中国矿业大学
计算机科学与技术学院

2017 级本科生课程设计报告

课程名称： 网络系统与安全实践

班　　级： 信息安全 2017-1 班

姓　　名： 袁孝健、张弘毅、杨唯

报告时间： 2020.6.28

任课教师： 王虎

分　　工

姓名	完成工作情况
袁孝健	拓扑设计、搭建拓扑、配置路由器、测试验证、撰写实验报告
张弘毅	拓扑设计、物理连线、配置三层交换机、测试验证
杨唯	拓扑设计、拓扑绘制、配置二层交换机、测试验证

2019-2020 学年第二学期
《网络系统与安全实践》课程评分表

（小组成员每人单独一页）

姓名 袁孝健 学号 06172151 班级 信息安全 2017-1 班

编号	课程教学目标	考查方式及考查点	占比	得分
1	（3.4）**目标 1**：根据指定场景，独立设计网络互联与网络安全解决方案。	方案答辩； 解决方案的合理性、可行性和完备性，文档规范性；	30%	
2	（4.2）**目标 2**：掌握网络互联设备网络安全设备的配置和使用，能综合运用网络与安全的基本原理和技术，解决网络互联与网络安全的具体实践问题，具备设计与实施网络互联与网络安全工程的能力。	现场检查实验结果和配置文件，问题提问；	40%	
3	（9.1）**目标 3**：通过分组实验，锻炼合作团队意识，锻炼组长的统筹协调能力。	分工情况是否合理，各组员所完成任务情况；	10%	
4	（10.3）**目标 4**：能够就课程方案设计的过程撰写较为规范的课程报告，在课程报告审查和方案答辩过程中清晰表达设计思路、出现的问题及解决方案。	课程报告； 课程报告的规范性和正确性；	20%	
	总分			

评阅人：_____

2019-2020 学年第二学期
《网络系统与安全实践》课程评分表

（小组成员每人单独一页）

姓名 张弘毅 学号 06172149 班级 信息安全 2017-1 班

编号	课程教学目标	考查方式及考查点	占比	得分
1	（3.4）**目标 1**：根据指定场景，独立设计网络互联与网络安全解决方案。	方案答辩； 解决方案的合理性、可行性和完备性，文档规范性；	30%	
2	（4.2）**目标 2**：掌握网络互联设备网络安全设备的配置和使用，能综合运用网络与安全的基本原理和技术，解决网络互联与网络安全的具体实践问题，具备设计与实施网络互联与网络安全工程的能力。	现场检查实验结果和配置文件，问题提问；	40%	
3	（9.1）**目标 3**：通过分组实验，锻炼合作团队意识，锻炼组长的统筹协调能力。	分工情况是否合理，各组员所完成任务情况；	10%	
4	（10.3）**目标 4**：能够就课程方案设计的过程撰写较为规范的课程报告，在课程报告审查和方案答辩过程中清晰表达设计思路、出现的问题及解决方案。	课程报告； 课程报告的规范性和正确性；	20%	
	总分			

评阅人：_____

2019-2020 学年第二学期
《网络系统与安全实践》课程评分表

（小组成员每人单独一页）

姓名 杨 唯 学号 08172855 班级 信息安全 2017-1 班

编号	课程教学目标	考查方式及考查点	占比	得分
1	（3.4）**目标 1**：根据指定场景，独立设计网络互联与网络安全解决方案。	方案答辩； 解决方案的合理性、可行性和完备性，文档规范性；	30%	
2	（4.2）**目标 2**：掌握网络互联设备网络安全设备的配置和使用，能综合运用网络与安全的基本原理和技术，解决网络互联与网络安全的具体实践问题，具备设计与实施网络互联与网络安全工程的能力。	现场检查实验结果和配置文件，问题提问；	40%	
3	（9.1）**目标 3**：通过分组实验，锻炼合作团队意识，锻炼组长的统筹协调能力。	分工情况是否合理，各组员所完成任务情况；	10%	
4	（10.3）**目标 4**：能够就课程方案设计的过程撰写较为规范的课程报告，在课程报告审查和方案答辩过程中清晰表达设计思路、出现的问题及解决方案。	课程报告； 课程报告的规范性和正确性；	20%	
	总分			

评阅人：_____

目 录

1 实验设备

（1）RSR20 路由器：2 台。
（2）S3760E 三层交换机：3 台。
（3）S2628G-I 二层交换机：2 台。
（4）Windows PC 机：4 台。

2 实验背景

某一大型企业内部网络由母公司 Site 1 和子公司 Site 2 组成，其中母公司 Site 1 下包含部门 Office 1 和 Office 2，子公司 Site 2 下包含部门 Office 3 和 Office 4，总部与分公司间网络通过 Tunnel 打通路由。

3 网络连通性要求

3.1 Site 1

母公司 Site 1 的部门 Office 1 和 Office 2 分别隶属于 VLAN 10 和 VLAN 20，它们的网关分别指向 SW1 的 SVI 10、SVI 20 接口。SW1 和边界路由器 R1 之间启用动态路由协议 OSPF，并在 area 0 中宣告所有本地路由。SW1 中开启快速生成树协议（RSTP）来提高网络的可靠性，同时避免环路。

