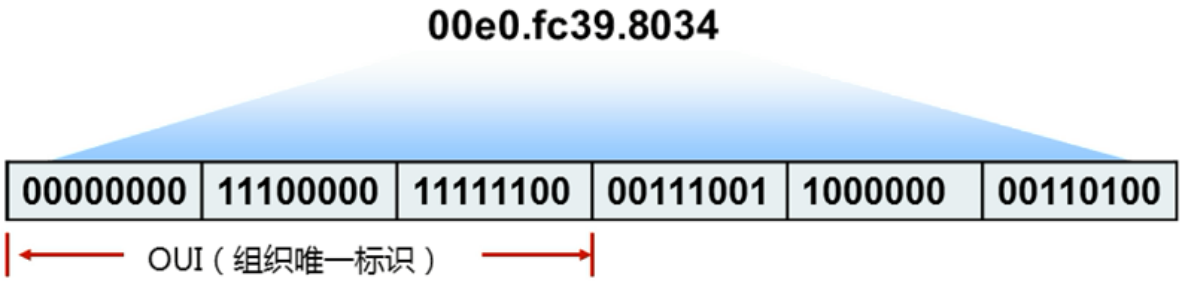
MAC地址

- mac地址有48位，通常表示为点分16进制
- mac地址全球唯一，由IEEE对这些地址进行管理
- 每个地址由两部分租场，分别是供应商代码和序列号。其中前24位代表的是供应商，剩下的24位代表的是序列号



交换机工作机制

1. 读取数据帧头部的源MAC地址，并且将接口和MAC地址记录到MAC地址表
2. 读取数据帧头部的目的MAC地址，并且和自己的MAC地址表进行对应
3. 1. 匹配成功，则从对应的接口发出
   2. 没有匹配成功，则会从收到的那个接口之外的所有接口发出
4. 1. 广播或者组播地址，则会从收到的那个接口之外的所有接口发出

路由器工作机制

1. 读取数据帧头部的源MAC地址，并且将接口和MAC地址记录到MAC地址表
2. 读取数据帧头部的目的MAC地址，并且和自己的MAC地址表进行对应
3. 匹配成功。打开IP报文，继续取IP地址
4. 1. 如果是自己则接受
5. 2. 如果不是自己根据路由表转发
6. 1. 没有匹配成功
7. 1. 丢弃

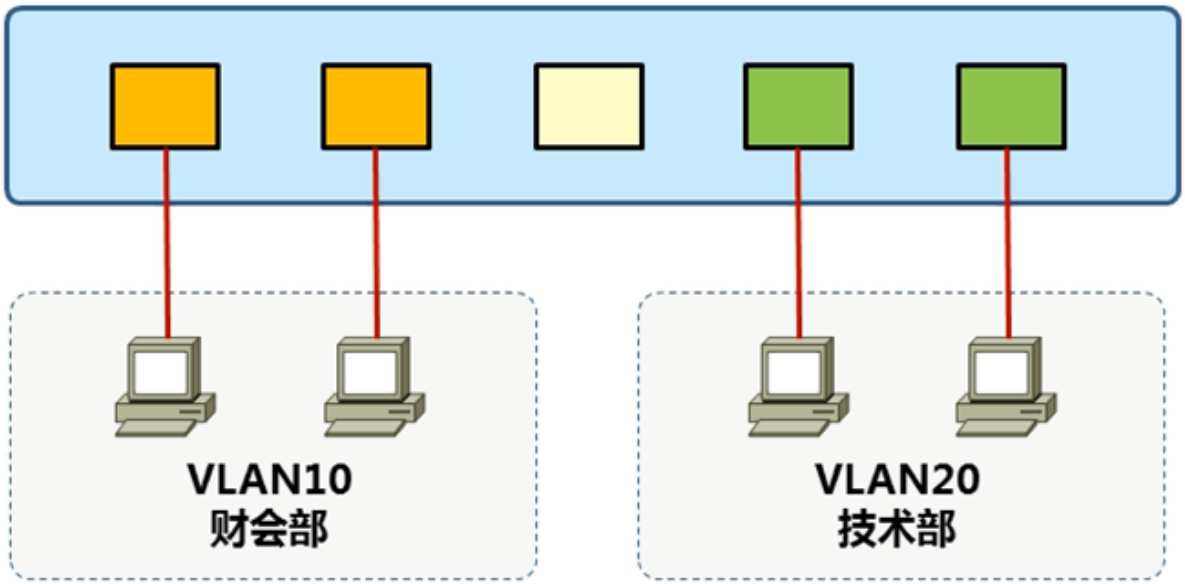
LAN

- 一般指局域网
- 通常会使用到的设备有网卡、集线器、交换机
- 一般是指私有网络

VLAN

- 一个VLAN中所有设备都是在同一个广播域内，不同的VLAN是不同的广播域
- VLAN之间相互隔离，广播域不能跨越VLAN传播，不同的VLAN间需要通告三层设备实现互相通信
- 一个VLAN一般位一个逻辑子网，由多个VLAN设备
- VLAN多基于交换机的接口分配，划分VLAN成员就是对交换机的接口划分
- VLAN工作于OSI参考模型第二层

- VLAN是二层交换机的一个根本的工作机制

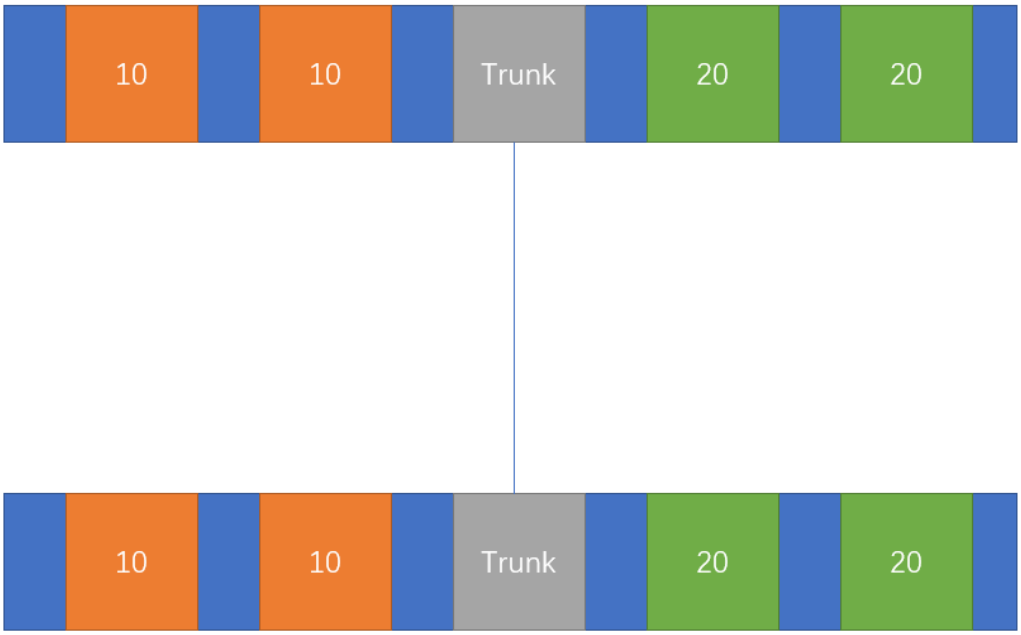


VLAN范围	作用
0，4095	保留，系统使用
1	cisco默认VLAN
2-1001	For Ethernet VLANs
1002-1005	Cisco默认为FDDI及TokenRing定义
1006-4094	只能为Ethernet使用，在一些特殊平台被保留使用

