IPv6 学习笔记

基础版

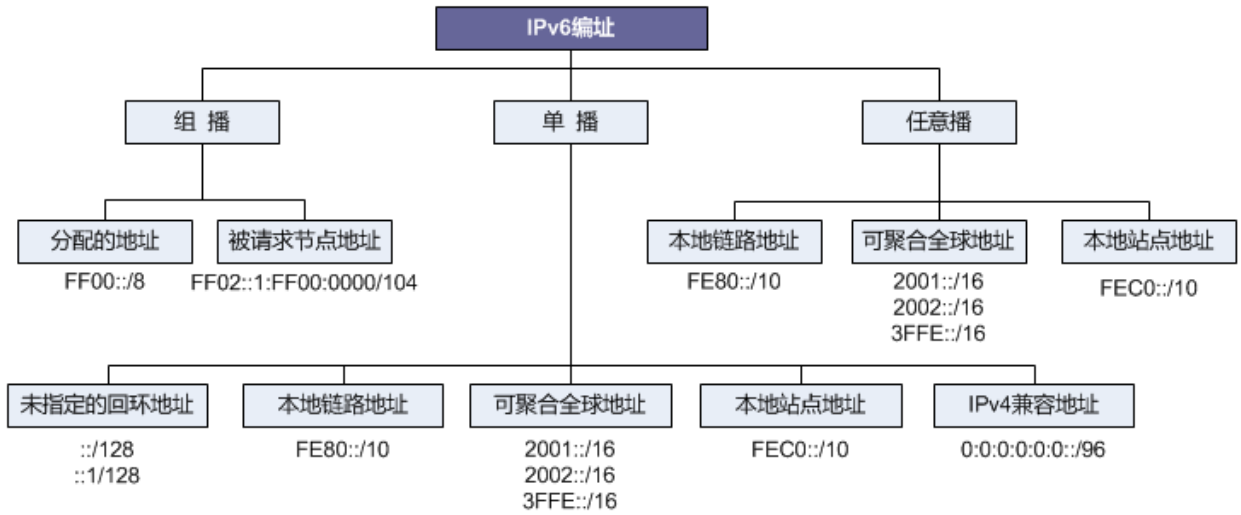
http://t.sina.com/vinsoney

1 Ipv6 编址

1.1 基础

1. 地址在简写的时候多个**前导** 0 可以省略成一个 0
   一个或多个连续的 16 比特字段为 0 时，可用 : : 表示，但只允许一个 : :
2. Ipv6 地址的 URL 表示： http:// **[ipv6 地址]** : 8080 //必须用 [ ] 括起来

1.2 地址分类



1.2.1 单播地址

Prefix	Hex	Size	Allocation
0000 0000	0000 – 00FF	1/256	Reserved
0000 0001	0100 – 01FF	1/256	Unassigned
0000 0010	0200 – 03FF	1/128	NSAP
0000 0100	0400 – 05FF	1/128	IPX -> moving to Unassigned
0000 0110	0600 – 07FF	1/128	Unassigned
0000 1000	0800 – 0FFF	1/32	Unassigned
0001 0000	1000 – 1FFF	1/16	Unassigned
0010 0000	2000 – 3FFF	1/8	**Aggregatable: IANA to registers**

1. **可聚合全局单播地址 (Aggregatable global unicast address)**
   目前已经分配的全球单播地址前缀都是以 001 开头的，(2000 : : 到 3FFF:3FFF:….FFFF)

Unicast Global [001]		
2001::/16	0010 0000 0000 0001	IPv6 InternetARIN, APNIC, RIPE NCC, LACNIC

2002::/16	0010 0000 0000 0010	6 to 4 transition mechanisms
2003::/16	0010 0000 0000 0011	IPv6 InternetRIPE NCC
2400:0000::/19 2400:2000::/19 2400:4000::/21	0010 0100 0000 0000	IPv6 InternetAPNIC
2600:0000::/22 2604:0000::/22 2608:0000::/22 260C:0000::/22	0010 0110 0000 0000 0010 0110 0000 0100 0010 0110 0000 1000 0010 0110 0000 1100	IPv6 InternetARIN
2A00:0000::/21 2A01:0000::/23	0010 1010 0000 0000 0010 1010 0000 0001	IPv6 InternetRIPE NCC
3FFE::/16		6bone

2. **本地站点地址 (Site-local address)**

类似 IPV4 中的私有地址

以 FEC0：: 为前缀。其中前十 bits 固定为 1111111011，紧跟在后面的是连续 38bits 0。

对于站点本地地址来说，前 48bits 总是固定的。在接口 ID 和 48bits 特定前缀之间有 16bits 子网 ID 字段，供机构在内部构建子网。站点本地地址不是自动生成的。

本地站点地址永远不会用于与全球 ipv6 因特网通信。一般用于内网通信

3. **链路本地地址 (Link-local address)**

以 FE80::/10 为前缀，11-64 位为 0，只能在连接到同一本地链路的节点之间使用，主要是用于 IPv6 的一些协议中（比如邻居发现协议：NDP）。 当一个节点启动 IPv6 协议栈时，节点的每个接口会自动配置一个链路本地地址。这种机制使得两个连接到同一链路的 IPv6 节点不需要做任何配置就可以通信 。缺省网关使用链路本地地址

4. **特殊的地址**

a) 未指定地址：全 0 地址 :: 0:0:0:0:0:0/128 或者 :/128

该地址可以表示某个接口或者节点还没有 IP 地址，可以作为某些报文的源 IP 地址（比如作为 DAD 报文的源 IP DHCP）。用于 IPv6 节点在没有获取 IPv6 地址时的

b) 回环地址：::1 0:0:0:0:0:1/128 或者::1/128

用于 IPv6 节点发送报文给自己

c) ipv4 兼容地址

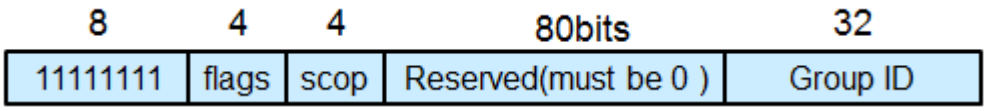
在过渡技术中会使用到一些包含 IPv4 地址的 IPv6 地址，为了让 IPv4 地址显得更加突出一些，定义了内嵌 IPv4 地址的 IPv6 地址格式。内嵌 IPv4 地址格式是过渡机制中使用的一种特殊表示方法。在这种表示方法中，IPv6 地址的部分使用十六进制表示，IPv4 地址部分是十进制格式。

0:0:0:0:0:0:192.168.1.2 或者::192.168.1.2（96 个 0）

自动 ipv4 兼容隧道 nat-pt

1.2.2 Multicast

1. **地址结构**



组播地址最高位前 8 位为 1　　FFXX : :

Flags　　为 0000 永久分配的 ；　　为 0001 临时分配的

Scop　　用来限制组播数据流在网络中发送的范围。

0：预留；
I：节点本地范围；单个接口有效，仅用于 Loopback 通讯
2：链路本地范围； FF02::1
5：站点本地范围；
8：组织本地范围；
E：全球范围；
F：预留。

Group-ID 该字段长度可以为 112 位，用来标识组播组，而 112 位最多可以生成 2112 个组 ID ，RFC2373 并没有将所有的 112 位都定义成组标识，而是建议仅使用该 112 位的最低 32 位组 ID，将剩余的 80 位都置 0。

