



G-CNA v2.0课程

讲师：沈老师



- 动态主机配置协议（DHCP）
- 访问控制列表（ACL）
- 网络地址转换（NAT）



前言

- 随着网络规模的不断扩大，网络复杂度不断提升，网络中的终端设备例如主机、手机、平板等，位置经常变化。终端设备访问网络时需要配置IP地址、网关地址、DNS服务器地址等。采用手工方式为终端配置这些参数非常抵效且不够灵活。
- 在大型企业网络中，会有大量的主机或设备需要获取IP地址等网络参数。如果采用手工配置，工作量大且不好管理，如果有用户擅自修改网络参数，还有可能会造成IP地址冲突等问题。使用动态主机配置协议DHCP（Dynamic Host Configuration Protocol）来分配IP地址等网络参数，可以减少管理员的工作量，避免用户手工配置网络参数时造成的地址冲突。



手动配置网络参数的问题 (1)

参数多、理解难

IPv4地址配置：

IP地址

. . .

掩码

. . .

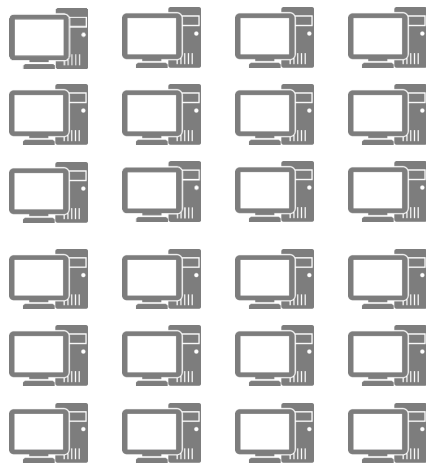
网关

. . .



- 普通用户对于网络参数不了解，经常配置错误，导致无法正常访问网络。随意配置IP地址导致地址冲突更是时常发生。

工作量大



本周工作计划

- ☐ 地址分配
- ☐ 地址分配
- ☐ 地址配置
- ☐ 地址配置



网络管理员

- 交由网络管理员统一配置，工作量巨大，属于重复性劳动。
- 网络管理员需要提前对IP地址进行规划、分配到个人。



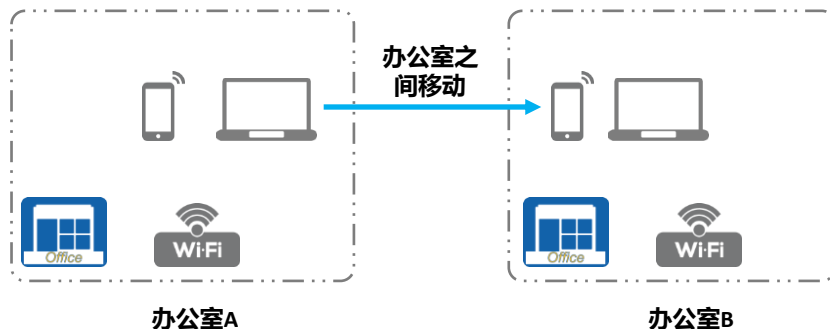
手动配置网络参数的问题 (2)

利用率低



- 企业网中每个人固定使用一个IP地址，IP地址利用率低，有些地址可能长期处于未使用状态。

灵活性差

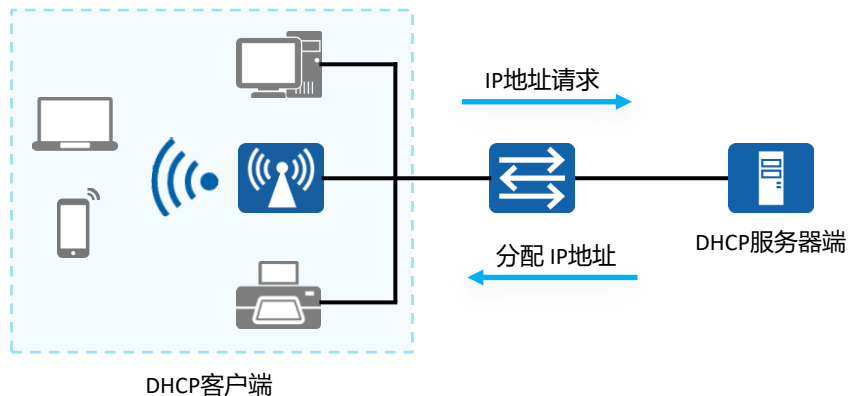


- WLAN (Wireless Local Area Network, 无线局域网) 的出现使终端位置不再固定，当无线终端移动到另外一个无线覆盖区域时，可能需要再次配置IP地址。



DHCP基本概念

DHCP工作示意图



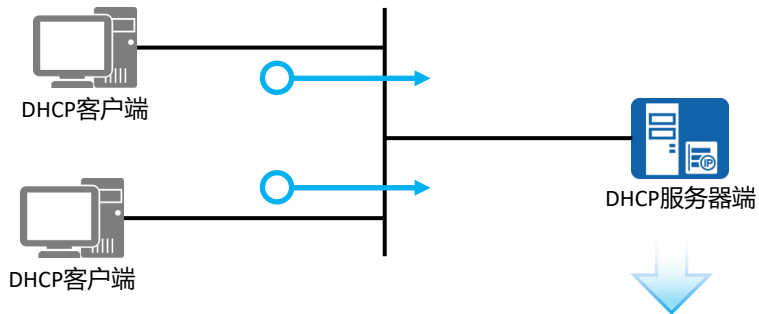
- 为解决传统的静态手工配置方式的不足，DHCP（Dynamic Host Configuration Protocol，动态主机配置协议）应运而生，其可以实现网络动态合理地分配IP地址给主机使用。
- DHCP采用C/S构架，主机无需配置，从服务器端获取地址，可实现接入网络后即插即用。
- 协议报文基于UDP的方式进行交互，采用67（DHCP服务器）和68（DHCP客户端）两个端口号：
 - 正常工作时由客户端向服务器提出配置申请。
 - 服务器返回为客户端分配的IP地址等相应配置信息



DHCP优点

统一管理

○ DHCP地址请求



Pool-No 1

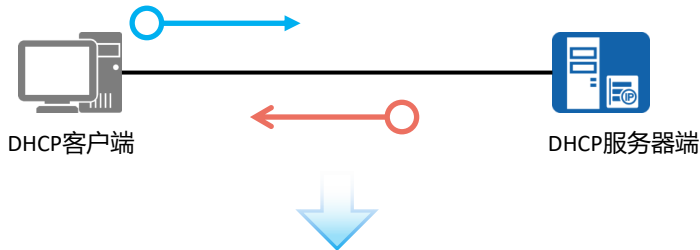
DNS-server 10.1.1.2 | Gateway 10.1.2.1
Network 10.1.2.0 | Mask 255.255.255.0
Total Used
252 2

- IP地址由从服务器端的地址池中获取，服务器端会记录维护IP地址的使用状态，做到IP地址统一分配、管理。

地址租期

○ DHCP地址请求

○ DHCP地址应答



IP:192.168.1.10
Network mask:24
Gateway:192.168.1.1
DNS: 114.114.114.114
Lease: 8 hour

- DHCP提出了租期的概念，可有效提高地址利用率。



DHCP报文类型

报文类型	含义
DHCP DISCOVER	客户端用来寻找DHCP服务器。
DHCP OFFER	DHCP服务器用来响应DHCP DISCOVER报文，此报文携带了各种配置信息。
DHCP REQUEST	客户端请求配置确认，或者续借租期。
DHCP ACK	服务器对REQUEST报文的确认响应。
DHCP NAK	服务器对REQUEST报文的拒绝响应。
DHCP RELEASE	客户端要释放地址时用来通知服务器。



DHCP包交互过程（路由器做server）

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	0.0.0.0	255.255.255.255	DHCP	346	DHCP Discover - Transaction ID 0xde03de03
2	0.001512	10.0.0.1	10.0.99.2	DHCP	308	DHCP Offer - Transaction ID 0xde03de03
3	0.002302	0.0.0.0	255.255.255.255	DHCP	346	DHCP Request - Transaction ID 0xde03de03
4	0.009811	10.0.0.1	10.0.99.2	DHCP	338	DHCP ACK - Transaction ID 0xde03de03

▶ Frame 1: 346 bytes on wire (2768 bits), 346 bytes captured (2768 bits)

▶ Ethernet II, Src: Lite-OnC_30:c8:db (00:a0:cc:30:c8:db), Dst: Broadcast (ff:ff:ff:ff:ff:ff)

▶ Internet Protocol Version 4, Src: 0.0.0.0, Dst: 255.255.255.255

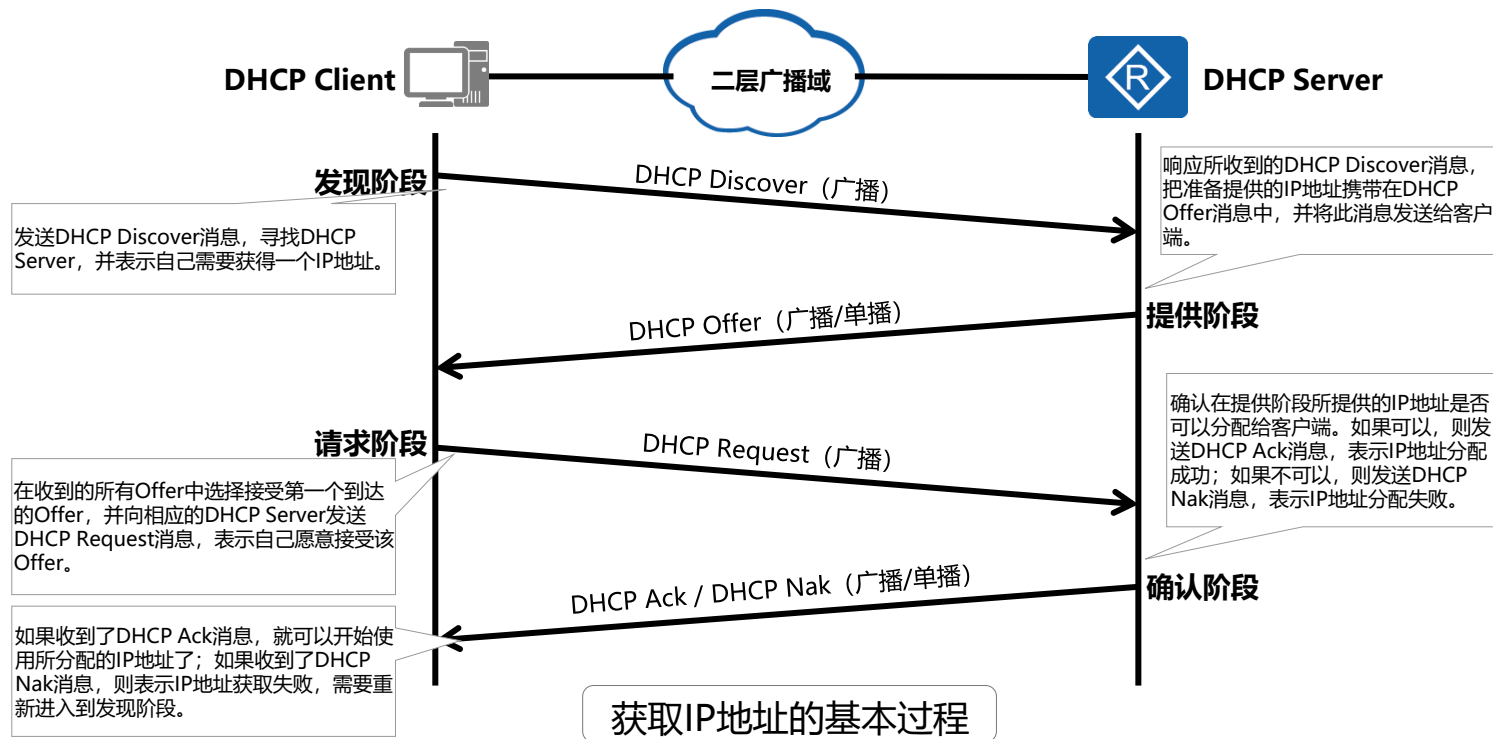
▶ User Datagram Protocol, Src Port: 68, Dst Port: 67

▶ Bootstrap Protocol (Discover)

讨论：通过抓包我们可以发现dhcp获取地址的过程是通过广播进行，那么跨网段如何获取地址呢？

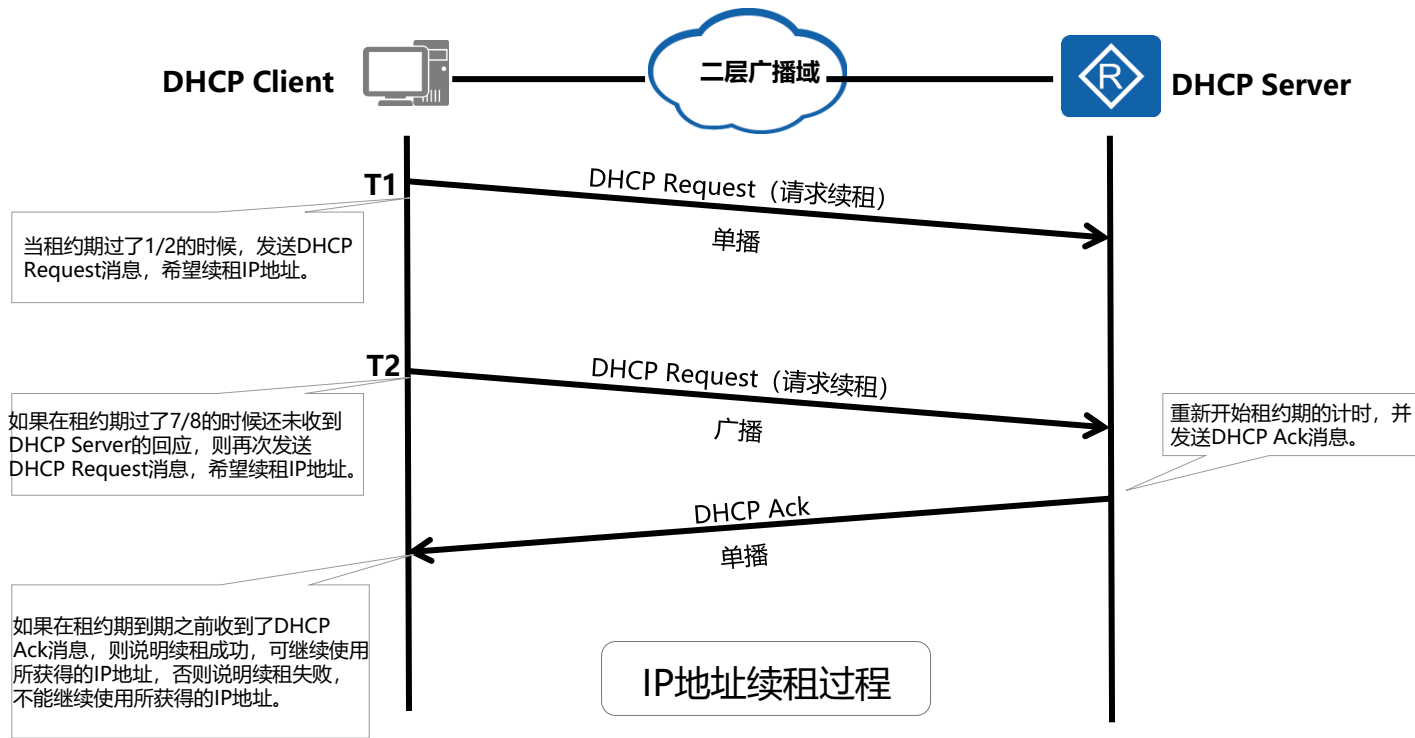


DHCP基本工作过程 (1)



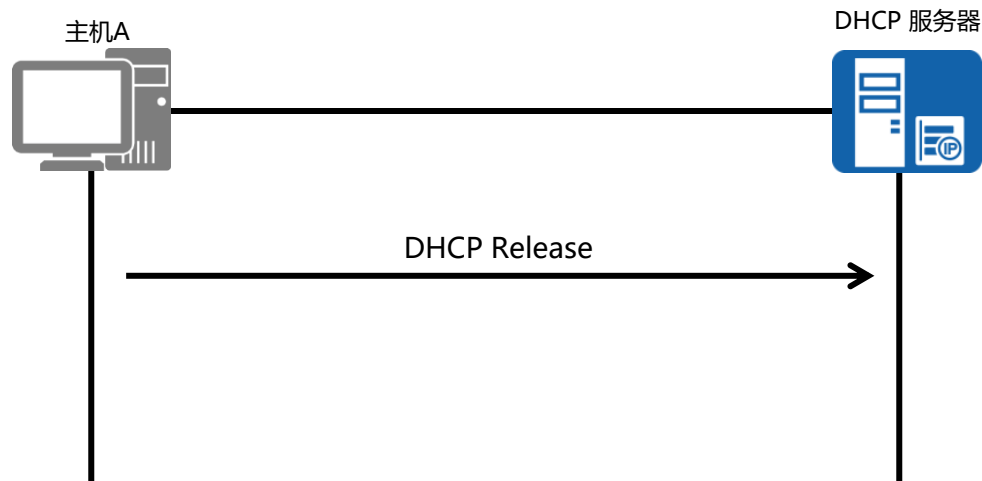


DHCP基本工作过程 (2)





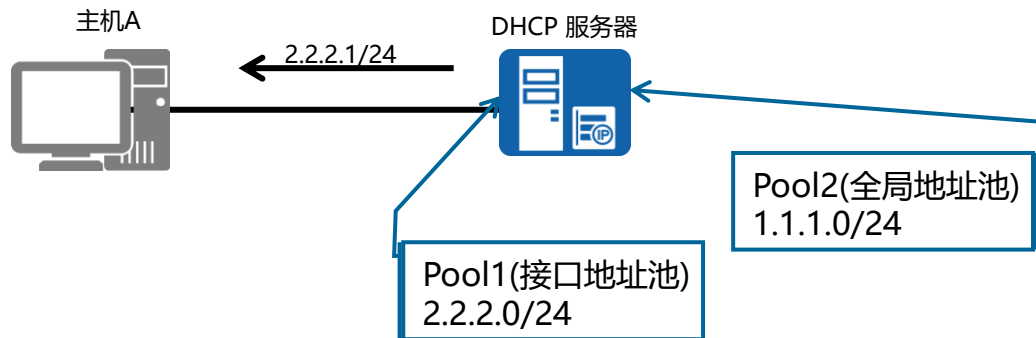
IP地址释放



- 如果IP租约到期前都没有收到服务器响应，客户端停止使用此IP地址。
- 如果DHCP客户端不再使用分配的IP地址，也可以主动向DHCP服务器发送DHCP RELEASE报文，释放该IP地址。



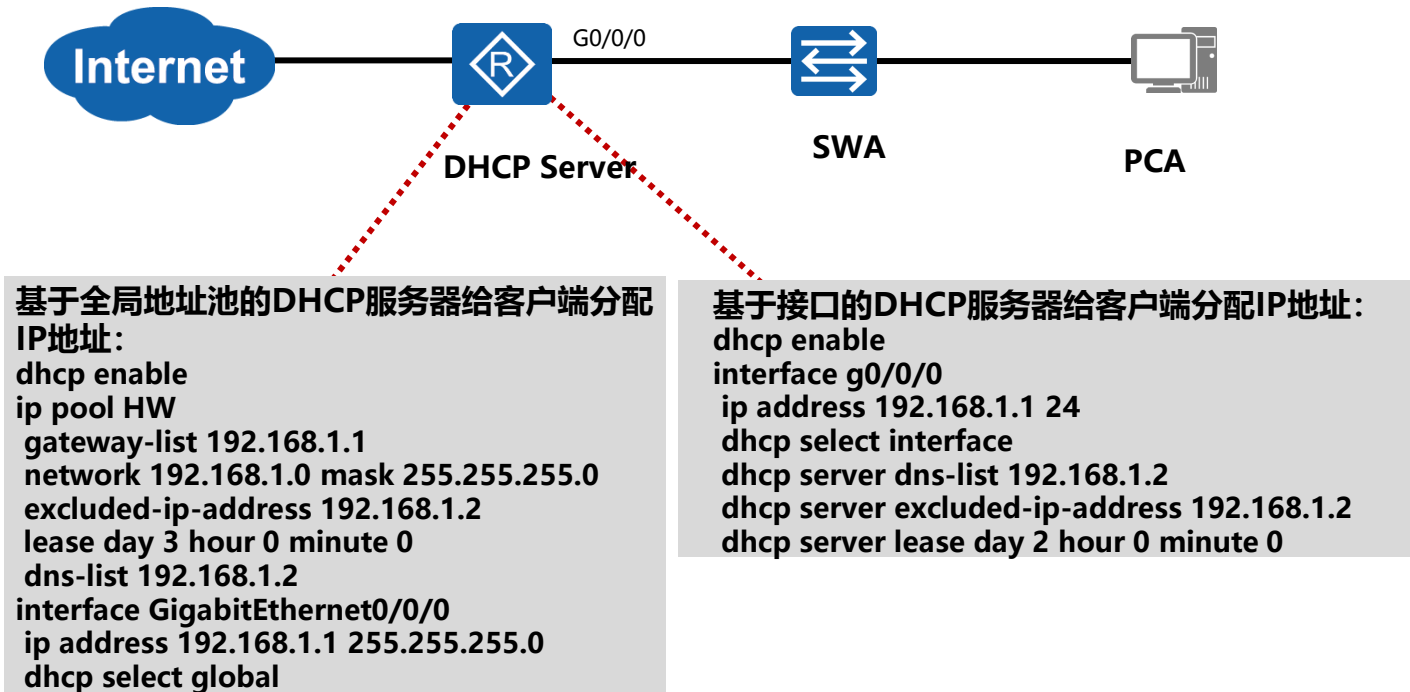
地址池



- 华为路由器支持两种地址池：全局地址池和接口地址池。



DHCP配置实现





配置验证

```
[Huawei]display ip pool
Pool-name      : GigabitEthernet0/0/0
Pool-No       : 0
Position      : Interface      Status      : Unlocked
Gateway-0     : 10.1.1.1
Mask          : 255.255.255.0
VPN instance  : --

IP address Statistic
Total         :253
Used          :1      Idle          :252
Expired       :0      Conflict      :0      Disable    :1
```



配置验证

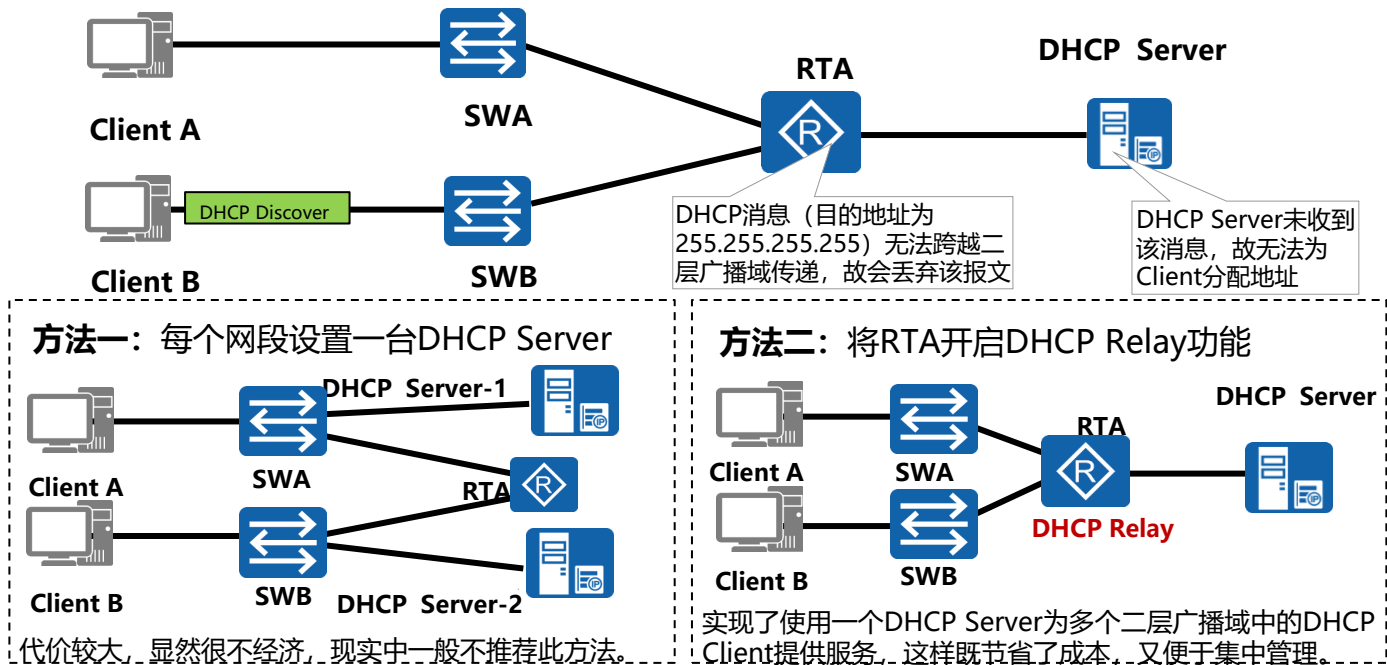
```
[Huawei]display ip pool
```

```
-----  
Pool-name       : pool2  
Pool-No         : 0  
Position        : Local           Status           : Unlocked  
Gateway-0       : 1.1.1.1  
Mask            : 255.255.255.0  
VPN instance    : --  
IP address Statistic  
Total           :253  
Used            :1             Idle             :252  
Expired         :0             Conflict         :0             Disable      :0
```



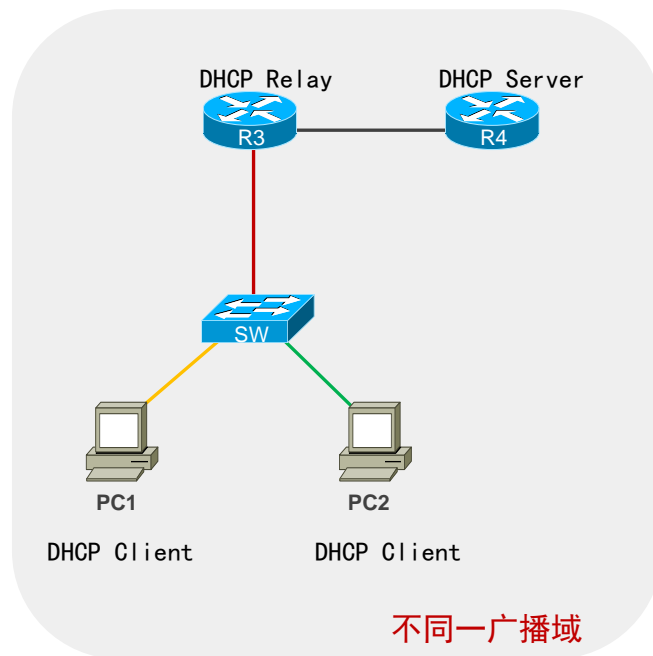
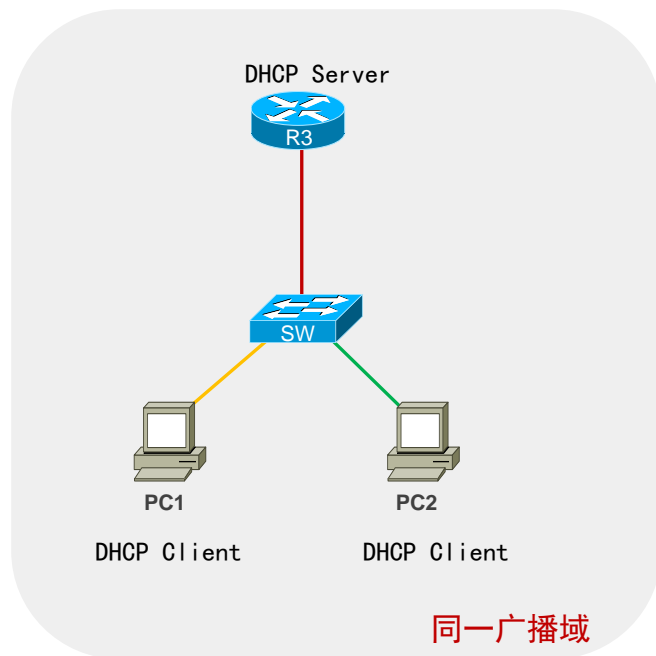

为什么需要DHCP Relay?

