168.255.1: bytes=32 time=1ms TTL=251
Reply from 192.168.255.1: bytes=32 time=1ms TTL=251
Reply from 192.168.255.1: bytes=32 time=1ms TTL=251
Reply from 192.168.255.1: bytes=32 time=1ms TTL=251

Ping statistics for 192.168.255.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms
```

PCA 访问 PCB，不能互通：

```
C:\Documents and Settings\user>ping 172.32.254.1

Pinging 172.32.254.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 172.32.254.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

PCA 访问 PCD，不能互通：

```
C:\Documents and Settings\user>ping 172.32.255.1

Pinging 172.32.255.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 172.32.255.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

PCB 访问 PCC，不能互通：

```
C:\Documents and Settings\user>ping 192.168.255.1

Pinging 192.168.255.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.255.1
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

PCB 访问 PCD，可以互通：

```
C:\Documents and Settings\user >ping 172.32.255.1
```

```
Pinging 172.32.255.1 with 32 bytes of data:

Reply from 172.32.255.1: bytes=32 time=1ms TTL=251
Reply from 172.32.255.1: bytes=32 time=1ms TTL=251
Reply from 172.32.255.1: bytes=32 time=1ms TTL=251
Reply from 172.32.255.1: bytes=32 time=1ms TTL=251

Ping statistics for 172.32.255.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms
```

PCC 访问 PCD，不能互通：

```
C:\Documents and Settings\user>ping 172.32.255.1

Pinging 172.32.255.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 172.32.255.1
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

实验结果与组网需求相符。

5.5 实验中的命令列表

表5-2 实验命令列表

命令	描述
mpls lsr-id *lsr-id*	配置本节点的LSR ID
mpls ldp	使能LDP能力
ip vpn-instance *vpn-instance-name*	创建VPN实例，并进入VPN实例视图
route-distinguisher *route-distinguisher*	配置VPN实例的RD
vpn-target *vpn-target*&<1-8> [**both** | **export-extcommunity** | **import-extcommunity**]	将当前VPN实例与一个或多个VPN Target相关联
ip binding vpn-instance *vpn-instance-name*	将当前接口与VPN实例关联
ospf [*process-id* | **router-id** *router-id* | **vpn-instance** *vpn-instance-name*]	创建PE-CE间的OSPF实例
address-family vpnv4	进入BGP-VPNv4子地址族视图
ip vpn-instance *vpn-instance-name*	进入BGP-VPN实例视图

5.6 思考题

1. 为什么配置 PE1 与 PE2 建立 BGP 邻居的时候一定要配置 connect-interface 为 LoopBack 0？如果不配置，或者配置采用接口地址作为建立 BGP 邻居的地址，那么结果会怎样？

答：配置 PE1 和 PE2 建立 BGP 邻居采用 LoopBack 0 作为 connect-interface 的原因有两个。

其一，这样能够有效增强 BGP 邻居的健壮性，因为实际网络中 PE1 和 PE2 可能有多条路径可达，如果采用接口地址建立邻居，可能因为该接口的故障导致 BGP 邻居中断，而此时 PE1 和 PE2 还是有其他路径可达的。

其二，因为在 BGP MPLS VPN 的组网中，PE 设备将本地的私网路由传递给对端 PE 时，路由中的下一跳地址就是与对端 PE 建立邻居的地址。这个地址如果不是 loopback0，在缺省的 MPLS LDP 标签分配方法中，将不会有到这个下一跳地址的隧道，私网数据就无法转发了。如上面的实验中，PE1 上的 LSP 情况如下表所示，只有到 PE2 的 loopback 接口地址的隧道，而没有到其他地址的隧道。

```
[PE1]dis mpls ldp lsp
Status Flags: * - stale, L - liberal, B - backup
FECs: 3         Ingress: 2         Transit: 2         Egress: 1