经配置，应该满足位于不同部门的 PC1、PC2 可以相互通信，并且 R1 与 SW1 建立路由邻居并收到 vlan10、20 的路由明细。

3.2 Site 2

子公司 Site 2 的部门 Office 3 和 Office 4 分别隶属于 vlan30、vlan40。SW2、SW3 之分别与边界路由器 R2 建立 OSPF 邻居，在区域 0 中宣告所有本地直连路由。除此之外，SW2 和 SW3 之间启用 Trunk 放行 VLAN，并用了多生成树协议（MST），能够通过 trunks 建立多个生成树，关联 VLANs 到相关的生成树进程，提供了多个数据转发路径和负载均衡，提高了网络容错能力。

经配置，位于不同部门的 PC3、PC4 可以相互通信，R2 与 SW2、SW3 建立 OSPF 邻居并收到 VLAN30、VLAN40 的路由明细。

3.3 Tunnel

GRE（通用路由封装）协议是 tunnel 技术中的一种协议，它对某些网络层协议的数据报文进行封装，使这些被封装的数据报文能够在另一个网络层协议中传输。GRE 提供了一条通路使封装的数据报文能够在这个通路上传输，并且在一个 Tunnel 的两端分别对数据报进行封装及解封装。

母公司（Site 1）于子公司（Site 2）通过在 R1、R2 上起 Tunnel 来打通路由，源目的地址分别为自己和对端的串口，并 Tunnel 隧道建立 OSPF 邻居。

经配置， R1、R2 可以成功建立 OSPF 邻居，Site 1、Site 2 互传路由明细，PC1、PC2、PC3、PC4 四个部门可以相互通信。

3.4 NATP

NAT（Network Address Translation，网络地址转换），可以使用少量的公有 IP 地址代表较多的私有 IP 地址的方式，将有助于减缓可用的 IP 地址空间的枯竭。除此之外，NAT 还能够有效地避免来自网络外部的攻击，隐藏并保护网络内部的计算机。

由于 NAT 实现是私有 IP 和 NAT 的公共 IP 之间的转换，那么，私有网中同时与公共网进行通信的主机数量就受到 NAT 的公共 IP 地址数量的限制。为了克服这种限制，NAT 被进一步扩展到在进行 IP 地址转换的同时进行 Port 的转换，这就是网络地址端口转换 NAPT（Network Address Port Translation）技术。

首先需要在 R2 的 lo 0 接口模拟公网 IP：8.8.8.8，而 R1 作为 Site 1 唯一网络出口，默认路由指向外网接口 s2/0，并下发默认路由。R1 的 s2/0 口上开启端口复用 NAT 对所有来自 Site 1 内部访问外网（8.8.8.8）的流量进行地址转换。

经配置，从 Site 1 去往外部网络的流量均实现了 NATP 地址转换。

3.5 ACL

ACL（访问控制列表）是指对流经路由器或交换机的数据包根据一定的规则进行过滤的控制，可以提高网络可管理性和安全性。

根据需求，编写标准 ACL 在 SW2 的入方向放行部门 Office 3 到所有目标地址的流量。编写拓展 ACL 接口下调用在 SW3 入方向只拒绝部门 Office 4 访问外网 8.8.8.8 的流量。

经配置，四个部门的所有 PC 可以相互通信；除 PC4 之外，均能访问公网地址 8.8.8.8。

4 网络安全性要求

4.1 VRRP

VRRP（虚拟路由冗余网关）是一种容错协议，它保证当主机的下一跳路由器出现故障时，由另一台路由器来代替出现故障的路由器进行工作，从而保持网络通信的连续性和可靠性。

在 SW3 和 SW4 上配置 VRRP，VLAN 30 的主虚拟网关位于 SW3，VLAN 40 的主虚拟网关位于 SW4。当交换机检测上行链路转发故障时自动降低本地 VRRP 进程优先级，虚拟网关身份切换到 peer 端。

4.2 IPSec

互联网安全协议（英语：Internet Protocol Security，缩写为 IPsec），是一个协议包，通过对 IP 协议的分组进行加密和认证来保护 IP 协议的网络传输协议族。IPsec 主要由以下协议组成：

- 认证头（AH）：为 IP 数据报提供无连接数据完整性、消息认证以及防重放攻击保护。
- 封装安全载荷（ESP）：提供机密性、数据源认证、无连接完整性、防重放和有限的传输流（traffic-flow）机密性。
- 安全关联（SA）：提供算法和数据包，提供 AH、ESP 操作所需的参数。

在该企业网络中，为了保证母公司与子公司之间路由通信的安全，用 IPSEC 加密 Tunnel 隧道，模式为隧道模式。规定 IKE 第一阶段采用预共享密钥的方式建立安全关联，IKE 第二阶段采用 AES-256 加密数据、Sha-HMAC 用于数据哈希校验。