2. **IPv6 有一些特殊的组播地址，这些地址有特别的含义**

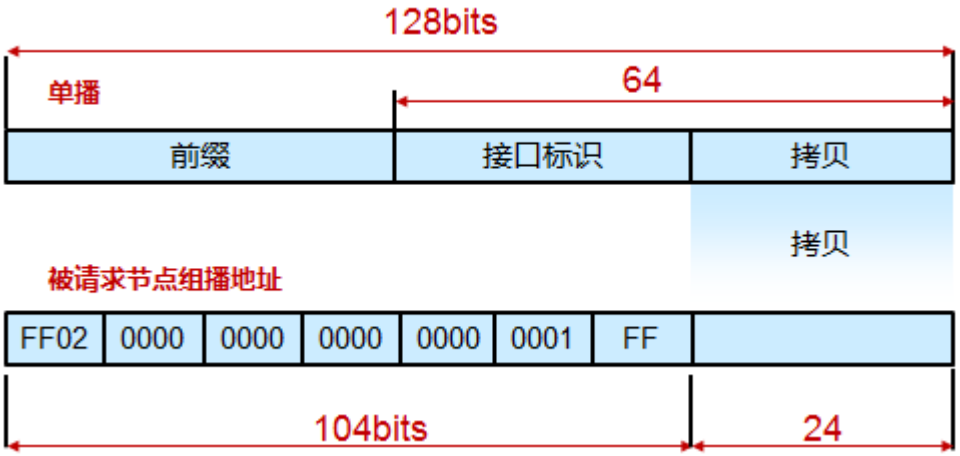
FF01::1(节点本地范围所有节点组播地址)；
FF01 ::2(节点本地范围所有路由器组播地址)；
FF02::1(链路本地范围所有节点组播地址)；
FF02::2(链路本地范围所有路由器组播地址)；
……

3. **被请求组播地址**

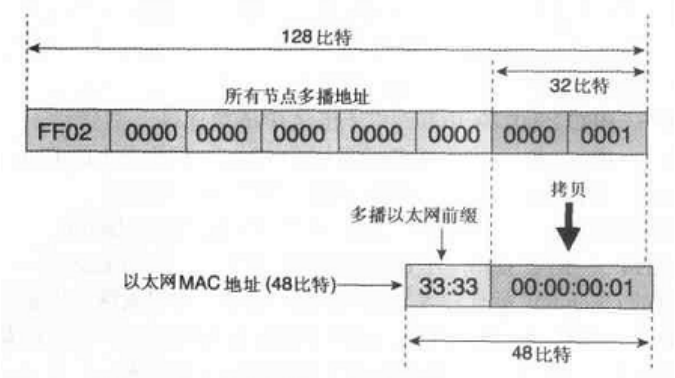
在 IPv6 组播地址中，有一种特别的组播地址，叫做 Solicited-node 地址(请求节点组播地址)。

Solicited-node 地址是一种特殊用途的地址．主要用于重复地址检测（DAD）和替代 ipv4 中的 ARP。

Solicited-node 地址由前缀 FF02::1:FF00:0 / 104 和 ipv6 单播地址的最后 24 位组成。一个 ipv6 单播地址对应一个 Solicited-node 地址。Solicited-node 地址受限范围为本地链路范围。 FF02:0:0:0:0:1:FFXX:XXXX，用于地址解析、DAD



4. **多播的 MAC 地址映射**



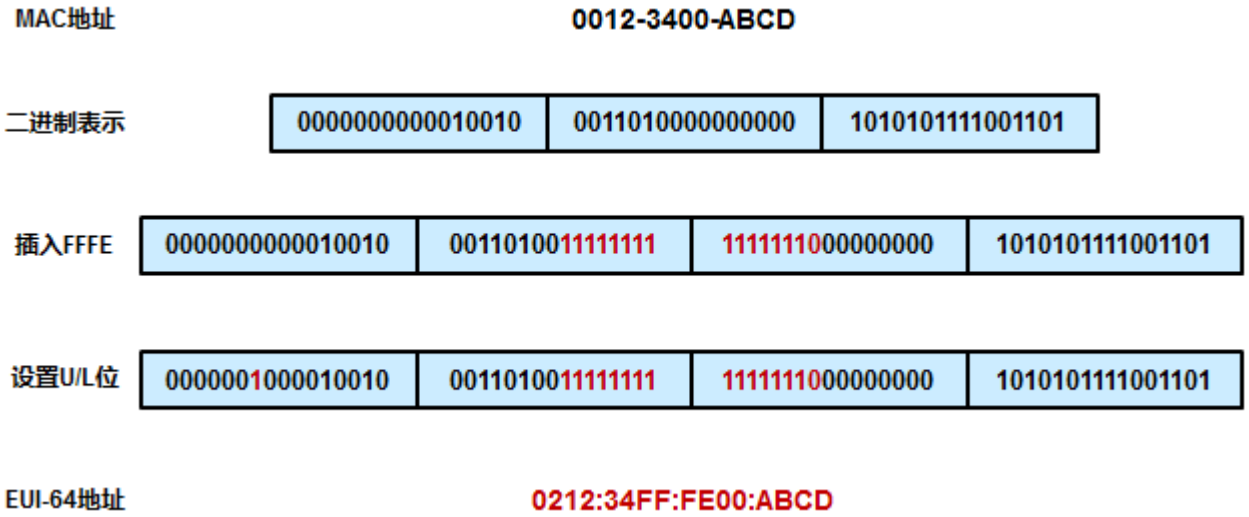
1.2.3 Anycast

适合于 One-to-One-of-Many(一对一组中的一个)的通信场合。接收方只需要是一组接口中的一个即可，如移动用户上网就需要因地理位置的不同而接入离用户最近的一个接收站，这样才可以使移动用户在地理位置上不受太多的限制。

1.2.4 接口ID

接口 ID 可以根据 IEEE　EUI - 64 规范将 48 比特的 MAC 地址转化为 64 比特的接口 ID

- MAC 地址的唯一性保证了接口 ID 的唯一性
- 设备自动生成，不需人为干预



U/L 位为 1 表示全局唯一，为 0 表示本地唯一

- 接口 ID 也可由设备随机生成或者手工配置

1.2.5 所有节点必须具备的地址

链路本地地址	FE80::/10
回环地址	::1

所有节点组播地址	FF01::1；FF02::1
分配的可聚合全球单播地址	2000::/3
所使用的每个单播和任意播地址对应的被请求节点组播地址	FF02::1:FFxx:xxxx 其中 xx:xxxx 是每个单播或任意播地址的低 24 比特
主机所属的所有组的组播地址	FF00::/8

以上是主机节点必须具备的地址，路由器的有所差异。

1.2.6 ipv6 地址配置方法

手工配置

自动配置

- 有状态地址自动配置（DHCPv6）
- 无状态地址自动配置

为了自动获得这个前缀，只要在路由器和主机之间运行一个协议即可。使用 NDP 协议的 Router Solicitation 恳求和 Router Advertisement 通告消息。前者用于发现路由器，并促使路由器发送 Router Advertisement 消息通报前缀信息

RA 源地址：目的地址：FF02::1 链路本地范围所有节点组播地址

RS 的源 ip 是由设备自动产生的 link-local 地址，目标 ip FF02::2 链路本地范围所有路由器组播地址

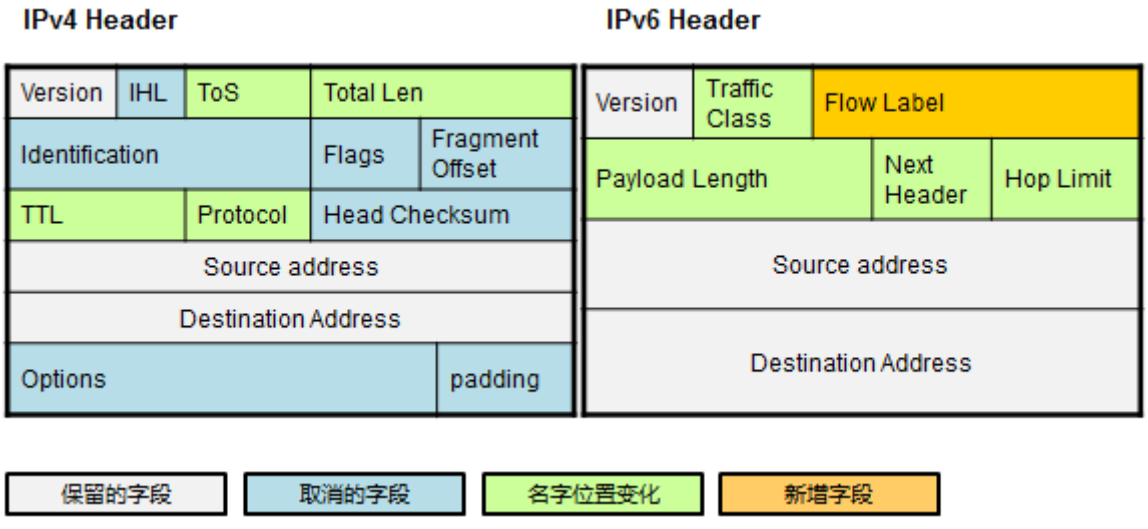
RA(ICMP 134) 接口自动产生 link-local 地址，具备 IP 连接能力 RS 133