- 随着网络规模的扩大，网络中就会出现用户处于不同网段的情况：



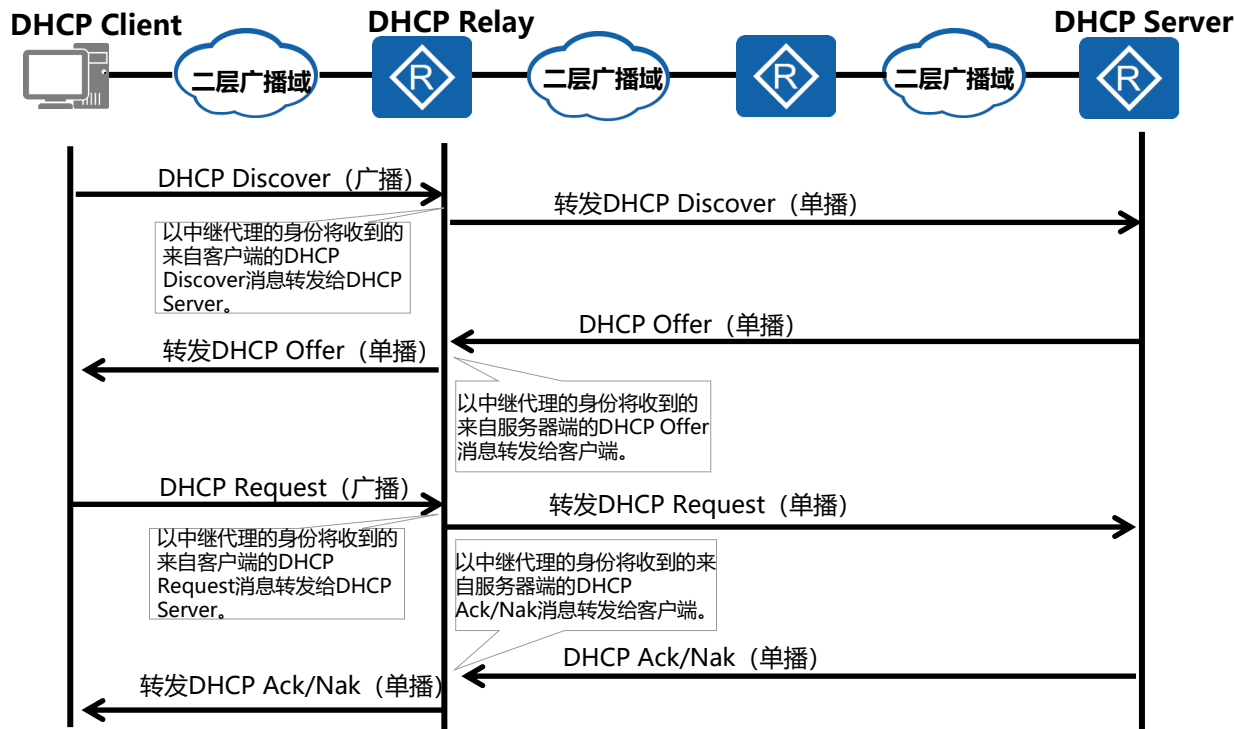


DHCP两种获取地址的架构



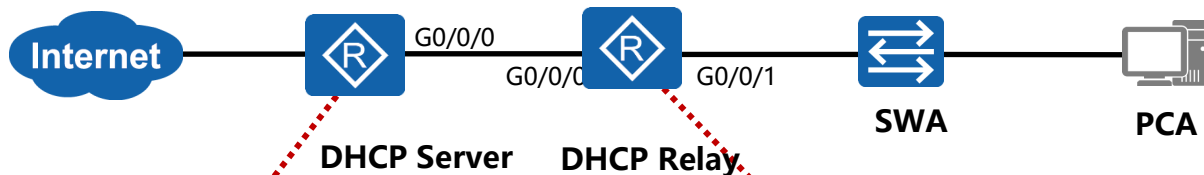


DHCP Relay基本原理





DHCP Relay配置实现



配置DHCP-Server:

(以基于全局地址池分配地址为例)

```
dhcp enable
ip pool DHCP-relay
gateway-list 192.168.1.1
network 192.168.1.0 mask 24
dns-list 10.1.1.1
interface g0/0/0
ip address 10.1.1.1 24
dhcp select global
ip route-static 192.168.1.0 24 10.1.1.2
```

配置DHCP中继 (GW) :

```
dhcp server group DHCP
dhcp-server 10.1.1.1
dhcp enable
interface g0/0/1
ip address 192.168.1.1 24
dhcp select relay
dhcp relay server-select DHCP
interface g0/0/0
ip address 10.1.1.2 24
```



DHCP-Relay.rar



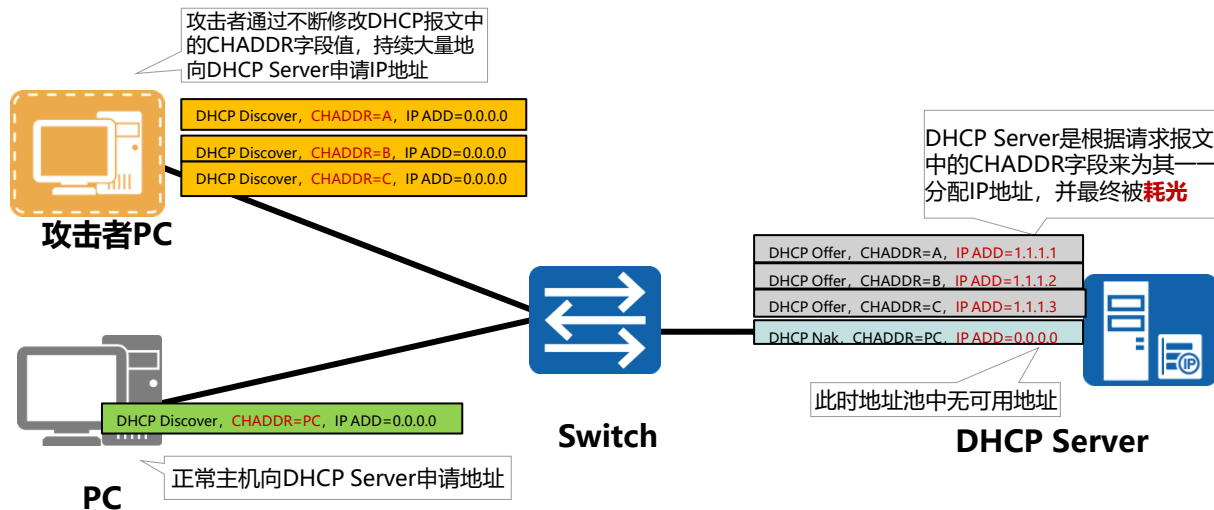
DHCP面临的安全威胁

- 网络攻击行为无处不在，针对DHCP的攻击行为也不例外。例如，某公司突然出现了大面积用户无法上网的情况，经检查用户终端均未获取到IP地址，且DHCP Server地址池中的地址已经全部被分配出去了，这种情况很有可能就是DHCP受到了饿死攻击而导致的。
- DHCP在设计上未充分考虑到安全因素，从而留下了许多安全漏洞，使得DHCP很容易受到攻击。实际网络中，针对DHCP的攻击行为主要有以下三种：
 - DHCP饿死攻击
 - 假冒DHCP Server攻击
 - DHCP中间人攻击



DHCP饿死攻击

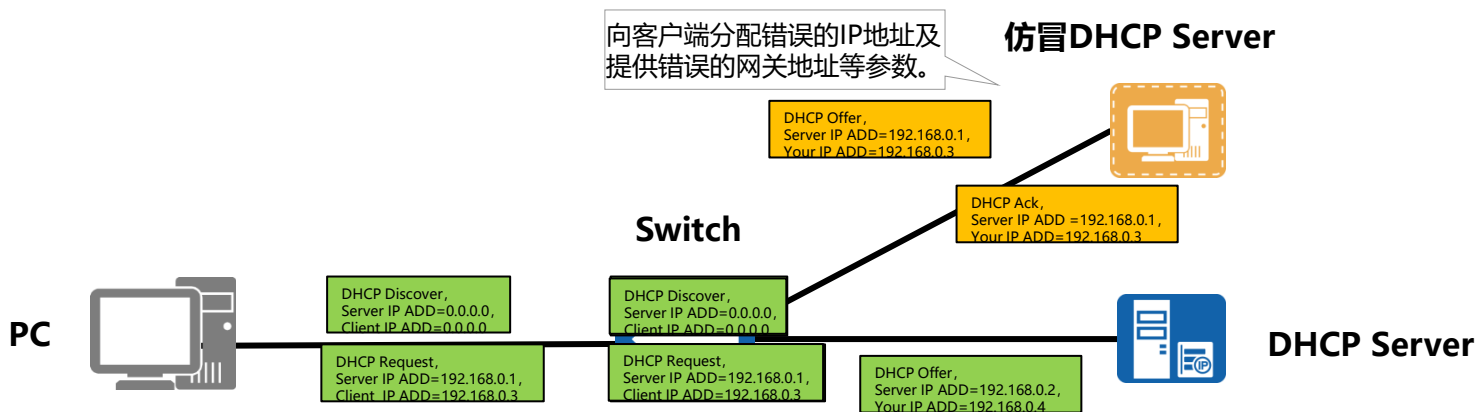
- 攻击原理：攻击者持续大量地向DHCP Server申请IP地址，直到耗尽DHCP Server地址池中的IP地址，导致DHCP Server不能给正常的用户进行分配。
- 漏洞分析：DHCP Server向申请者分配IP地址时，无法区分正常的申请者与恶意的申请者。





假冒DHCP Server攻击

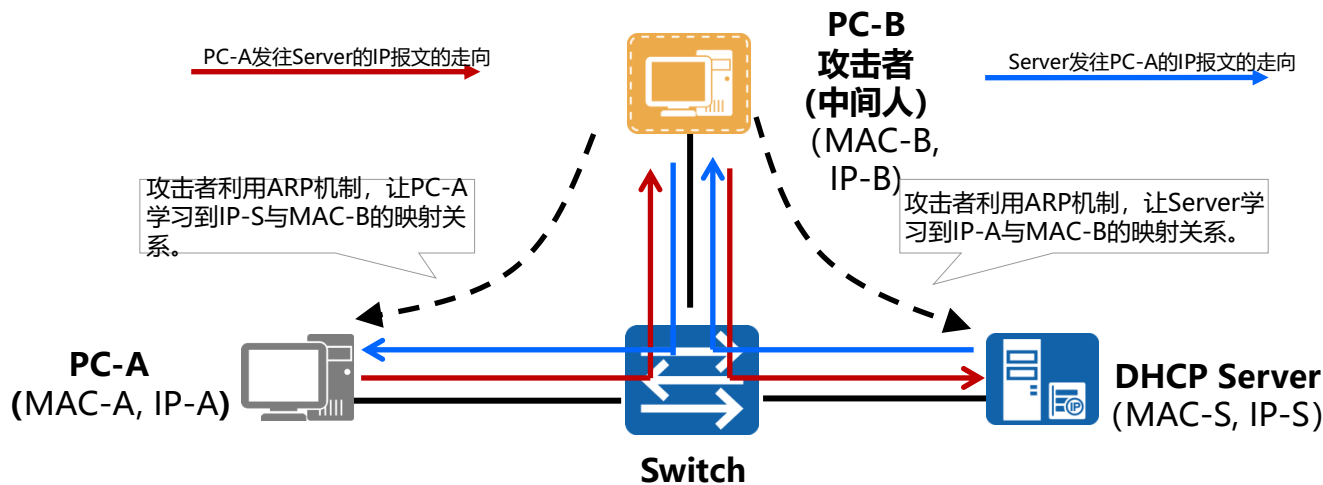
- 攻击原理：攻击者假冒DHCP Server，向客户端分配错误的IP地址及提供错误的网关地址等参数，导致客户端无法正常访问网络。
- 漏洞分析：DHCP客户端接收到来自DHCP Server的DHCP消息后，无法区分这些DHCP消息是来自假冒的DHCP Server，还是来自合法的DHCP Server。





DHCP中间人攻击

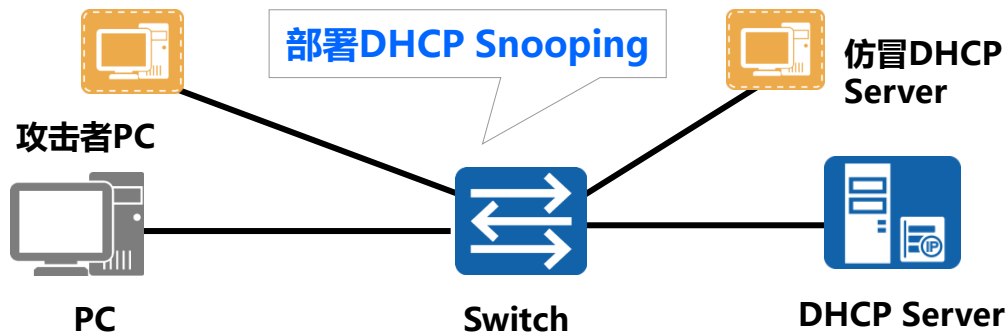
- 攻击原理：攻击者利用ARP机制，让PC-A学习到IP-S与MAC-B的映射关系，又让Server学习到IP-A与MAC-B的映射关系。如此一来，PC-A与Server之间交互的IP报文都会经过攻击者中转。
- 漏洞分析：从本质上讲，中间人攻击是一种Spoofing IP/MAC攻击，中间人利用了虚假的IP地址与MAC地址之间的映射关系来同时欺骗DHCP的客户端和服务端。





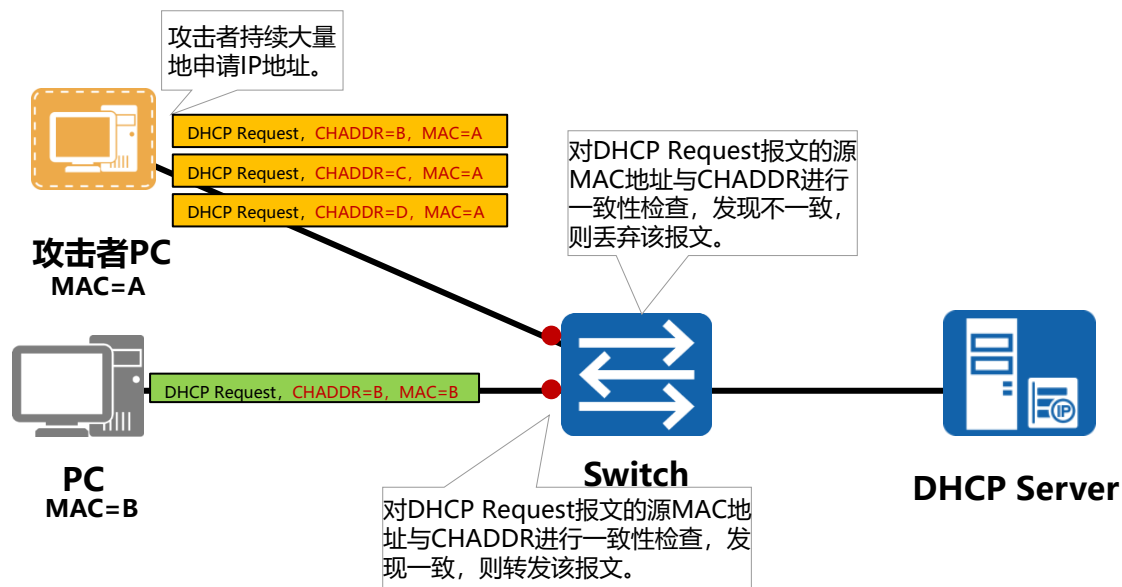
DHCP Snooping技术的出现

- 为了增强网络安全，防止DHCP受到攻击，一种称为DHCP Snooping的技术应运而生。DHCP Snooping不是一种标准技术，尚未有统一的标准规范，不同的网络设备制造商在DHCP Snooping的实现上也不尽相同。
- DHCP Snooping部署在交换机上，其作用类似于在DHCP客户端与DHCP服务器端之间构筑了一道虚拟的防火墙。





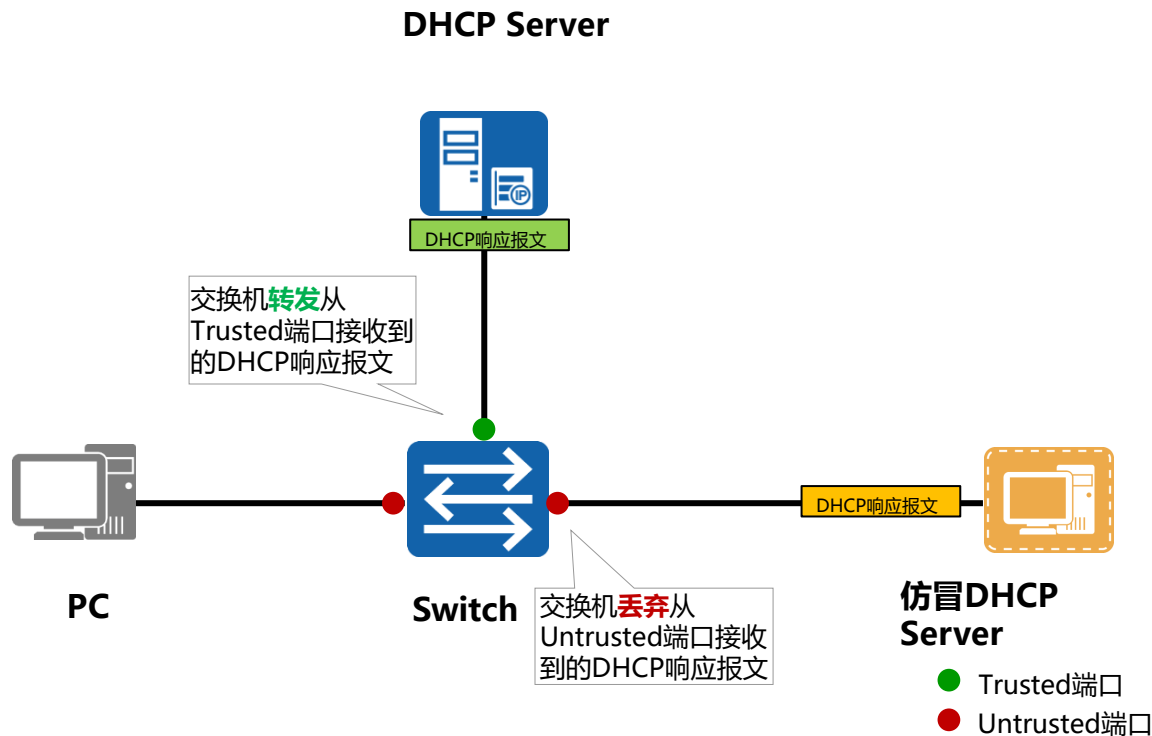
DHCP Snooping用于防止DHCP饿死攻击



● 开启dhcp-chaddr检查

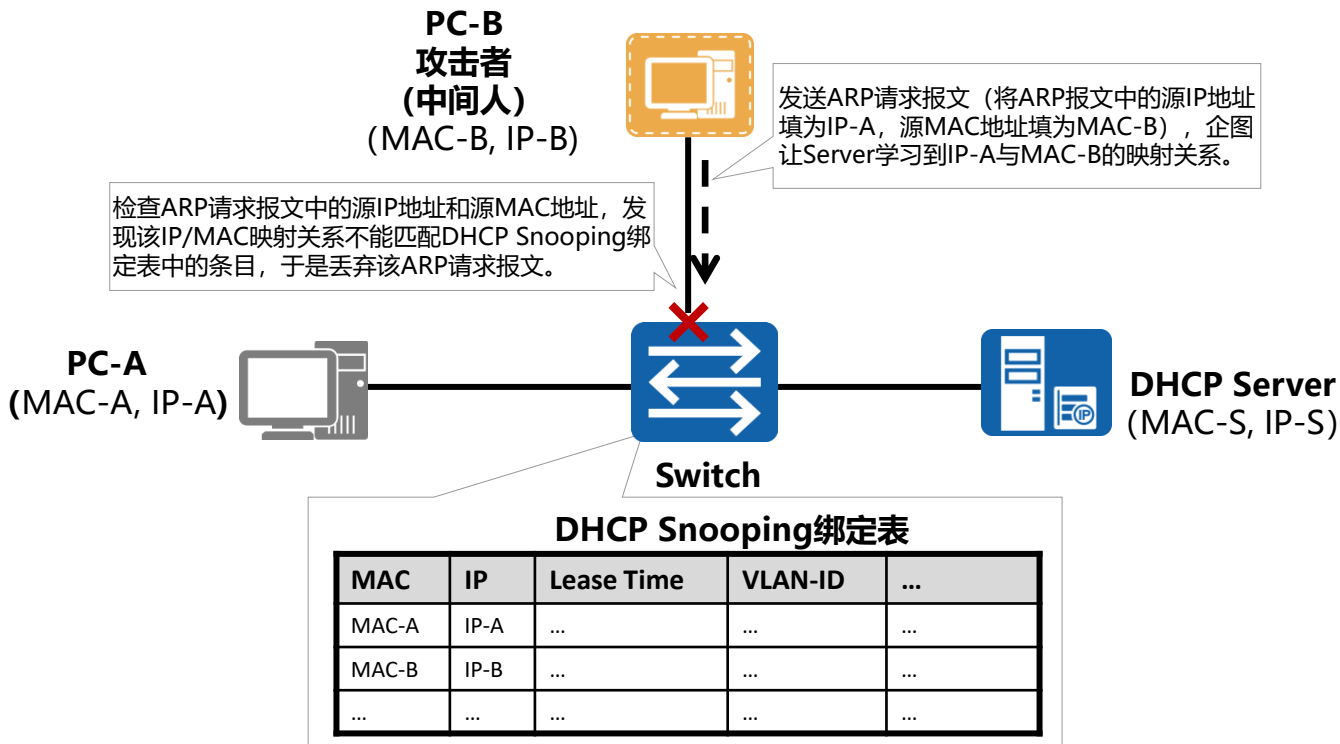


DHCP Snooping用于防止仿冒DHCP Server攻击



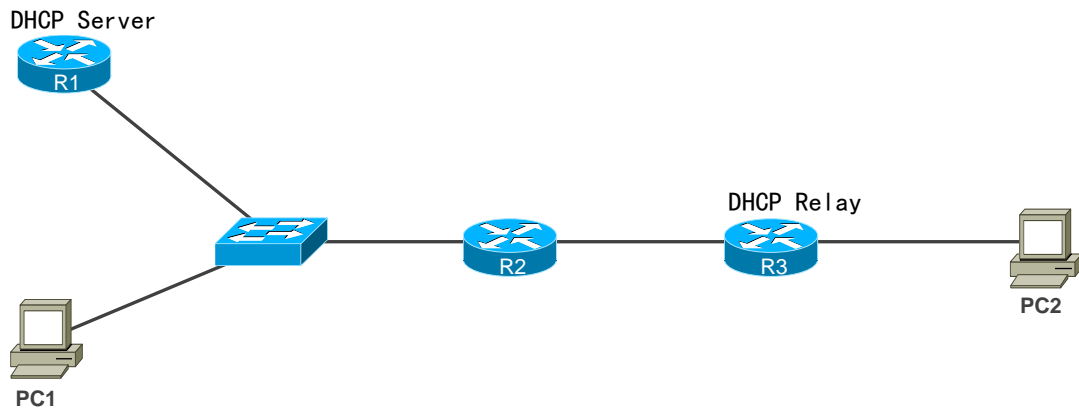


DHCP Snooping用于防止DHCP中间人攻击





课堂实验十



实验目的：

PC1和PC2通过dhcp server获取地址，且能互相访问

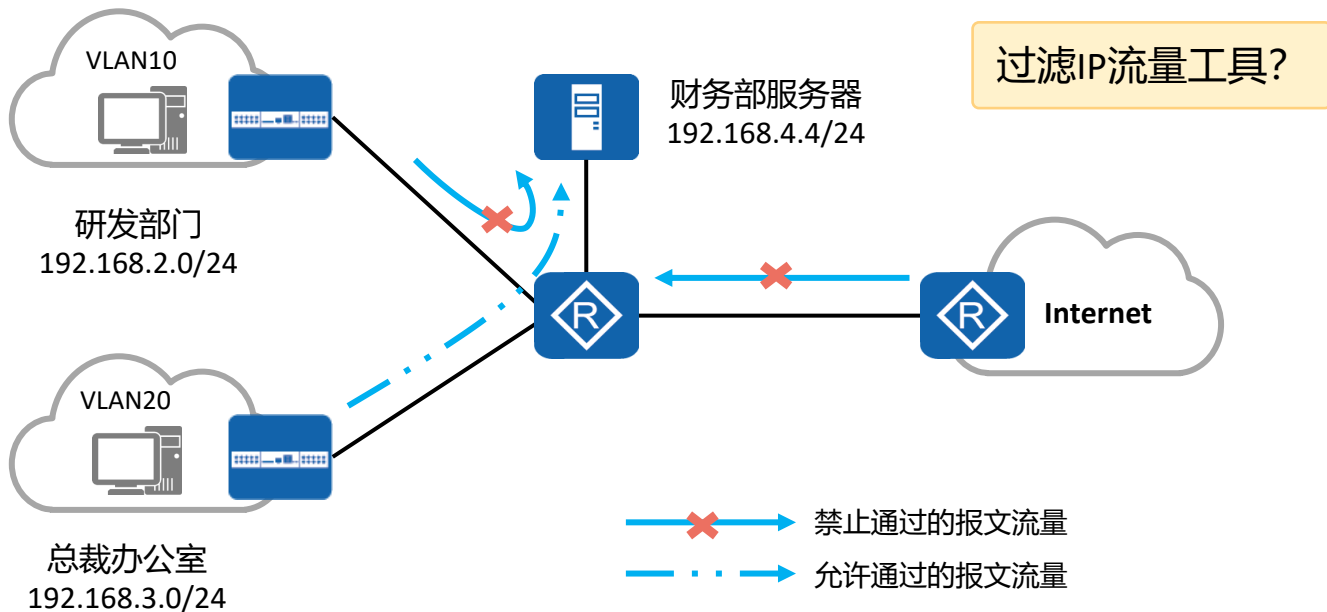


前言

- 随着网络的飞速发展，网络安全和网络服务质量QoS (Quality of Service) 问题日益突出。访问控制列表 (ACL, Access Control List) 是与其紧密相关的一个技术。
- **ACL**可以通过对网络中报文流的精确识别，与其他技术结合，达到控制网络访问行为、防止网络攻击和提高网络带宽利用率的目的，从而切实保障网络环境的安全性和网络服务质量的可靠性。



技术背景：需要一个工具，实现流量过滤

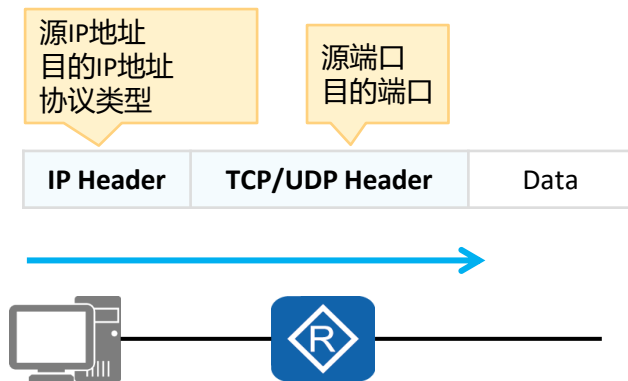


- 某公司为保证财务数据安全，禁止研发部门访问财务服务器，但总裁办公室不受限制。



ACL概述

- ACL是由一系列permit或deny语句组成的、有序规则的列表。
- ACL是一个匹配工具，能够对报文进行匹配和区分。



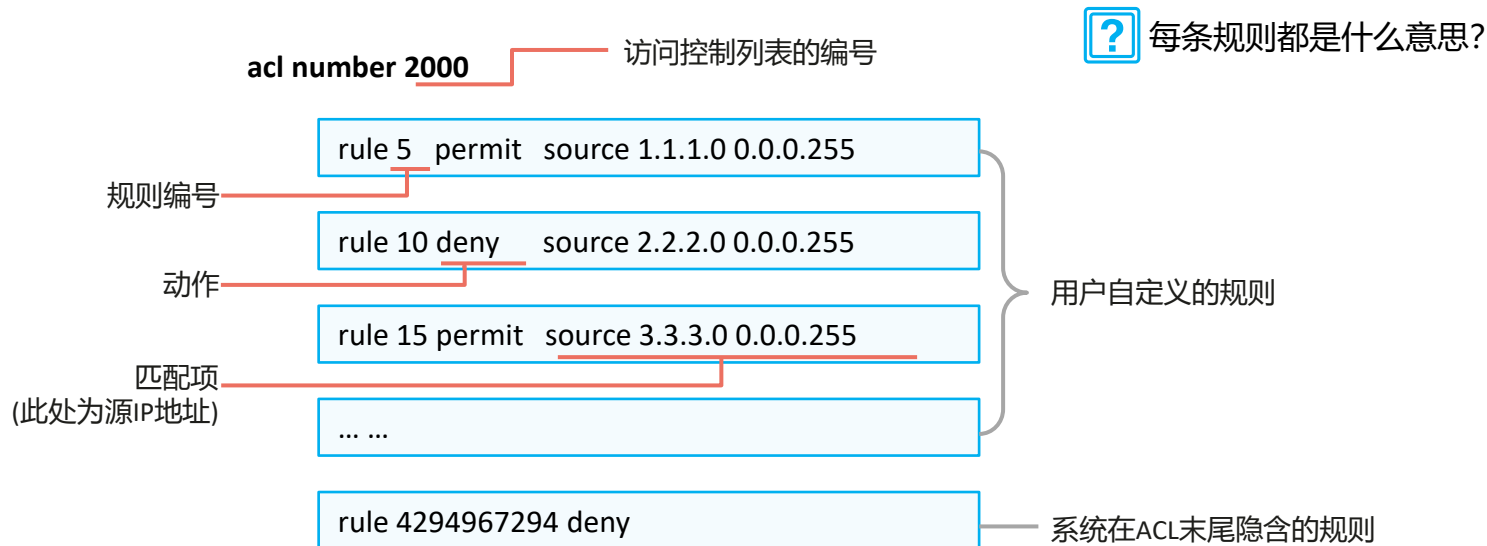
ACL应用

- 匹配IP流量
- 在Traffic-filter中被调用
- 在NAT（Network Address Translation）中被调用
- 在路由策略中被调用
- 在防火墙的策略部署中被调用
- 在QoS中被调用
- 其他.....