FEC              In/Out Label       Nexthop         OutInterface
1.1.1.1/32         3/-
                  -/1151(L)
1.1.1.2/32         -/1150            100.0.0.2       GE0/0
                  1150/1150         100.0.0.2       GE0/0
1.1.1.3/32         -/3               100.0.0.2       GE0/0
                  1151/3            100.0.0.2       GE0/0
```

实验6 配置流量监管

6.1 实验内容与目标

完成本实验，您应该能够：

- 深入理解流量监管工具的作用
- 配置 CAR 进行流量监管和标记并查看相关信息

6.2 预备知识和技能

掌握 CAR 进行流量监管和标记的基本原理，熟悉 H3C 设备的基本操作和相关命令。

6.3 实验组网图

图6-1 实验组网

实验组网如图 6-1 所示。PCA 和 PCB 为两台主机，通过 RTA 和 RTB 两台路由器相连。路由器之间为 64Kbps 的串行线路，PC 与路由器之间为以太网。网络地址如上图所示。

6.4 实验设备和器材

本实验所需之主要设备器材如表 6-1 所示。

表6-1 实验设备和器材

名称和型号	版本	数量	描述
MSR36-20	CMW7.1.049-R0106P15	2	
PC	Windows 7	2	

名称和型号	版本	数量	描述
V.24 DTE串口线	--	1	
V.24 DCE串口线	--	1	
第5类UTP以太网连接线	--	2	

6.5 实验过程

实验任务一：配置入方向的流量监管

在本实验中，PCB 作为一台 FTP 客户端，从作为服务器的 PCA 上进行文件下载。验证 CAR 不但可以对特定的流量进行限速，而且也能对其进行重标记。

步骤一：搭建试验环境，进行基本连通性配置

搭建如图所示的试验环境，RTA 和 RTB 之间用 V.24 电缆连接，在接口采用 PPP 协议。在 RTA 和 RTB 上分别配置静态路由，在 PCA 和 PCB 上分别配置缺省网关，以保证两台 PC 之间互相能够 Ping 通。

接口地址和协议的配置：

```
[RTA] interface Serial 1/0
[RTA-Serial1/0] ip address 1.1.1.1 24
[RTA-Serial1/0] link-protocol ppp
[RTA] interface GigabitEthernet 0/0
[RTA-GigabitEthernet0/0] ip address 192.168.1.1 24

[RTB] interface Serial 1/0
[RTB-Serial1/0] ip address 1.1.1.2 24
[RTB-Serial1/0] link-protocol ppp
[RTB] interface GigabitEthernet 0/0
[RTB-GigabitEthernet0/0] ip address 192.168.2.1 24
```

静态路由的配置：

```
[RTA] ip route-static 192.168.2.0 255.255.255.0 1.1.1.2

[RTB] ip route-static 192.168.1.0 255.255.255.0 1.1.1.1
```

根据图 6-1 在 PCA 和 PCB 上配置 IP 地址，将其缺省网关分别配置为 192.168.1.1 和 192.168.2.1。

验证连通性：

```
C:\>ping 192.168.2.2

Pinging 192.168.2.2 with 32 bytes of data:

Reply from 192.168.2.2: bytes=32 time=1ms TTL=253
Reply from 192.168.2.2: bytes=32 time=1ms TTL=253
Reply from 192.168.2.2: bytes=32 time=1ms TTL=253
Reply from 192.168.2.2: bytes=32 time=7ms TTL=253

Ping statistics for 192.168.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

```
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 7ms, Average = 2ms

C:\>
```

步骤二：观察不配置 CAR 时的下载速率

如下图，可见下载 684KB 大小文件耗时 91s，平均速率 7.70KByte/sec，约为 61.6Kbps。

```
D:\>ftp 192.168.1.2
Connected to 192.168.1.2.
220 Serv-U FTP-Server v2.4 for WinSock ready...
User (192.168.1.2:(none)): tyuan
331 User name okay, need password.
Password:
230 User logged in, proceed.
ftp> get rfc-index.txt
200 PORT Command successful.
150 Opening ASCII mode data connection for rfc-index.txt (701206 bytes).
226 Transfer complete.
ftp: 701206 bytes received in 91.09Seconds 7.70Kbytes/sec.
ftp>
```