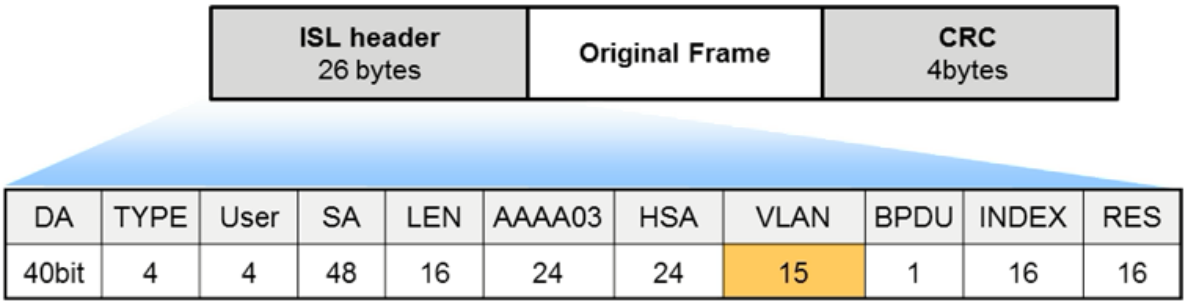
- 如果需要将两台交换机连接到一起，那么每个vlan都用一根网线连接起来是现实的，多一个可以使用Trunk解决这个问题

Trunk

- 当一条链路，需要承载多条vlan信息的时候，需要使用trunk来实现
- trunk两端采用相同的干道协议
- 一般见于交换机之间或者交换机于路由器、服务器之间