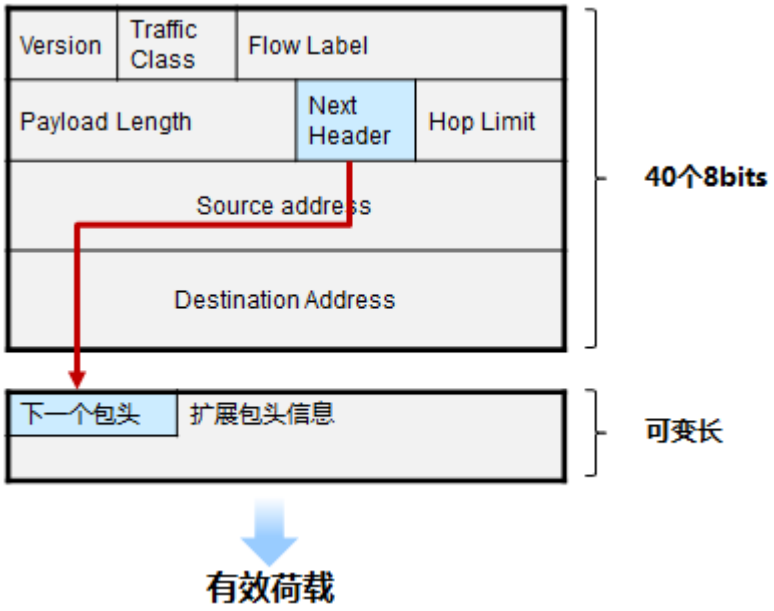
RA 中包含前缀、生存期、缺省网关等信息

缺省情况下 ra 路由通告关闭，在接口下使用 no ipv6 nd suppress-ra

2 Header

1. 报头格式





a) 基本的 IPv4 报头包含 12 的字段，20 个字节长。options 和 padding 字段在需要时添加。
IHL 包头长度；
Total Length 总长度；

b) 基本的 IPv6 报头 40 个 8 位 bit,即 40 个字节长。

c) IPv6 数据包由一个基本报头加上 0 个或多个扩展报头再加上上层协议单元构成。

d) 几个字段的含义

下一个报头(Next Header)： 该字段定义了紧跟在 IPv6 报头后面的第一个扩展报头(如果存在)的类型，或者上层协议数据单元中的协议类型。

跳限制(Hop Limit)： 类似于 IPv4 中的 TTL 字段。它定义了 IP 数据包所能经过的最大跳数。每经过一个路由器，该数值减去 1，当该字段的值为 0 时，数据包将被丢弃

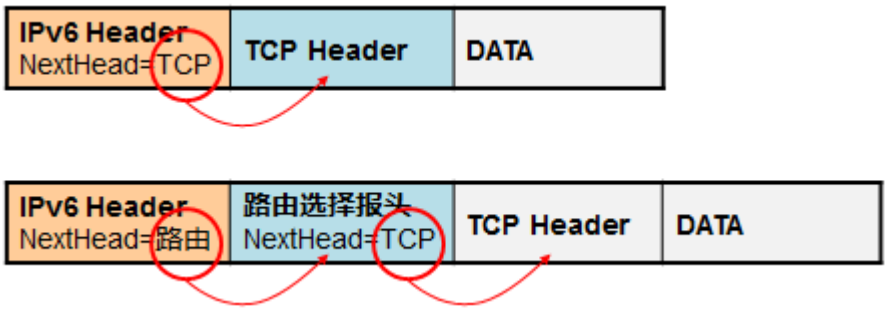
FlowLabel：

2. 扩展报头

在 IPv6 中，v4 中的选项被移到了扩展报头中。中间路由器就不需要处理每一个可能出现的选项，提高了路由器处理数据包的速度，提高了其转发性能。在扩展报头链的最后就是有效负载。

扩展报头可选，只有需要该扩展报头对应的功能，发送主机才会添加相应扩展报头

- 逐跳选项报头(Hop-by—Hop Options Header)
  传送路径上每个路由器都要处理，用于巨型数据包和路由器警报。如 RSVP 资源预留协议
- 目标选项报头(Destination Options Header) 向目标通告信息
- 路由报头(Routing Header) 强制经过特定路由器
- 分段报头(Fragment Header)
- 认证报头(Authentication Header)
- 封装安全有效载荷报头(Encapsulating Security Payload Header)



3. **MTU**

IPv4 中，一个链路的最小 MTU 长度是 68 个字节，一个 IPv4 的最大长度是 60 个字节。

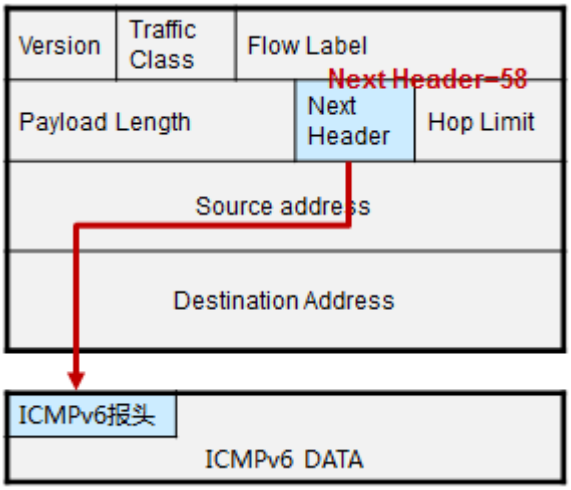
IPv6 最小 MTU 长度是 1280 个字节

3 ICMPv6（rfc2463）

3.1 Foundation

ICMPv6 是 IPv6 的基础协议之一。协议号 58，该协议号在下一个包头字段中。

ICMP 报文有两种：差错消息及信息消息

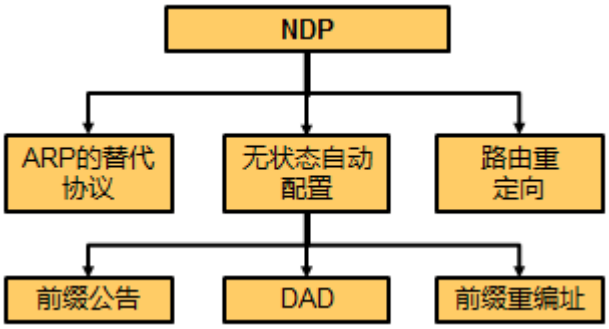


3.2 消息类型

消息类型	TYPE	名称	CODE
差错消息	1	目的不可达	0 无路由
			1 因管理原因禁止访问
			2 未指定
			3 地址不可达

			4 端口不可达
	2	数据包过长	
	3	超时	0 跳数到 0
			1 分片重组超时
	4	参数错误	0 错误的包头字段
			1 无法识别的下一包头类型
			2 无法识别的 ipv6 选项
信息消息	128	回应请求	
	129	回应应答	

3.3 NDP 邻居发现协议



使用 ICMPv6 报文实现以下功能

1. **地址解析(代替 ARP 协议，使用 ICMP 完成地址解析)**
   IPV6 取消了 arp 使用 NS 和 NA 来做，利用的是 solicited address
   通过邻居请求（NS）和邻居通告（NA）报文来解析三层地址对应的链路层地址。
2. **跟踪邻居的状态**
   Show ipv6 neighbor
3. **无状态自动配置**
4. **重复地址检测（DAD）**
5. **路由器重定向**
   路由器向一个 Ipv6 节点发送 icmpv6 消息，通知它在相同的本地链路上才能在一个更好的到达目的地望楼的路由器地址