4.3 MAC 地址绑定

交换机的端口安全，是一种交换机的过滤策略，即为交换机的某个端口绑定一个固定的 MAC 地址，使其他的 MAC 地址访问的时候触发策略，关闭掉端口或者拒绝服务。

在 SW2 和 SW3 交换口上设置端口安全自动 MAC 地址绑定，如果检测到主机 MAC 改动立即关闭端口，从而严格控制了输入，防止了身份的伪造。

4.4 端口限速

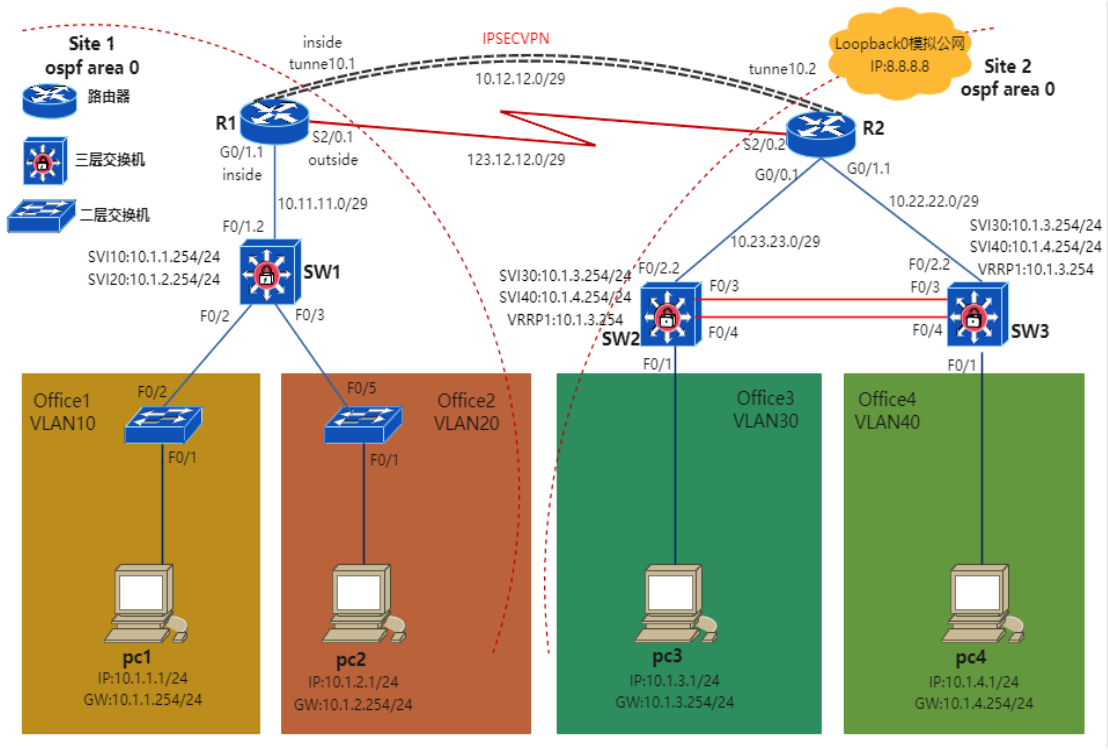
母公司部门 Office 1 的网络管理员最近收到很多员工的投诉，抱怨网络变得很慢， 不论是收发邮件还是上网查资料都很慢，影响了工作效率。对此，网络管理员进行了调查，发现有交换机的某些端口的数据流量很大，严重影响了网络性能， 于是决定对这个交换机端口进行速率限制，从而改进网络性能。

在 SW1 中定义分类映射图（class-map）和策略映射图（policy-map），设置带宽限制、猝发数据量限制，若用户使用超过此限制则直接丢弃数据包，在 f0/2 端口启用 QoS，并设置接口的 QoS 信任模式为 cos。