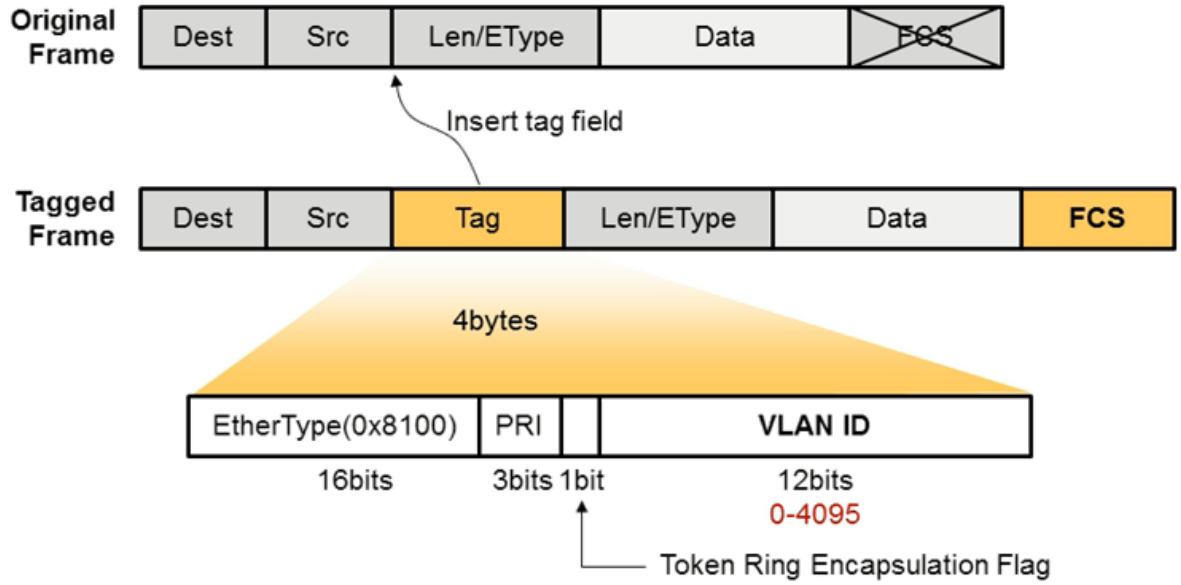


ISL



- cisco私有
- 支持PVST
- 在原始的数据帧基础上装上ISL头以及新的FCS
- 处理效率比802.1Q低
- 最多支持1024个vlan

802.1Q



- IEEE共有协议
- 插入tag字段，同时重新计算FCS

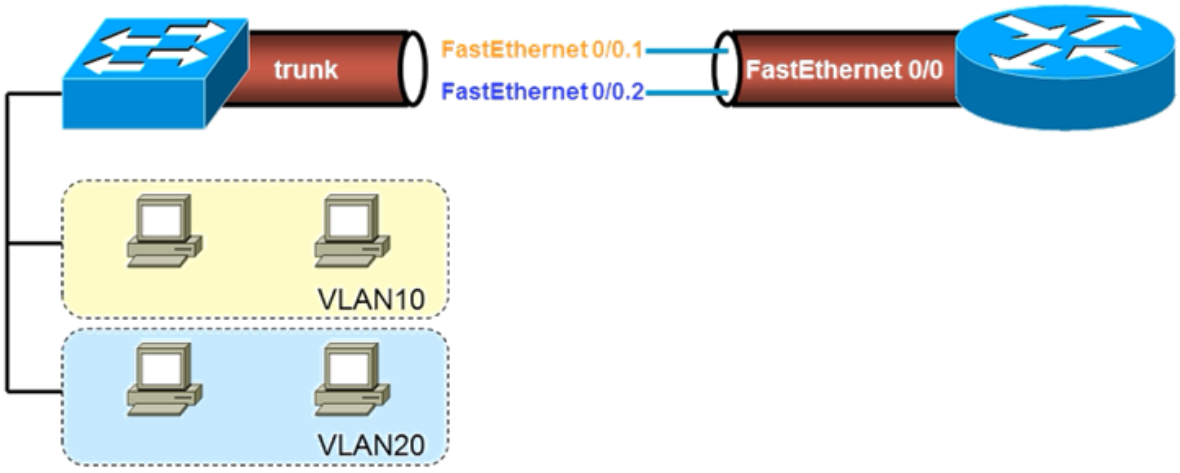
三层交换

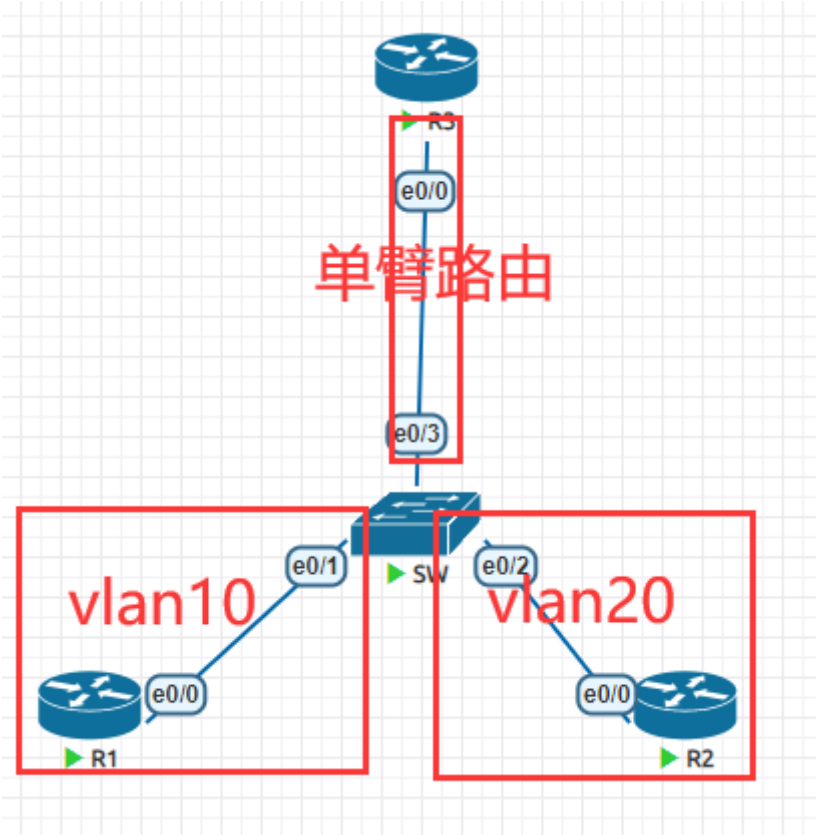
vlan间的通信

- vlan是将一个广播域给隔离成多个广播域互不干扰
- 不同广播域的PC往往在可控的情况下是需要进行通信的，所以需要解决vlan之间的通信

单臂路由

- 单臂路由可以解决vlan间的通信问题
- 但是这种方式第一不够安全可靠，成本较高
- 这种方式往往是临时解决问题



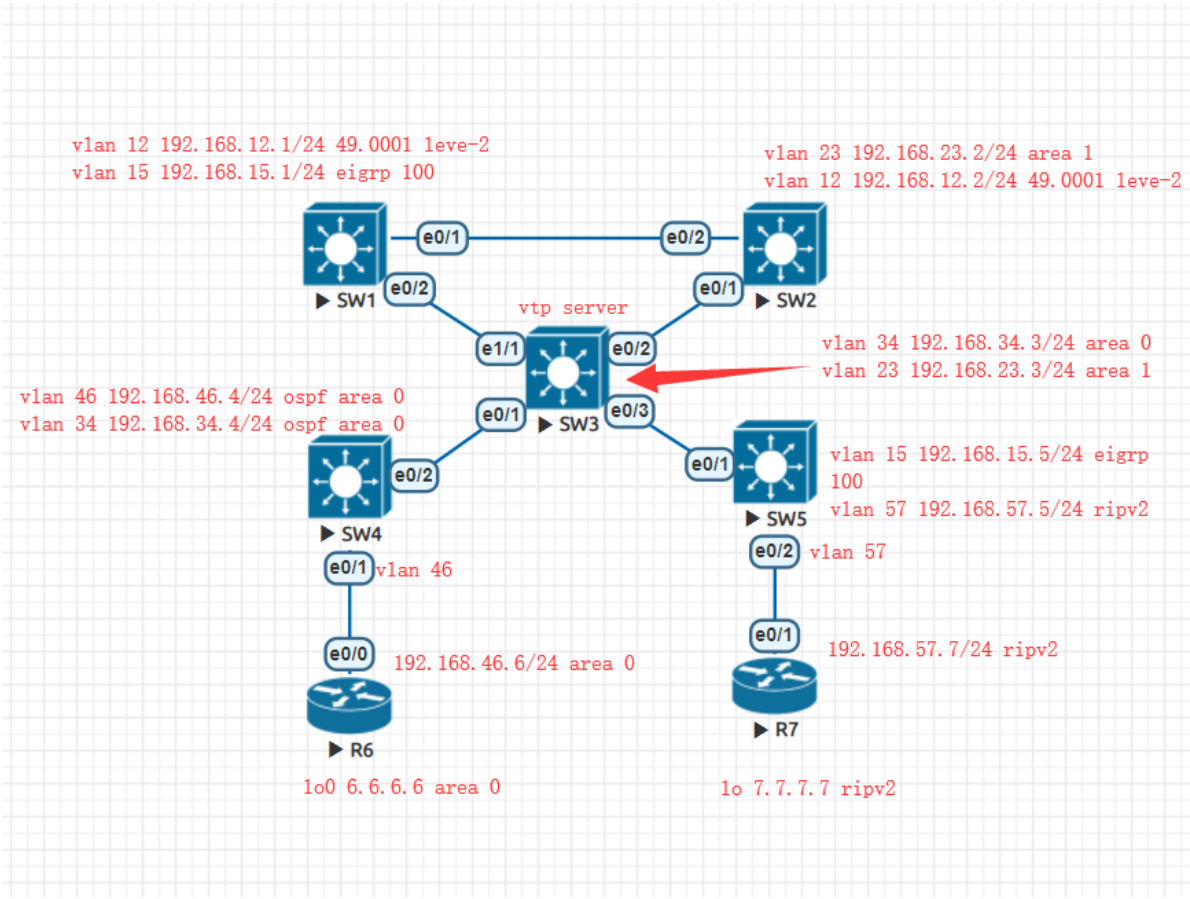



```
第一步是将R1、R2配置不同网段的地址
第二步在交换机上将e-/1，e0/2接口划分至不同的vlan，将e0/3接口配置成窗口
第四步将R3的e0/0接口no sh，然后配置子接口的单臂路由
interface Ethernet0/0.10
 encapsulation dot1Q 10
 ip address 192.168.1.3 255.255.255.0
 no shutdown
!
interface Ethernet0/0.20
 encapsulation dot1Q 20
 ip address 192.168.2.3 255.255.255.0
 no shutdown
```

三层交换机

- SVI交换式虚拟接口
- - 可以通过int vlan 10的方式定义交换虚拟接口
  - 被定义的该SVI接口是被自动放到指定的vlan中
- - 默认如果没有物理接口在指定的vlan中时，SVI接口无法开启