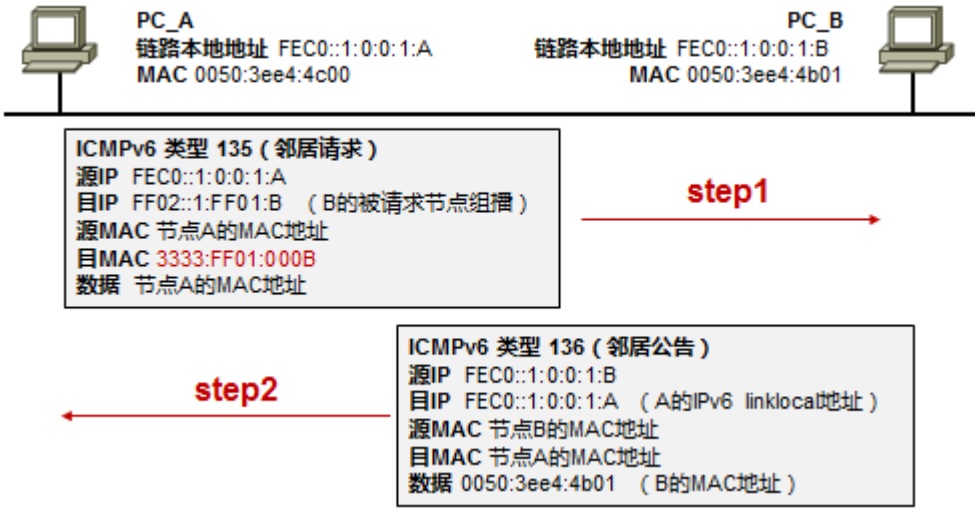
3.3.1 为NDP定义的icmpv6 消息

ICMP 133 RS 134 RA 135 NS 136 NA 137 重定向

机制	RS 133	RA 134	NS 135	NA 136	重定向消息 137
报文介绍	主机可以发送 RS 要求路由器立即产生 RA	包含 MTU、前缀信息等	用来判断邻居的链路层地址，也用于 DAD 等		

替代 ARP			X	X	
前缀公告	X	X			
前缀重新编制	X	X			
DAD			X		
路由重定向					X

3.3.2 用NDP代替ARP



其中 33:33:FF:01:00:0B 是 ipv6 目的地址 FF02::1:FF01:B 的多播映射。

使用 show ipv6 neighbors 可以查看邻居表项

使用 ipv6 neighbor ipv6 地址 接口编号 mac 地址 可实现静态绑定

3.3.3 无状态自动配置

涉及机制

前缀公告

DAD

前缀重新编址

3.3.3.1 前缀公告

1. **前缀公告相关概念**

Show ipv6 interface xxx perfix //显示接口上公告的前缀参数

前缀参数更改则进入接口，ipv6 nd prefix ?

在无状态自动配置中，前缀长度为 64 比特

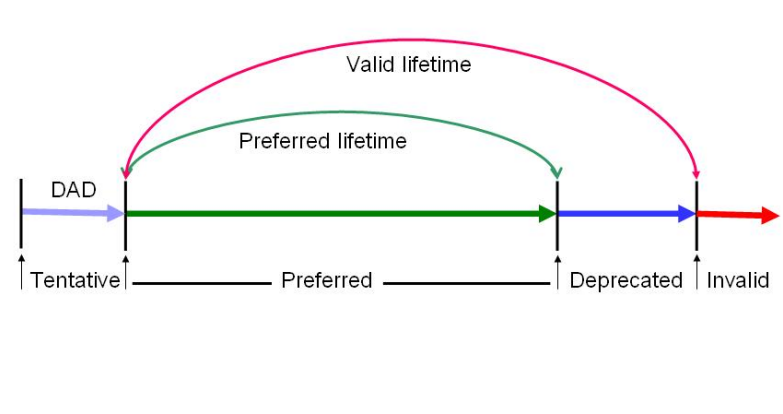
【前缀的两个时间】

IPv6 地址可以通过 state configuration 和 stateless configuration 获得到，简单的说

状态化配置是通过手工添加或是通过 dhcpv6 获得，无状态化是通过 prefix advertisement 获得。不管哪种方式获得的

地址，都会有几个时间状态。

其实 IPv6 地址的 lifetime 主要有两个，一个是 **Valid lifetime** ，另一个是 **preferred lifetime**。首先在 DAD 检查阶段。在 DAD success 前，地址一直处于 tentative (试验状态)状态，这时的地址是不可用状态，直到 DAD success。然后进入 Preferred 状态，也就是首选状态，这个状态过程中的 IPv6 address 的有效性是由 preferred lifetime 决定的。Valid lifetime 是一个类似于 dhcp 中的 lease time。如果地址超过了 valid lifetime ，节点就会把这个地址置为 inactive 状态，然后 delete。在 preferred time 和 valid lifetime 之间叫做 deprecated 状态，这种状态算是一种 buffer，大概意思叫做不赞成使用的状态，当地址达到这个时间段的时候，地址不能主动的发起连接只能是被动的接受连接，这也是为了保证上层应用而设计的，但是过了 valid lifetime 时间地址就变为 invalid，这时任何连接就会 down 掉。



Ipv6 nd suppress-ra 在接口上关闭路由器公告

2. **调整前缀公告的参数**

R1(config-if)#ipv nd prefix 2001::/64 ?

<0-4294967295>	Valid Lifetime (secs)
at	Expire prefix at a specific time/date
infinite	Infinite Valid Lifetime
no-advertise	Do not advertise prefix
no-autoconfig	Do not use prefix for autoconfiguration
	// 当在特定前缀后开启该参数后，该前缀不能用于无状态自动配置
no-rtr-address	Do not send full router address in prefix advert
off-link	Do not use prefix for onlink determination

3.3.3.2 调整ipv6 nd参数

- **ipv nd prefix 2001::/64 30 15** // 30 为 Valid Lifetime, 15 为 Preferred Lifetime

配置该命令后，邻居收到 prefix，显示如下：

```
R2#sh ipv int f 0/0
    FastEthernet0/0 is up, line protocol is up
    IPv6 is enabled, link-local address is FE80::CE00:DFF:FE48:0
    Global unicast address(es):
    2001::CE00:DFF:FE48:0, subnet is 2001::/64 [PRE]    // 状态为 prefer
    valid lifetime 25 preferred lifetime 10                    // 两个时间在不断递减
    当 preferred lifetime 先到 0 时，状态变为[DEP]，当 valid time 变 0 时，地址抹去
```

- **ipv6 nd managed-config-flag**

配置这条命令后，该接口上的前缀信息将不能被链路上的主机用于无状态自动配置。

- **ipv nd other-config-flag**

  Hosts should use DHCP for non-address config

  主机需使用 dhcp 配置除了 ipv6 地址外的其他信息

3.3.3.3 DAD Duplicate Address Detection

- 无状态配置和节点启动时的一个 NDP 机制。
- 通过 NS（ ICMP 135 ）
- 使用源地址（ ::），目的地址为获取到的 v6 地址对应的被请求节点组播地址的 NS 报文
- 一个地址在通过重复地址检测之前称为“tentative 地址”，试验地址。接口不能使用这个试验地址进行单播通讯，但是会加入和 tentative 地址所对应的 Solicited-Node 组播组。重复地址检测：节点向一个自己将使用的 tentative 地址所在的组播组发送一个 Neighbor Solicitation，如果收到某个其他站点回应的 Neighbor dvertisement，就证明该地址已被网络上使用，节点将不能使用该 tentative 地址通讯。
- Solicited-Node 组播地址生成过程

  前缀 FF02:0:0:0:0:1:FF 104 位固定

  接口 ID 的后 24 位：XX:XXXX

  FF02:0:0:0:0:1:FFXX:XXXX

  Ipv6 nd dad attempts x //用来定义邻居请求消息数目，为 0 则关闭 DAD。
- 接口模式下 ipv6 nd dad ？ 可修改 DAD 参数

3.3.3.4 前缀重新编址

前缀重新编址允许从以前的网络平稳过渡到新的前缀，站点内节点使用无状态自动配置（或者其他重编址方法），路由器接口配置新老两个前缀，并且进行公告，老前缀的生存期较短，这样节点使用两个地址，当老前缀失效后，即可实现切换。

首选，站点中所有路由器继续公告当前的前缀，但是有效和首选生存期被减小到接近于 0 的一个值，然后，路由器开始在本地链路公告新的前缀，因此每个本地链路上至少有两个前缀存在。当旧的前缀完全被废止时（生存期已过），路由器公告消息仅包括新前缀。