步骤三：配置 CAR 限速和标记

配置 CAR 限速为 32Kbps，同时对允许通过的报文重标记 IP Precedence 为 5。

```
[RTA]acl basic 2000
[RTA-acl-ipv4-basic-2000]rule permit source 192.168.1.2 0
[RTA-acl-ipv4-basic-2000]quit
[RTA]interface GigabitEthernet0/0
[RTA-GigabitEthernet0/0]qos car inbound acl 2000 cir 32 green remark-prec-pass
5 red discard
[RTA-GigabitEthernet0/0]
```

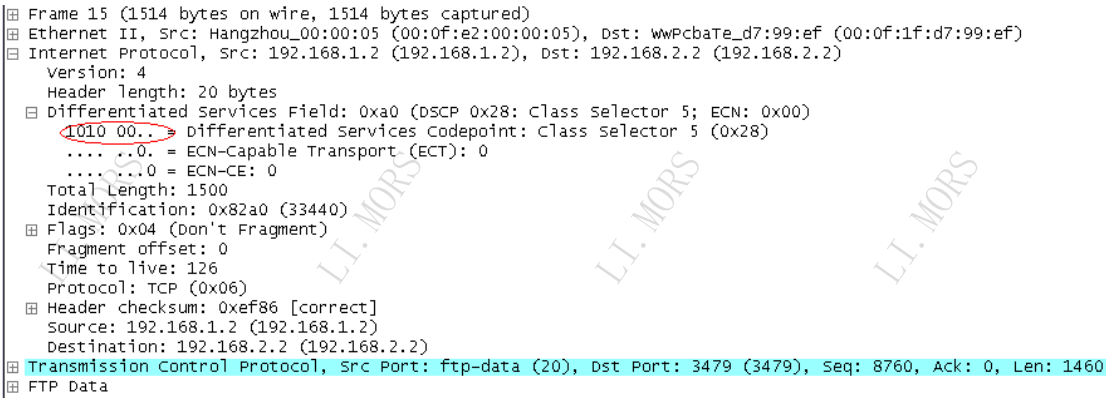
步骤四：观察配置 CAR 之后的下载速率

观察配置 CAR 之后的下载速率，如下图所示。同时在 PCB 上抓包查看报文 IP Precedence。

```
D:\>ftp 192.168.1.2
Connected to 192.168.1.2.
220 Serv-U FTP-Server v2.4 for WinSock ready...
User (192.168.1.2:(none)): tyuan
331 User name okay, need password.
Password:
230 User logged in, proceed.
ftp> get rfc-index.txt
200 PORT Command successful.
150 Opening ASCII mode data connection for rfc-index.txt (701206 bytes).
226 Transfer complete.
ftp: 701206 bytes received in 296.84Seconds 2.36Kbytes/sec.
ftp> bye
221 Goodbye!
```

可见下载 684KB 大小文件耗时 296s，平均速率 2.36KByte/sec，约为 18.88Kbps。

用抓包工具（如 Ethereal）抓取 PCA 传送给 PCB 的包，可以看到报文 IP Precedence 被修改为 5：



步骤五：在设备上查看流量监管的统计信息

在设备上查看流量监管的统计信息，可以看到被标记为 Red（红色）和 Green（绿色）的报文统计。

```
[RTA]display qos car interface GigabitEthernet 0/0
Interface: GigabitEthernet0/0
 Direction: inbound
  Rule: If-match acl 2000
   CIR 32 (kbps), CBS 2000 (Bytes), EBS 0 (Bytes)
   Green action  : remark dscp 5 and pass
   Yellow action : pass
   Red action    : discard
   Green packets : 529 (Packets), 733817 (Bytes)
   Yellow packets: 0 (Packets), 0 (Bytes)
   Red packets   : 291 (Packets), 402968 (Bytes)
```

6.6 实验中的命令列表

表6-2 命令列表

命令	描述
qos car { **inbound** | **outbound** } { **any** | **acl** [**ipv6**] acl-number | **carl** carl-index } **cir** committed-information-rate [**cbs** committed-burst-size [**ebs** excess-burst-size]] [**pir** peak-information-rate] [**green** action] [**red** action]	端口下启用CAR，对符合监管条件的报文进行度量和操作。
display qos car interface [interface-type interface-number]	查看接口下配置CAR的统计信息。