综合实验




```
配置trunk
SW1(config)#int range e0/1 -2
SW1(config-if-range)#sw tr en do
SW1(config-if-range)#sw mo tr
==================================
SW2(config)#int range e0/1 -2
SW2(config-if-range)#sw tr en do
SW2(config-if-range)#sw mo tr
==================================
SW3(config)#int range e0/1 -3, e1/1
SW3(config-if-range)#sw tr en do
SW3(config-if-range)#sw mo tr
==================================
SW4(config)#int e0/2
SW4(config-if)#sw tr en do
SW4(config-if)#sw mo tr
==================================
SW5(config)#int e0/1
SW5(config-if)#sw tr en do
SW5(config-if)#sw mo tr

配置vtp
SW3(config)#vtp do cisco
SW3(config)#vtp mo ser
============================
除了Sw3外的交换机
vtp do cisco
vtp mo cl
创建vlan
SW3(config)#vlan 46,34,23,12,15,57
==================================
去Sw3以外的交换机上检查是否同步vlan
```

```
show vlan
将R6，R7对应的接口加入vlan
SW4(config)#int e0/1
SW4(config-if)#sw ac vl 46
==================
SW5(config)#int e0/2
SW5(config-if)#sw ac vl 57

配置路由协议
R6(config)#int lo0
R6(config-if)#ip add 6.6.6.6 255.255.255.0
R6(config-if)#ip ospf 1 area 0
R6(config-if)#int e0/0
R6(config-if)#ip add 192.168.46.6 255.255.255.0
R6(config-if)#no sh
R6(config-if)#ip ospf 1 area 0
=====================================
SW4(config-if)#int vlan 46
SW4(config-if)#ip add 192.168.46.4 255.255.255.0
SW4(config-if)#no sh
SW4(config-if)#int lo0
SW4(config-if)#ip add 4.4.4.4 255.255.255.0
SW4(config-if)#int range lo0 , vl 46
SW4(config-if-range)#ip ospf 1 area 0
SW4(config-if-range)#int vl 34
SW4(config-if)#ip add 192.168.34.4 255.255.255.0
SW4(config-if)#no sh
SW4(config-if)#ip ospf 1 area 0
=======================================
SW3(config)#int lo0
SW3(config-if)#ip add 3.3.3.3 255.255.255.0
SW3(config-if)#no sh
SW3(config-if)#ip ospf 1 area 0
SW3(config-if)#int vlan 34
SW3(config-if)#ip add 192.168.34.3 255.255.255.0
SW3(config-if)#no sh
SW3(config-if)#ip ospf 1 area 0
SW3(config-if)#int vl 23
SW3(config-if)#ip add 192.168.23.3 255.255.255.0
SW3(config-if)#no sh
SW3(config-if)#ip ospf 1 area 1
=============================================
SW2(config)#int lo0
SW2(config-if)#ip add 2.2.2.2 255.255.255.0
SW2(config-if)#ip ospf 1 area 1
SW2(config-if)#int vl 23
SW2(config-if)#ip add 192.168.23.2 255.255.255.0
SW2(config-if)#no sh
SW2(config-if)#ip ospf 1 area 1
SW2(config-if)#int vl 12
SW2(config-if)#ip add 192.168.12.2 255.255.255.0
SW2(config-if)#no sh
SW2(config-if)#router isis
SW2(config-router)#net 49.0001.0000.0000.0002.00
SW2(config-router)#is-type level-2
SW2(config-router)#int vl 12
SW2(config-if)#ip router isis
=============================================
```

```
SW1(config)#router isis
SW1(config-router)#net 49.0001.0000.0000.0001.00
SW1(config-router)#is-type level-2
SW1(config-router)#int lo0
SW1(config-if)#ip add 1.1.1.1 255.255.255.0
SW1(config-if)#ip router isis
SW1(config-if)#int vl 12
SW1(config-if)#ip add 192.168.12.1 255.255.255.0
SW1(config-if)#no sh
SW1(config-if)#ip router isis
SW1(config-if)#int vl 15
SW1(config-if)#ip add 192.168.15.1 255.255.255.0
SW1(config-if)#no sh
SW1(config-if)#router eigrp 100
SW1(config-router)#net 192.168.15.0
================================================
SW5(config-if)#int lo0
SW5(config-if)#ip add 5.5.5.5 255.255.255.0
SW5(config-if)#int vl 15
SW5(config-if)#ip add 192.168.15.5 255.255.255.0
SW5(config-if)#no sh
SW5(config-if)#int vl 57
SW5(config-if)#ip add 192.168.57.5 255.255.255.0
SW5(config-if)#no sh
SW5(config-if)#router eigrp 100
SW5(config-router)#net 5.0.0.0
SW5(config-router)#net 192.168.15.0
SW5(config-router)#router rip
SW5(config-router)#ver 2
SW5(config-router)#no au
SW5(config-router)#net 192.168.57.0
================================================
R7(config)#int lo0
R7(config-if)#ip add 7.7.7.7 255.255.255.0
R7(config-if)#int e0/1
R7(config-if)#ip add 192.168.57.7 255.255.255.0
R7(config-if)#no sh
R7(config-if)#router rip
R7(config-router)#ver 2
R7(config-router)#no au
R7(config-router)#net 7.0.0.0
R7(config-router)#net 192.168.57.0

检查ospf
SW4#sh ip ospf nei

Neighbor ID     Pri   State           Dead Time   Address         Interface
3.3.3.3           1   FULL/BDR        00:00:38    192.168.34.3    Vlan34
6.6.6.6           1   FULL/DR         00:00:39    192.168.46.6    Vlan46
=======================================
SW3#sh ip ospf nei

Neighbor ID     Pri   State           Dead Time   Address         Interface
4.4.4.4           1   FULL/DR         00:00:36    192.168.34.4    Vlan34
2.2.2.2           1   FULL/BDR        00:00:38    192.168.23.2    Vlan23
=======================================
R6#sh ip route ospf
      2.0.0.0/32 is subnetted, 1 subnets
```

```
O IA     2.2.2.2 [110/13] via 192.168.46.4, 00:00:32, Ethernet0/0
      3.0.0.0/32 is subnetted, 1 subnets
O        3.3.3.3 [110/12] via 192.168.46.4, 00:00:32, Ethernet0/0
      4.0.0.0/32 is subnetted, 1 subnets
O        4.4.4.4 [110/11] via 192.168.46.4, 00:00:32, Ethernet0/0
O IA  192.168.23.0/24 [110/12] via 192.168.46.4, 00:00:32, Ethernet0/0
O     192.168.34.0/24 [110/11] via 192.168.46.4, 00:00:32, Ethernet0/0
检查isis
SW2#sh ip route isis
      1.0.0.0/24 is subnetted, 1 subnets
i L2     1.1.1.0 [115/20] via 192.168.12.1, 00:05:36, Vlan12
检查eigrp
SW1#sh ip route eigrp
      5.0.0.0/24 is subnetted, 1 subnets
D        5.5.5.0 [90/130816] via 192.168.15.5, 00:04:30, Vlan15
检查rip
SW5#sh ip route rip
      7.0.0.0/24 is subnetted, 1 subnets
R        7.7.7.0 [120/1] via 192.168.57.7, 00:00:09, Vlan57

做双向重发步
SW2(config)#router ospf 1
SW2(config-router)#red isis level-2 subnets
SW2(config-router)#router isis
SW2(config-router)#redistribute ospf 1 level-2
==========================
SW1(config)#router isis
SW1(config-router)#redistribute eigrp 100 level-2
SW1(config-router)#router eigrp 100
SW1(config-router)#redistribute isis metric 10000 100 255 1 1500
==========================
SW5(config)#router rip
SW5(config-router)#redistribute eigrp 100 metric 5
SW5(config-router)#router eigrp 100
SW5(config-router)#redistribute rip metric 10000 100 255 1 1500

检查路由
R6#sh ip route ospf
      1.0.0.0/24 is subnetted, 1 subnets
O E2     1.1.1.0 [110/20] via 192.168.46.4, 00:01:25, Ethernet0/0
      2.0.0.0/32 is subnetted, 1 subnets
O IA     2.2.2.2 [110/13] via 192.168.46.4, 00:04:45, Ethernet0/0
      3.0.0.0/32 is subnetted, 1 subnets
O        3.3.3.3 [110/12] via 192.168.46.4, 00:04:45, Ethernet0/0
      4.0.0.0/32 is subnetted, 1 subnets
O        4.4.4.4 [110/11] via 192.168.46.4, 00:04:45, Ethernet0/0
      5.0.0.0/24 is subnetted, 1 subnets
O E2     5.5.5.0 [110/20] via 192.168.46.4, 00:01:01, Ethernet0/0
      7.0.0.0/24 is subnetted, 1 subnets
O E2     7.7.7.0 [110/20] via 192.168.46.4, 00:00:19, Ethernet0/0
O E2  192.168.15.0/24 [110/20] via 192.168.46.4, 00:01:01, Ethernet0/0
O IA  192.168.23.0/24 [110/12] via 192.168.46.4, 00:04:45, Ethernet0/0
O     192.168.34.0/24 [110/11] via 192.168.46.4, 00:04:45, Ethernet0/0
O E2  192.168.57.0/24 [110/20] via 192.168.46.4, 00:00:19, Ethernet0/0

检查连通性
R6#ping 7.7.7.7
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 7.7.7.7, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/3/4 ms
R6#tr
R6#traceroute 7.7.7.7
Type escape sequence to abort.
Tracing the route to 7.7.7.7
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.46.4 1 msec 0 msec 0 msec
  2 192.168.34.3 1 msec 1 msec 1 msec
  3 192.168.23.2 2 msec 2 msec 2 msec
  4 192.168.12.1 3 msec 2 msec 1 msec
  5 192.168.15.5 2 msec 3 msec 2 msec
  6 192.168.57.7 3 msec *  5 msec

思考：真实的流量是按照traceroute的结果走的吗？如何解决？
```