2. 配置前缀重新编制

   Ipv6 nd prefix xxxx **at 时间**

3.3.4 邻居状态

Incomplete	邻居请求已经发送到目标节点的请求组播地址，但没有收到邻居的通告
Reachable	收到确认，不续再发包确认
Stale	从收到上一次可达性确认后过了超过 30s。
Delay	在 stale 状态后发送过一个报文，并且 5s 内没有可达性确认
Probe	每隔 1s 重传邻居请求来主动请求可达性确认，直到收到确认

4 其他应用

1. **域名系统**

   IPv4 中 A 记录用来将主机名称映射到一个 V4 地址，AAAA 资源记录将主机名映射到一个 IPv6 地址。

   Ipv6 host xxx 地址

   Ipv6 name-sever xxxx

2. **ACL**

   Ipv6 access-list x　　进入 acl 配置模式

   Ipv6 traffic-filter　　在接口下应用

5 动态路由协议（V6）

5.1 RIPng

1. **Foundation**

   使用 UDP521

   Distance 默认 120

2. **Config**

   a) 接口模式下 ipv6 rip 进程名 enable 即可激活

   【注意】进程名 本地有效，如果 Router 两个接口，分别启用 RIPng，使用的是两个不同的进程名，则两进程互相独立。

   b) ipv rip 1　default-information originate　　//接口模式下 ,直接重发布默认路由进 RIP 进程(本地无需静态默认路由)

   ipv rip 1　default-information only　　//接口模式下，只发布默认路由，禁止发布其他路由信息

   c) 调整 RIPng 进程

```
Ipv6 router rip
        ?
```

5.2 OSPFv3 （RFC2740）

1. **Foundation**

   a) 使用 32 位地址作为 router-id，以点分十进制形式表示，如果没有 ROUTER 没有配置 IPv4 地址，则需手工指定 router-id

   b) LINKID 也是 32 位的

   c) 使用 FF02：：5 及 FF02：：6

   d) 使用与 OSPFv2 基本相同的数据包类型

   e) 新增链路 LSA 及区内前缀 LSA

2. **配置**

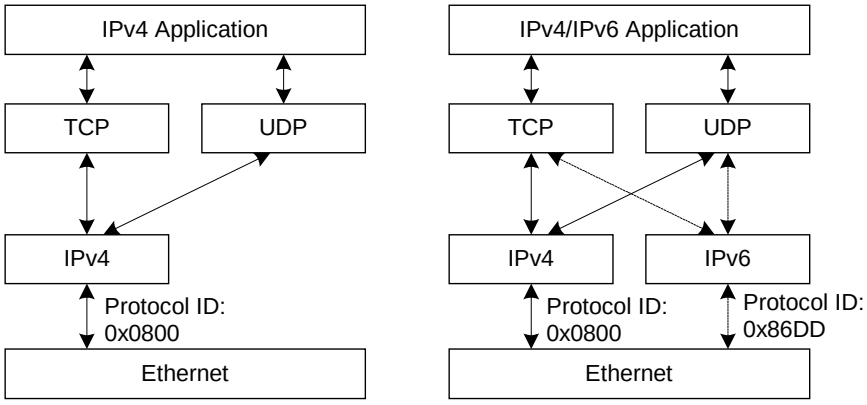
```
Interface vlan 10
Ipv6 enable
Ipv6 address 2001:0:0:10::1/64
Ipv6 ospf 1 area 10

Ipv6 router ospf 1
Router-id 1.1.1.1
redis connect
area 1 range xxx
redistribute rip rip1 metric 10 include-connected
```

6 过渡技术

6.1 Foundation

1. **双栈（dual stack）**

   在双栈设备上，上层应用会优先选择 IPv6 协议栈，而不是 IPv4。 比如，一个同时支持 v4 和 v6 的应用请求地址，会先请求 AAAA 记录，如果没有，则在请求 A 记录。



2. **隧道 tunnel**

   用于在现有网络（v4）中传输不兼容的协议（v6）或者特殊的数据，

   - 手动隧道技术：GRE 隧道、手工隧道
   - 自动隧道技术：6to4 隧道、IPv4 兼容 IPv6 自动隧道、ISATAP 隧道
   - 6PE 6PE 技术依赖于 BGP，BGP 的 Peer 是需要手工指定的，可以算是一种半自动隧道技术。

3. **v6 v4 之间互通**

   NAT-PT (Network Address Translation-Protocol Translation)

6.2 隧道

6.2.1 GRE隧道

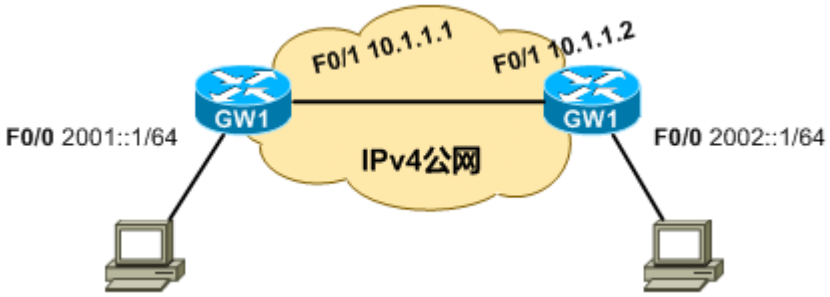
IPv6 报文被包含在 GRE 报文中，作为 GRE 的载荷，优点是通用性比较好

IPv4 报头	GRE 报头	IPv6 报头	数据

配置方式

```
Interface tun 0
    Ipv6 address xxx
    tun source 211.11.11.11
    tun destination 11.21.32.1
    tun mode gre ipv6
```

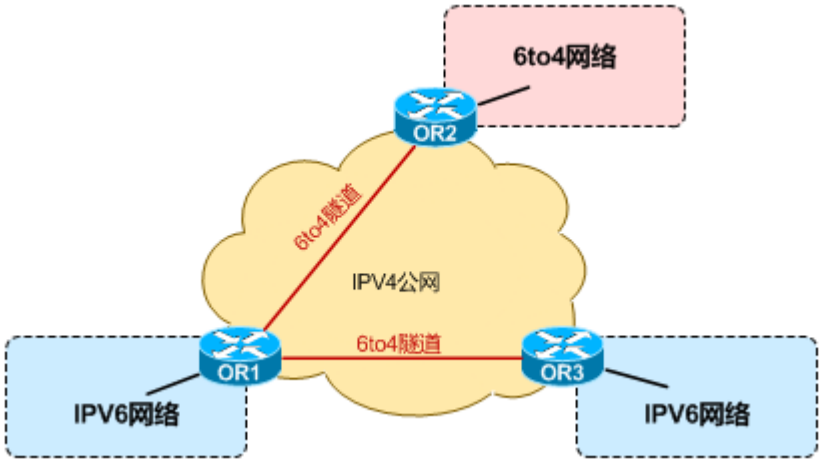
6.2.2 6to4 隧道（手工）



IPv6 报文被包含在 IPv4 报文中作为 IPv4 的载荷

IPv4 报头	IPv6 报头	数据

6.2.3 6to4 隧道（自动）



1. **地址空间**
   a) 6to4 地址空间 2002：：/16
   b) 2002：**IPv4 地址**：子网 ID：：接口 ID
2. **原理**

   其中 **IPv4 地址**是为 IPv6 孤岛申请的公有地址，在 IPv6/IPv4 边界路由器（出口路由器）连接 Ipv4 公网的接口上配置该 v4 地址。内部节点如果使用 6to4 地址，目标网络可以是 6to4 网络，也可以是普通的 v6 网络（非 6to4 网络）

   隧道的源 ipv4 地址手工指定，目标 ipv4 地址不需要显式配置（路由器动态建立隧道），根据通过隧道转发的报文决定，如果报文的目的地是 6to4 地址，则从目的 ip 中提取 ipv4 地址，如果目的地不是 6to4 地址，但下一跳是 6to4 地址，则从下一跳中提取（6to4 中继）V6 报文到达边界设备，查 v6 路由表，出口在隧道口，如果数据流的目标 ip 或者下一跳是 6to4 地址，则 6to4 隧道。

   内部节点不是用 6to4 地址，而用普通 ipv6 地址情况类似

6.2.4 ISATAP

ISATAP（Intra-Site Automatic Tunnel Addressing Protocol）

1. 主机根据配置指定的路由器（在主机上配置该路由器的 IPV4 地址），建立 isatap 隧道
2. 使用内嵌 ipv4 地址的链路本地地址，获得路由器下发的前缀，接口标识中嵌入主机的 ipv4 地址 0000：5efe：ab：cd
3. 通过建立的隧道，传输 ipv6 数据。