5 网络拓扑

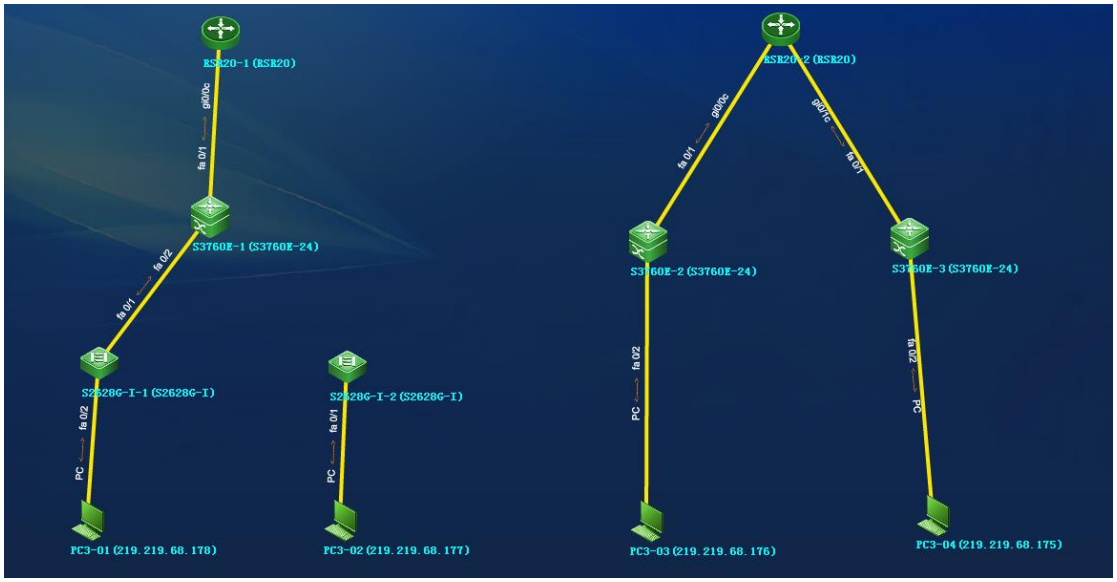
5.1 拓扑构建

根据企业网络架构，设计出的网络拓扑图如下：



5.2 拓扑仿真

在锐捷仿真平台上绘制的实验拓扑如下：



除了上图中通过平台连线之外，下面这些线直接在机柜中进行物理线路的连接：

- RSR20-1 的 s2/0 口与 RSR20-2 的 s2/0 口
- S3760E-1 的 3 口与 S2628G-1-2 的 5 口

- S3760E-2 与 S3760E-3 的 3 口
- S3760E-2 与 S3760E-3 的 4 口

6 网络配置

6.1 连通性配置

6.1.1 R1

```
enable
configure terminal
hostname R1
interface gi0/1 //给接口配置 ip
ip address 10.11.11.1 255.255.255.248
no shutdown
interface s2/0
ip address 123.12.12.1 255.255.255.248
no shutdown
interface tunnel 0
//配置 tunnel 口，设置模式、协议、IP 地址、源目
tunnel mode gre ip
tunnel source 123.12.12.1
tunnel destination 123.12.12.2
ip address 10.12.12.1 255.255.255.248
no shutdown
router ospf 1 //ospf 进程 1
network 10.11.11.0 0.0.0.7 area 0 //宣告接口
network 10.12.12.0 0.0.0.7 area 0
default-info originate //给邻居下发默认路由
ip route 0.0.0.0 0.0.0.0 ser2/0 //配置静态默认路由
ip access-list extend NAT //拓展 ACl NAT
permit ip 10.1.0.0 0.0.255.255 hostnamet
8.8.8.8 //允许源自 10.1.0.0/
16 的 ip 层流量访问主机 8.8.8.8
exit //退出
ip nat inside source list NAT interface s2/0
overload //动态 nat 在 s2/0 接口端口复用
interface s2/0
ip nat outside //nat 流量为出方向
interface tunnel0
ip nat inside //nat 流量进方向
interface gi0/1
ip nat inside //nat 流量进方向
```

6.1.2 R2

```
enable
configure terminal
hostname R2
interface gi0/0 //打开接口配置 ip
ip address 10.22.22.1 255.255.255.248
no shutdown
interface gi0/1
ip address 10.23.23.1 255.255.255.248
no shutdown
interface s2/0
ip address 123.12.12.2 255.255.255.248
no shutdown
interface tunnel 0 //进入 tunnel 口 0
tunnel mode gre ip //tunnel 模式为 gre，ip 支持 ipv4
tunnel source 123.12.12.2 //设置 tunnel 源
tunnel destination 123.12.12.1 //设置 tunnel 目的
ip address 10.12.12.2 255.255.255.248 //给 tunnel 口配置 ip 地址
no shutdown //开启接口
interface lo 0 //进入环回接口 loopback0
ip address 8.8.8.8 255.255.255.255 //配置 ip
router ospf 1 //ospf 进程 1
network 10.22.22.0 0.0.0.7 area 0 //在 areaa 0 宣告路由
network 10.23.23.0 0.0.0.7 area 0
network 10.12.12.0 0.0.0.7 area 0
```

6.1.3 SW1

```
enable //修改主机名
configure terminal
hostname switch1
spanning-treenableing-tree //开启生成树
spanning-treenableing-tree mode rstp
vlan 10 //创建 vlan
vlan 20
interface f0/2 //划分 vlan
switch mode access
switch access vlan 10
no shutdown
interface f0/3
switch mode access
switch access vlan 20
no shutdown
interface vlan 10 //进入 svi 口
```

```
ip address 10.1.1.254 255.255.255.0 //设置 svi 的 ip 地址
no shutdown //打开接口
interface vlan 20 //设置 svi 口
ip address 10.1.2.254 255.255.255.0
no shutdown
interface f0/1 //进入接口
no switch //关闭交换功能（打开路由功能）
ip address 10.11.11.2 255.255.255.248 //配置 ip
no shutdown //开启接口
router ospf 1 //开启 ospf 进程 1
network 10.1.1.0 0.0.0.255 area 0 //
在 area0 中宣告网段 10.1.1.0/24
network 10.1.2.0 0.0.0.255 area 0 //宣告网段 10.1.2.0/24
network 10.11.11.0 0.0.0.7 area 0 //宣告网段
10.11.11.0/29
```

6.1.4 SW2

```
enable //修改主机名
configure terminal
hostname switch2
vlan 30 //创建 vlan
vlan 40
interface vlan 30
ip address 10.1.3.254 255.255.255.0
no shutdown
interface r f0/3-4
switch mode trunk
no shutdown
interface f0/1 //vlan 划分
switch mode access
switch access vlan 30
no shutdown
spanning-tree //开启生成树
spanning-tree mode mst //生成树模式 mst
spanning-tree mst conf //配置 mst
instance 1 vlan 30 //划分 vlan30 到 mst 实例 1
instance 2 vlan 40
spanning-tree mst 1 prio 0 //配置实例 1 优先级（本地最高）
spanning-tree mst 2 prio 4096 //配置实例 2 优先级
interface f0/2 //关闭交换功能配置三层 ip
no switch
ip address 10.22.22.2 255.255.255.248
no shutdown
router ospf 1 //开启 ospf 进程并在 areaa 0 中宣告路由
```

```
network 10.22.22.0 0.0.0.7 area 0
network 10.1.3.0 0.0.0.255 area 0
network 10.1.4.0 0.0.0.255 area 0
ip access-list stand 10 //标准的访问控制列表 10
permit hostnamet 10.1.3.1
//放行源地址是 10.1.3.1 的所有流量
interface f0/1 //进入接口
ip access-group 10 in //将 ACL10 接口下调用在接口的入向
```