ACL的组成

- ACL由若干条permit或deny语句组成。每条语句就是该ACL的一条规则，每条语句中的permit或deny就是与这条规则相对应的处理动作。





规则编号

acl number 2000

	规则编号		
rule	5	deny	source 10.1.1.1 0
rule	10	deny	source 10.1.1.2 0
rule	15	permit	source 10.1.1.0 0.0.0.255

步长=5



如果希望增加1条规则，该如何处理？

rule 11 deny source 10.1.1.3 0

acl number 2000

rule	5	deny	source 10.1.1.1 0
rule	10	deny	source 10.1.1.2 0
rule	11	deny	source 10.1.1.3 0
rule	15	permit	source 10.1.1.0 0.0.0.255

规则编号与步长

- **规则编号 (Rule ID) :**

一个ACL中的每一条规则都有一个相应的编号。

- **步长 (Step) :**

步长是系统自动为ACL规则分配编号时，每个相邻规则编号之间的差值，缺省值为5。步长的作用是为了方便后续在旧规则之间，插入新的规则。

- **Rule ID分配规则:**

系统为ACL中首条未手工指定编号的规则分配编号时，使用步长值（例如步长=5，首条规则编号为5）作为该规则的起始编号；为后续规则分配编号时，则使用大于当前ACL内最大规则编号且是步长整数倍的最小整数作为规则编号。



通配符 (1)

acl number 2000

rule	5	deny	source 10.1.1.1 0
rule	10	deny	source 10.1.1.2 0
rule	15	permit	source 10.1.1.0 0.0.0.255


通配符

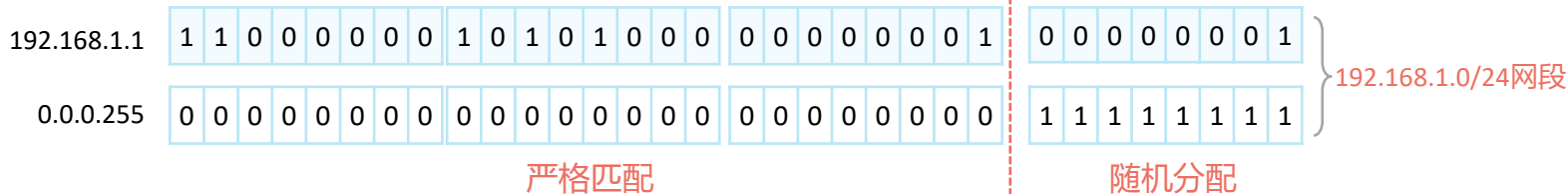
通配符 (Wildcard)

- 通配符是一个32比特长度的数值，用于指示IP地址中，哪些比特位需要严格匹配，哪些比特位无需匹配。
- 通配符通常采用类似网络掩码的点分十进制形式表示，但是含义却与网络掩码完全不同。

匹配规则：

“0”表示“匹配”；“1”表示“随机分配”

 如何匹配192.168.1.1/24对应网段的地址？





通配符 (2)

- 匹配192.168.1.0/24这个子网中的奇数IP地址，例如192.168.1.1、192.168.1.3、192.168.1.5等。

严格匹配

随机分配

严格匹配

192.168.1	1							
192.168.1	0	0	0	0	0	0	0	1

192.168.1	3							
192.168.1	0	0	0	0	0	0	1	1

192.168.1	5							
192.168.1	0	0	0	0	0	1	0	1

对应通配符

0.0.0.	1	1	1	1	1	1	1	0
--------	---	---	---	---	---	---	---	---

答案：192.168.1.1 0.0.0.254

通配符中的1或者0可以不连续

特殊的通配符

- 精确匹配192.168.1.1这个IP地址
192.168.1.1 0.0.0.0 = 192.168.1.1 0
- 匹配所有IP地址
0.0.0.0 255.255.255 = any



ACL的分类与标识

• 基于ACL规则定义方式的分类

分类	编号范围	规则定义描述
基本ACL	2000~2999	仅使用报文的源IP地址、分片信息和生效时间段信息来定义规则。
高级ACL	3000~3999	可使用IPv4报文的源IP地址、目的IP地址、IP协议类型、ICMP类型、TCP源/目的端口号、UDP源/目的端口号、生效时间段等来定义规则。
二层ACL	4000~4999	使用报文的以太网帧头信息来定义规则，如根据源MAC地址、目的MAC地址、二层协议类型等。
用户自定义ACL	5000~5999	使用报文头、偏移位置、字符串掩码和用户自定义字符串来定义规则。
用户ACL	6000~6999	既可使用IPv4报文的源IP地址或源UCL（User Control List）组，也可使用目的IP地址或目的UCL组、IP协议类型、ICMP类型、TCP源端口/目的端口、UDP源端口/目的端口号等来定义规则。

• 基于ACL标识方法的分类

分类	规则定义描述
数字型ACL	传统的ACL标识方法。创建ACL时，指定一个唯一的数字标识该ACL。
命名型ACL	通过名称代替编号来标识ACL。



ACL的类型

■ 标准 ACL

- 检查源地址
- 通常允许或拒绝整个协议簇

■ 扩展 ACL

- 检查源地址和目的地址
- 通常允许或拒绝特定协议和应用程序

■ 有两种用于标识

标准 ACL 和扩展 ACL 的方法：

- 编号 ACL 使用编号进行标识
- 命名 ACL 使用描述性名称或编号进行标识

IPv4 ACL 类型	编号范围/标识符
采用数字编号的标准 ACL、采用数字编号的扩展 ACL、命名 ACL（标准和扩展）	1–99, 1300–1999 100–199, 2000–2699 名称

- 标准编号 IP列表（1–99）可测试源地址的所有 IP 数据包的条件。扩展范围是（1300–1999）
- 扩展编号 IP列表（100–199）可测试源地址和目的地址、特定 TCP/IP 协议和目的端口的条件。扩展范围是（2000–2699）
- 命名 ACL 用字母数字字符串（名称）标识 IP 标准 ACL 和扩展 ACL



基本ACL&高级ACL

- 基本ACL

编号范围:
2000-2999

源IP地址

IP Header		TCP/UDP Header		Data
acl number 2000				
rule	5	deny	source 10.1.1.1 0	
rule	10	deny	source 10.1.1.2 0	
rule	15	permit	source 10.1.1.0 0.0.0.255	

- 高级ACL

编号范围:
3000-3999

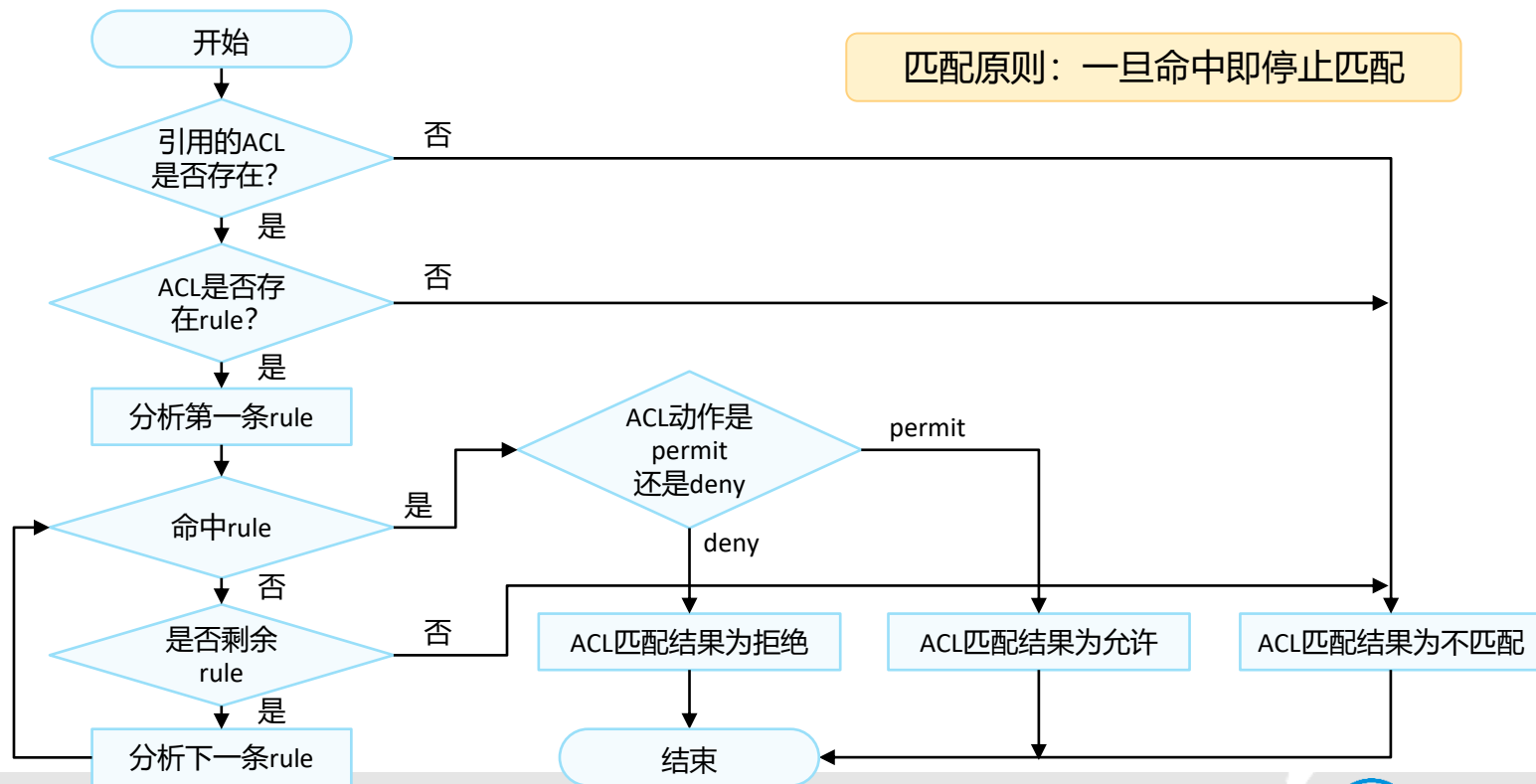
源IP地址目的IP
地址协议类型

源端口
目的端口

IP Header	TCP/UDP Header	Data
acl number 3000		
rule 5 permit ip source 10.1.1.0 0.0.0.255 destination 10.1.3.0 0.0.0.255		
rule 10 permit tcp source 10.1.2.0 0.0.0.255 destination 10.1.3.0 0.0.0.255 destination-port eq 21		



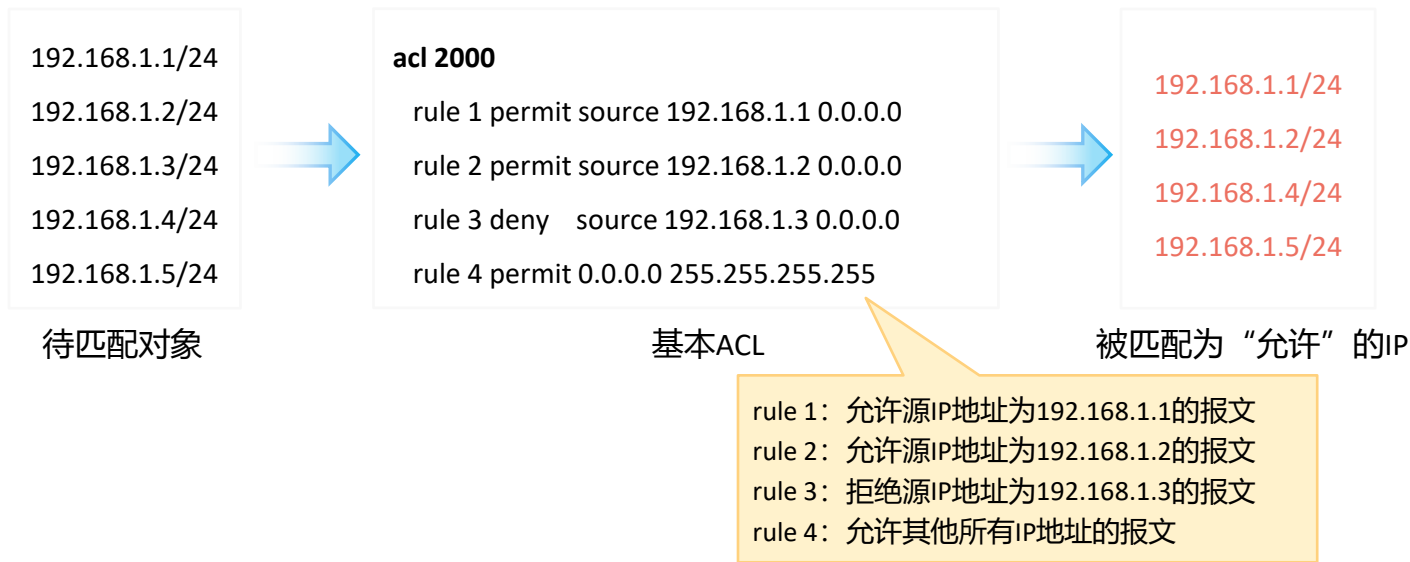
ACL的匹配机制





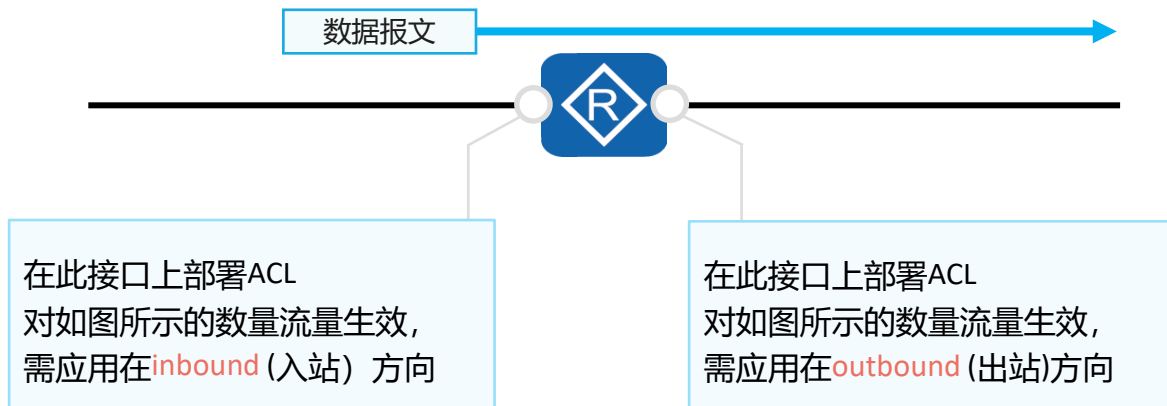
ACL的匹配顺序及匹配结果

- 配置顺序（config模式）
 - 系统按照ACL规则编号从小到大的顺序进行报文匹配，规则编号越小越容易被匹配。



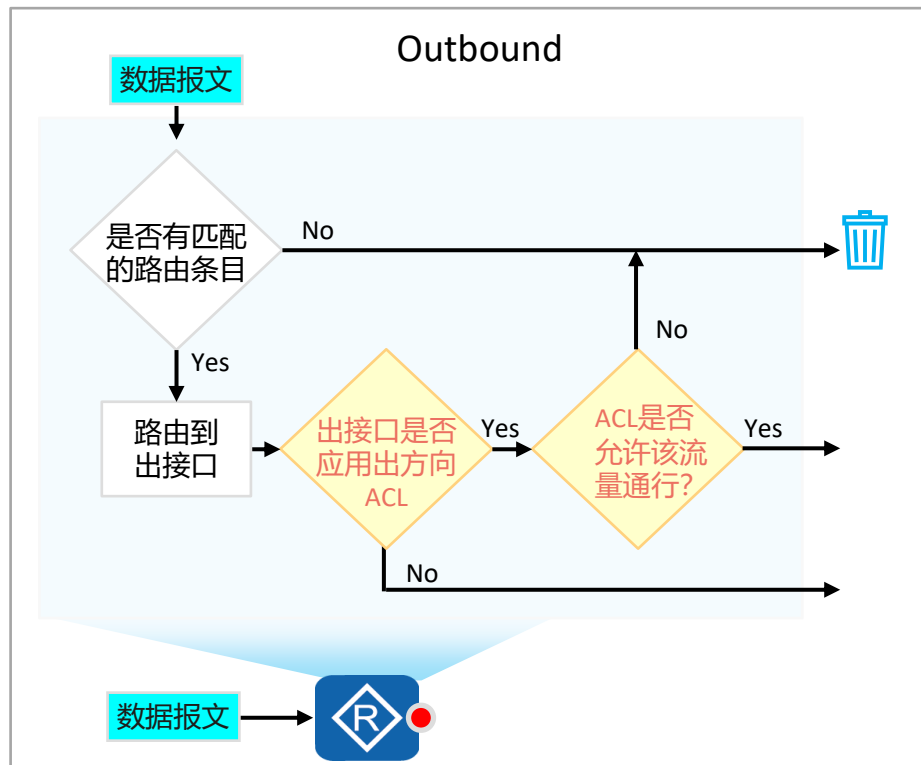
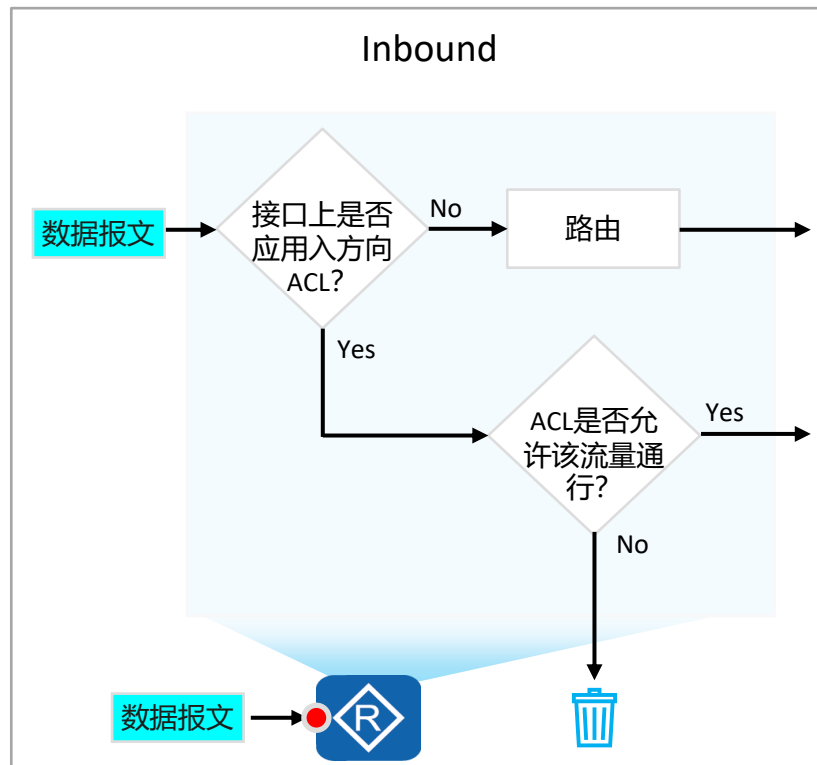


ACL的匹配位置





入站 (Inbound) 及出站 (Outbound) 方向





ACL配置的指导原则

- 每个接口、协议、方向只允许有一个 ACL
- ACL 语句的顺序控制着测试，因此最具体的语句位于列表顶部
- 最后的 ACL 测试始终是隐式拒绝其它所有语句，因此每个列表需要至少一条 permit 语句
- 在全局范围内创建 ACL，然后将其应用到入站流量或出站流量的接口
- 将 ACL 置于网络中时：
 - 扩展 ACL 应靠近源地址
 - 标准 ACL 应靠近目的地址



基本ACL的基础配置命令

1. 创建基本ACL

```
[Huawei] acl [ number ] acl-number [ match-order config ]
```

使用编号（2000 ~ 2999）创建一个数字型的基本ACL，并进入基本ACL视图。

```
[Huawei] acl name acl-name { basic | acl-number } [ match-order config ]
```

使用名称创建一个命名型的基本ACL，并进入基本ACL视图。

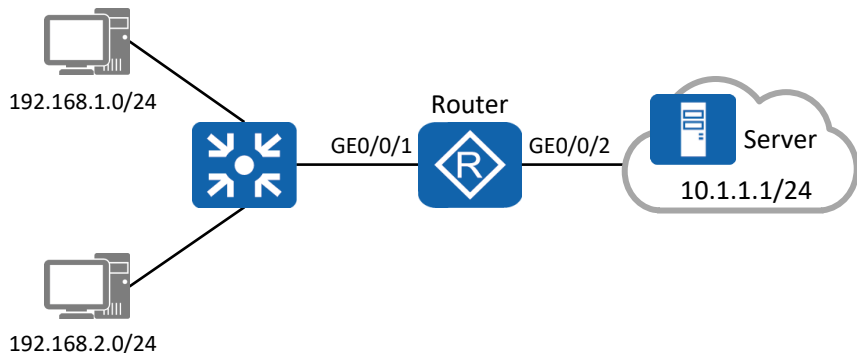
2. 配置基本ACL的规则

```
[Huawei-acl-basic-2000] rule [ rule-id ] { deny | permit } [ source { source-address source-wildcard | any } | time-range time-name ]
```

在基本ACL视图下，通过此命令来配置基本ACL的规则。



案例：使用基本ACL过滤数据流量



• 配置需求：

在Router上部署基本ACL后，ACL将试图穿越Router的源地址为192.168.1.0/24网段的数据包过滤掉，并放行其他流量，从而禁止192.168.1.0/24网段的用户访问Router右侧的服务器网络。

1、Router已完成IP地址和路由的相关配置

2、在Router上创建基本ACL，禁止192.168.1.0/24网段访问服务器网络：

```
[Router] acl 2000
[Router-acl-basic-2000] rule deny source 192.168.1.0 0.0.0.255
[Router-acl-basic-2000] rule permit source any
```

3、由于从接口GE0/0/1进入Router，所以在接口GE0/0/1的入方向配置流量过滤：

```
[Router] interface GigabitEthernet 0/0/1
[Router-GigabitEthernet0/0/1] traffic-filter inbound acl 2000
[Router-GigabitEthernet0/0/1] quit
```



高级ACL的基础配置命令 (1)

1. 创建高级ACL

```
[Huawei] acl [ number ] acl-number [ match-order config ]
```

使用编号（3000 ~ 3999）创建一个数字型的高级ACL，并进入高级ACL视图。

```
[Huawei] acl name acl-name { advance | acl-number } [ match-order config ]
```

使用名称创建一个命名型的高级ACL，进入高级ACL视图。



高级ACL的基础配置命令 (2)

2. 配置基本ACL的规则

根据IP承载的协议类型不同，在设备上配置不同的高级ACL规则。对于不同的协议类型，有不同的参数组合。

- 当参数protocol为IP时，高级ACL的命令格式为

```
rule [ rule-id ] { deny | permit } ip [ destination { destination-address destination-wildcard | any } | source { source-address source-wildcard | any } | time-range time-name | [ dscp dscp | [ tos tos | precedence precedence ] ] ]
```

在高级ACL视图下，通过此命令来配置高级ACL的规则。

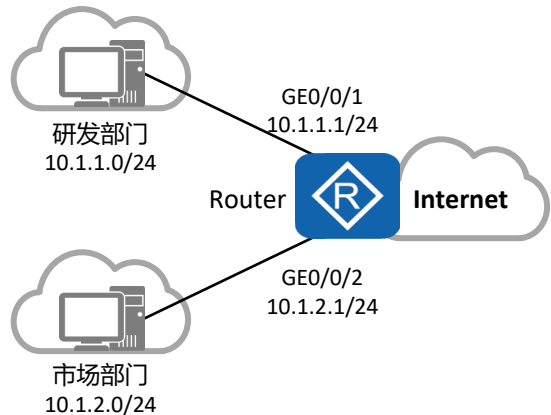
- 当参数protocol为TCP时，高级ACL的命令格式为

```
rule [ rule-id ] { deny | permit } { protocol-number | tcp } [ destination { destination-address destination-wildcard | any } | destination-port { eq port | gt port | lt port | range port-start port-end } | source { source-address source-wildcard | any } | source-port { eq port | gt port | lt port | range port-start port-end } | tcp-flag { ack | fin | syn } * | time-range time-name ] *
```

在高级ACL视图下，通过此命令来配置高级ACL的规则。



案例：使用高级ACL限制不同网段的用户互访（1）



配置需求：

- 某公司通过Router实现各部门之间的互连。为方便管理网络，管理员为公司的研发部和市场部规划了两个网段的IP地址。
- 现要求Router能够限制两个网段之间互访，防止公司机密泄露。

1、Router已完成IP地址和路由的相关配置。

2、创建高级ACL 3001并配置ACL规则，拒绝研发部访问市场部的报文：

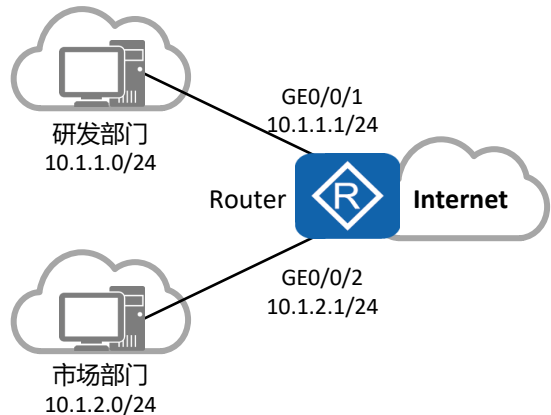
```
[Router] acl 3001
[Router-acl-adv-3001] rule deny ip source 10.1.1.0 0.0.0.255
destination 10.1.2.0 0.0.0.255
[Router-acl-adv-3001] quit
```

3、创建高级ACL 3002并配置ACL规则，拒绝市场部访问研发部的报文：

```
[Router] acl 3002
[Router-acl-adv-3002] rule deny ip source 10.1.2.0 0.0.0.255
destination 10.1.1.0 0.0.0.255
[Router-acl-adv-3002] quit
```



案例：使用高级ACL限制不同网段的用户互访（2）



配置需求：

- 某公司通过Router实现各部门之间的互连。为方便管理网络，管理员为公司的研发部和市场部规划了两个网段的IP地址。
- 现要求Router能够限制两个网段之间互访，防止公司机密泄露。