   与 6to4 地址类似，ISATAP 地址中也内嵌了 IPv4 地址，它的隧道封装也是根据此内嵌 IPv4 地址来进行的，只是两种地址格式不同。6to4 是使用 IPv4 地址做为网络 ID，而 ISATAP 用 IPv4 地址做为接口 ID。其接口标识符是用修订的 EUI-64 格式构造的。如：fe80::5efe:10.10.10.1，其中 10.10.10.1 为 PC 接口 Ipv4 地址

   ISATAP 地址的前 64 位是通过向 ISATAP 路由器发送请求来得到的，它可以进行地址自动配置。

   在 ISATAP 隧道的两端设备之间可以运行 ND 协议。

   ISATAP 隧道将 IPv4 网络看作一个非广播的点到多点的链路（NBMA）。ISATAP 过渡机制允许在现有的 IPv4 网络内部部署 IPv6，该技术简单而且扩展性很好，可以用于本地站点的过渡。

配置方式：

- 首先配置 *ISATAP* 隧道接口，这时会根据 *IPv4* 地址生成 *ISATAP* 类型的接口 *ID*；
- 根据接口 *ID* 生成一个 *ISATAP* 链路本地 *IPv6* 地址，生成链路本地地址以后，主机就有了 *IPv6* 连接功能；
- 进行主机自动配置，主机获得全局 *IPv6* 地址、站点本地地址等；
- 当主机与其它 *IPv6* 主机进行通讯时 ,从隧道接口转发 ,将从报文的下一跳 *IPv6* 地址中取出 *IPv4* 地址作为 *IPv4* 封装的目的地址。如果目的主机在本站点内，则下一跳就是目的主机本身，如果目的主机不在本站点内，则下一跳为 *ISATAP* 路由器的地址。

6.2.5 Ipv4 兼容Ipv6 自动隧道技术

IP V4 兼容 IP V6 自动隧道技术

目的地址为 IPv4 兼容 IPv6 地址，包含的 IPv4 地址即为隧道对端

IPv4 兼容 IPv6 地址:　0:0:0:0:0:0:a.b.c.d（a.b.c.d 是 ip v4 地址）

适用于不经常性的 IPv6 节点连接需求

IPv4 兼容隧道是通过 Tunnel 虚接口实现的，如果一个 Tunnel 口的封装模式是 IPv4 兼容隧道，则只需配置隧道的源地址，而目的地址是在转发报文时 ,从 IPv6 报文的目的地址中取得的。从 IPv4 兼容隧道转发的 IPv6 报文的目的地址必须是 IPv4 兼容的 IPv6 地址，隧道的目的地址就是 IPv4 兼容地址的后 32 位。如果一个 IPv6 报文的目的地址不是 IPv4 兼容地址，则不能从 IPv4 兼容隧道转发出去。

7 IPv6 基础实验

7.1 主机相关

1. **基本配置命令**

   PC 安装 ipv6 协议栈

Ipv6 install	// 安装完后重启，默认开启无状态自动获取地址
ipv6 if	// 查看 ipv6 接口配置信息
ping *IPv6Address* [%*ZoneID*]	// ping 链路本地地址和站点本地地址可能要加索引号，全局单播地址不需要

2. **PC 常用 ipv6 命令集合**

   Netsh 可以用来配置、管理、重置 IPV4 V6 协议栈

netsh interface ipv6 show interface	查看接口索引

ipv6 if x	
netsh interface ipv6 show neighbors	
netsh interface ipv6 add address *接口索引号 ipv6 地址* 或 ipv6 adu x ……	为接口添加 ipv6 地址
netsh interface ipv6 show address	查看 ipv6 地址（ipconfig 也可）
netsh interface ipv6 show route	查看路由
netsh interface ipv6 add route ::/0 “本地连接” 2010::1 或 ipv6 rtu	添加路由
netsh interface ipv6 reset	Ipv6 接口复位 （手工配置的 ipv6 地址会丢失）
netsh interface ipv6 renew	Ipv6 接口重启
ipv6 adu ifindex/address [life validlifetime]	给某个接口添加 IPv6 地 如给接口 4 添加 IPv6 地址 2001：：1/64， 命令： ipv6 adu 4/2001::1

3. Ipv6 网络应用基本模型

Ipv6 主机（xp 系统 ie6.0）访问 ipv6 WEB 站点

IE6.0 暂不支持地址栏输入 rfc2732 定义的 literal ipv6 address 的方式访问 ipv6 站点。

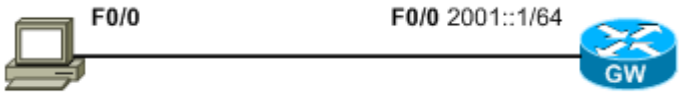
解决办法，通过域名来访问，修改 hosts 文件 **C:/windows/system32/drivers/etc/hosts**

```
# 102.54.94.97      rhino.acme.com          # source server
# 38.25.63.10       x.acme.com              # x client host

127.0.0.1           localhost
::1    test.com
```

7.2 路由器基本配置

```
Router(config)#ipv6 unicast-routing
Router(config)#intertface f0/1
Router(config-if)#ipv6 enable
Router(config-if)#ipv6 adress 2001::1/64
```

7.3 无状态自动获取



GW 的配置

no ipv6 nd suppress-ra	接口地址 2001::1/64， 同时开启路由器通告（默认关闭）
Ipv6 nd managed-config-flag	用来启用接口的有状态自动配置（如通过 DHCPv6 获取地址），默认是关闭的。
Ipv6 nd other-config-flag	默认关闭，关闭时用来使节点不使用有状态配置机制来获取除了 v6 地址以外的其他参数。

客户端的配置

ipv6 address autoconfig　　使用 EUI64 地址构建 Ipv6 地址；如果加 default 关键字，则会添加默认网关（默认路由）

7.4 DHCP自动获取

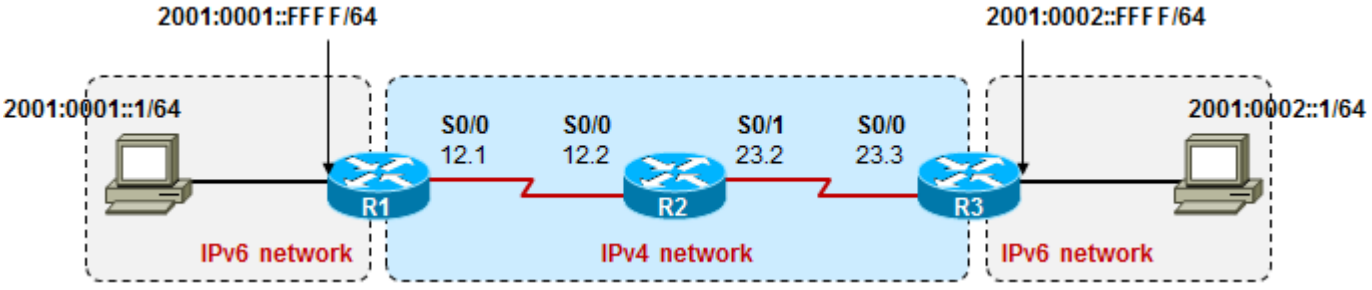
GW 的配置

```
ipv6 dhcp pool DHCP-pool
    prefix-delegation pool dhcppool lifetime 1800 600      // 调用ipv6 local pool 并设定lifetime
    dns-server 2000::8
    domain-name HelloWord
ipv6 local pool dhcppool 2001::/64 64                      // 定义准备通告的Ipv6前缀

interface FastEthernet0/0
    ipv6 enable
    ipv6 address 2001::1/64
    ipv6 dhcp server DHCP-pool                             // 在接口上开启ipv6 DHCP，并调用池
```

PC 的配置

```
interface FastEthernet0/0
    ipv6 enable                          // 接口开启IPV
    ipv6 dhcp client pd test             // 配置DHCP client，并且指定获取到的前缀名称为test（本地有效）
    ipv6 address test ::/64 eui-64       // 使用获取到的前缀（test），加上本接口的EUI64，构成接口全局ipv6地址
```