6.7 思考题

1. 为什么在采用 CAR 之后，FTP 的平均流量并没有达到配置的 32Kbps，而是小于这个值？

答：这是因为被流量监管所丢弃的报文被上层协议重传了；同时 TCP 自身的窗口机制也适时的调整了发送速率，导致实际的传输速率在 32Kbps 以下进行波动；另外 TCP 的连接确认机制也要消耗一些时间。

实验7 配置拥塞管理

7.1 实验内容与目标

完成本实验，您应该能够：

- 深入理解并掌握 QoS policy 的配置方法
- 配置 QoS policy 实现 CBQ 并查看相关信息

7.2 实验组网图

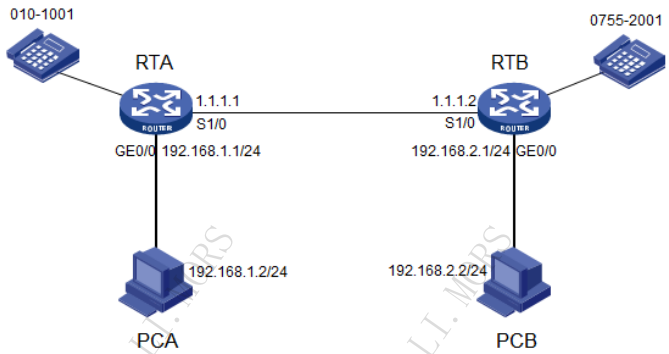


图7-1 配置拥塞管理实验环境图

实验组网如图 7-1 所示。由 2 台 MSR3620（RTA、RTB）路由器、2 台电话机、2 台 PC（PCA、PCB）组成，互连方式和 IP 地址分配参见图 7-1。

路由器 RTA 和 RTB 通过低速串口连接，RTA 和 RTB 的语音模块和电话机连接，提供 VoIP 服务。RTA 的 GE0/0 端口和 PCA 连接，RTB 的 GE0/0 端口和 PCB 连接。PCA 和 PCB 上安装 FTP server 软件，通过 FTP 协议互相传输大的数据文件。

7.3 实验设备与版本

本实验所需之主要设备器材如表 7-1 所示。

表7-1 实验设备和器材

名称和型号	版本	数量	描述
MSR3620	CMW 7.1.049 R0106P15	2	各带一块FXS语音模块
电话机	--	2	
PC	Windows 7	2	安装FTP Server软件
第5类UTP以太网连接线	--	4	其中包括交叉线2根

名称和型号	版本	数量	描述
RJ11接头电话线	--	2	

7.4 实验过程

本实验要求配置 CBQ 队列，在发生拥塞时，保证对语音业务的加速转发，对数据传输业务确保转发。

注意：

请勿带电插拔语音模块，否则极易损坏设备。

实验任务一：配置 CBQ

步骤一：连接设备，执行基本配置

首先，依照图示搭建实验环境，完成路由器 RTA 与 RTB 的接口 IP 地址的配置，为了使 PCA 和 PCB 可以互相访问。在 RTA 上配置通往 192.168.2.0 网段的静态路由，下一跳为 1.1.1.2，在 RTB 上配置通往 192.168.1.0 网段的静态路由，下一跳为 1.1.1.1。将 RTA 和 RTB 之间的串口速率限制在 128K。

```
[RTA]interface Serial 1/0
[RTA-Serial1/0]ip address 1.1.1.1 24
[RTA-Serial1/0]interface GigabitEthernet0/0
[RTA-Serial1/0]ip add 1.1.1.2 24
[RTA-Serial1/0] qos lr outbound cir 128
[RTA-GigabitEthernet0/0]ip address 192.168.1.1 24
[RTA-GigabitEthernet0/0]ip route-static 192.168.2.0 24 1.1.1.2

[RTB]interface Serial 1/0
[RTB-Serial1/0]ip address 1.1.1.2 24
[RTB-Serial1/0]interface Ethernet0/0
[RTA-Serial1/0] qos lr outbound cir 128
[RTB-GigabitEthernet 0/0]ip address 192.168.2.1 24
[RTB-GigabitEthernet0/0]ip route-static 192.168.1.0 24 1.1.1.1
```