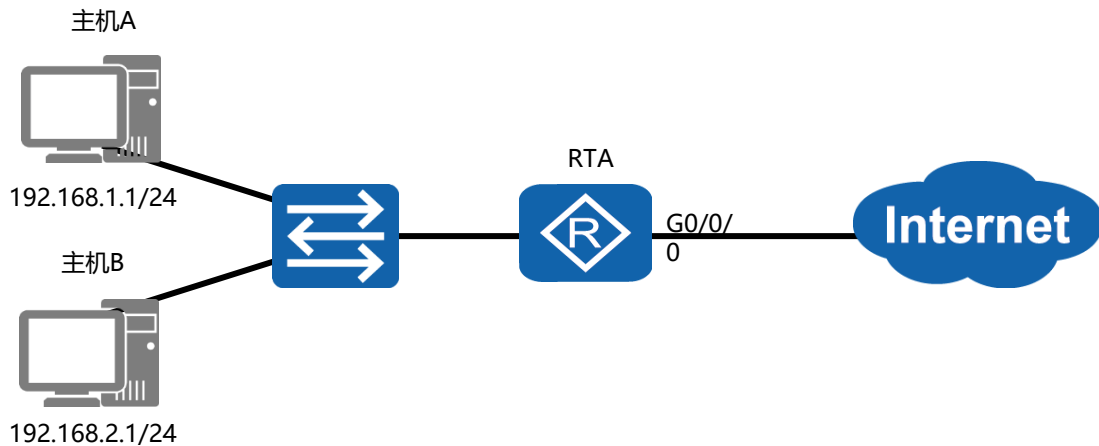
4、由于研发部和市场部互访的流量分别从接口GE0/0/1和GE0/0/2进入Router，所以在接口GE0/0/1和GE0/0/2的入方向配置流量过滤：

```
[Router] interface GigabitEthernet 0/0/1
[Router-GigabitEthernet0/0/1] traffic-filter inbound acl 3001
[Router-GigabitEthernet0/0/1] quit

[Router] interface GigabitEthernet 0/0/2
[Router-GigabitEthernet0/0/2] traffic-filter inbound acl 3002
[Router-GigabitEthernet0/0/2] quit
```



基本ACL配置



```
[RTA]acl 2000
[RTA-acl-basic-2000]rule deny source 192.168.1.0 0.0.0.255
[RTA]interface GigabitEthernet 0/0/0
[RTA-GigabitEthernet 0/0/0]traffic-filter outbound acl 2000
```



配置确认

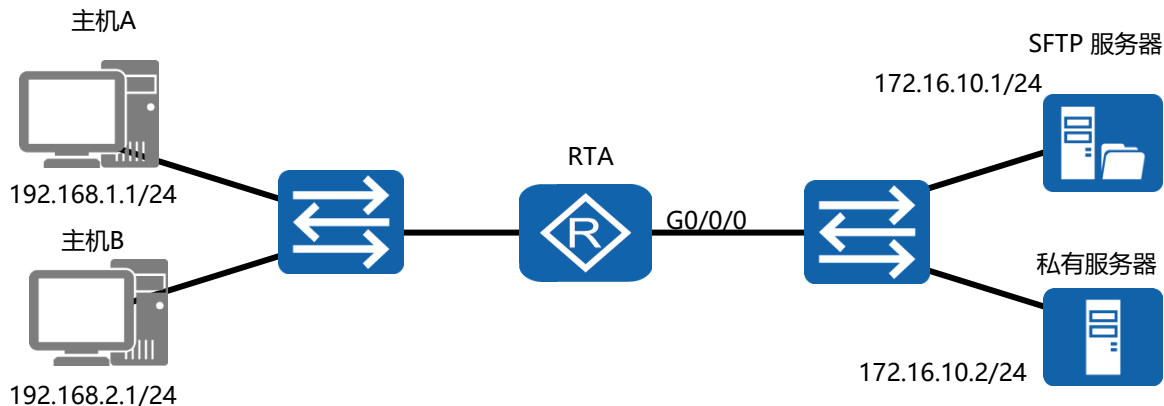
```
[RTA]display acl 2000  
Basic ACL 2000, 1 rule  
Acl's step is 5  
rule 5 deny source 192.168.1.0 0.0.0.255
```

```
[RTA]display traffic-filter applied-record
```

Interface	Direction	AppliedRecord
GigabitEthernet0/0/0	outbound	acl 2000



高级ACL配置



```
[RTA]acl 3000
[RTA-acl-adv-3000]rule deny tcp source 192.168.1.0 0.0.0.255
destination 172.16.10.1 0.0.0.0 destination-port eq 21
[RTA-acl-adv-3000]rule deny tcp source 192.168.2.0 0.0.0.255
destination 172.16.10.2 0.0.0.0
[RTA-acl-adv-3000]rule permit ip
[RTA-GigabitEthernet 0/0/0]traffic-filter outbound acl 3000
```



配置验证

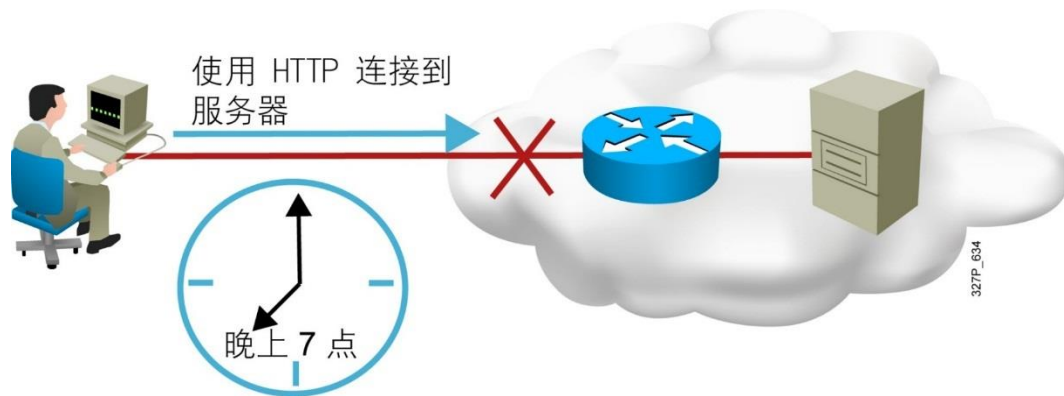
```
[RTA]display acl 3000
Advanced ACL 3000, 3 rules
Acl's step is 5
rule 5 deny tcp source 192.168.1.0 0.0.0.255 destination 172.16.10.1 0
destination-port eq sftp
rule 10 deny tcp source 192.168.2.0 0.0.0.255 destination 172.16.10.2 0
rule 15 permit ip
```

```
[RTA]display traffic-filter applied-record
```

Interface	Direction	AppliedRecord
GigabitEthernet0/0/0	outbound	acl 3000



基于时间的ACL（补充）

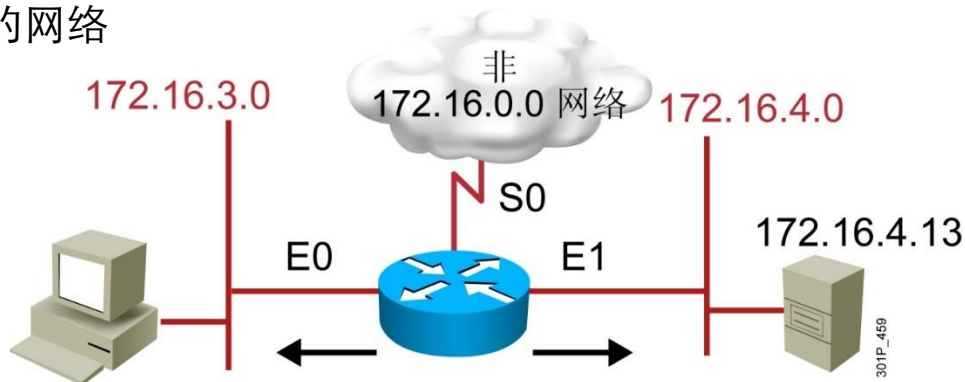


基于时间的 ACL：允许根据天数和周数控制访问



标准ACL实例一

- 仅允许我的网络



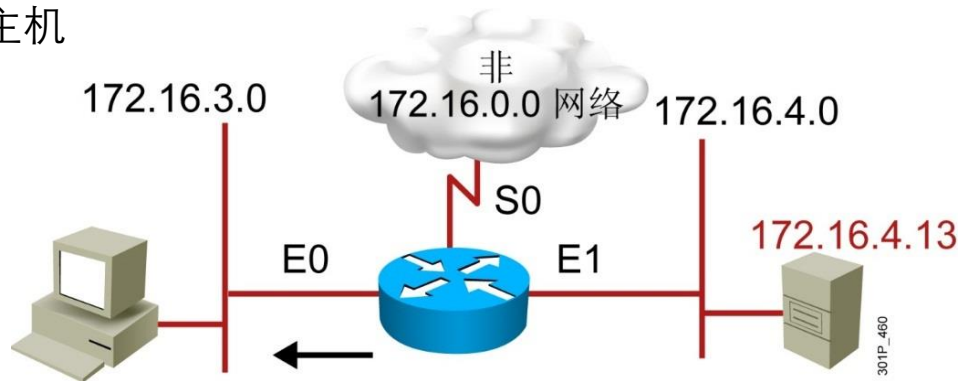
```
RouterX(config)# access-list 1 permit 172.16.0.0 0.0.255.255  
(隐式拒绝所有 — 在列表中不可见)  
(access-list 1 deny 0.0.0.0 255.255.255.255)
```

```
RouterX(config)# interface ethernet 0  
RouterX(config-if)# ip access-group 1 out  
RouterX(config)# interface ethernet 1  
RouterX(config-if)# ip access-group 1 out
```




标准ACL实例二

- 拒绝特定主机



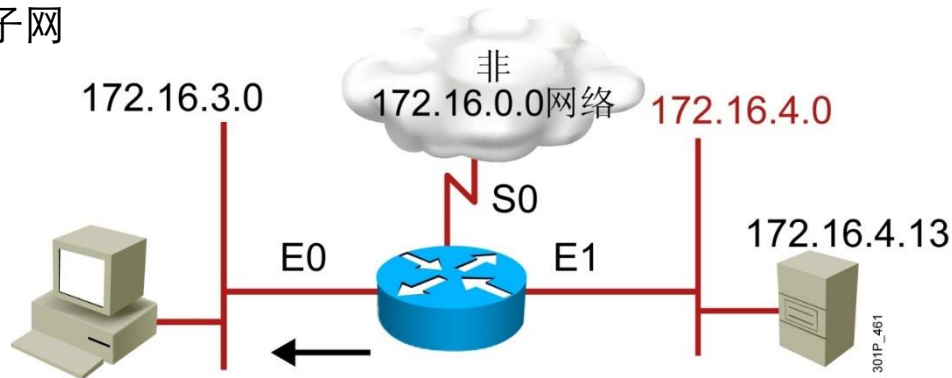
```
RouterX(config)# access-list 1 deny 172.16.4.13 0.0.0.0
RouterX(config)# access-list 1 permit 0.0.0.0 255.255.255.255
(隐式拒绝所有)
(access-list 1 deny 0.0.0.0 255.255.255.255)

RouterX(config)# interface ethernet 0
RouterX(config-if)# ip access-group 1 out
```



标准ACL实例三

- 拒绝特定子网



```
RouterX(config)# access-list 1 deny 172.16.4.0 0.0.0.255
RouterX(config)# access-list 1 permit any
(隐式拒绝所有)
(access-list 1 deny 0.0.0.0 255.255.255.255)

RouterX(config)# interface ethernet 0
RouterX(config-if)# ip access-group 1 out
```



标准ACL实例四（控制VTY访问）

示例：

```
access-list 12 permit 192.168.1.0 0.0.0.255  
（隐式拒绝所有）
```

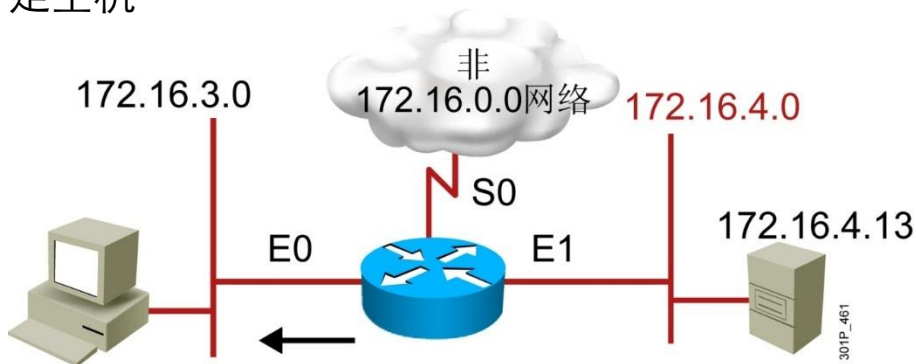
```
line vty 0 4  
access-class 12 in
```

- 仅允许网络 192.168.1.0 0.0.0.255 中的主机连接到路由器的 vty 线路



标准命名ACL实例

- 拒绝特定主机

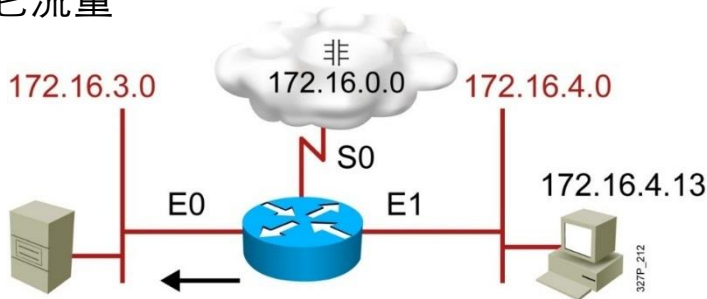


```
RouterX(config)#ip access-list standard troublemaker
RouterX(config-std-nacl)#deny host 172.16.4.13
RouterX(config-std-nacl)#permit 172.16.4.0 0.0.0.255
RouterX(config-std-nacl)#interface e0
RouterX(config-if)#ip access-group troublemaker out
```



扩展ACL实例一

- 拒绝从子网 172.16.4.0 到子网 172.16.3.0 经 E0 流出的 FTP 流量
- 允许所有其它流量



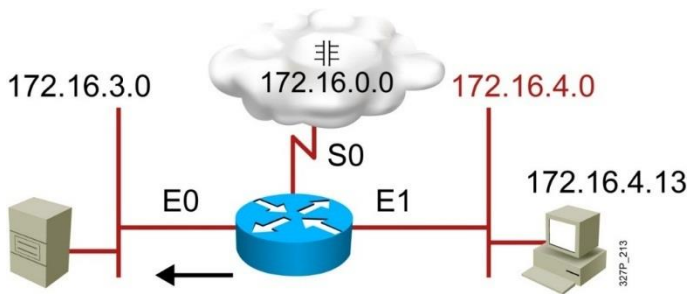
```
RouterX(config)# access-list 101 deny tcp 172.16.4.0 0.0.0.255 172.16.3.0 0.0.0.255 eq 21
RouterX(config)# access-list 101 permit ip any any
(隐式拒绝所有)
(access-list 101 deny ip 0.0.0.0 255.255.255.255 0.0.0.0 255.255.255.255)
```

```
RouterX(config)# interface ethernet 0
RouterX(config-if)# ip access-group 101 out
```



扩展ACL实例二

- 仅拒绝来自子网 172.16.4.0 经 E0 流出的 Telnet 流量
- 允许所有其它流量



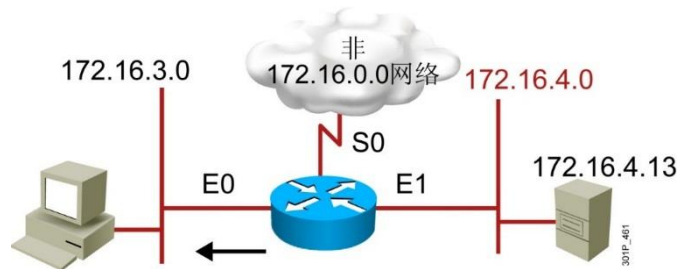
```
RouterX(config)# access-list 101 deny tcp 172.16.4.0 0.0.0.255 any eq 23
RouterX(config)# access-list 101 permit ip any any
(隐式拒绝所有)
```

```
RouterX(config)# interface ethernet 0
RouterX(config-if)# ip access-group 101 out
```



扩展命名ACL实例

- 拒绝来自特定子网的 Telnet 流量
- 允许所有其它流量



```
RouterX(config)#ip access-list extended badgroup
RouterX(config-ext-nacl)#deny tcp 172.16.4.0 0.0.0.255 any eq 23
RouterX(config-ext-nacl)#permit ip any any
RouterX(config-ext-nacl)#interface e0
RouterX(config-if)#ip access-group badgroup out
```



ACL注释和检查

ACL 注释

```
ip access-list {standard|extended} name  
remark remark
```

或

```
access-list access-list-number remark remark
```

ACL 检查

```
RouterX# show access-lists
```

Standard IP access list SALES

10 deny 10.1.1.0, wildcard bits 0.0.0.255

20 permit 10.3.3.1

30 permit 10.4.4.1

40 permit 10.5.5.1

Extended IP access list ENG

10 permit tcp host 10.22.22.1 any eq telnet (25 matches)

20 permit tcp host 10.33.33.1 any eq ftp

30 permit tcp host 10.44.44.1 any eq ftp-data



课堂实验十一

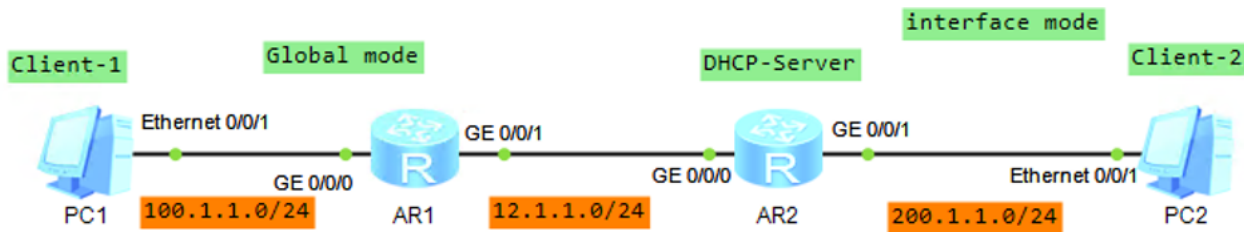


实验目的：

1. R1/R3各虚拟3个环回口模拟主机，用静态路由实现网络互通
2. 通过标准只允许R1的loop0和loop1网段访问R3
3. 用命名的ACL只允许R1 loop0访问R3 loop0的ICMP协议
4. 用扩展的ACL只允许R1 loop1，在每个工作日的9点到18点，访问R3 loop1的Telnet协议



课后实验



1. PC1和PC2都是通过DHCP获取IP地址，R2是DHCP-Server，华为的按照图的要求实现DHCP的地址获取，思科正常配置地址获取
2. 按照访问控制进行数据的控制
允许PC1pingPC2，拒绝PC1访问其他网段，允许R1telnetR2在工作日的9：00-18：00
分别用思科和华为的设备进行实验



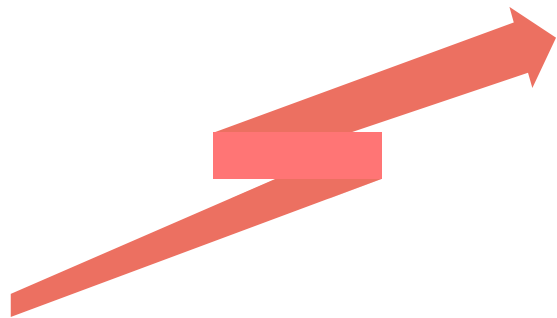
前言

- 随着Internet的发展和网络应用的增多，有限的IPv4公有地址已经成为制约网络发展的瓶颈。为解决这个问题，NAT（Network Address Translation，网络地址转换）技术应运而生。
- NAT技术主要用于实现内部网络的主机访问外部网络。一方面NAT缓解了IPv4地址短缺的问题，另一方面NAT技术让外网无法直接与使用私有地址的内网进行通信，提升了内网的安全性。

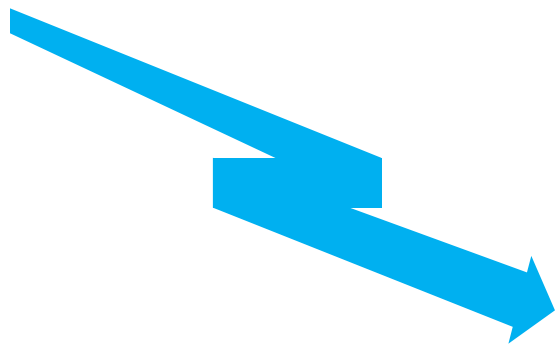


NAT产生背景

- 随着互联网用户的增多，IPv4的公有地址资源显得越发短缺。
- 同时IPv4公有地址资源存在地址分配不均的问题，这导致部分地区的IPv4可用公有地址严重不足。
- 为解决该问题，使用过渡技术解决IPv4公有地址短缺就显得尤为必要。



互联网用户



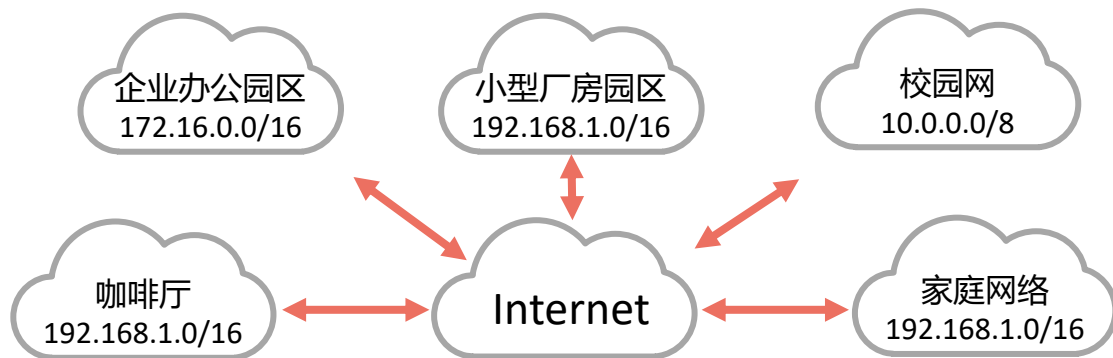
公有IPv4地址

0



私网IP地址

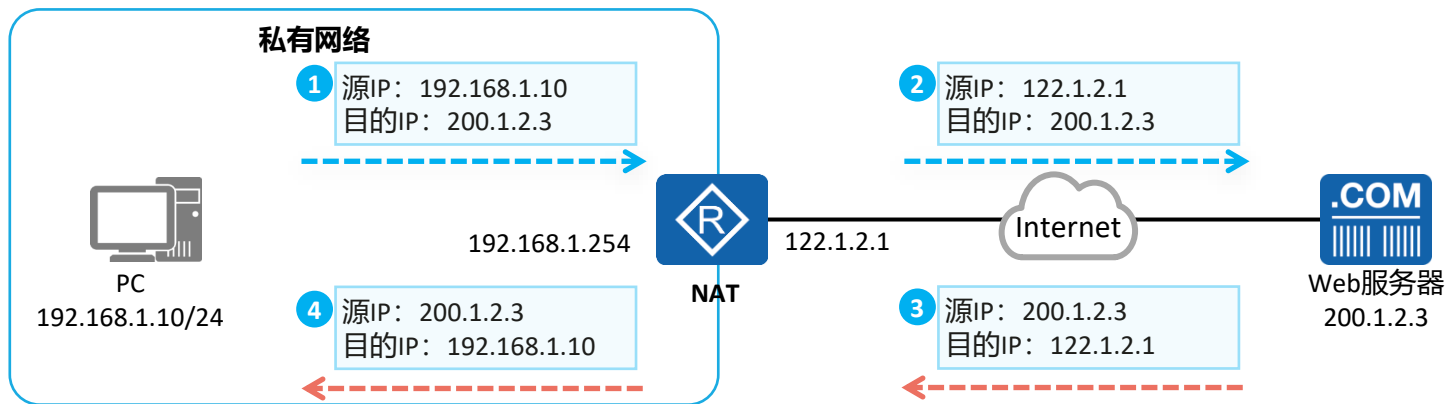
- 公有地址：由专门的机构管理、分配，可以在Internet上直接通信的IP地址。
- 私有地址：组织和个人可以任意使用，无法在Internet上直接通信，只能在内网使用的IP地址。
- A、B、C类地址中各预留了一些地址专门作为私有IP地址：
 - A类：10.0.0.0 ~ 10.255.255.255
 - B类：172.16.0.0 ~ 172.31.255.255
 - C类：192.168.0.0 ~ 192.168.255.255





NAT技术原理

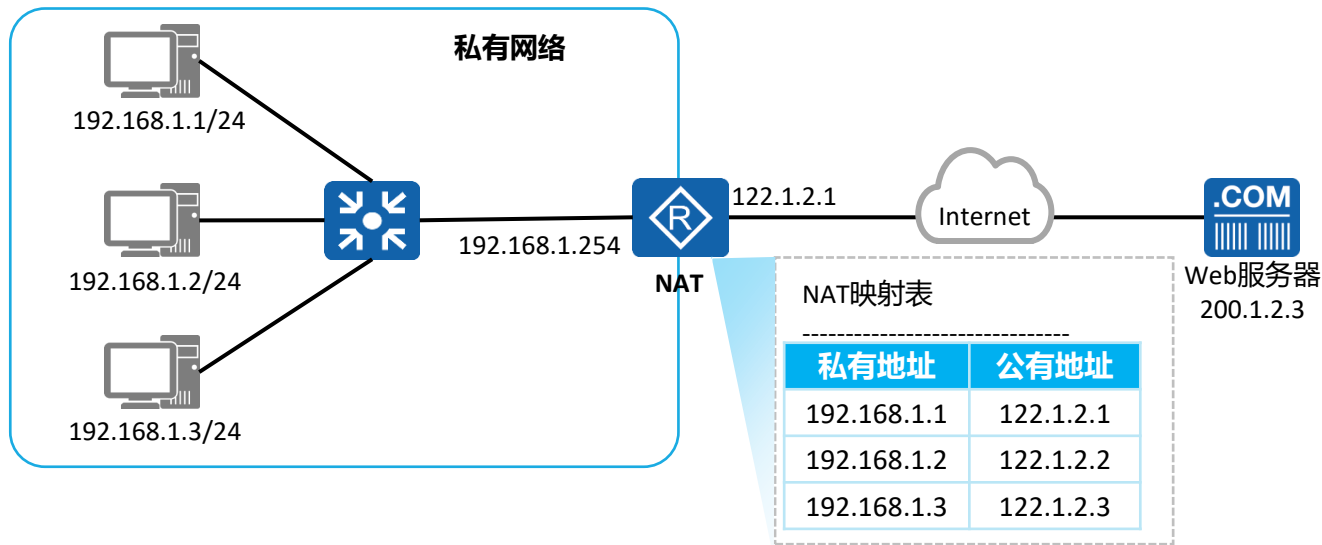
- **NAT:** 对IP数据报文中的IP地址进行转换，是一种在现网中被广泛部署的技术，一般部署在网络出口设备，例如路由器或防火墙上。
- **NAT的典型应用场景:** 在私有网络内部（园区、家庭）使用私有地址，出口设备部署**NAT**，对于“从内到外”的流量，网络设备通过**NAT**将数据包的源地址进行转换（转换成特定的公有地址），而对于“从外到内的”流量，则对数据包的目的地址进行转换。
- 通过私有地址的使用结合**NAT**技术，可以有效节约公网IPv4地址。