6.1.5 SW3

```
enable //修改主机名
configure terminal
hostname switch3
vlan 30 //
vlan 40 //创建 vlan40 并设置 svi40 接口
interface vlan 40
ip address 10.1.4.254 255.255.255.0
no shutdown
interface r f0/3-4
switch mode trunk
no shutdown
interface f0/1 //vlan 划分
switch mode access
switch access vlan 40
no shutdown
spanning-tree //配置 mst 生成树
spanning-tree mode mst
spanning-tree mst conf
instance 2 vlan 40
instance 1 vlan 30
spanning-tree mst 2 prio 0
spanning-tree mst 1 prio 4096
interface f0/2 //关闭交换功能，打开路由功能
no switch
ip address 10.23.23.2 255.255.255.248
no shutdown
router ospf 1 //开启 ospf 进程 1 并宣告网段
network 10.23.23.0 0.0.0.7 area 0
network 10.1.4.0 0.0.0.255 area 0
network 10.1.3.0 0.0.0.255 area 0
ip access-list extenabled 100 //拓展访问控制列表 100
deny ip hostnamet 10.1.4.1 host 8.8.8.8
//拒绝主机 10.1.4.1 访问主机 8.8.8.8
permit ip any any //放行所有流量
```

```
interface f0/1 //进入接口 f0/1 并在入方向接口下调用 ACL100
ip access-group 100 in
```

6.1.6 SW11

```
enable
configure terminal //特权模式
hostname switch11 //命名
vlan 10 //创建 vlan10
spanning-tree //开启生成树
spanning-tree mode rstp //
设置生成树模式 rstp
interface f0/1 //进入接口
switch mode access //设置接口模式
switch access vlan 10 //给接口划分 vlan
no shutdown //打开接口
interface f0/2 //划分 vlan
switch mode access
switch access vlan 10
no shutdown
```

6.1.7 SW12

```
enable //进入特权模式修改主机名
configure terminal
hostname switch12
vlan 20 //创建 vlan
spanning-tree //开启生成树
spanning-tree mode rstp
interface f0/1 //划分 vlan
switch mode access
switch access vlan 20
no shutdown
interface f0/5 //划分 vlan
switch mode access
switch access vlan 20
no shutdown
```

6.2 安全性配置

6.2.1 VRRP

（1）SW2

```
int vlan 30
ip address 10.1.3.252 255.255.255.0
vrrp 1 version 2 //vrrp 进程 1 版本 2
```

vrrp 1 ip 10.1.3.254 //虚拟网关 10.1.3.254
vrrp 1 prio 100 //本地进程优先级 100（主）
vrrp 1 preempt //开启抢占，进程优先级高的会抢占成为主设备
vrrp 1 track f0/2 20 //监控 f0/2 状态，如果异常优先级降低 20
Int vlan40
Ip add 10.1.4.252 255.255.255.0
vrrp 2 version 2 //进程 1 版本 2
vrrp 2 ip 10.1.4.254 //虚拟网关 10.1.4.254
vrrp 2 prio 99 //本地进程优先级 99（备）
vrrp 2 preempt //开启抢占
vrrp 2 track f0/2 20 /监控 f0/2 口状态，异常降低优先级

（2）SW3

int vlan 30
ip address 10.1.3.253 255.255.255.0
vrrp 1 version 2 //版本
vrrp 1 ip 10.1.3.254 //虚拟网关
vrrp 1 prio 99 //优先级（备）
vrrp 1 pre //抢占
vrrp 1 track f0/2 20 //监控端口
int vlan 40
ip add 10.1.4.253 255.255.255.0
vrrp 2 version 2 //版本
vrrp 2 ip 10.1.4.254 //虚拟网关
vrrp 2 prio 100 //优先级（主）
vrrp 2 pre //抢占
vrrp 2 track f0/2 20 //监控端口

6.2.2 IPSec

（1）R1

ip access-list extend 100 //拓展 ACL 抓取加密感兴趣流
permit ip 10.0.0.0 0.0.0.255 host 10.0.0.0
crypto iskamp policy 10 //ike 第一阶段 策略 10
encry 3des //加密算法 3des
authen pre-share //协商方法预共享密钥
group 2 //密钥长度 1024
crypto iskamp key 7 ruijie add 10.12.12.2 //加密的共享密钥 ruijie，对端 ip10.12.12.2
crypto ipsec transform-set IPSEC esp-aes-256 esp-sha-hmac //ike 第二阶段 设置传输集 IPSEC，约定 esp 协议封装数据包、加密算法 256 位 aes、哈希算法 sha
mode tunnel //加密模式位传输
crypto map VPN 1 ipsec-iskamp //配置加密映射表 VPN 策略 1
set transform-set IPSEC //设定传输集 IPSEC

```
set peer 10.12.12.2 //设置对端 ip10.12.12.2
match add 100 //匹配感兴趣流量
int tunnel0
crypto map VPN //接口下调用加密策略
```

（2）R2

```
ip access-list extend 100 //同上
permit ip 10.0.0.0 0.0.0.255 host 10.0.0.0
crypto iskamp policy 10
encry 3des
authen pre-share
group 2
crypto iskamp key 7 ruijie add 10.12.12.1
crypto ipsec transform-set IPSEC esp-aes-256 esp-sha-hmac
mode tunnel
crypto map VPN 1 ipsec-iskamp
set transform-set IPSEC
set peer 10.12.12.1
match add 100
int tunnel0
crypto map VPN
Port-security
```