VPN的两种连接方式

- 站点到站点，主要是用于公司重要站点之间的连接
  - IPsecVPN、MPLSVPN
- 远程连接，常用于企业出差人员，在任意一个可以接入互联网的地方，连进企业内部网络
  - SSLVPN、IPsecVPN、VPDN

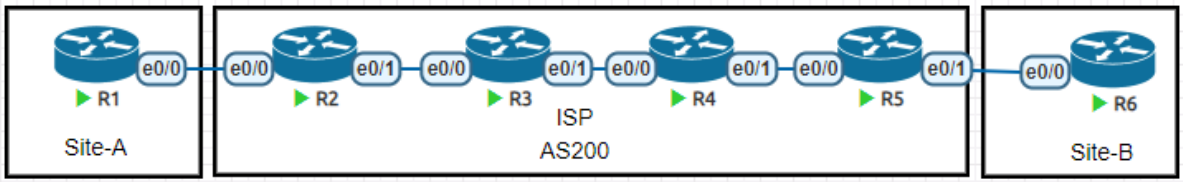
MPLS VPN和IPsecVPN

- MPLS VPN
  - 优点
    - 作为客户部署简单
    - 带宽和延迟得到保障
    - 跨地域连接也能得到保障稳定性
    - 灵活扩展
  - 缺点
    - 策略需要与运营商协商
    - 费用较为昂贵
- IPsec VPN
  - 优点
    - 比较经济
    - 灵活扩展
    - 数据的加密安全
  - 缺点
    - 稳定性依赖于宽带服务商
    - 作为客户需要专业技术人员部署