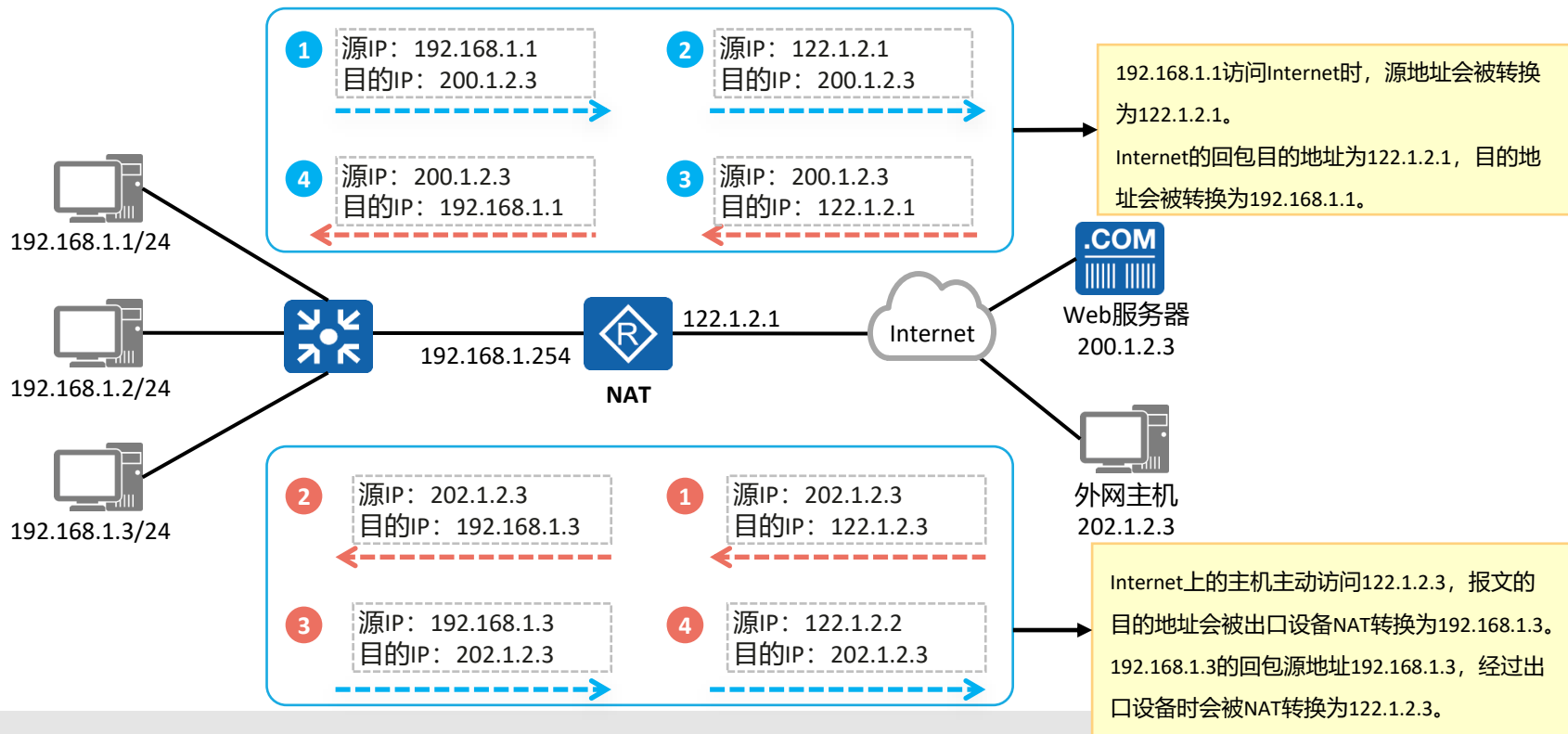
静态NAT原理

- 静态NAT：每个私有地址都有一个与之对应并且固定的公有地址，即私有地址和公有地址之间的关系是一对一映射。
- 支持双向互访：私有地址访问Internet经过出口设备NAT转换时，会被转换成对应的公有地址。同时，外部网络访问内部网络时，其报文中携带的公有地址（目的地址）也会被NAT设备转换成对应的私有地址。





静态NAT转换示例





静态NAT配置介绍

1. 方式一：接口视图下配置静态NAT

```
[Huawei-GigabitEthernet0/0/0] nat static global { global-address } inside { host-address }
```

global参数用于配置外部公有地址，inside参数用于配置内部私有地址。

2. 方式二：系统视图下配置静态NAT

```
[Huawei] nat static global { global-address } inside { host-address }
```

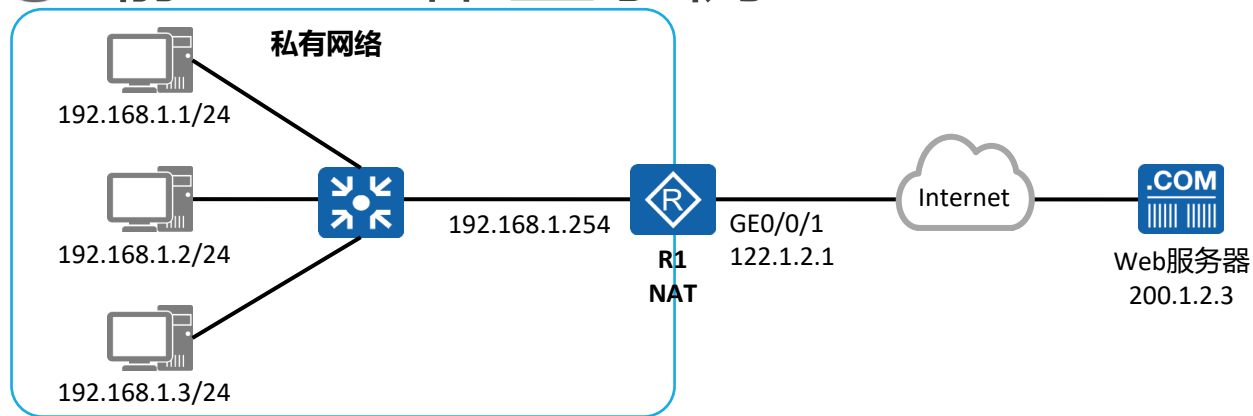
配置命令相同，视图为系统视图，之后在具体的接口下开启静态NAT。

```
[Huawei-GigabitEthernet0/0/0] nat static enable
```

在接口下使能nat static功能。



静态NAT配置示例



- 在R1上配置静态NAT将内网主机的私有地址一对一映射到公有地址。

```
[R1]interface GigabitEthernet0/0/1
[R1-GigabitEthernet0/0/1]ip address 122.1.2.1 24
[R1-GigabitEthernet0/0/1]nat static global 122.1.2.1 inside 192.168.1.1
[R1-GigabitEthernet0/0/1]nat static global 122.1.2.2 inside 192.168.1.2
[R1-GigabitEthernet0/0/1]nat static global 122.1.2.3 inside 192.168.1.3
```



配置验证

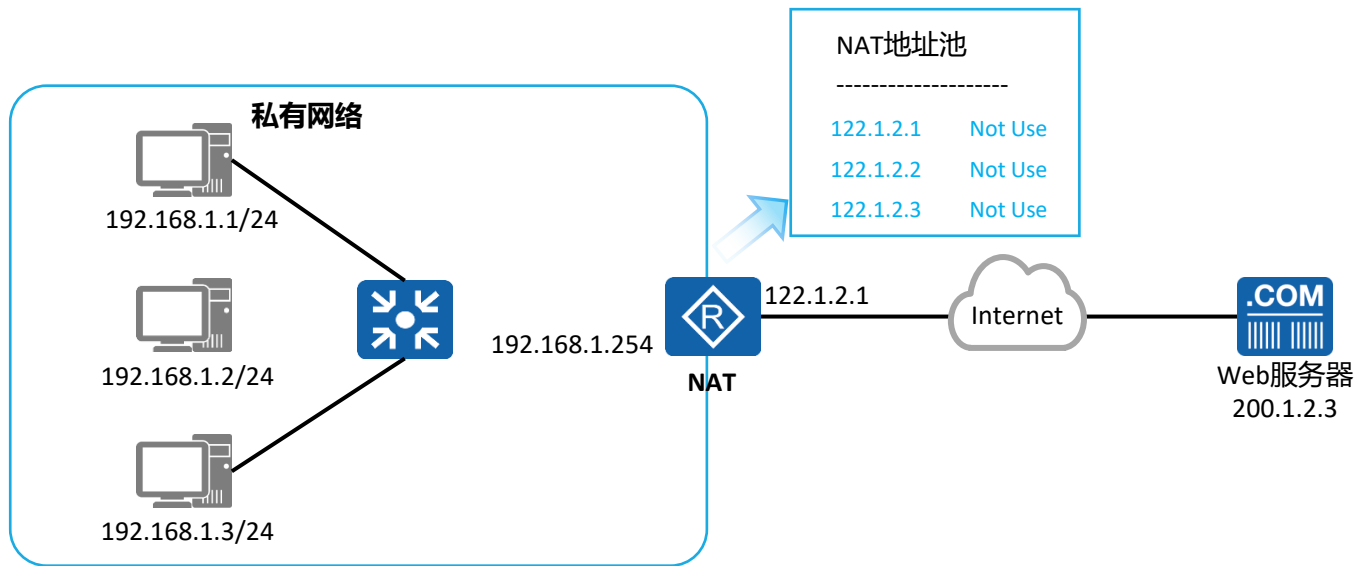
```
[RTA]display nat static
Static Nat Information:
Interface   : Serial1/0/0
Global IP/Port      : 202.10.10.1/----
Inside IP/Port      : 192.168.1.1/----
.....
Global IP/Port      : 202.10.10.2/----
Inside IP/Port      : 192.168.1.2/----
.....

Total :      2
```



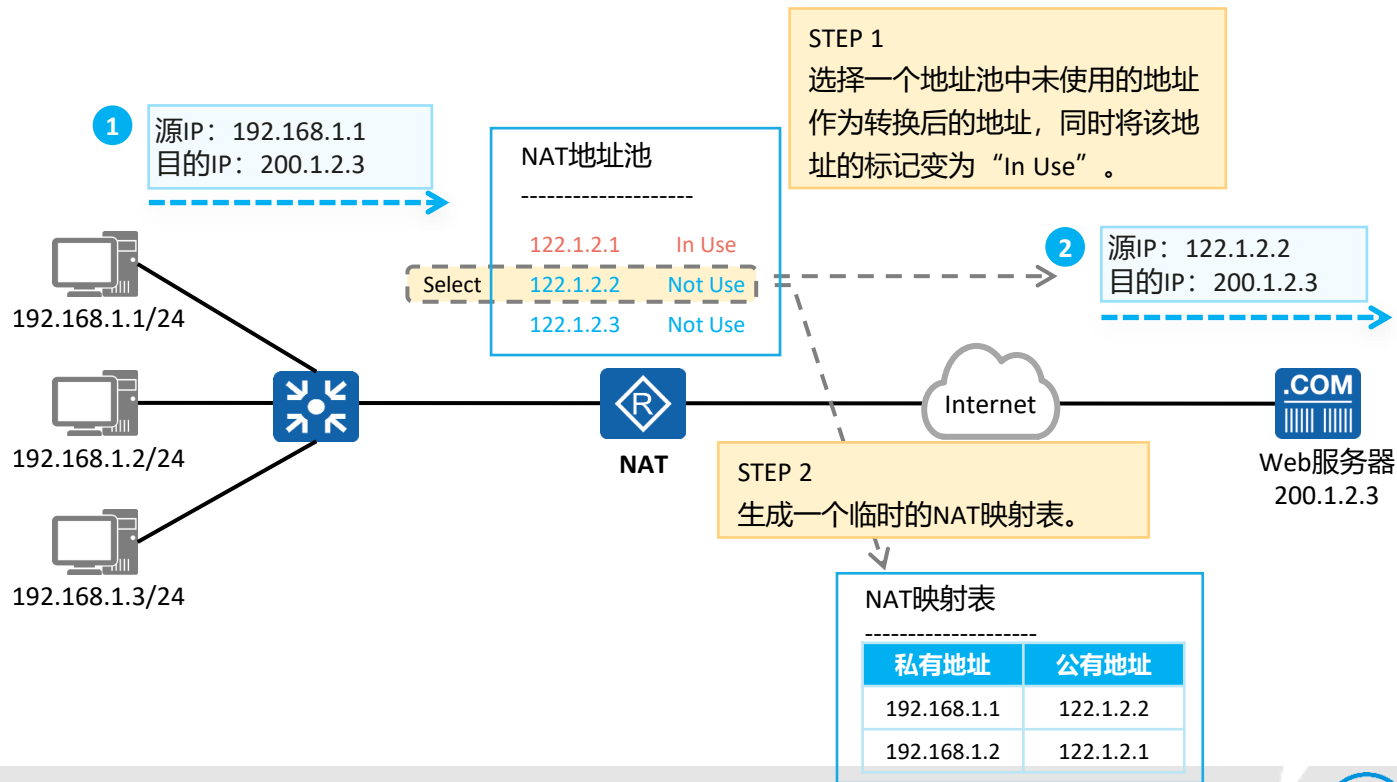
动态NAT原理

- 动态NAT：静态NAT严格地一对一进行地址映射，这就导致即便内网主机长时间离线或者不发送数据时，与之对应的公有地址也处于使用状态。为了避免地址浪费，动态NAT提出了地址池的概念：所有可用的公有地址组成地址池。
- 当内部主机访问外部网络时临时分配一个地址池中未使用的地址，并将该地址标记为“**In Use**”。当该主机不再访问外部网络时回收分配的地址，重新标记为“**Not Use**”。



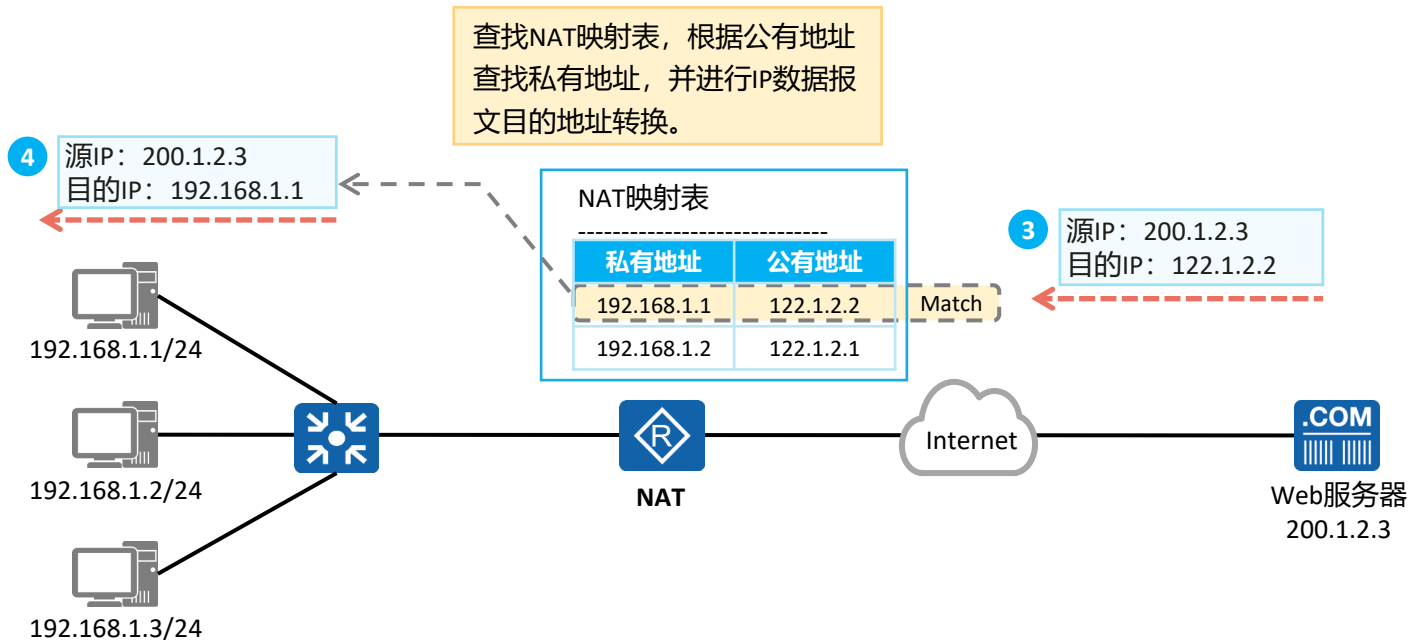


动态NAT转换示例 (1)





动态NAT转换示例 (2)





动态NAT配置介绍

1. 创建地址池

```
[Huawei] nat address-group group-index start-address end-address
```

配置公有地址范围，其中group-index为地址池编号，start-address、end-address分别为地址池起始地址、结束地址。

2. 配置地址转换的ACL规则

```
[Huawei] acl number
```

```
[Huawei-acl-basic-number] rule permit source source-address source-wildcard
```

配置基础ACL，匹配需要进行动态转换的源地址范围。

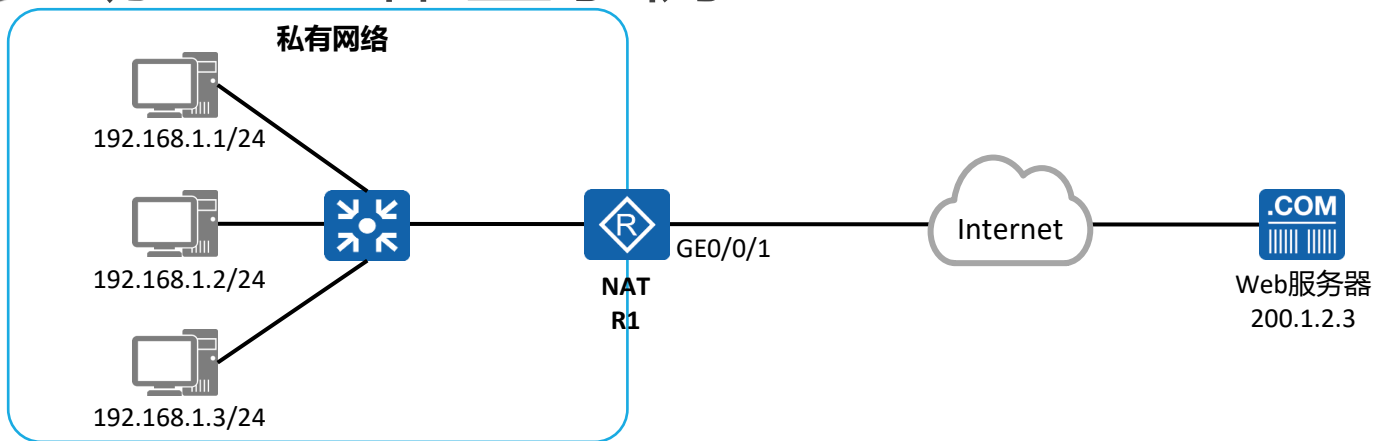
3. 接口视图下配置带地址池的NAT Outbound

```
[Huawei-GigabitEthernet0/0/0] nat outbound acl-number address-group group-index [ no-pat ]
```

接口下关联ACL与地址池进行动态地址转换，no-pat参数指定不进行端口转换。



动态NAT配置示例



- 在R1上配置动态NAT将内网主机的私有地址动态映射到公有地址。

```
[R1]nat address-group 1 122.1.2.1 122.1.2.3
[R1]acl 2000
[R1-acl-basic-2000]rule 5 permit source 192.168.1.0 0.0.0.255
[R1-acl-basic-2000]quit
[R1]interface GigabitEthernet0/0/1
[R1-GigabitEthernet0/0/1]nat outbound 2000 address-group 1 no-pat
```




配置验证

```
[RTA]display nat address-group 1
```

```
NAT Address-Group Information:
```

Index	Start-address	End-address
1	200.10.10.1	200.10.10.200

```
[RTA]display nat outbound
```

```
NAT Outbound Information:
```

Interface	Acl	Address-group/IP/Interface	Type

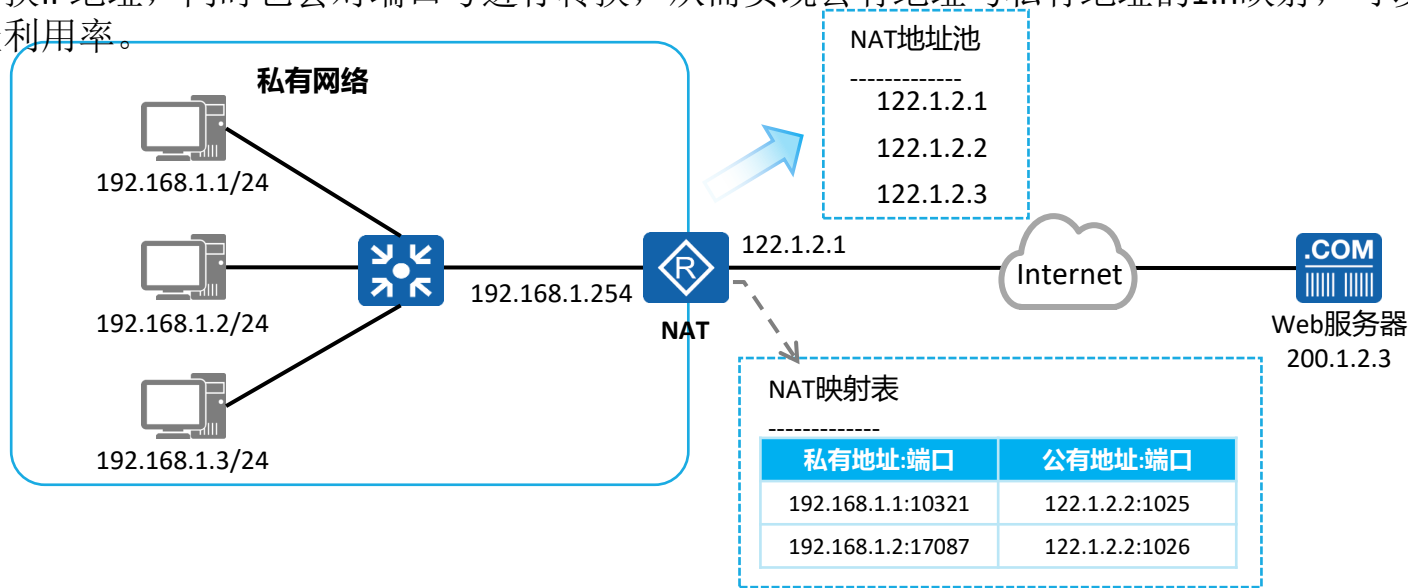
Serial1/0/0	2000	1	no-pat

Total : 1			



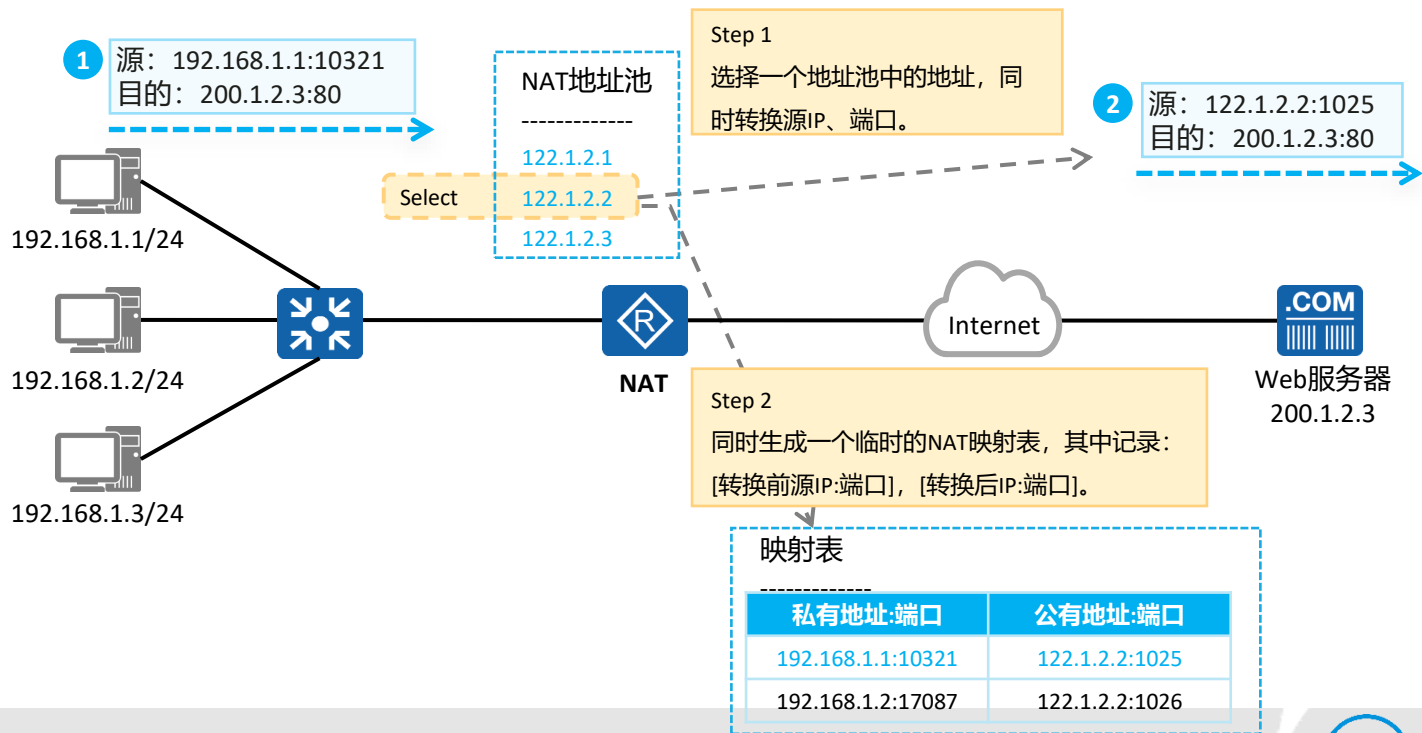
NAPT原理

- 动态NAT选择地址池中的地址进行地址转换时不会转换端口号，即No-PAT（No-Port Address Translation，非端口地址转换），公有地址与私有地址还是1:1的映射关系，无法提高公有地址利用率。
- NAPT（Network Address and Port Translation，网络地址端口转换）：从地址池中选择地址进行地址转换时不仅转换IP地址，同时也会对端口号进行转换，从而实现公有地址与私有地址的1:n映射，可以有效提高公有地址利用率。



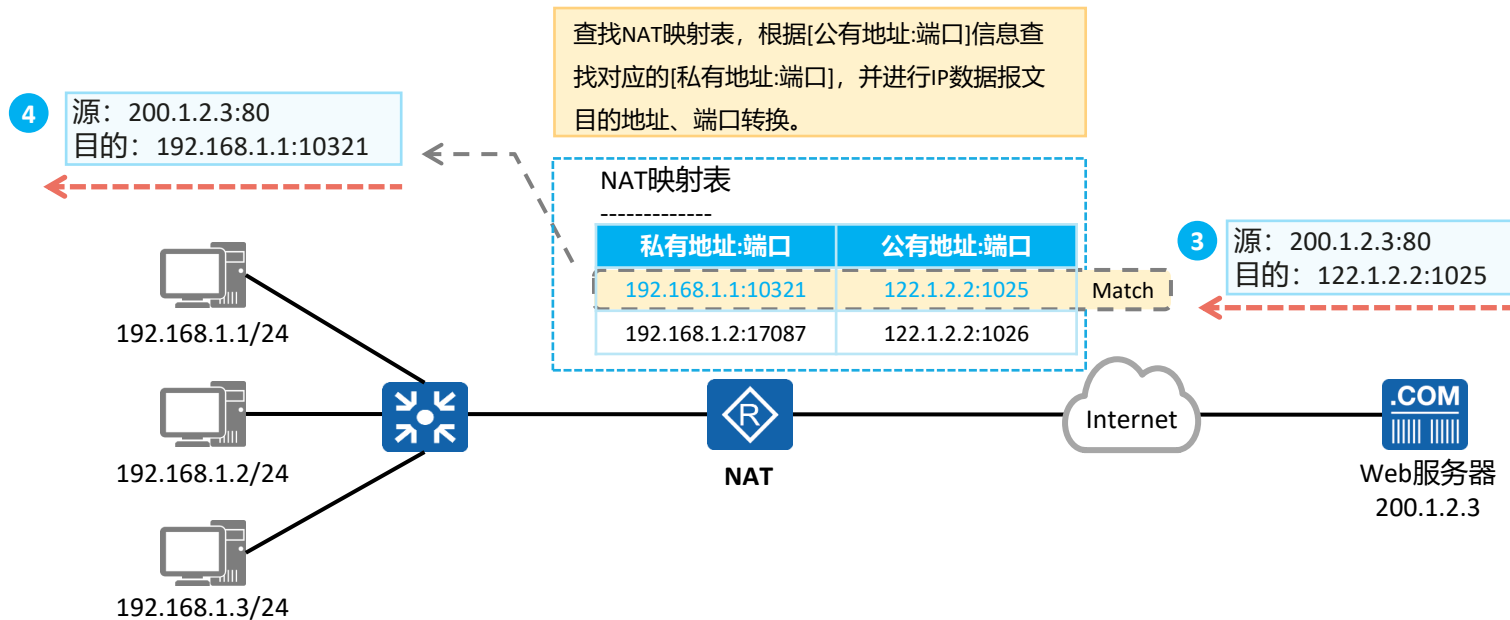


NAPT转换示例 (1)