6.2.3 MAC 地址绑定

（1）SW2

```
interface f0/1
sw port-sec mac-address sticky //端口安全自动绑定 mac
sw port-sec violation shutdown //发生违规自动关闭窗口
```

（2）SW3

```
interface f0/1
sw port-sec mac-address sticky //端口安全自动绑定 mac
sw port-sec violation shutdown //发生违规自动关闭窗口
```

6.2.4 端口限速

（1）SW1

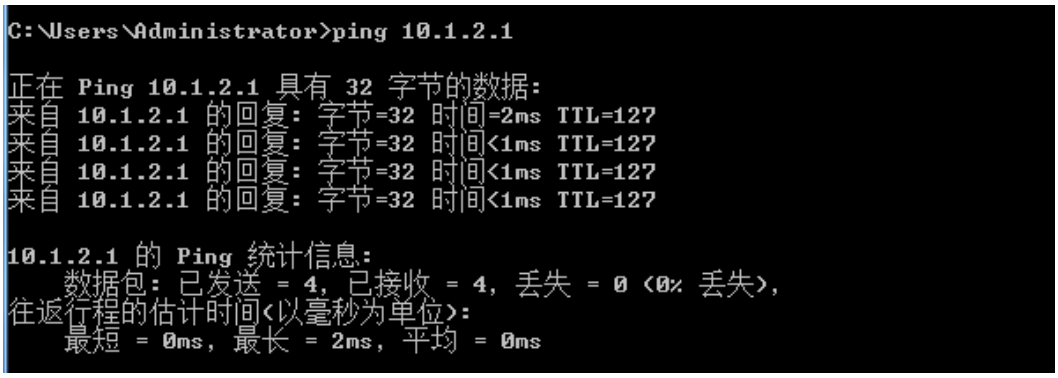
```
enable
configure terminal
ip access-list standard qoslimit1 //定义访问控制列表
permit host 10.1.1.254 //定义需要限速的数据流
exit
class-map classmap1 //设置分类映射图
match access-group qoslimit1 //匹配访问控制列表
exit
```

```
policy-map policymap1 //设置策略映射图
class classmap1 //匹配分类映射图
police 1000000 65536 exceed-action drop //带宽限制为 1Mbps，猝
发数据量为 64k/sec
exit
interface fa0/2
mls qos trust cos //启动 Qos，并且设置信任模式为 cos
service-policy input policymap1 //应用策略
```

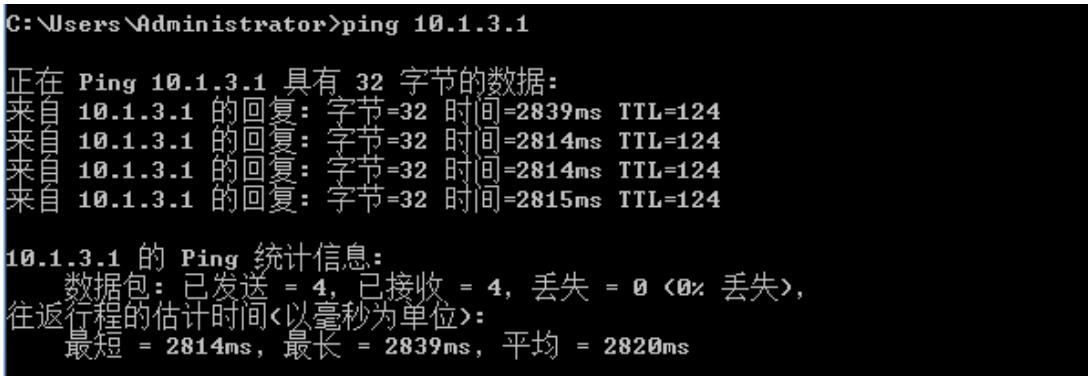
7 结果验证

7.1 验证 PC1 与各关键节点互通

（1）PC1 → PC2



（2）PC1 → PC3



（3）PC1 → PC4

（4）PC1 → SW1

```
C:\Users\Administrator>ping 10.1.1.254

正在 Ping 10.1.1.254 具有 32 字节的数据:
来自 10.1.1.254 的回复: 字节=32 时间=4ms TTL=64
来自 10.1.1.254 的回复: 字节=32 时间=3ms TTL=64
来自 10.1.1.254 的回复: 字节=32 时间=3ms TTL=64
来自 10.1.1.254 的回复: 字节=32 时间=7ms TTL=64

10.1.1.254 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 = 3ms, 最长 = 7ms, 平均 = 4ms
```

（5）PC1 → R1

```
C:\Users\Administrator>ping 10.11.11.1

正在 Ping 10.11.11.1 具有 32 字节的数据:
来自 10.11.11.1 的回复: 字节=32 时间<1ms TTL=63
来自 10.11.11.1 的回复: 字节=32 时间<1ms TTL=63
来自 10.11.11.1 的回复: 字节=32 时间<1ms TTL=63
来自 10.11.11.1 的回复: 字节=32 时间<1ms TTL=63

10.11.11.1 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
```