配置主机 PCA 的 IP 地址为 192.168.1.2/24，网关为 192.168.1.1，配置主机 PCB 的 IP 地址为 192.168.2.2/24，网关为 192.168.2.1。配置完成后，主机 PCA 和 PCB 之间可以 ping 通：

```
C:\>ping 192.168.2.2

Pinging 192.168.2.2 with 32 bytes of data:

Reply from 192.168.2.2: bytes=32 time<1ms TTL=126
Reply from 192.168.2.2: bytes=32 time<1ms TTL=126
Reply from 192.168.2.2: bytes=32 time=1ms TTL=126
Reply from 192.168.2.2: bytes=32 time<1ms TTL=126

Ping statistics for 192.168.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

步骤二：配置 VoIP

RTA 设备 VoIP 配置：

配置到 RTB 的语音实体

```
[RTA] voice-setup
[RTA-voice] dial-program
[RTA-voice-dial] entity 0755 voip
[RTA-voice-dial-entity755] match-template 0755....
[RTA-voice-dial-entity755] address ip 1.1.1.2
[RTA-voice-dial-entity755] quit
```

配置本地 FXS 端口 Line 1/0 对应的 POTS 语音实体

```
[RTA-voice-dial] entity 1001 pots
[RTA-voice-dial-entity1001] match-template 0101001
[RTA-voice-dial-entity1001] line 1/0
[RTA-voice-dial-entity1001] quit
[RTA-voice-dial]default entity compression 2nd-level g711alaw
```

RTB 设备 VoIP 配置：

配置到 RTA 设备的 VoIP 语音实体。

```
[RTB] voice-setup
[RTB-voice] dial-program
[RTB-voice-dial] entity 010 voip
[RTB-voice-dial-entity10] match-template 010....
[RTB-voice-dial-entity10] address ip 1.1.1.1
[RTB-voice-dial-entity10] quit
```

配置本地 FXS 端口 Line 1/0 对应 POTS 语音实体。

```
[RTB-voice-dial] entity 2001 pots
[RTB-voice-dial-entity1001] match-template 07552001
[RTB-voice-dial-entity1001] line 1/0
[RTB-voice-dial-entity1001] quit
[RTB-voice-dial]default entity compression 2nd-level g711alaw
```

以上 VoIP 配置中使用 G.711a 语音编码，占用 64K 带宽，通话质量最好。配置完成后，用 RTA 设备的电话拨打 07552001，或用 RTB 设备的电话拨打 0101001，通话正常，声音清晰。

步骤三：检查拥塞时的语音效果

PCA 使用 FTP 载 PCB 上的一个大文件，造成 RTB 设备串口 Serial0/1 出方向拥塞。此时用 RTB 设备的电话拨打 0101001，连续说话，应该无法保持持续清晰。

步骤四：配置 CBQ

配置 CBQ，在路由器串口上为语音数据提供 EF 服务，为文件传输提供 AF 服务。

配置匹配语音流的访问控制列表

```
[RTB]acl basic 2000
[RTB-acl-ipv4-basic-2000]rul 0 per source 1.1.1.1 0
```

配置匹配 ftp 数据流的访问控制列表

```
[RTB]acl basic 2001
[RTB-acl-ipv4-basic-2001]rule permit source 192.168.2.2 0
```

配置匹配语音流的类

```
[RTB]traffic classifier EF-voice
[RTB-classifier-EF-voice]if-match acl 2000
```

配置匹配 ftp 数据流的类

```
[RTB]traffic classifier AF-ftp
[RTB-classifier-AF-ftp]if-match acl 2001
```

配置 EF 队列，对语音流分配 64K 带宽

```
[RTB]traffic behavior EF-voice
[RTB-behavior-EF-voice]queue ef bandwidth 64
```

配置 AF 队列，对 ftp 数据流保证 50K 带宽

```
[RTB]traffic behavior AF-ftp
[RTB-behavior-AF-ftp]queue af bandwidth 50
```

配置 QoS 策略，把类和流行位绑定