GRE

- 轻量级的隧道协议

- 头部
  - 外部IP头部
  - GRE头部
  - 内层IP头部
  - 实际传输的数据



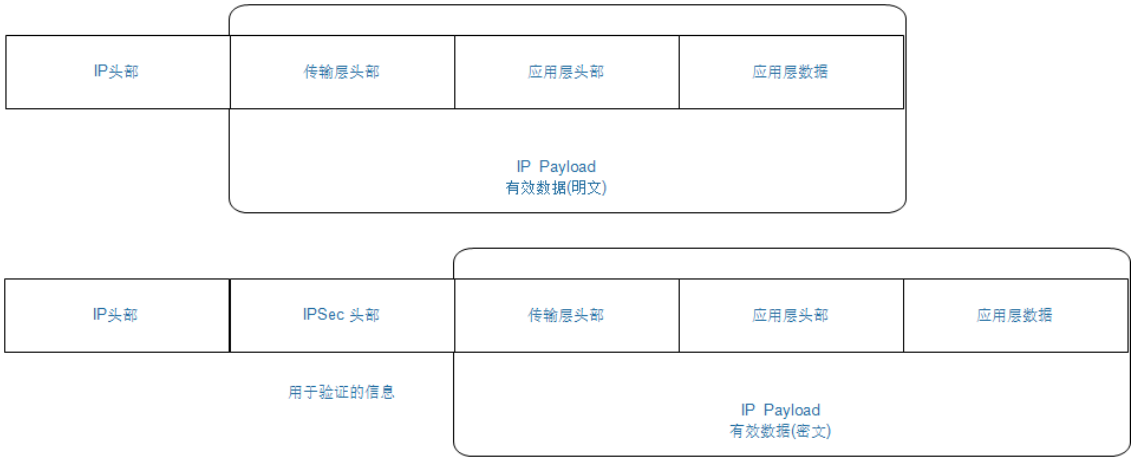

```
配置运营商网络
R2
interface Loopback0
 ip address 2.2.2.2 255.255.255.255
 ip ospf 1 area 0
!
interface Ethernet0/0
 ip address 192.168.12.2 255.255.255.0
!
interface Ethernet0/1
 ip address 192.168.23.2 255.255.255.0
 ip ospf 1 area 0
router ospf 1
 redistribute connected subnets
========================
R3
interface Loopback0
 ip address 3.3.3.3 255.255.255.0
 ip ospf 1 area 0
!
interface Ethernet0/0
 ip address 192.168.23.3 255.255.255.0
 ip ospf 1 area 0
!
interface Ethernet0/1
 ip address 192.168.34.3 255.255.255.0
 ip ospf 1 area 0
========================
R4
interface Loopback0
 ip address 4.4.4.4 255.255.255.0
 ip ospf 1 area 0
!
interface Ethernet0/0
 ip address 192.168.34.4 255.255.255.0
 ip ospf 1 area 0
!
interface Ethernet0/1
 ip address 192.168.45.4 255.255.255.0
 ip ospf 1 area 0
========================
R5
interface Loopback0
 ip address 5.5.5.5 255.255.255.0
 ip ospf 1 area 0
```

```
!
interface Ethernet0/0
 ip address 192.168.45.5 255.255.255.0
 ip ospf 1 area 0
!
interface Ethernet0/1
 ip address 192.168.56.5 255.255.255.0
router ospf 1
 redistribute connected subnets
配置客户网络
R1
interface Loopback0
 ip address 1.1.1.1 255.255.255.0
 ip ospf 1 area 0
!
interface Tunnel0
 ip address 172.16.16.1 255.255.255.0
 ip ospf 1 area 0
 tunnel source Ethernet0/0
 tunnel destination 192.168.56.6
!
interface Ethernet0/0
 ip address 192.168.12.1 255.255.255.0
==================
R6
interface Loopback0
 ip address 6.6.6.6 255.255.255.0
 ip ospf 1 area 0
!
interface Tunnel0
 ip address 172.16.16.6 255.255.255.0
 ip ospf 1 area 0
 tunnel source Ethernet0/0
 tunnel destination 192.168.12.1
!
interface Ethernet0/0
 ip address 192.168.56.6 255.255.255.0
 检查
 R1#sh ip ospf nei

Neighbor ID     Pri   State           Dead Time   Address         Interface
6.6.6.6           0   FULL/  -        00:00:31    172.16.16.6     Tunnel0
R1#sh ip route ospf
      6.0.0.0/32 is subnetted, 1 subnets
O        6.6.6.6 [110/1001] via 172.16.16.6, 00:02:20, Tunnel0
R1#ping 6.6.6.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 6.6.6.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

IPsec

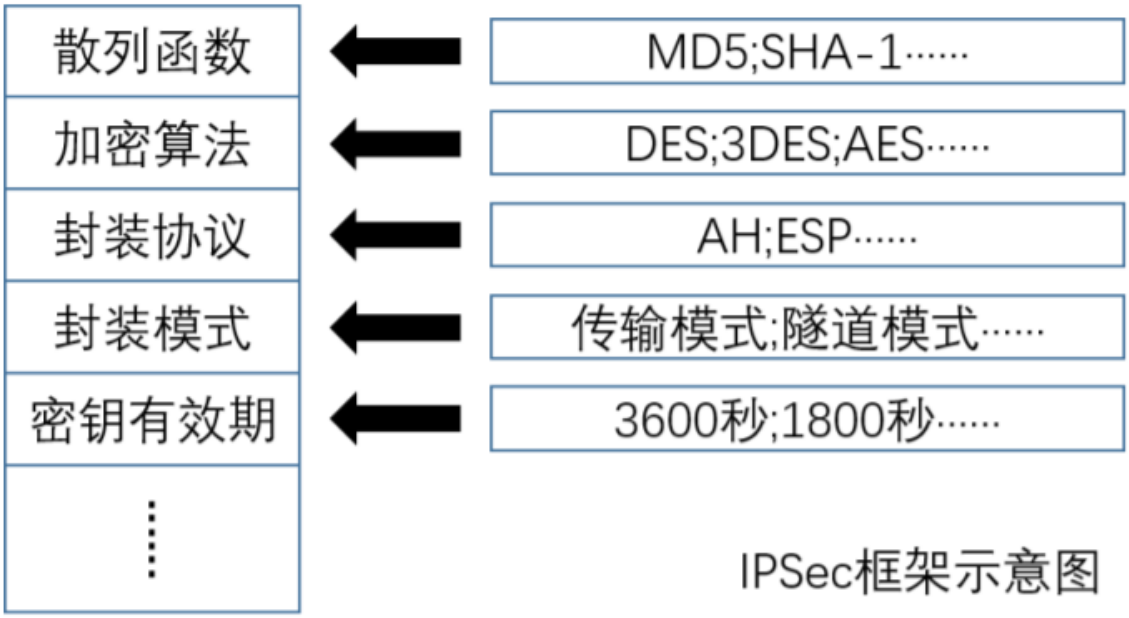
- 是一项标准的安全技术，它是通过在数据包中插入一个预定义头部的方式来保障网络层数据安全



- 私密性
  - 对数据进行加密
  - 即使第三方截获数据，第三方也没有能力将其恢复为明文
- 完整性
  - 确保数据传输过程中不会被篡改
- 源认证
  - 确保是合法的源发送的此数据
  - 接收方能够知道数据发送方是谁

IPsec框架

- 传统的一些安全技术，比如https等使用固定的加密算法，当某天这个加密算法被破解就会有严重问题
- IPsec框架不定义具体的加密算法，只是提供一个框架，具体加密方式和算法只在双方进行会话时协商
- 框架
  - 散列函数
    - MD5
    - SHA
  - 加密算法
    - DES
    - 3DES
    - AES
  - 封装协议
    - AH
    - ESP
  - 封装模式
    - 传输模式
    - 隧道模式
  - 密钥有效期
    - 3600秒
    - 1800秒



IPSec框架示意图

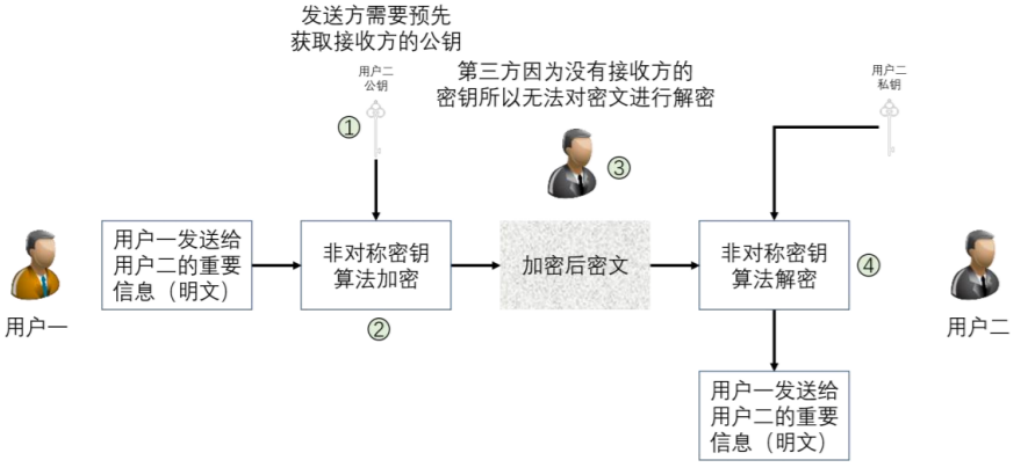
散列函数

- 散列函数也叫hash函数，主流的算法有md5和sha1，通常是用来获得散列值，这个散列值可以唯一标识原数据