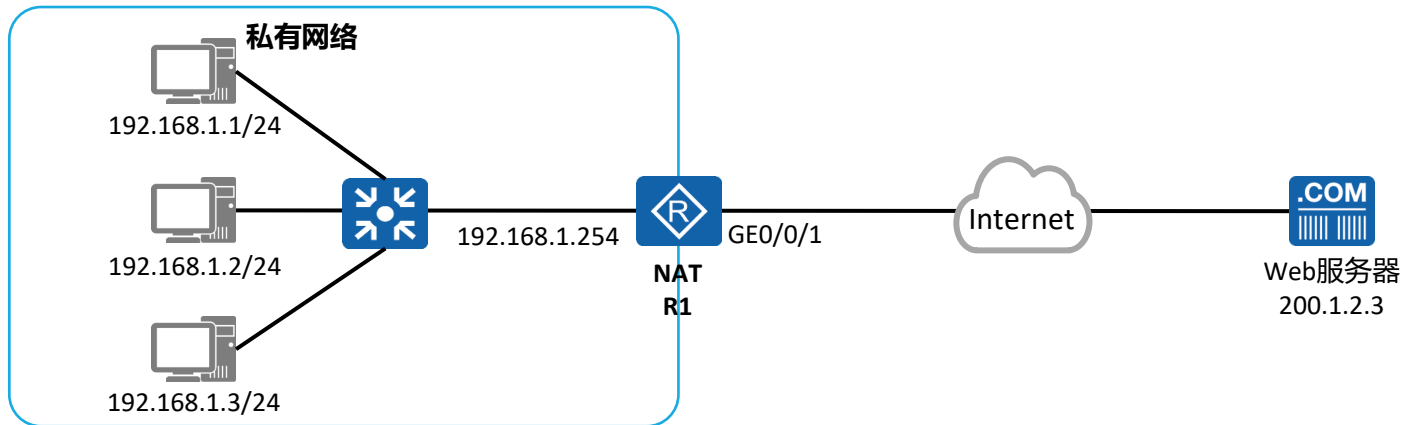


NAPT转换示例 (2)





NAPT配置示例



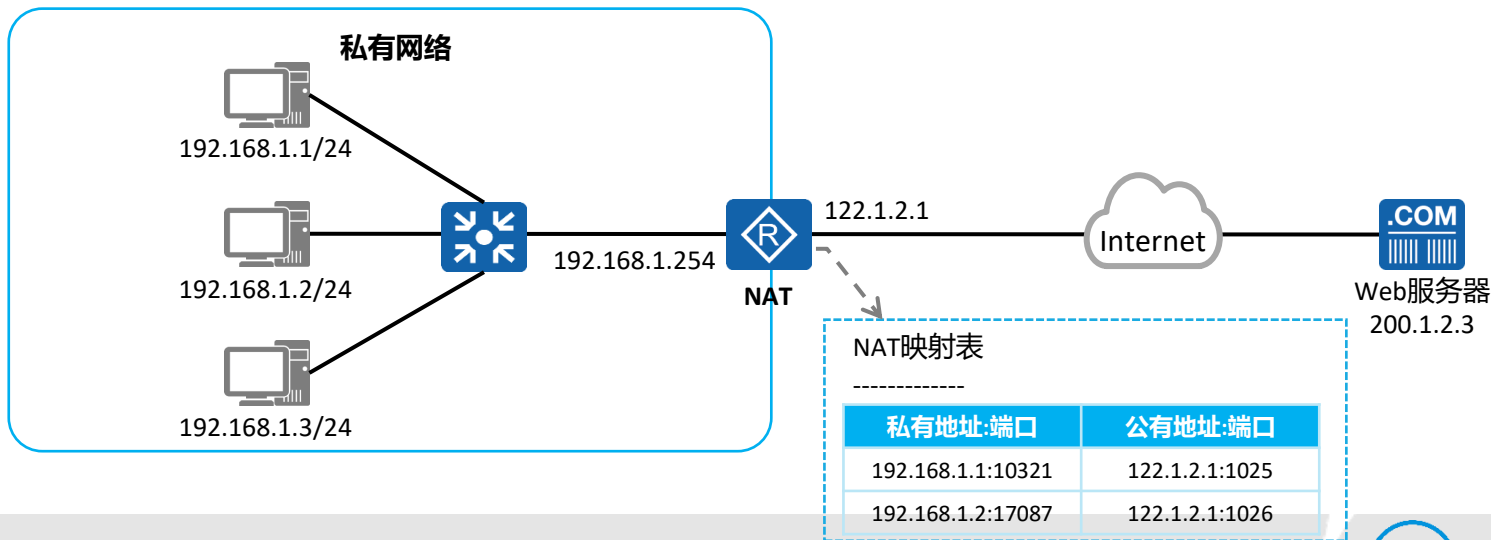
- 在R1上配置NAPT让内网所有私有地址通过122.1.2.1访问公网。

```
[R1]nat address-group 1 122.1.2.1 122.1.2.1
[R1]acl 2000
[R1-acl-basic-2000]rule 5 permit source 192.168.1.0 0.0.0.255
[R1-acl-basic-2000]quit
[R1]interface GigabitEthernet0/0/1
[R1-GigabitEthernet0/0/1]nat outbound 2000 address-group 1
```



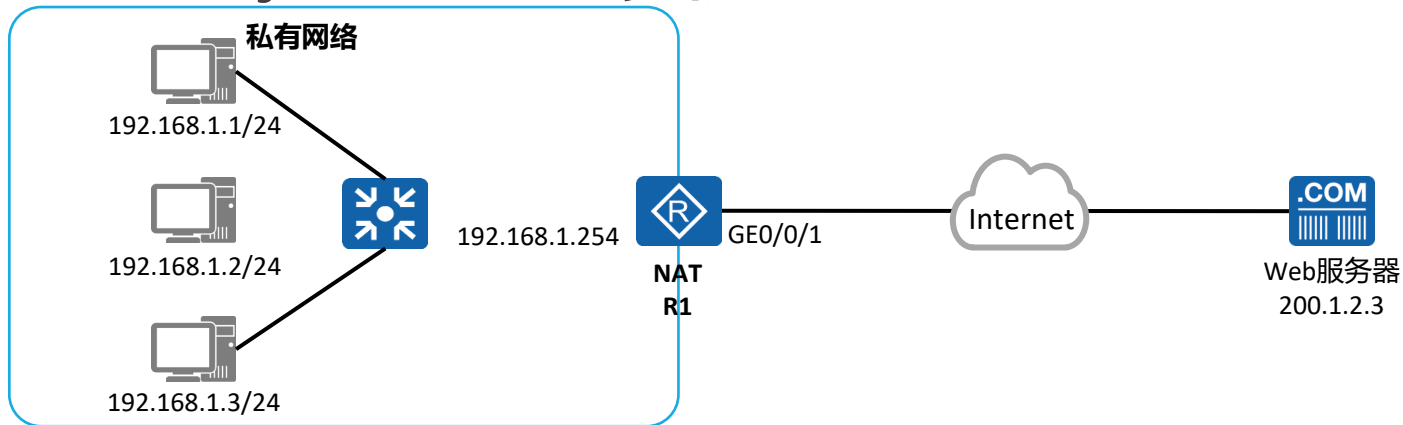
Easy IP

- **Easy IP:** 实现原理和NAPT相同，同时转换IP地址、传输层端口，区别在于Easy IP没有地址池的概念，使用接口地址作为NAT转换的公有地址。
- **Easy IP适用于不具备固定公网IP地址的场景：**如通过DHCP、PPPoE拨号获取地址的私有网络出口，可以直接使用获取到的动态地址进行转换。





Easy IP配置示例



- 在R1上配置Easy-IP让内网所有私有地址通过122.1.2.1访问公网。

```
[R1]acl 2000
[R1-acl-basic-2000]rule 5 permit source 192.168.1.0 0.0.0.255
[R1-acl-basic-2000]quit
[R1]interface GigabitEthernet0/0/1
[R1-GigabitEthernet0/0/1]nat outbound 2000
```



配置验证

```
[RTA]display nat outbound
```

```
NAT Outbound Information:
```

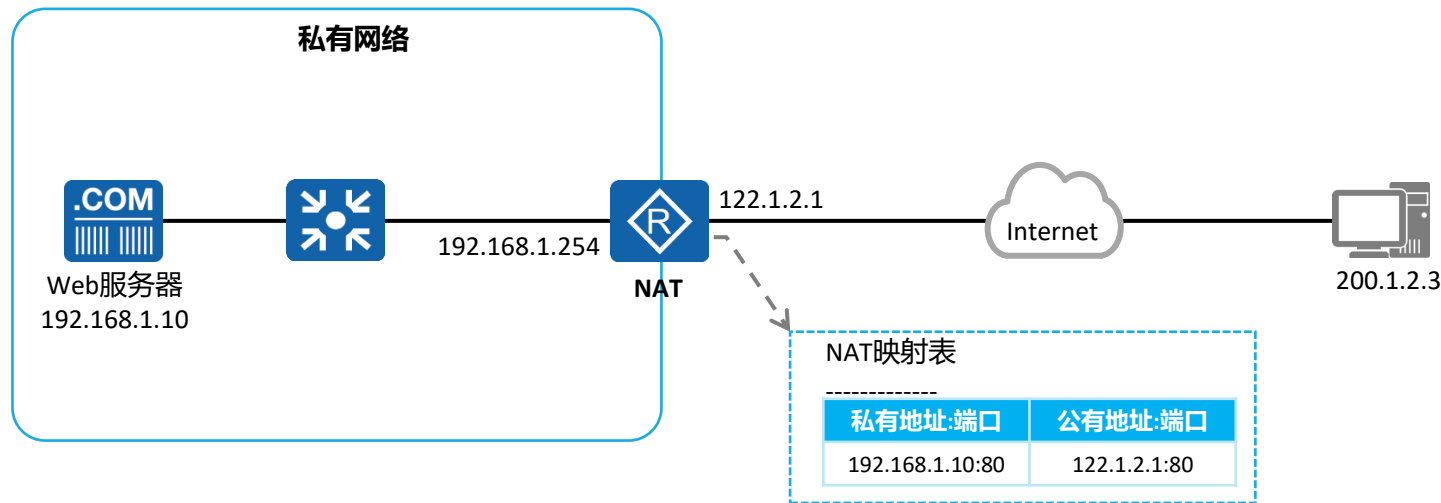
Interface	Acl	Address-group/IP/Interface	Type
Serial1/0/0	2000	200.10.10.1	easyip

```
Total : 1
```



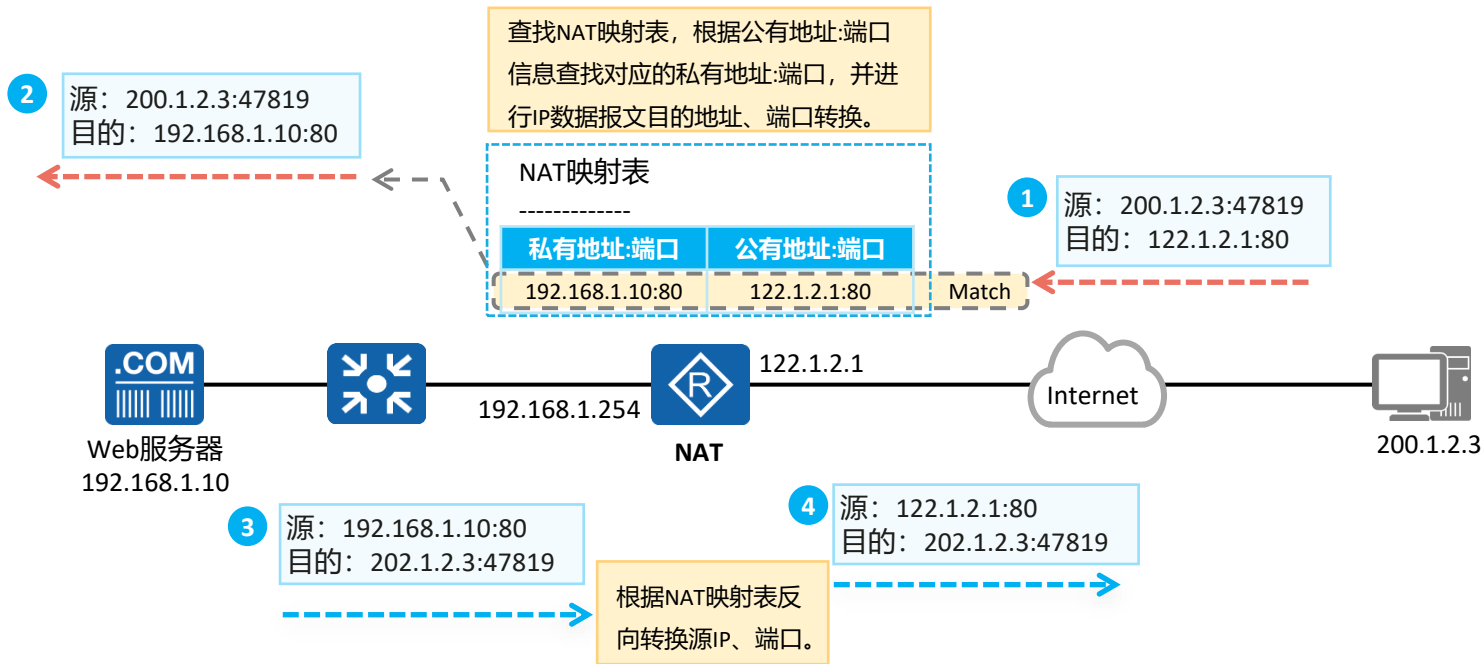

NAT Server使用场景

- **NAT Server:** 指定[公有地址:端口]与[私有地址:端口]的一对一映射关系, 将内网服务器映射到公网, 当私有网络中的服务器需要对公网提供服务时使用。
- 外网主机主动访问[公有地址:端口]实现对内网服务器的访问。



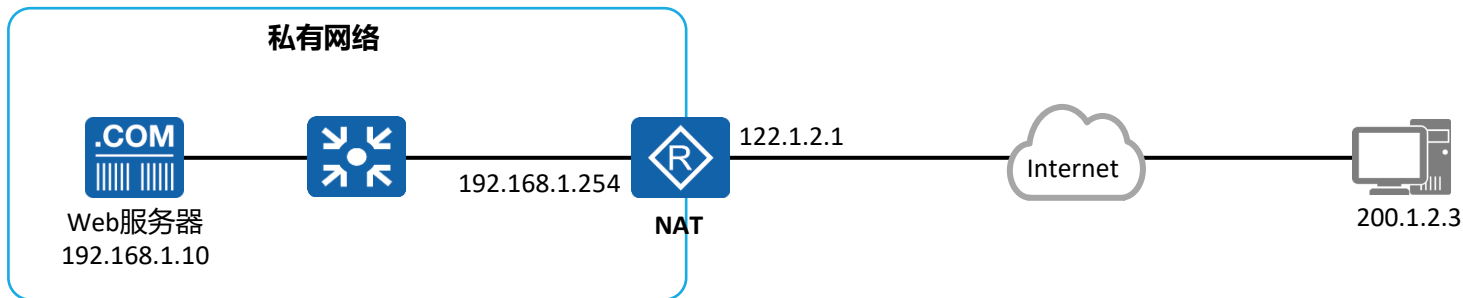


NAT Server转换示例





NAT Server配置示例



- 在R1上配置NAT Server将内网服务器192.168.1.10的80端口映射到公有地址122.1.2.1的8080端口。

```
[R1]interface GigabitEthernet0/0/1
```

```
[R1-GigabitEthernet0/0/1]ip address 122.1.2.1 24
```

```
[R1-GigabitEthernet0/0/1]nat server protocol tcp global 202.10.10.1 www inside 192.168.1.1 8080
```



配置验证

```
[RTA]display nat server
Nat Server Information:
Interface   : Serial1/0/0
Global IP/Port      : 202.10.10.1/80 (www)
Inside IP/Port      : 192.168.1.1/8080
Protocol   : 6(tcp)
VPN instance-name   : ----
Acl number          : ----
Description         : ----

Total :      1
```



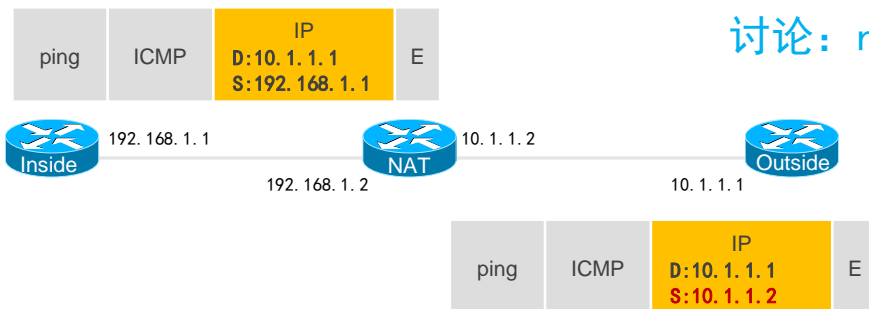
总结

- 在私有网络内使用私有地址，并在网络出口使用**NAT**技术，可以有效减少网络所需的**IPv4**公有地址数目，**NAT**技术有效地缓解了**IPv4**公有地址短缺的问题。
- 动态**NAT**、**NAPT**、**Easy IP**为私网主机访问公网提供源地址转换。
- **NAT Server**实现了内网主机对公网提供服务。
- 静态**NAT**提供了一对一映射，支持双向互访。



源网络地址转换S-NAT

1. 作用：员工上网，家庭上网
2. 功能：基于inside到outside的源地址转换
3. 注意：外部主动向内部发起流量，nat不做操作
4. 模式：多对一、多对多



讨论：nat的作用和应用场景？



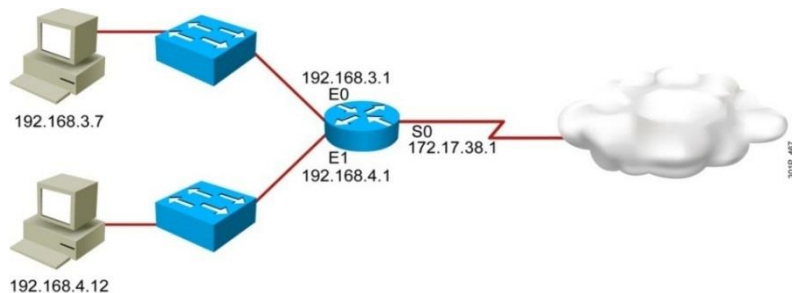
源网络地址转换S-NAT

S-NAT几种类型：

1. 动态多对一nat
2. 动态多对多nat
3. 静态nat
4. 静态PAT



S-NAT配置（动态多对一）



```
hostname RouterX
!
interface Ethernet0
 ip address 192.168.3.1 255.255.255.0
 ip nat inside
!
interface Ethernet1
 ip address 192.168.4.1 255.255.255.0
 ip nat inside
!
interface Serial0
 description To ISP
 ip address 172.17.38.1 255.255.255.0
 ip nat outside
!
ip nat inside source list 1 interface Serial0 overload
!
ip route 0.0.0.0 0.0.0.0 Serial0
!
access-list 1 permit 192.168.3.0 0.0.0.255
access-list 1 permit 192.168.4.0 0.0.0.255
```




S-NAT配置（动态多对多）

R3(config)#**ip nat in source list 1 pool Cisco ?**

mapping-id Associate a mapping id to this mapping

oer Use with vtemplate only. On new translation, if OER BR is UP, OER will select IP from outgoing Interface. All packets matching translation are forwarded over Interface for duration of translation.

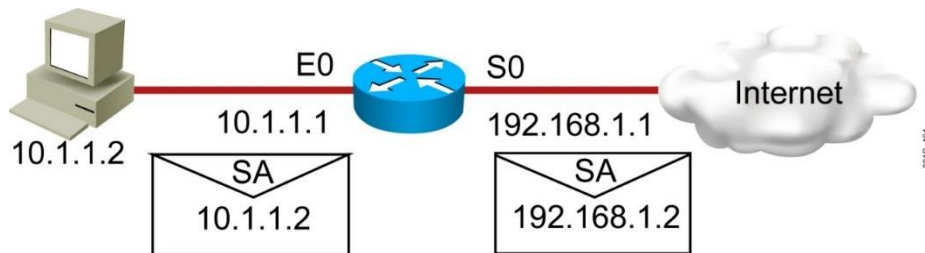
overload Overload an address translation

reversible Allow out->in traffic //相当于Ip nat inside destination 15.1被取消

vrf Specify vrf



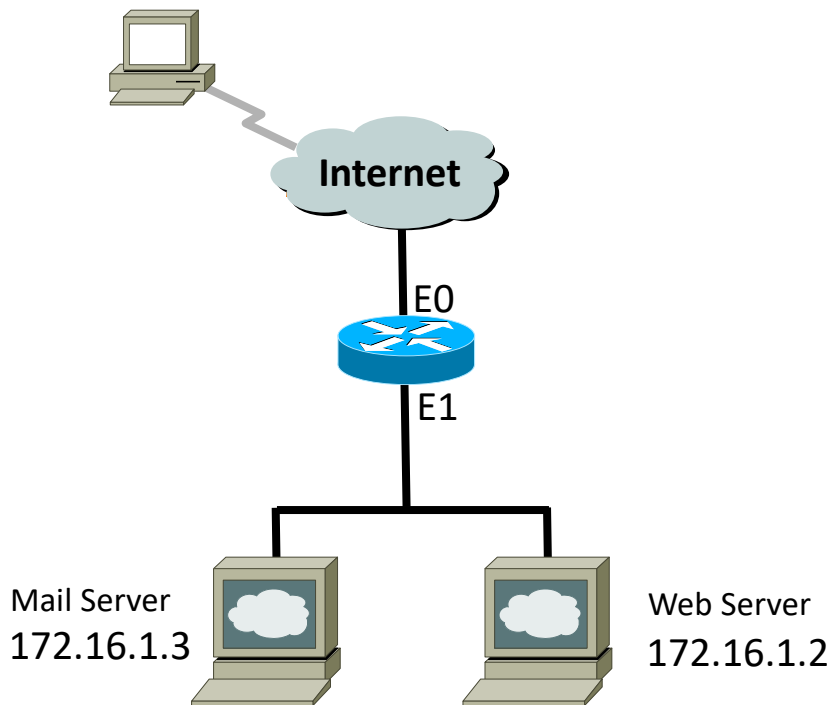
S-NAT配置（静态NAT）



```
interface s0
ip address 192.168.1.1 255.255.255.0
ip nat outside
!
interface e0
ip address 10.1.1.1 255.255.255.0
ip nat inside
!
ip nat inside source static 10.1.1.2 192.168.1.2
```



D-NAT



```
interface ethernet 0
```

```
ip address 200.1.1.1 255.255.255.252
```

```
ip nat outside
```

```
!
```

```
interface ethernet 1
```

```
ip address 172.16.1.1 255.255.255.0
```

```
ip nat inside
```

```
!
```

```
ip nat inside source static tcp 172.16.1.2 80 e0 80
```

```
ip nat inside source static tcp 172.16.1.3 25 e0 25
```

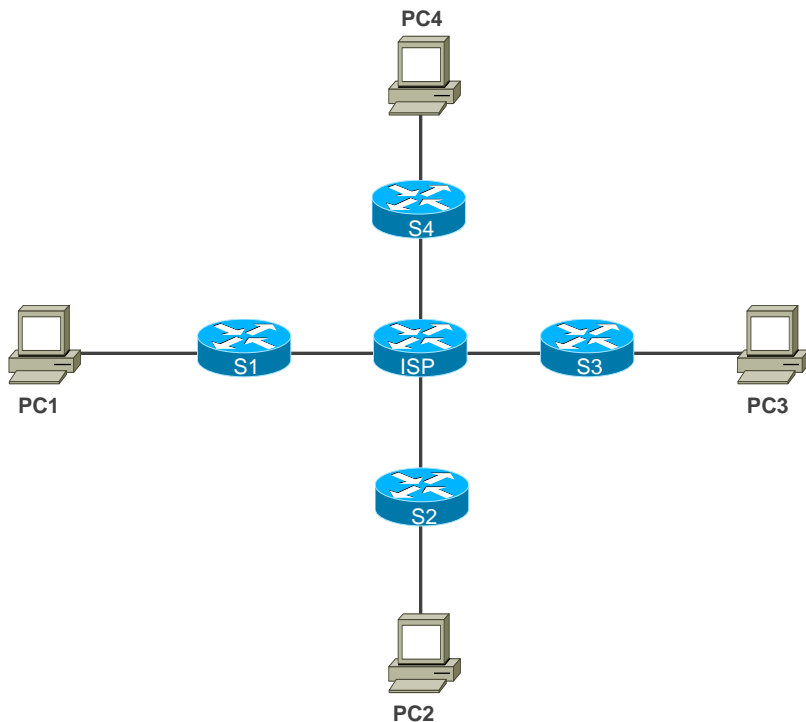


源网络地址转换S-NAT讨论

1. 多对一的情况，路由器如何区分不同的内部主机
2. 多对多的情况，例如100台主机，5个公网地址，nat如何转换？
3. NAT的优点和缺点？
4. 如何理解inside local、inside global、outside local、outside global？



课堂实验十二



实验目的：

1. S1做静态nat
2. S2做动态nat，申请到公网地址pool是202.100.2.0/29
3. S3做动态nat，但是只有一个公网地址
4. S4做静态PAT，内部23到外部2323的转换

结论：静态和多对多动态（除了最后一个）可以实现双向转换；多对一只能实现单向转换。



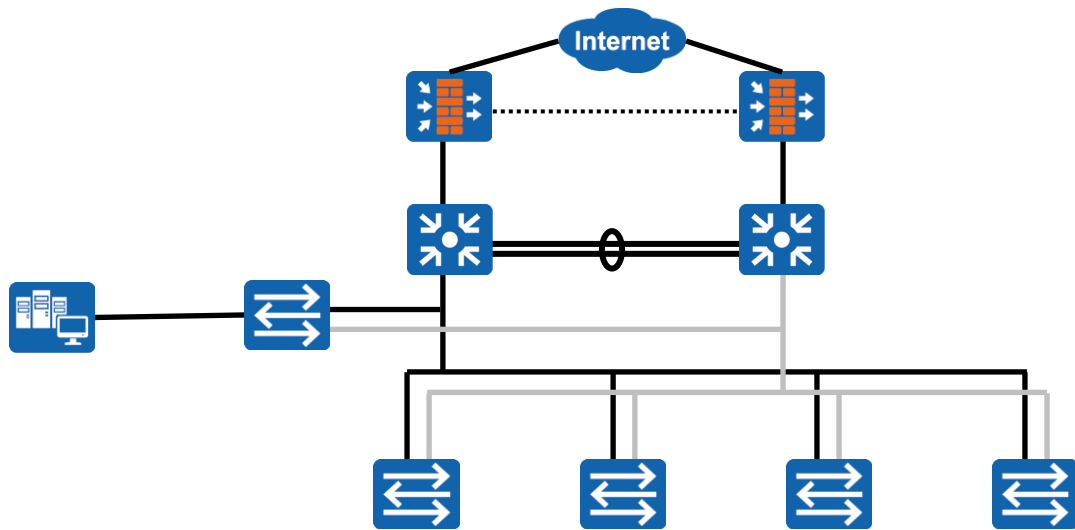
什么是网络管理？

- 网络管理是通过对网络中设备的管理，保证设备工作正常，使通信网络正常地运行，以提供高效、可靠和安全的通信服务，是通信网络的重要组成部分。

网络管理员管理和维护网络，保证网络的稳定运行。



网络管理员



常见企业网络架构



网络管理基本功能

配置
管理

性能
管理

故障
管理

安全
管理

计费
管理

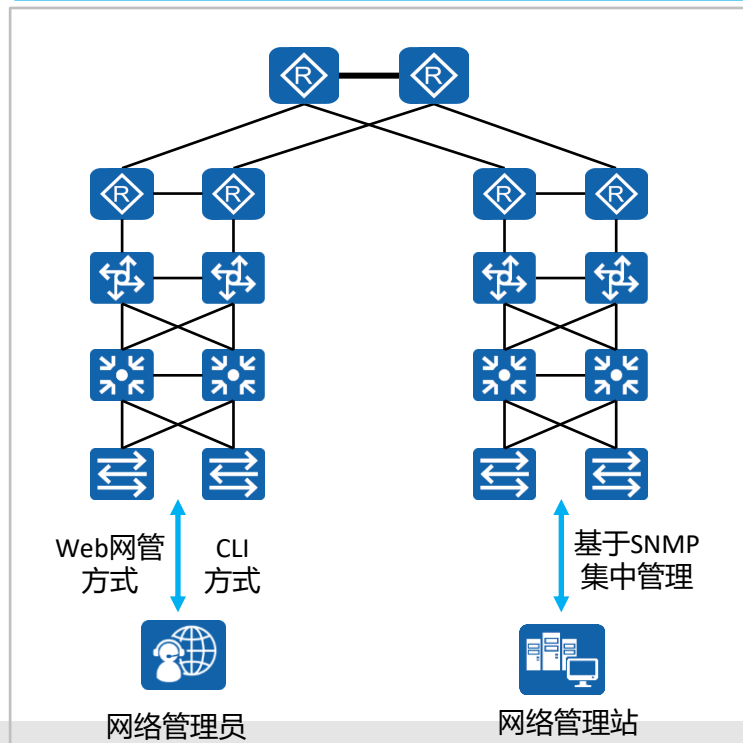
OSI定义了网络管理的五大功能模型：

- 配置管理（Configuration Management）：配置管理负责监控网络的配置信息，使网络管理人员可以生成、查询和修改硬件、软件的运行参数和条件，并可以进行相关业务的配置。
- 性能管理（Performance Management）：性能管理以网络性能为准则，保证在使用较少网络资源和具有较小时延的前提下，网络能够提供可靠、连续的通信能力。
- 故障管理（Fault Management）：故障管理的主要目标是确保网络始终可用，并在发生故障时尽快将其修复。
- 安全管理（Security Management）：安全管理可以保护网络和系统免受未经授权的访问和安全攻击。
- 计费管理（Accounting Management）：记录用户使用网络资源的情况并核收费用，同时也统计网络的利用率。