（6）PC1 → 8.8.8.8

```
C:\Users\Administrator>ping 8.8.8.8

正在 Ping 8.8.8.8 具有 32 字节的数据:
来自 8.8.8.8 的回复: 字节=32 时间=837ms TTL=62
来自 8.8.8.8 的回复: 字节=32 时间=630ms TTL=62
来自 8.8.8.8 的回复: 字节=32 时间=964ms TTL=62
来自 8.8.8.8 的回复: 字节=32 时间=828ms TTL=62

8.8.8.8 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 = 630ms, 最长 = 964ms, 平均 = 814ms
```

7.2 验证 PC2 与各关键节点互通

（1）PC2 → PC1

```
C:\Users\Administrator>ping 10.1.1.1

正在 Ping 10.1.1.1 具有 32 字节的数据:
来自 10.1.1.1 的回复: 字节=32 时间<1ms TTL=127
来自 10.1.1.1 的回复: 字节=32 时间<1ms TTL=127
来自 10.1.1.1 的回复: 字节=32 时间<1ms TTL=127
来自 10.1.1.1 的回复: 字节=32 时间<1ms TTL=127

10.1.1.1 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
```

（2）PC2 → PC3

```
C:\Users\Administrator>ping 10.1.3.1

正在 Ping 10.1.3.1 具有 32 字节的数据:
来自 10.1.3.1 的回复: 字节=32 时间=3443ms TTL=124
来自 10.1.3.1 的回复: 字节=32 时间=3291ms TTL=124
来自 10.1.3.1 的回复: 字节=32 时间=3276ms TTL=124
来自 10.1.3.1 的回复: 字节=32 时间=3165ms TTL=124

10.1.3.1 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 = 3165ms, 最长 = 3443ms, 平均 = 3293ms
```

（3）PC2 → PC4

```
C:\Users\Administrator>ping 10.1.4.1

正在 Ping 10.1.4.1 具有 32 字节的数据:
请求超时。
来自 10.1.4.1 的回复: 字节=32 时间=3059ms TTL=124
来自 10.1.4.1 的回复: 字节=32 时间=3073ms TTL=124
来自 10.1.4.1 的回复: 字节=32 时间=3032ms TTL=124

10.1.4.1 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 = 3032ms, 最长 = 3073ms, 平均 = 3054ms
```

（4）PC2 → SW1

```
C:\Users\Administrator>ping 10.1.2.254

正在 Ping 10.1.2.254 具有 32 字节的数据:
来自 10.1.2.254 的回复: 字节=32 时间=1ms TTL=64
来自 10.1.2.254 的回复: 字节=32 时间=1ms TTL=64
来自 10.1.2.254 的回复: 字节=32 时间=1ms TTL=64
来自 10.1.2.254 的回复: 字节=32 时间=1ms TTL=64

10.1.2.254 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 = 1ms, 最长 = 1ms, 平均 = 1ms
```

（5）PC2 → R1

```
C:\Users\Administrator>ping 10.11.11.1

正在 Ping 10.11.11.1 具有 32 字节的数据:
来自 10.11.11.1 的回复: 字节=32 时间<1ms TTL=63
来自 10.11.11.1 的回复: 字节=32 时间<1ms TTL=63
来自 10.11.11.1 的回复: 字节=32 时间<1ms TTL=63
来自 10.11.11.1 的回复: 字节=32 时间<1ms TTL=63

10.11.11.1 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
```

（6）PC2 → 8.8.8.8

```
C:\Users\Administrator>ping 8.8.8.8

正在 Ping 8.8.8.8 具有 32 字节的数据:
来自 8.8.8.8 的回复: 字节=32 时间=587ms TTL=62
来自 8.8.8.8 的回复: 字节=32 时间=517ms TTL=62
来自 8.8.8.8 的回复: 字节=32 时间=656ms TTL=62
来自 8.8.8.8 的回复: 字节=32 时间=596ms TTL=62

8.8.8.8 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 = 517ms, 最长 = 656ms, 平均 = 589ms
```

7.3 验证 PC3 与各关键节点互通

（1）PC3 → PC1

```
C:\Users\Administrator>ping 10.1.1.1

正在 Ping 10.1.1.1 具有 32 字节的数据:
来自 10.1.1.1 的回复: 字节=32 时间=3301ms TTL=124
来自 10.1.1.1 的回复: 字节=32 时间=3345ms TTL=124
来自 10.1.1.1 的回复: 字节=32 时间=3353ms TTL=124
来自 10.1.1.1 的回复: 字节=32 时间=3310ms TTL=124

10.1.1.1 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 = 3301ms, 最长 = 3353ms, 平均 = 3327ms
```

（2）PC3 → PC2

```
C:\Users\Administrator>ping 10.1.2.1

正在 Ping 10.1.2.1 具有 32 字节的数据:
来自 10.1.2.1 的回复: 字节=32 时间=3082ms TTL=124
来自 10.1.2.1 的回复: 字节=32 时间=3090ms TTL=124
来自 10.1.2.1 的回复: 字节=32 时间=3141ms TTL=124
来自 10.1.2.1 的回复: 字节=32 时间=3166ms TTL=124

10.1.2.1 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 = 3082ms, 最长 = 3166ms, 平均 = 3119ms
```