- 特点
  - 固定大小
    - 无论数据有多大，最终计算的散列值长度时固定的
  - 雪崩效应
    - 即使一个非常大的数据，你修改了微不足道的一小部分，都会导致散列值完全不一样
  - 单向
    - 只能从原始数据计算得到散列值，不可能从散列值恢复哪怕一个bit的原始数据
  - 冲突避免
    - 几乎不能够周到另外一个数据和当前数据的散列值完全相同
    - 散列值可以确保数据的唯一性
  - 验证数据完整性
    - 使用散列函数计算重要文件的散列值，记作A
    - 发给对方重要文件和散列值A
    - 对方对接受到的文件进行计算得到散列值，记作B
    - 如果A和B相等，则代表文件没有被篡改过，否则则被篡改过
  - HMAC验证数据发送源
    - 在计算散列值的过程中，把key加入计算过程

加密算法

- 把明文数据转换为密文数据，并且在拥有解密能力的情况下可以将数据解密回来
- 分类
  - 对称加密算法
    - 加密方和解密方使用相同的密钥
    - 优点
      - 速度快
      - 安全

        - 紧凑
    - 缺点
        - 明文传输共享密钥，容易出现中途劫持和窃听的问题
        - 随着参与者数量的增加，密钥数量急剧膨胀，对密钥的管理和存储则是一个严重问题
        - 不支持数字签名和不可否认性
    - 主流协议
        - DES
        - 3DES
        - AES
        - RC4
- 非对称加密算法
    - 所有参与者在通信前需要产生一对密钥对，包括一个公钥和一个私钥
    - 公钥可以放在公共的服务器上，任何人都可以获取
    - 私钥需要持有者严格保护，确保只有持有者才能唯一拥有
    - 公钥加密的数据只能私钥解密，私钥加密的数据只能公钥解密
- 加密过程
    - 发送方需要获取接收方的公钥
    - 使用公钥对发送的数据进行加密
    - 加密的数据如果中途被拦截，由于拦截方没有私钥则不能对其进行解密
    - 接收方接受到密文使用私钥解密得到明文



- 用来做数字签名，可以验证数据的合法来源，可以防止否认
    - 发送方计算得到重要文件的散列值
    - 发送方使用自己的私钥对散列值进行加密得到数字签名
    - 发送方将重要文件（明文）和数字签名发送给对方
    - 接收方使用相同的散列函数对重要文件进行计算得到散列值1
    - 接收方使用发送方的公钥解密数字签名得到散列值2
    - 如果散列值1和散列值2相等，数字签名验证成功
- 常见算法
    - RSA
        - 常见于数字签名
        - 网址证书
    - DH
        - 常见于IPsec中交换密钥
    - ECC
        - 椭圆曲线算法

- 优点
  - 安全
  - 私钥不担心泄露，因为无须共享给任何人
  - 密钥数数量和参与者数量相同，密钥管理相对比较轻松
  - 交换密钥之前不用预先建立信任关系，公钥直接上传PKI/CA，对称解密需要事先共享密钥
  - 支持数字签名和不可否认
- 缺点
  - 加密速度很慢
  - 密文会变长
  - 所以在IPsec中，使用非对称加密算法加密对称加密算法的密钥

封装协议

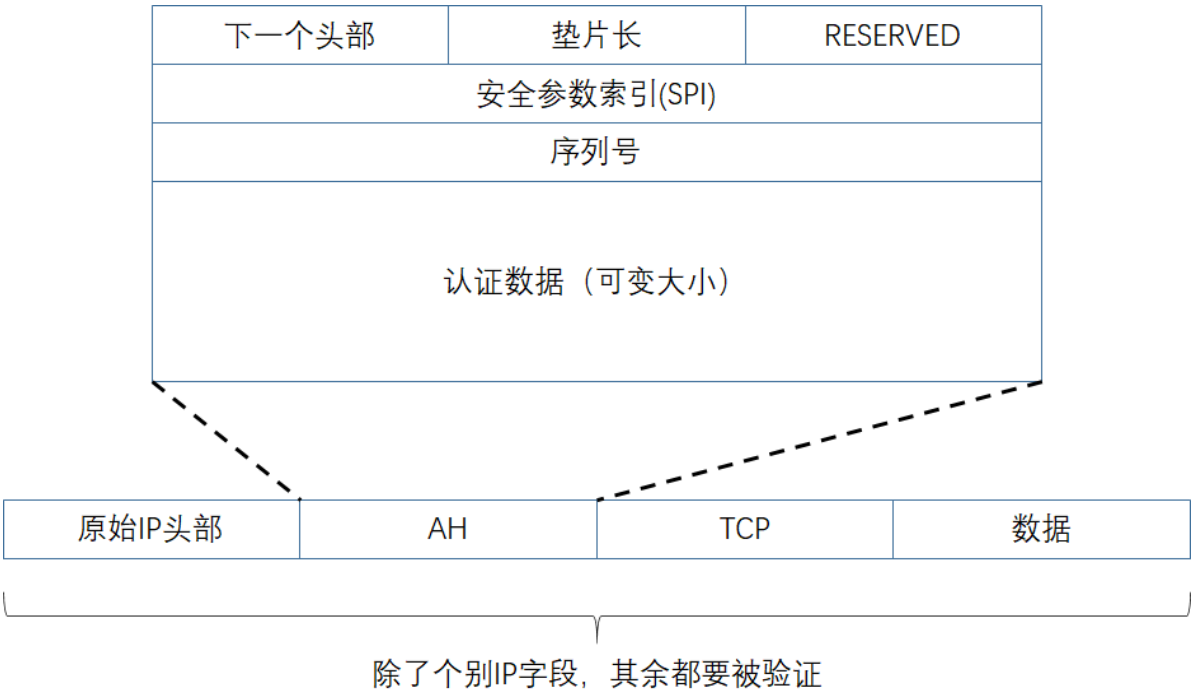
- 通常由ESP和AH两种协议
- ESP协议
  - 协议号我i50
  - 能够提供私密性、完整性、源认证
  - 能够抵御重放攻击
  - 只能保护IP数据部分，不能保护IP头部部分

- 安全参数索引SPI
- 一个单调增长的序列号
- 初始化向量
  - 每一个使用cbc加密的数据包都会产生一个随机数
- 负载数据
  - 需要被加密保护传输的数据
- 垫片
  - 为了补齐固定长度
- 垫片长度
  - 告诉接收方垫片数据有多长
  - 认证数据

- 对数据进行验证，通过HAMC的散列计算得到散列值，将散列值放在认证数据这个部分

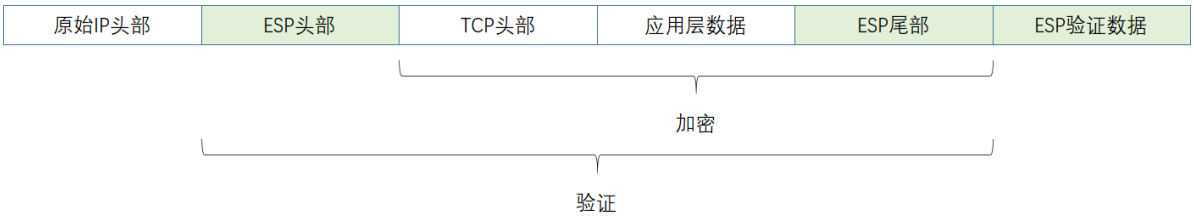
AH协议

- AH协议由于没有提供数据加密功能，所以没有被广泛采用

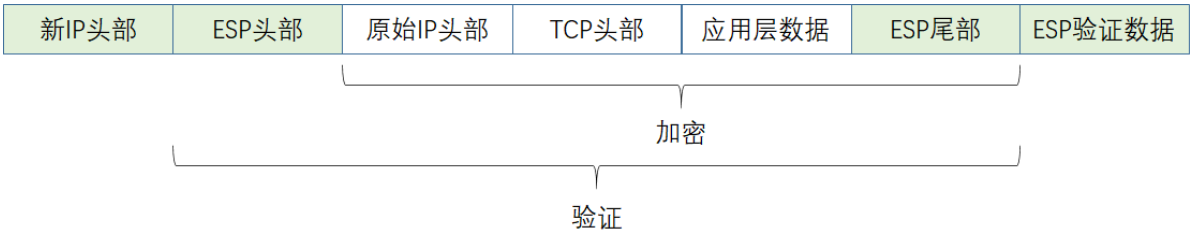


封装模式

- 传输模式



- 隧道模式



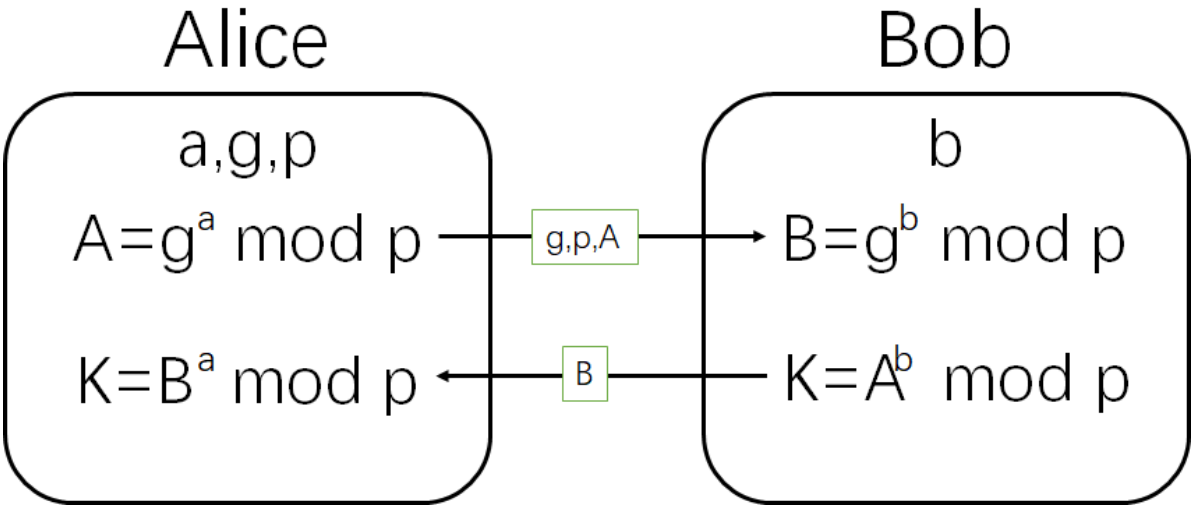
IKE互联网密钥交换协议

- **对建立IPsec的双方进行认证**
- **通过密钥交换，产生用于加密和HMAC的随机密钥**
- **协商协议参数**
  - 加密协议
  - 散列函数
  - 封装协议
  - 封装模式

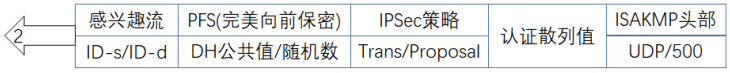
    - 密钥有效期
- 安全关联
    - IKE SA
        - 维护IKE协议运行安全
    - IPsec SA
        - 维护用户流量安全
- IKE协议由三个协议组成
    - SKEME：决定了IKE的密钥交换方式
    - Oakley：决定了IPsec的框架涉及
    - ISAKMP：IKE的本质协议，决定了包封装与交换还有模式切换

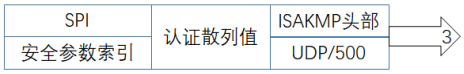
IKE的两个阶段

- 进入IKE SA状态，保证下一阶段的通信是安全的
    - 主模式
    - 主模式1-2包交换：这五个策略不是为了最终加密用户数据的，而是为了保障接下来协商过程的安全
        - 加密策略
        - 散列函数
        - DH组
        - 认证方式
        - 密钥有效期
    - 主模式3-4包交换，为保护五六包的交换提供密钥



- 主模式5-6包交换
    - 本次交换用于验证密码的正确，并且确认双方的流量源目的地址
- 主动模式（也叫野蛮模式）
    - 第一阶段的主要任务是互相认证，互相协商好第二阶段采用的加密算法和加密的密钥，来保证第二阶段的通信是安全的，就是 IKE SA状态
    - 快速模式三个包交换进入IPsec SA状态
    -





- 快速模式主要是协商数据流量的安全策略
  - 感兴趣流量
  - 加密策略
  - 散列函数
  - 封装协议
  - 封装模式
  - 密钥有效期