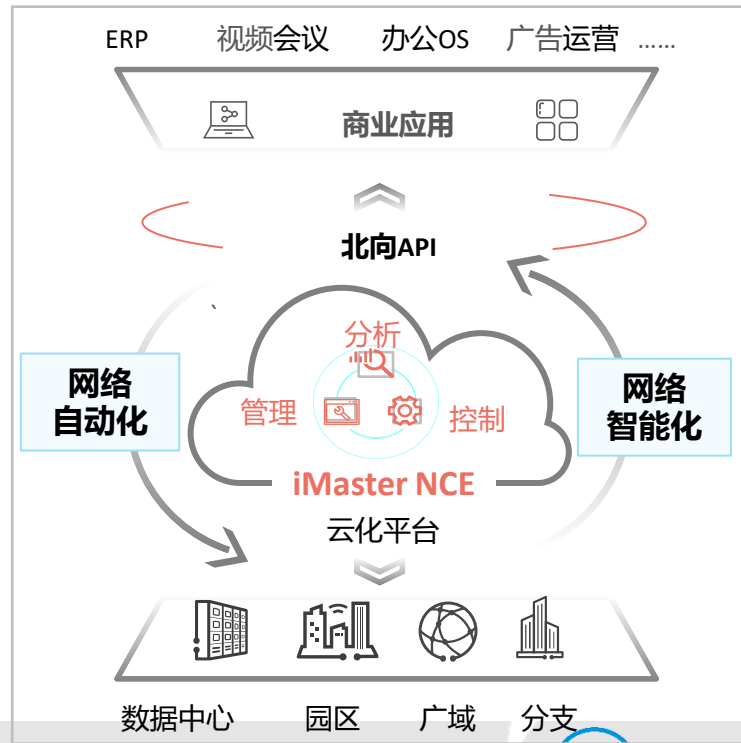


网络管理方式

传统网络管理



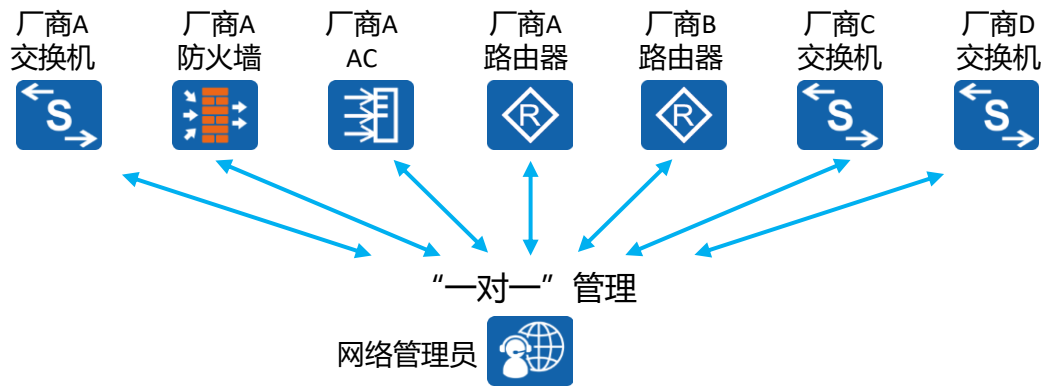
基于iMaster NCE的网络管理





通过CLI或Web进行管理

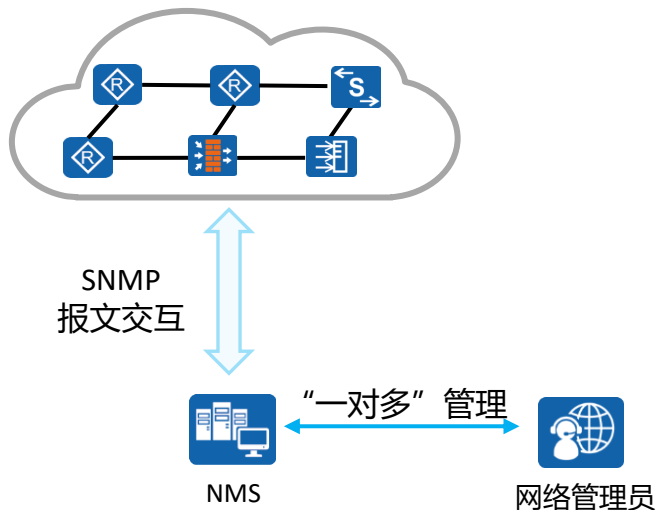
- 当网络规模较小时，CLI和Web方式是常见的网络管理方式。
 - 网络管理员可以通过HTTPS、Telnet、Console等方式登录设备后，对设备逐一进行管理。
 - 这种管理方式不需要在网络中安装任何程序或部署服务器，成本较低。
 - 网络管理员自身需要熟练掌握网络理论知识、各设备厂商网络配置命令。
 - 当网络规模较大，网络拓扑较为复杂时，这种方式的局限性较大。





基于SNMP的集中式管理

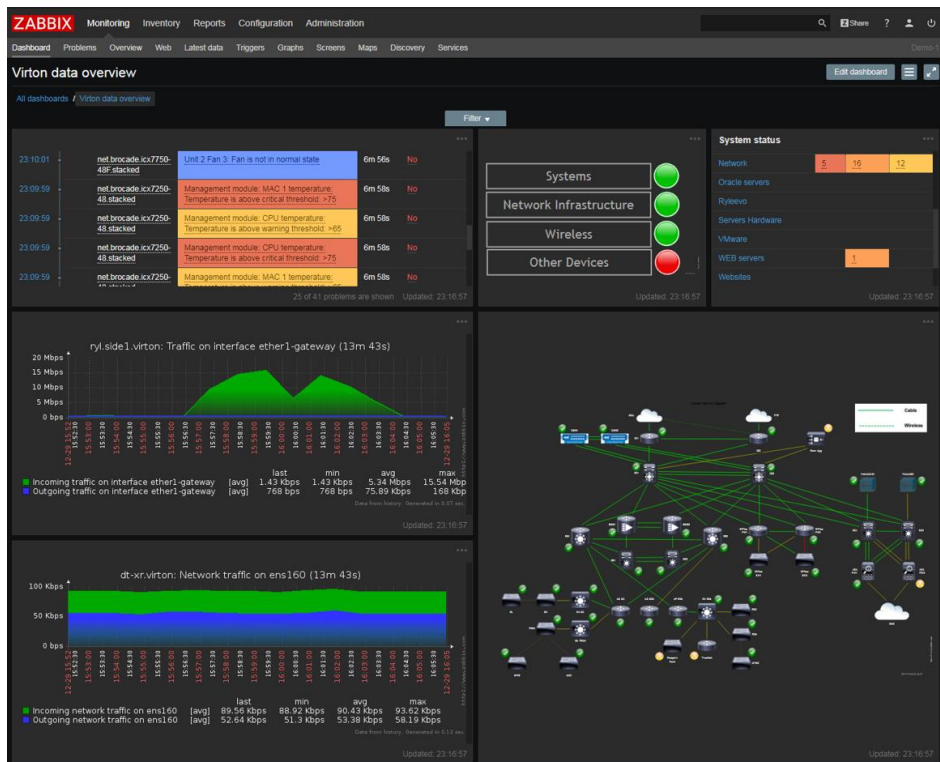
- SNMP (Simple Network Management Protocol, 简单网络管理协议) 是广泛用于TCP/IP网络的网络管理标准协议, 提供了一种通过运行网络管理软件的中心计算机, 即NMS (Network Management Station, 网络管理工作站) 来管理网元的方法。



- 网络管理员可以利用NMS在网络上的任意节点完成信息查询、信息修改和故障排查等工作, 提升工作效率。
- 屏蔽了不同产品之间的差异, 实现了不同种类和厂商的网络设备之间的统一管理。

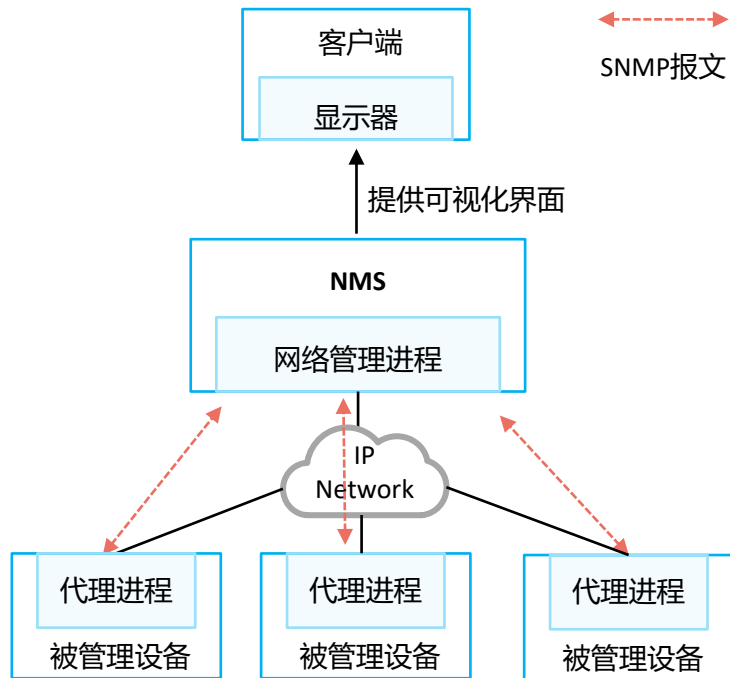


网管软件





SNMP典型架构




- 在基于SNMP进行管理的网络中，NMS是整个网络的网管中心，在它之上运行管理进程。每个被管理设备需要运行代理（Agent）进程。管理进程和代理进程利用SNMP报文进行通信。
- NMS是一个采用SNMP协议对网络设备进行管理/监控的系统，运行在NMS服务器上。
- 被管理设备是网络中接受NMS管理的设备。
- 代理进程运行于被管理设备上，用于维护被管理设备的信息数据并响应来自NMS的请求，把管理数据汇报给发送请求的NMS。



SNMP的信息交互

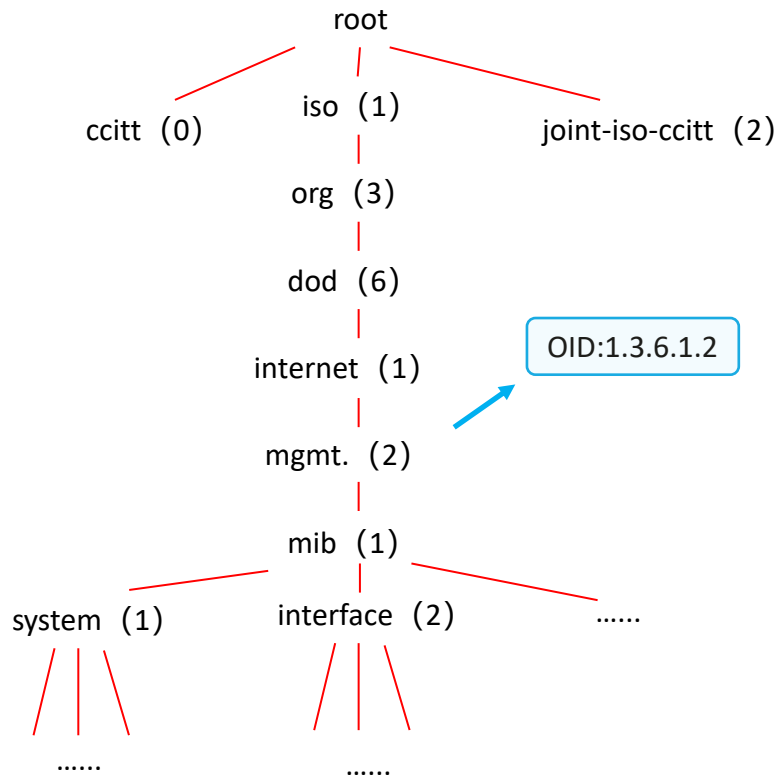


 被管理对象

- NMS和被管理设备的信息交互分为两种：
 - NMS通过SNMP协议给被管理设备发送修改配置信息请求或查询配置信息请求。被管理设备上运行的代理进程根据NMS的请求消息做出响应。
 - 被管理设备可以主动向NMS上报告警信息（Trap）以便网络管理员及时发现故障。
- 被管理对象（Managed object）：每一个设备可能包含多个被管理对象，被管理对象可以是设备中的某个硬件，也可以是在硬件、软件（如路由选择协议）上配置的参数集合。
- SNMP规定通过MIB（Management Information Base，管理信息库）去描述可管理实体的一组对象。



MIB



- MIB是一个数据库，指明了被管理设备所维护的变量（即能够被代理进程查询和设置的信息）。MIB在数据库中定义了被管理设备的一系列属性：
 - 对象标识符（Object Identifier, OID）
 - 对象的状态
 - 对象的访问权限
 - 对象的数据类型等
- MIB给出了一个数据结构，包含了网络中所有可能的被管理对象的集合。因为数据结构与树相似，MIB又被称为对象命名树。



常见MIB节点

- 用于查询或修改的节点：

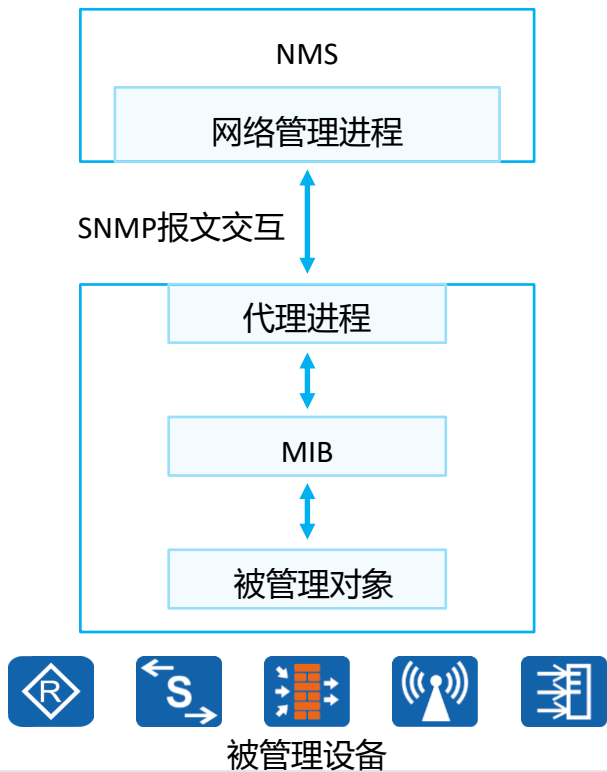
OID	节点名称	数据类型	最大访问权限	含义
1.3.6.1.2.1.2.1	ifNumber	Integer	read-only	系统中网络接口的数量（不关注接口当前状态）。
1.3.6.1.4.1.2011.5.25.41.1.2.1.1.2	ipAdEntIfIndex	IpAddress	read-create	IP地址的子网掩码。

- 用于告警通知的节点：

OID	节点名称	绑定变量	含义
3.6.1.6.3.1.1.5.3	linkDown	ifIndex ifAdminStatus ifOperStatus ifDesc	经检测到由于ifOperStatus节点中的其中一条通信链路已经从其他状态（但不是notPresent状态）进入Down状态。这里的其他状态由ifOperStatus的值显示。



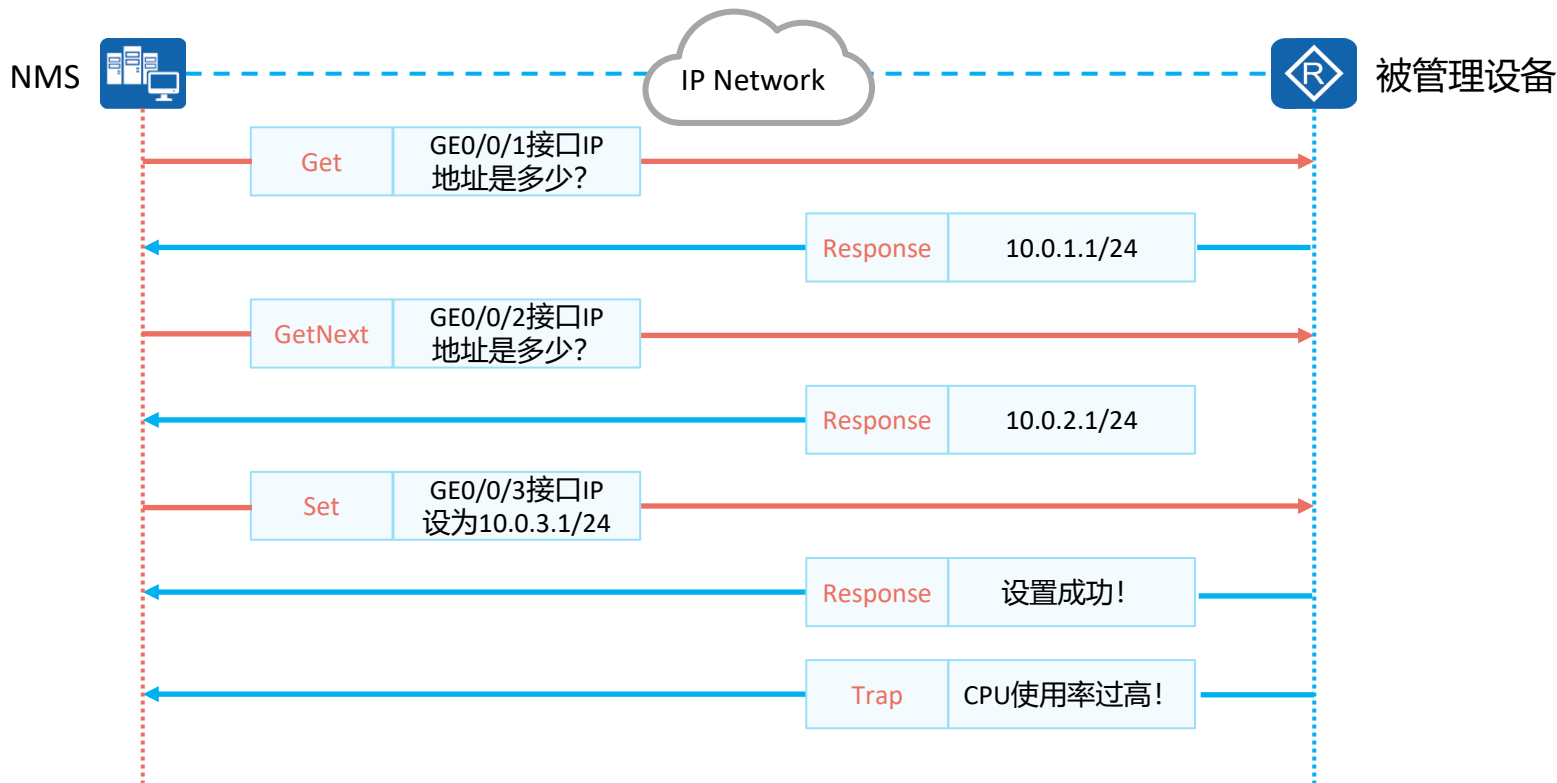
SNMP管理模型



- 查询/修改操作：
 - NMS作为管理者，向代理进程发送SNMP请求报文。
 - 代理进程通过设备端的MIB找到所要查询或修改的信息，向NMS发送SNMP响应报文。
- 告警操作：
 - 设备端的模块由于达到模块定义的告警触发条件，通过代理进程向NMS发送消息，告知设备侧出现的情况，这样便于网络管理人员及时对网络中出现的情况进行处理。

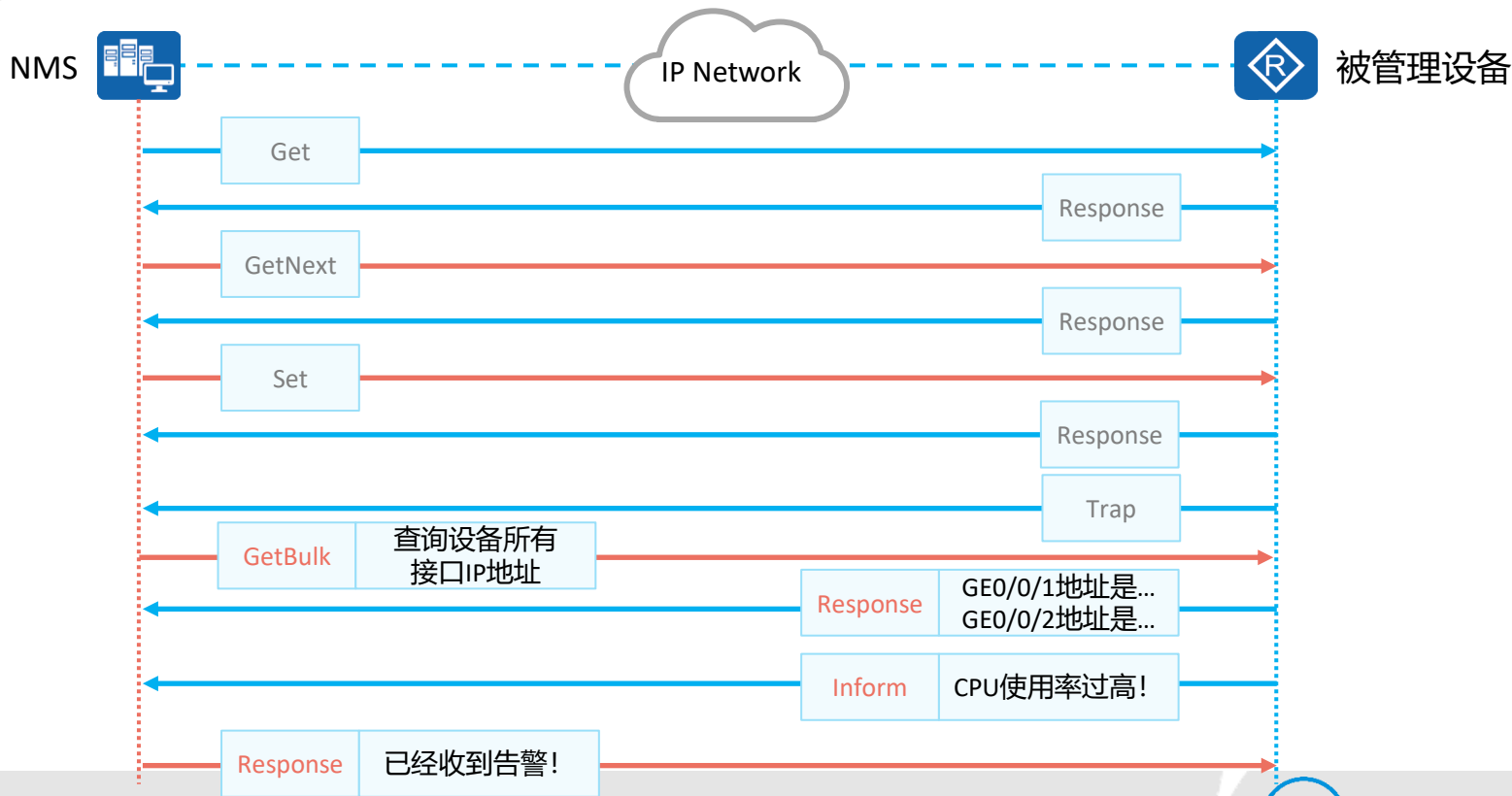


SNMPv1





SNMPv2c





SNMPv3

- SNMPv3与SNMPv1和SNMPv2c的工作机制基本一致但添加了报头数据和安全参数。
- SNMPv3报文具有身份验证和加密处理的功能。
- SNMPv3适用于各种规模的网络，安全性极高。





SNMP小结

- SNMP的特点如下：
 - 简单：SNMP采用轮询机制，提供基本的功能集，适合快速、低价格的场景使用，而且SNMP以UDP报文为承载，因而得到绝大多数设备的支持。
 - 强大：SNMP的目标是保证管理信息在任意两点传送，便于管理员在网络上的任何节点检索信息，进行故障排查。
- SNMPv1版本适用于小型网络。组网简单、安全性要求不高或网络环境比较安全且比较稳定的网络，比如校园网，小型企业网。
- SNMPv2c版本适用于大中型网络。安全性要求不高或者网络环境比较安全，但业务比较繁忙，有可能发生流量拥塞的网络。
- SNMPv3版本作为推荐版本，适用于各种规模的网络。尤其是对安全性要求较高，只有合法的管理员才能对网络设备进行管理的网络。



SNMP基本配置 (1)

1. 使能SNMP代理功能

```
[Huawei] snmp-agent
```

2. 配置SNMP的版本

```
[Huawei] snmp-agent sys-info version [v1 | v2c | v3]
```

用户可以根据自己的需求配置对应的SNMP版本，但设备侧使用的协议版本必须与网管侧一致。

3. 创建或者更新MIB视图的信息

```
[Huawei] snmp-agent mib-view view-name { exclude | include } subtree-name [mask mask]
```

4. 增加一个新的SNMP组，将该组用户映射到SNMP视图

```
[Huawei] snmp-agent group v3 group-name { authentication | noauth | privacy } [ read-view view-name | write-view view-name | notify-view view-name ]
```

该命令用于SNMPv3版本中创建SNMP组，指定认证加密方式、只读视图、读写视图、通知视图。是安全性需求较高的网管网络中的必需指令。



SNMP基本配置 (2)

5. 为一个SNMP组添加一个新用户

```
[Huawei] snmp-agent usm-user v3 user-name group group-name
```

6. 配置SNMPv3用户认证密码

```
[Huawei] snmp-agent usm-user v3 user-name authentication-mode { md5 | sha | sha2-256 }
```

7. 配置SNMPv3用户加密密码

```
[Huawei] snmp-agent usm-user v3 user-name privacy-mode { aes128 | des56 }
```

8. 配置设备发送Trap报文的参数信息

```
[Huawei] snmp-agent target-host trap-paramsname paramsname v3 securityname securityname { authentication | noauthnopriv | privacy }
```



SNMP基本配置 (3)

9. 配置Trap报文的目的主机

```
[Huawei] snmp-agent target-host trap-hostname hostname address ipv4-address trap-paramsname paramsname [ notify-filter-profile profile-name ]
```

10. 打开设备的所有告警开关

```
[Huawei] snmp-agent trap enable
```

注意该命令只是打开设备发送Trap告警的功能，要与snmp-agent target-host协同使用，由snmp-agent target-host指定Trap告警发送给哪台设备。

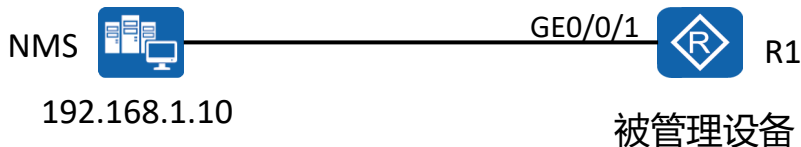
11. 配置发送告警的源接口。

```
[Huawei] snmp-agent trap source interface-type interface-number
```