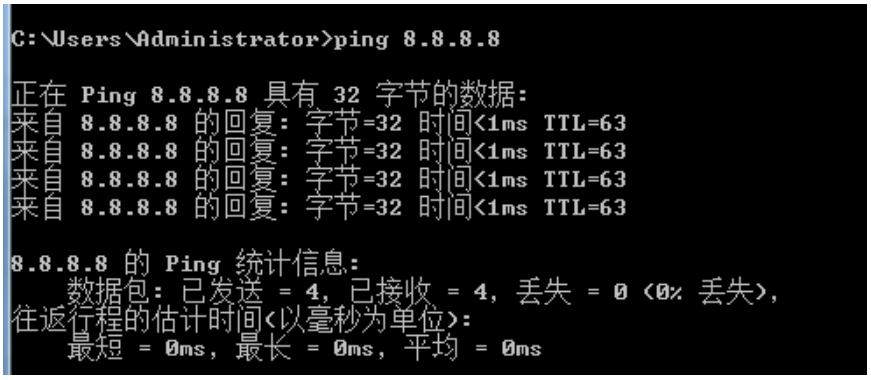
（3）PC3 → PC4

```
C:\Users\Administrator>ping 10.1.4.1

正在 Ping 10.1.4.1 具有 32 字节的数据:
来自 10.1.4.1 的回复: 字节=32 时间<1ms TTL=125
来自 10.1.4.1 的回复: 字节=32 时间<1ms TTL=125
来自 10.1.4.1 的回复: 字节=32 时间<1ms TTL=125
来自 10.1.4.1 的回复: 字节=32 时间<1ms TTL=125

10.1.4.1 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

C:\Users\Administrator>
```

（4）PC3 → SW2

（5）PC3 → R2

（6）PC3 → 8.8.8.8



7.4 验证 PC4 与各关键节点互通

（1）PC4 → PC1

（2）PC4 → PC2

```
C:\Users\Administrator>ping 10.1.2.1

正在 Ping 10.1.2.1 具有 32 字节的数据:
来自 10.1.2.1 的回复: 字节=32 时间=3014ms TTL=124
来自 10.1.2.1 的回复: 字节=32 时间=2978ms TTL=124
来自 10.1.2.1 的回复: 字节=32 时间=2973ms TTL=124
来自 10.1.2.1 的回复: 字节=32 时间=3021ms TTL=124

10.1.2.1 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 = 2973ms, 最长 = 3021ms, 平均 = 2996ms
```

（3）PC4 → PC3

```
C:\Users\Administrator>ping 10.1.3.1

正在 Ping 10.1.3.1 具有 32 字节的数据:
来自 10.1.3.1 的回复: 字节=32 时间<1ms TTL=125
来自 10.1.3.1 的回复: 字节=32 时间<1ms TTL=125
来自 10.1.3.1 的回复: 字节=32 时间<1ms TTL=125
来自 10.1.3.1 的回复: 字节=32 时间<1ms TTL=125

10.1.3.1 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
```

（4）PC4 → SW3

```
C:\Users\Administrator>ping 10.1.4.254

正在 Ping 10.1.4.254 具有 32 字节的数据:
来自 10.1.4.254 的回复: 字节=32 时间=1ms TTL=64
来自 10.1.4.254 的回复: 字节=32 时间=1ms TTL=64
来自 10.1.4.254 的回复: 字节=32 时间=1ms TTL=64
来自 10.1.4.254 的回复: 字节=32 时间=1ms TTL=64

10.1.4.254 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 = 1ms, 最长 = 1ms, 平均 = 1ms
```

（5）PC4 → R2

```
C:\Users\Administrator>ping 10.22.22.1

正在 Ping 10.22.22.1 具有 32 字节的数据:
来自 10.22.22.1 的回复: 字节=32 时间<1ms TTL=63
来自 10.22.22.1 的回复: 字节=32 时间<1ms TTL=63
来自 10.22.22.1 的回复: 字节=32 时间<1ms TTL=63
来自 10.22.22.1 的回复: 字节=32 时间<1ms TTL=63

10.22.22.1 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
```

（6）PC4 → 8.8.8.8（由于配置了 ACL，所以无 ping 通）

```
C:\Users\Administrator>ping 8.8.8.8

正在 Ping 8.8.8.8 具有 32 字节的数据:
请求超时。
请求超时。
请求超时。
请求超时。

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

7.5 IPsec Tunnel 验证

（1）PC2 → PC4

```
C:\Users\Administrator>tracert 10.1.4.1

通过最多 30 个跃点跟踪
到 WL21-16 [10.1.4.1] 的路由:

  1     1 ms     1 ms     1 ms  10.1.2.254
  2    <1 毫秒   <1 毫秒   <1 毫秒 10.11.11.1
  3  2659 ms  2696 ms  2698 ms  10.12.12.2
  4     *        *     3164 ms  10.22.22.2
  5  3047 ms     *     2971 ms  WL21-16 [10.1.4.1]

跟踪完成。
```

（2）PC3 → PC1

```
C:\Users\Administrator>tracert 10.1.1.1

通过最多 30 个跃点跟踪
到 WL21-19 [10.1.1.1] 的路由:

  1     7 ms     1 ms     1 ms  10.1.3.252
  2    <1 毫秒   <1 毫秒   <1 毫秒 10.23.23.1
  3  3447 ms     *        *     10.12.12.1
  4  3565 ms     *     3540 ms  10.11.11.2
  5  3451 ms  3328 ms  3464 ms  WL21-19 [10.1.1.1]

跟踪完成。
```