```
[RTB]qos policy CBQ
[RTB-qospolicy-CBQ]classifier EF-voice behavior EF-voice
[RTB-qospolicy-CBQ]classifier AF-ftp behavior AF-ftp
```

把 QoS 策略应用到端口

```
[RTB]interface Serial 1/0
[RTB-Serial0/1]qos apply policy CBQ outbound
```

步骤五：再次检查拥塞时的语音效果

再次在 PCA 上使用 FTP 协议下载 PCB 上的一个大文件，造成 RTB 设备串口 Serial0/1 出方向拥塞。此时用 RTB 设备的电话拨打 0101001，语音效果清晰，PCA 到 PCB 的文件下载正常。

通过显示命令，可以看到如下 CBQ 相关信息：

```
<RTA>display qos policy interface Serial 1/0

  Interface: Serial1/0

  Direction: Outbound

  Policy: CBQ
   Classifier: default-class
     Matched : 0(Packets) 0(Bytes)
     5-minute statistics:
       Forwarded: 0/0 (pps/bps)
       Dropped  : 0/0 (pps/bps)
     Operator: AND
     Rule(s) :
       If-match any
     Behavior: be
     Default Queue:
      Flow Based Weighted Fair Queuing
        Max number of hashed queues: 256
        Matched  : 0/0 (Packets/Bytes)
        Enqueued : 0/0 (Packets/Bytes)
        Discarded: 0/0 (Packets/Bytes)
        Discard Method: Tail
   Classifier: EF-voice
     Matched : 1193(Packets) 88429(Bytes)
     5-minute statistics:
       Forwarded: 3/2358 (pps/bps)
       Dropped  : 0/0 (pps/bps)
     Operator: AND
     Rule(s) : If-match acl 2000
     Behavior: EF-voice
      Expedited Forwarding:
```

```
    Bandwidth 64 (Kbps), CBS 1600 (Bytes)
    Matched  : 1095/81135 (Packets/Bytes)
    Enqueued : 1095/81135 (Packets/Bytes)
    Discarded: 0/0 (Packets/Bytes)
Classifier: AF-ftp
  Matched : 819(Packets) 1153392(Bytes)
  5-minute statistics:
    Forwarded: 2/26658 (pps/bps)
    Dropped  : 0/0 (pps/bps)
  Operator: AND
  Rule(s) : If-match acl 2001
  Behavior: AF-ftp
   Assured Forwarding:
    Bandwidth 50 (Kbps)
    Matched  : 639/883808 (Packets/Bytes)
    Enqueued : 639/883808 (Packets/Bytes)
    Discarded: 0/0 (Packets/Bytes)
    Discard Method: Tail
```

7.5 实验中的命令列表

表7-2 实验命令列表

命令	描述
entity *entity-number* { **pots** | **voip** }	创建语音实体
match-template *match-string*	配置语音实体的号码模板
line *line-number*	将语音实体与指定的语音用户线绑定
default entity compression	配置全局范围内编解码方式的缺省值
acl [**ipv6**] { **advanced** | **basic** } { *acl-number* | **name** *acl-name* } [**match-order** { **auto** | **config** }]	创建访问控制列表
traffic classifier *tcl-name* [**operator** { **and** | **or** }]	创建类
if-match *match-criteria*	为类定义规则
traffic behavior *behavior-name*	创建行为
queue ef bandwidth { *bandwidth* [**cbs** *burst*] | **pct** *percentage* [**cbs-ratio** *ratio*] }	配置EF队列
queue af bandwidth { *bandwidth* **| pct** *percentage* }	配置AF队列
qos policy *policy-name*	创建策略
classifier *tcl-name* **behavior** *behavior-name*	在策略中把类和行为绑定

7.6 思考题

1. 如果 PCA 和 PCB 通过 ftp 双向传输大文件，应该如何保证 RTA 和 RTB 之间的通话质量？

   答：需要在 RTA 和 RTB 设备的串口都进行拥塞管理配置，保证语音流优先。

2. 除了 CBQ 外，还可以使用哪种队列技术，既可以保证语音流量优先转发，又可以保证其它数据流可以分得合理的带宽？

   答：可以通过 RTPQ 队列和其它用户队列技术组合应用实现。