注意Trap告警无论从那个接口发出都必须有一个发送的源地址，因此源接口必须是已经配置了IP地址的接口。



SNMP配置举例（网络设备侧）



- 上述路由器R1上使能SNMP功能，配置版本为v3。
- 配置SNMPv3组名为test，加密认证方式为privacy。
- 创建SNMPv3用户，名为R1同时配置认证和加密密码为HCIA-Datacom123。
- 创建名为param的Trap参数信息，securityname为sec
- 设置SNMP告警主机地址为192.168.1.10。
- 打开告警开关，设置发送告警的源接口为GE0/0/1。

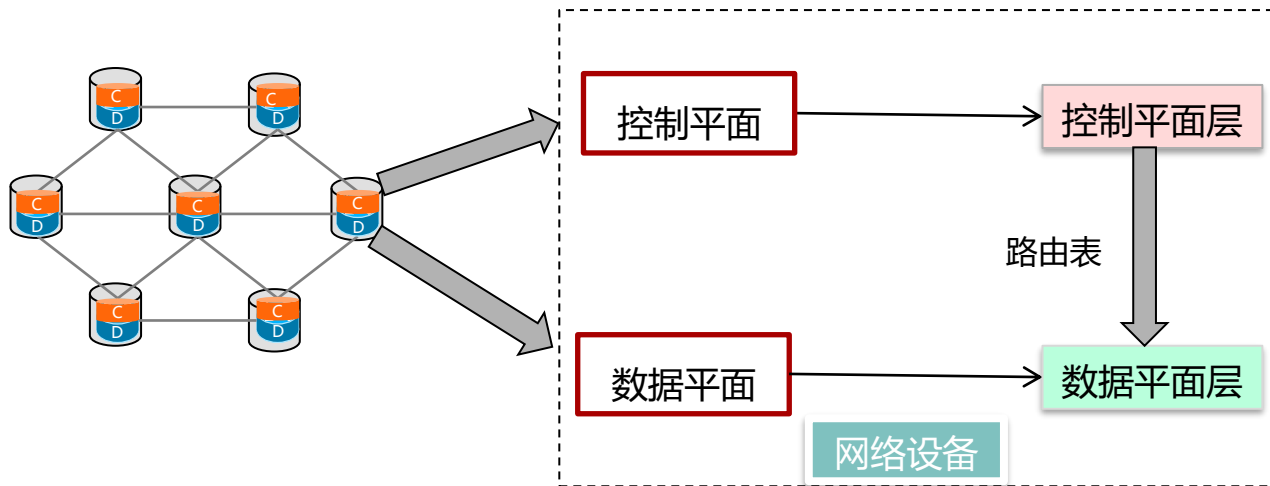
R1配置如下：

```
[R1]snmp-agent
[R1]snmp-agent sys-info version v3
[R1]snmp-agent group v3 test privacy
[R1]snmp-agent usm-user v3 R1 test authentication-mode md5
HCIA@Datacom123 privacy-mode aes128 HCIA-Datacom123
[R1]snmp-agent target-host trap-paramsname param v3
securityname sec privacy
[R1]snmp-agent target-host trap-hostname nms address
192.168.1.10 trap-paramsname param
[R1]snmp-agent trap source GigabitEthernet 0/0/1
[R1]snmp-agent trap enable
Info: All switches of SNMP trap/notification will be open.
Continue? [Y/N]:y
```




传统网络数据控制与转发

- 传统网络是分布式控制的架构，每台设备都包含独立的控制平面、数据平面。

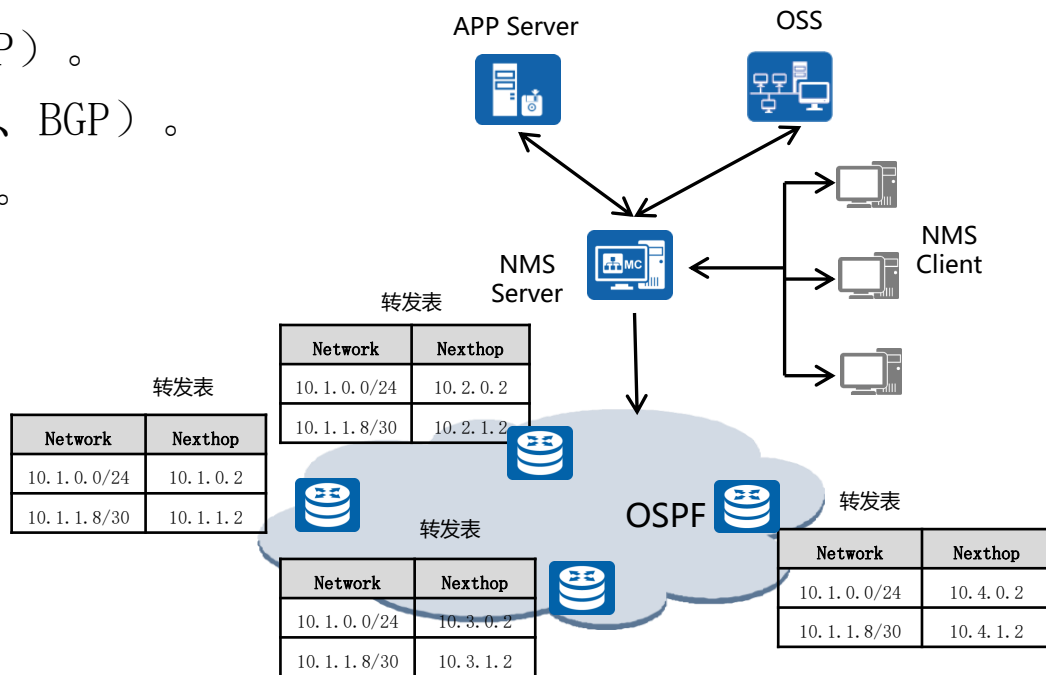




传统网络结构体系

- 传统网络的管理平面、控制平面、数据平面：

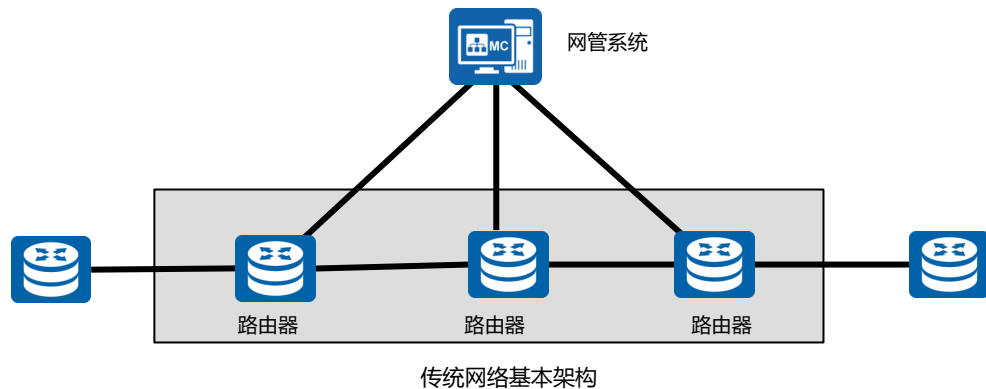
- 管理平面：设备管理（SNMP）。
- 控制平面：路由协议（IGP、BGP）。
- 数据平面：转发表（FIB）。





传统网络局限性

- 传统网络的局限性：
 - 流量路径的灵活调整能力不足。
 - 网络协议实现复杂，运维难度较大。
 - 网络新业务升级速度较慢。





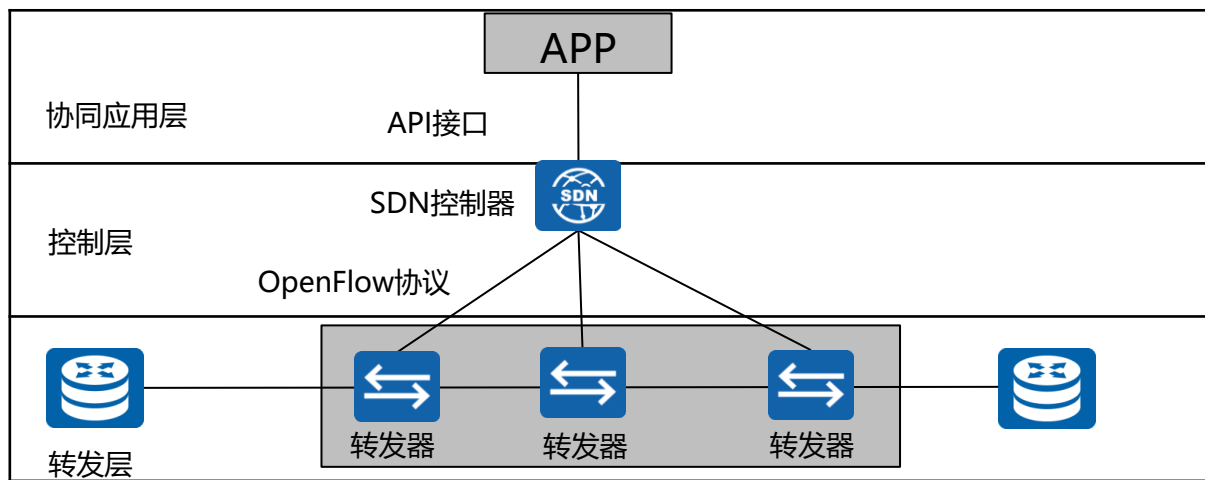
SDN概述

- SDN（Soft ware Defined Network）——软件定义网络。
 - 2006年，以斯坦福大学教授Nike McKeown为首的团队提出了OpenFlow的概念，并基于OpenFlow技术实现网络的可编程能力，使网络像软件一样灵活编程，SDN技术应运而生。
 - SDN的三个主要特征：
 - ✓ 转控分离。
 - ✓ 集中控制。
 - ✓ 开放接口。
- SDN控制器既不是网管，也不是规划工具。
 - 网管没有实现转控分离。
 - 规划工具的目的和控制器不同。



SDN网络体系架构

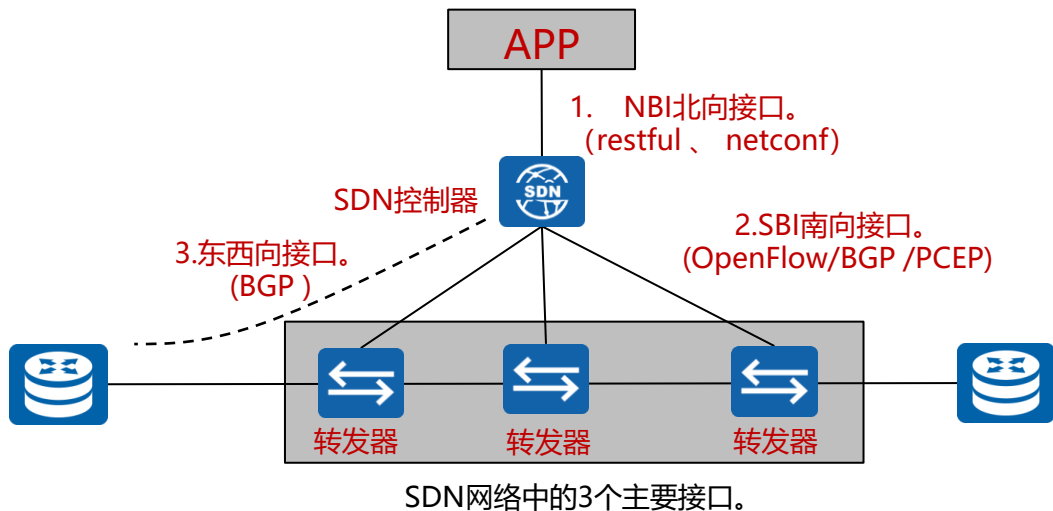
- SDN是对传统网络架构的一次重构，由原来的分布式控制的网络架构重构为集中控制的网络架构。
- SDN网络体系架构的三层模型：





SDN架构下的接口

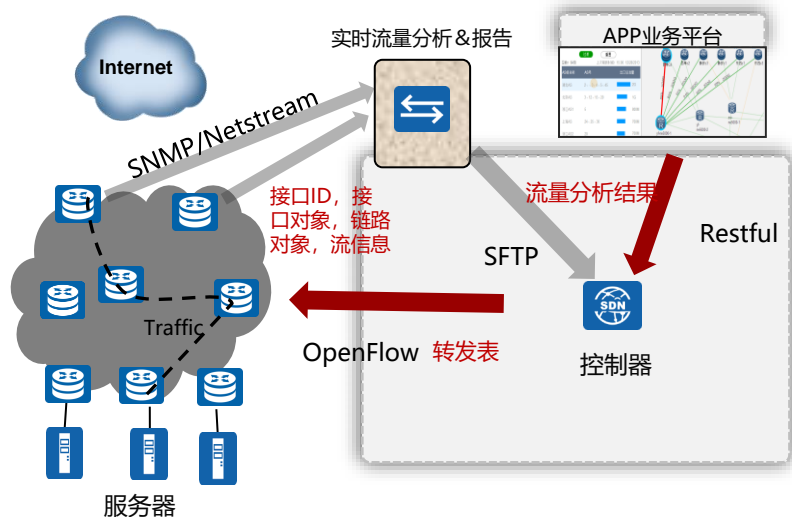
- NBI (North Bound Interface) 北向接口。
- SBI (South Bound Interface) 南向接口。





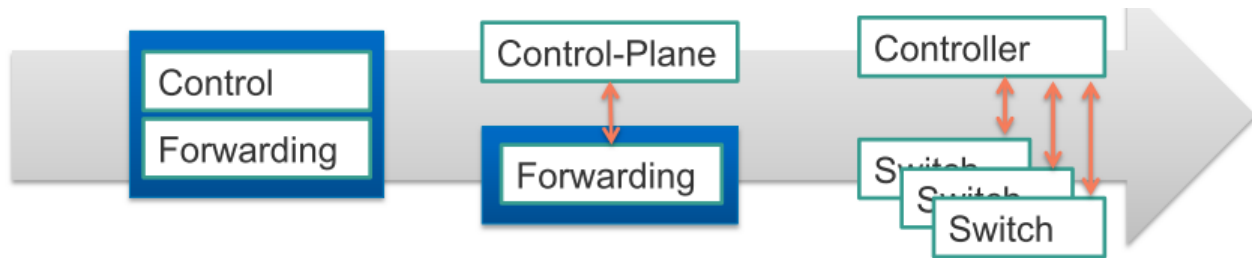
SDN基本工作原理

- 网元资源信息收集。
 - 转发器注册信息。
 - 上报资源过程。
 - MPLS标签信息。
 - VLAN资源信息。
 - 接口资源信息等。
- 拓扑信息搜集。
 - 节点对象、接口对象、链路对象（LLDP/IGP/BGP-LS）。
- SDN网络内部交换路由的生成。





OpenFlow的思想和功能



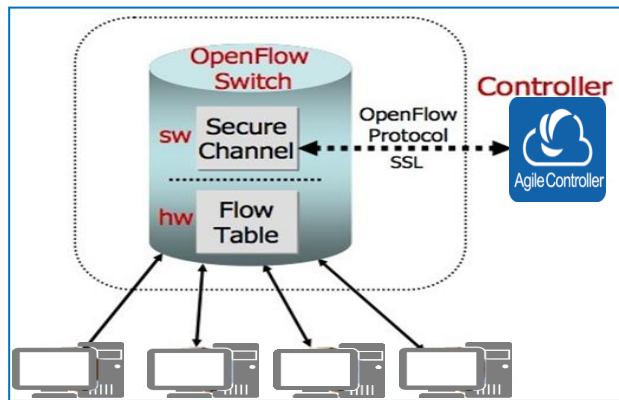
- 两个角色:

- OpenFlow Controller: 用于控制OpenFlow Switch, 计算路径, 维护状态和将流规则下发给交换机。
- OpenFlow Switch: 从OpenFlowController控制器接收命令或者流信息, 以及返回状态信息。
- OpenFlowSwitch基于流表并根据流规则进行转发、处理数据。



OpenFlow网络交换模型

- 该模型的指导思想是：底层的数据通信（交换机、路由器）是“简化的”，并定义一个对外开放的关于流表FlowTable的公用API（应用程序接口），同时采用控制器来控制整个网络。

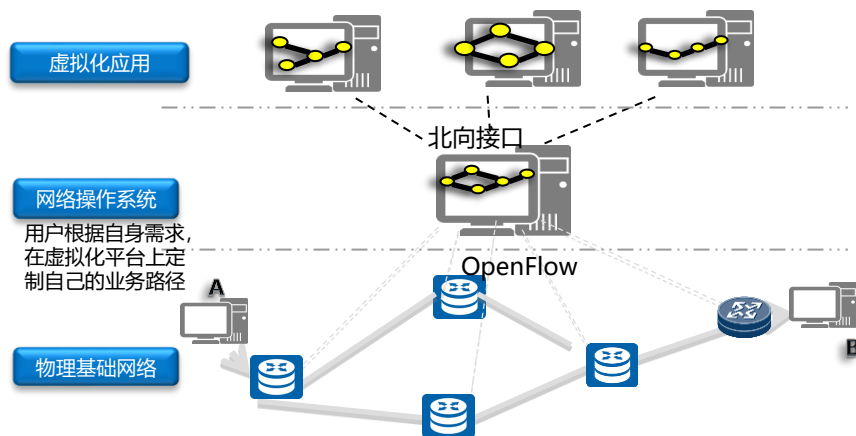


OpenFlow整体结构



网络业务快速创新

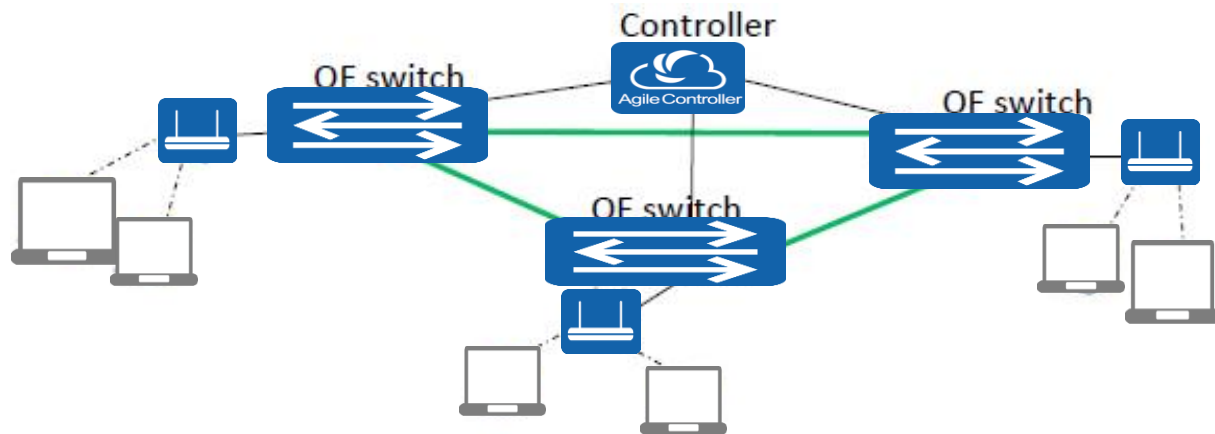
- SDN的可编程性和开放性，使得我们可以快速开发新的网络业务和加速业务创新。如果希望在网上部署新业务，可以通过针对SDN软件的修改实现网络快速编程，业务快速上线。





简化网络

- SDN的网络架构简化了网络，消除了很多IETF的协议。协议的去除，意味着学习成本的下降，运行维护成本下降，业务部署速度提升。这个价值主要得益于SDN网络架构下的网络集中控制和转控分离。





网络设备白牌化

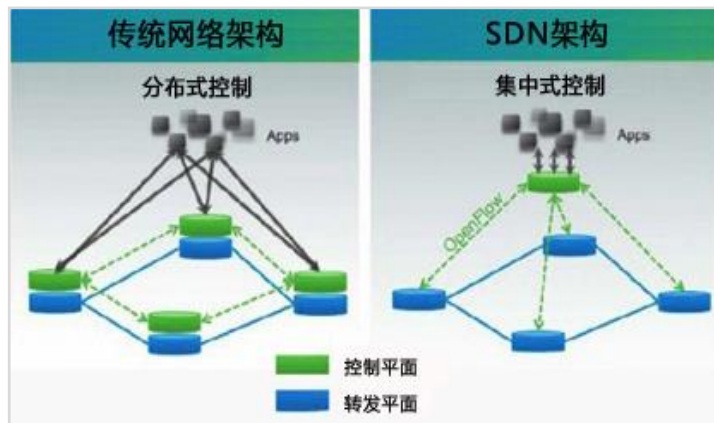
- 基于SDN架构，如果标准化了控制器和转发器之间的接口，比如Openflow协议逐渐成熟，那么网络设备的白牌化将成为可能，比如专门的Openflow转发芯片供应商，控制器厂商等，这也正是所谓的系统从垂直集成开发走向水平集成。

SDN 产业链		
类别	厂商	现状
芯片商	盛科、博通	盛科已经推出支持 OpenFlow 的交换机，并广泛应用于国内科研机构。博通推出 SDN 芯片解决方案
网络设备制造商	思科、华为、爱立信、阿朗	思科开放部分软件，推出针对性 SDN 产品；华为在硬件设备中增加对 OpenFlow 支持
IT 供应商	IBM、HP	推出控制器，支持 OpenFlow
创新公司	Nicira、Big Switch	Nicira 走在最前列，基于 VSwitch 的网络虚拟平台已服务于 AT&T、eBay、Fidelity、RackSpace 等公司



业务自动化

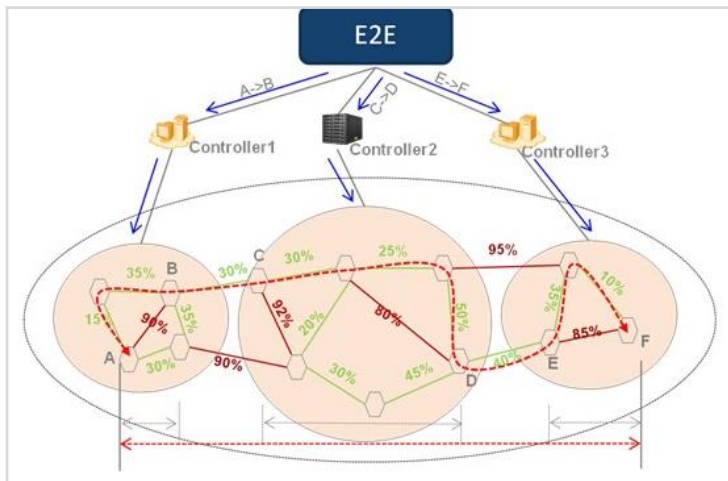
- SDN网络架构下，由于整个网络归属控制器控制，那么网络业务自动化就是理所当然的，不需要另外的系统进行配置分解。在SDN网络架构下，SDN控制器自己可以完成网络业务的部署，提供各种网络服务比如L2VPN、L3VPN等，屏蔽网络内部细节，提供网络业务自动化能力。





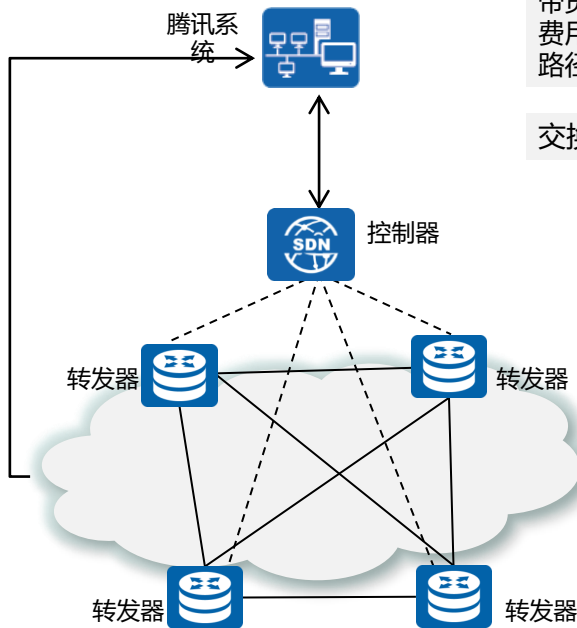
网络路径流量优化

- 通常传统网络的路径选择依据是通过路由协议计算出的“最优”路径，但结果可能会导致“最优”路径上流量拥塞，其它非“最优”路径空闲。当采用SDN网络架构时，SDN 控制器可以根据网络流量状态智能调整流量路径，提升网络利用率。





传统网络向SDN演进方式一：仅交换网SDN化

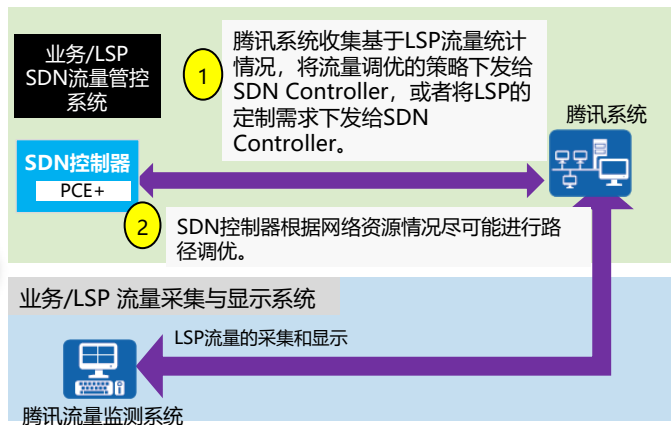


边界接入业务仍然是分布式接入到转发平面。

DC出口租用多条光纤，在DCI流量管理上存在以下问题：

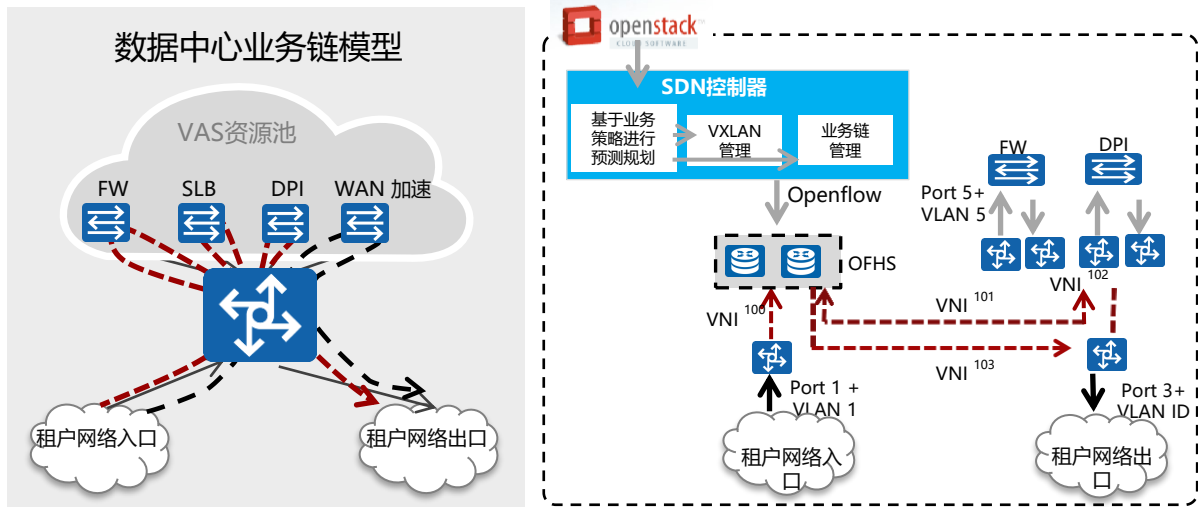
带宽利用率不高，但局部链路偶尔会拥塞（网络平均带宽利用率30%，每年的费用20多亿），腾讯的目标是：把带宽利用率提升到50%；目前网络是分布式路径计算，并产生了流量绕行等问题。

交换网SDN化，是指把域内交换网的路径计算功能进行集中控制。





方式二：仅业务SDN化



- 此方案仅仅将自治域AS所接入的业务由控制器接管，域内路径计算和控制依然由转发器负责。
- 统一部署增值业务**VAS资源池**，通过SDN Controller业务链解决方案，集中控制管理，同时实现**VAS资源共享**。
- 提升增值业务**快速创新**能力，提供新的创收来源。

THANK YOU

Ping 通您的梦想 ~

腾讯课堂交流群：17942636

ADD：苏州市干将东路666号和基广场401-402；Tel：0512-8188 8288；

课程咨询QQ：2853771087；官网：www.51glab.com