```
第一阶段
crypto isakmp policy 10
 encr 3des
 hash md5
 authentication pre-share
 group 2
 lifetime 1800
 crypto isakmp key cisco address 0.0.0.0 0.0.0.0
 第二阶段
 crypto ipsec transform-set cisco esp-aes esp-md5-hmac
 mode transport
crypto ipsec profile cisco
 set transform-set cisco
 set pfs group1
interface Tunnel0
 tunnel protection ipsec profile cisco
查看第一阶段isakmp sa的状态
R1#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status
192.168.56.6    192.168.12.1    QM_IDLE           1002 ACTIVE
192.168.12.1    192.168.56.6    QM_IDLE           1001 ACTIVE
IPv6 Crypto ISAKMP SA
查看第二阶段IPsec sa的状态
R1#sh crypto ipsec sa

interface: Tunnel0
    Crypto map tag: Tunnel0-head-0, local addr 192.168.12.1
查看IPSec VPN摘要
R1#show crypto engine connections active
Crypto Engine Connections

   ID  Type    Algorithm           Encrypt  Decrypt LastSeqN IP-Address
    3  IPsec   AES+MD5                   0       57       57 192.168.12.1
    4  IPsec   AES+MD5                  62        0        0 192.168.12.1
 1001  IKE     MD5+3DES                  0        0        0 192.168.12.1
 1002  IKE     MD5+3DES                  0        0        0 192.168.12.1

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.168.12.1/255.255.255.255/47/0)
```

```
   remote ident (addr/mask/prot/port): (192.168.56.6/255.255.255.255/47/0)
   current_peer 192.168.56.6 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 64, #pkts encrypt: 64, #pkts digest: 64
    #pkts decaps: 59, #pkts decrypt: 59, #pkts verify: 59
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 192.168.12.1, remote crypto endpt.: 192.168.56.6
     plaintext mtu 1458, path mtu 1500, ip mtu 1500, ip mtu idb Ethernet0/0
     current outbound spi: 0xC976B99E(3380001182)
     PFS (Y/N): Y, DH group: group1

     inbound esp sas:
      spi: 0x4FE3C503(1340327171)
        transform: esp-aes esp-md5-hmac ,
        in use settings ={Transport, }
        conn id: 3, flow_id: SW:3, sibling_flags 80004000, crypto map:
Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4343070/3389)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0xC976B99E(3380001182)
        transform: esp-aes esp-md5-hmac ,
        in use settings ={Transport, }
        conn id: 4, flow_id: SW:4, sibling_flags 80004000, crypto map:
Tunnel0-head-0
        sa timing: remaining key lifetime (k/sec): (4343069/3389)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     outbound ah sas:

     outbound pcp sas:
```

###