7.5 6to4 隧道（手工） ipv6 over ip



R1 的配置如下：

```
Ipv6 unicast-routing
Interface serial0/0
  ip address 10.1.12.1 255.255.255.0
Interface fa1/0
  Ipv6 enable
```

```
 Ipv6 address 2001:0001::FFFF/64

Interface tunnel 0
 Ipv6 enable
 tunnel mode ipv6ip
 Tunnel source serial 0/0
 Tunnel destination 10.1.23.3
ip route 0.0.0.0 0.0.0.0 10.1.12.2          // ipv4路由，使得路由能访问到tunnel destination，也就是10.1.23.3
Ipv6 route ::/0   tunnel 0                  // ipv6路由，前往IPv6网络的流量全部扔到tunnel
```

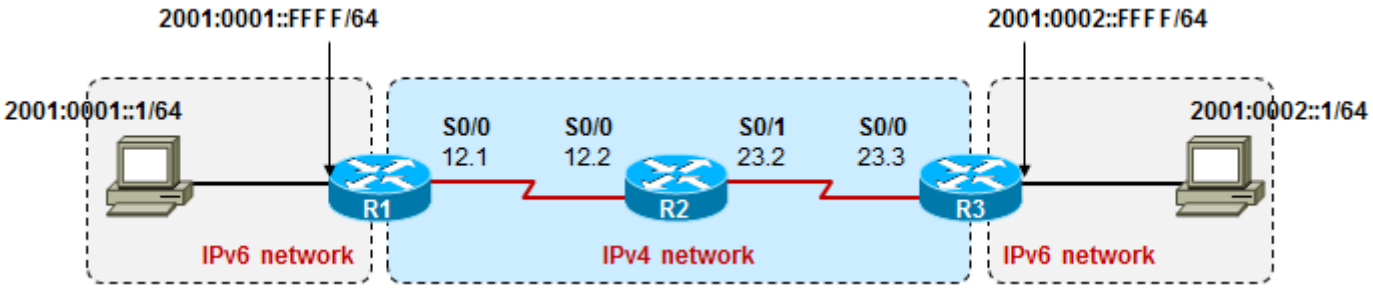
R3 的配置类似；R2 只需配置接口 IP 即可；PC1 及 PC2 各自配好 IPv6 地址，指网关即可。

Tunnel 接口未必一定需要 ipv6 address，当然，也可以配置 v6 地址，这不影响穿越路由器的流量

配置完成后，PC1 即可 ping 通 PC2，抓包如下：

```
⊞ Cisco HDLC
⊟ Internet Protocol, Src: 10.1.12.1 (10.1.12.1), Dst: 10.1.23.3 (10.1.23.3)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 120
    Identification: 0x0019 (25)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: IPv6 (0x29)
  ⊞ Header checksum: 0x843e [correct]
    Source: 10.1.12.1 (10.1.12.1)
    Destination: 10.1.23.3 (10.1.23.3)
⊟ Internet Protocol Version 6
  ⊞ 0110 .... = Version: 6
    .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00000000
    .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
    Payload length: 60
    Next header: ICMPv6 (0x3a)
    Hop limit: 63
    Source: 2001:1::1 (2001:1::1)
    Destination: 2001:2::1 (2001:2::1)
⊞ Internet Control Message Protocol v6
```

7.6 6to4 隧道（手工） GRE隧道



R1 的配置如下：

```
Ipv6 unicast-routing
Interface serial0/0
```

```
  ip address 10.1.12.1 255.255.255.0
Interface fa1/0
  Ipv6 enable
  Ipv6 address 2001:0001::FFFF/64

Interface tunnel 0
  Ipv6 enable
  tunnel mode gre ip                    // 注意隧道模式
  Tunnel source serial 0/0
  Tunnel destination 10.1.23.3
ip route 0.0.0.0 0.0.0.0 10.1.12.2      // ipv4路由，使得路由能访问到tunnel destination，也就是10.1.23.3
Ipv6 route ::/0   tunnel 0              // ipv6路由，前往IPv6网络的流量全部扔到tunnel
```

R3 的配置大同小异，只不过隧道接口的 destination 修改一下即可；

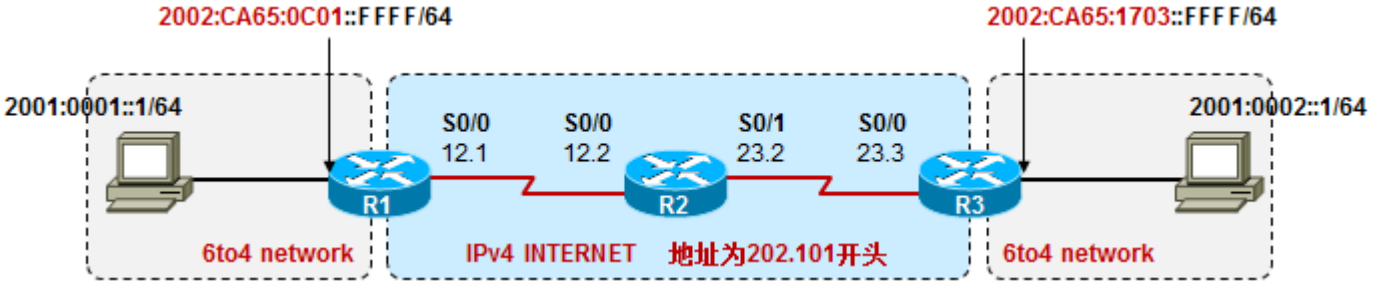
如果需要在 R1、R3 之间运行动态路由协议，则在 tunnel 接口上激活路由选择协议即可

报文抓包如下：

```
⊞ Cisco HDLC
⊟ Internet Protocol, Src: 10.1.12.1 (10.1.12.1), Dst: 10.1.23.3 (10.1.23.3)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 124
    Identification: 0x0030 (48)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: GRE (0x2f)
  ⊞ Header checksum: 0x841d [correct]
    Source: 10.1.12.1 (10.1.12.1)
    Destination: 10.1.23.3 (10.1.23.3)
⊞ Generic Routing Encapsulation (IPv6)
⊟ Internet Protocol Version 6
  ⊞ 0110 .... = Version: 6
    .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00000000
    .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
    Payload length: 60
    Next header: ICMPv6 (0x3a)
    Hop limit: 63
    Source: 2001:1::1 (2001:1::1)
    Destination: 2001:2::1 (2001:2::1)
⊞ Internet Control Message Protocol v6
```

7.7 6to4 隧道（自动）

【实验 1】：　两个内网都是用 6to4 地址（双方都是 6to4 网络）



R1 路由器申请的 ipv4 公网地址为 202.101.12.1；R3 申请的公网地址为 202.101.23.3

有了公网地址，我们也就有了根据公网地址计算得来的 6to4 的地址空间：

2002:IPV4 地址映射:子网 ID::/48　　前面 48 位是 2002+ipv4 的公网地址，后面 64 位是接口 ID

例如 R1 公网地址为：202.101.12.1，那么对应的 6to4 地址空间就是：2002:CA65:0C01::/48

这个地址空间加上子网 ID，就可以分配给内网用户了，这个实验我们内网用户采用无状态自动配置获取地址。

这时候，如果 R1 下有用户要访问 R3 下的 v6 网络，那么目的地址肯定是 R3 的 6to4 地址空间，数据到了 R1 后，R2 一看，发现是 2002 的 ipv6 地址，于是它就直接去读 2002 后面的 32 位，将其转换成 ipv4 地址，并作为 6to4 的 tunnel destination。

R1 的配置如下：

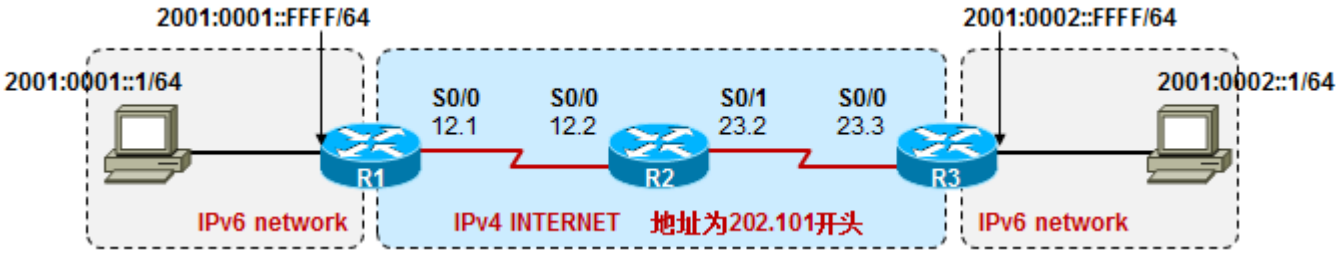
```
ipv6 unicast-routing
interface Tunnel0
  ipv6 enable
  tunnel source Serial0/0
  tunnel mode ipv6ip 6to4
interface fast1/0
  ipv6 address 2002:CA65:0C01::FFFF/64
  ipv6 enable
  no ipv6 nd suppress-ra
interface Serial0/1
  ip address 202.101.12.1 255.255.255.0
ipv6 route 2002::/16 Tunnel0
```

R2 的配置如下：

```
ipv6 unicast-routing
interface Tunnel0
  ipv6 enable
  tunnel source Serial0/0
  tunnel mode ipv6ip 6to4
interface fast1/0
  ipv6 address 2002:CA65:1703::FFFF/64
  ipv6 enable
  no ipv6 nd suppress-ra
interface Serial0/1
  ip address 202.101.23.3 255.255.255.0
ipv6 route 2002::/16 Tunnel0
```

PC 使用无状态自动获取地址： ipv6 address autoconfig default

【实验 2】： 两边的 IPv6 孤岛都是使用普通的 IPv6 全局地址



两端均为普通 Ipv6 网络，使用常规的全局 Ipv6 地址。

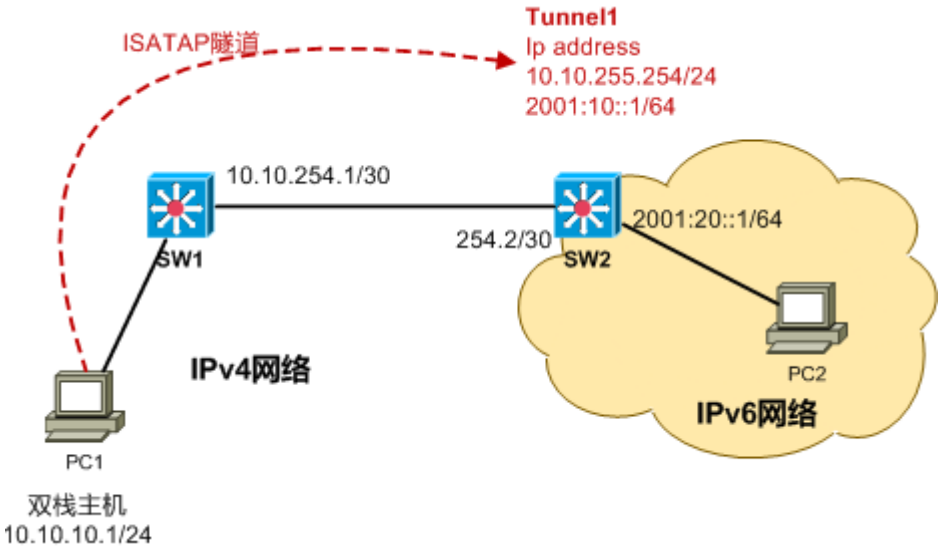
R1 及 R3 分别申请公网 Ipv4 地址，配置动态 6to4 隧道。

以 R1 为例，配置 tunnel0，分配一个 Ipv6 地址（6to4 地址），这个地址正是 R1 的 V4 外网地址对应的 IPV6 6to4 地址也即 202.101.12.1 对应 2002:**CA65:0C01**::1/48，R3 同理。接下来其实就是静态路由的把戏了。

这时，内网有数据去往 2001:0002::/16 网络时，数据被送到 2002:CA65:1703::FFFF（下一跳，也就是 R3 的 tunnel 6to4 地址）路由器通过递归查询到，下一跳 2002:CA65:1703::FFFF 应该扔到 tunnel0，而 tunnel0 是 6to4 隧道，于是将 2002:CA65:1703::FFFF 翻译成对应的 V4 地址，也就是 202.101.23.3

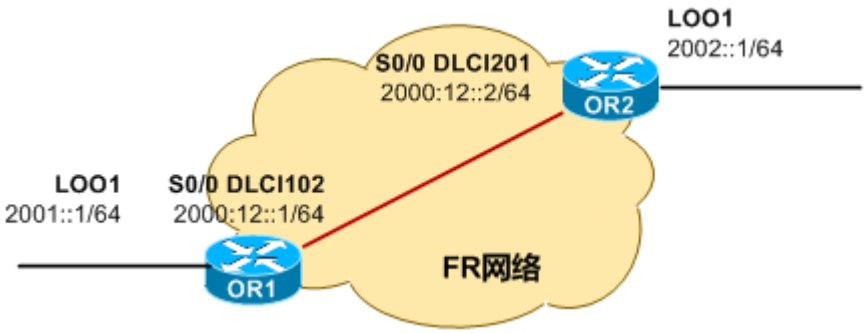
R1	R3
ipv6 unicast-routing	ipv6 unicast-routing
interface Tunnel0	interface Tunnel0
ipv6 address **2002:CA65:C01**::FFFF/64	ipv6 address **2002:CA65:1703**::FFFF/64
ipv6 enable	ipv6 enable
tunnel source Serial0/0	tunnel source Serial0/0
tunnel mode ipv6ip 6to4	tunnel mode ipv6ip 6to4
interface fast1/0	interface fast1/0
ipv6 address 2001:0001::FFFF/64	ipv6 address 2001:0002::FFFF/64
ipv6 enable	ipv6 enable
interface Serial0/0	interface Serial0/0
ip address 202.101.12.1 255.255.255.0	ip address 202.101.23.3 255.255.255.0
ip route 0.0.0.0 0.0.0.0 202.101.12.2	ip route 0.0.0.0 0.0.0.0 202.101.23.2
ipv6 route 2001::/16 2002:CA65:1703::FFFF	ipv6 route 2001::/64 2002:CA65:C01::FFFF
ipv6 route 2002:CA65:1703::/48 Tunnel0	ipv6 route 2002:CA65:C01::/48 Tunnel0

7.8 ISATAP



PC	SW2
netsh interface ipv6 isatap set router **10.10.255.254**	ipv6 unicast-routing interface Tunnel1 ip address **10.10.255.254** 255.255.255.0 ipv6 address 2001:10::1/64 ipv6 enable no ipv6 nd suppress-ra tunnel source **10.10.255.254** tunnel mode ipv6ip isatap interface FastEthernet0/1 switchport access vlan 20 // PC2 的 VLAN interface Vlan20 ipv6 address 2001:20::1/64 ipv6 enable no ipv6 nd suppress-ra
实验现象： PC 上获取到了 IPV6 地址： 隧道适配器 isatap.{CC1CCF67-E12C-4BD9-849F-ECB2EADA3747}: 连接特定的 DNS 后缀 . . . . . : IPv6 地址 . . . . . : 2001:10::5efe:10.10.10.1 本地链接 IPv6 地址. . . . . : fe80::5efe:10.10.10.1%20 默认网关. . . . . : fe80::5efe:10.10.255.254%20 并且能访问到 PC2	

7.9 FR下的OSPFv3



OR1 的配置

```
ipv6 unicast-routing
interface Serial0/0
  encapsulation frame-relay
  no frame-relay inverse-arp
  ipv6 enable
  ipv6 address 2000:12::1/64
  ipv6 ospf network point-to-point
  ipv6 ospf 1 area 0
  frame-relay map ipv6 2000:12::2 102 broadcast
```

OR2 的配置

```
ipv6 unicast-routing
interface Serial0/0
  encapsulation frame-relay
  no frame-relay inverse-arp
  ipv6 enable
  ipv6 address 2000:12::2/64
  ipv6 ospf network point-to-point
  ipv6 ospf 1 area 0
  frame-relay map ipv6 2000:12::1 201 broadcast
```

完成此步骤后，OR1 及 OR2 能建立起 OSPF 邻居关系，互相之间的全局 IPV6 单播地址也都能 ping 通

此时分别在 OR1 及 OR2 上开启 loopback 接口，并配置 Ipv6 地址，然后宣告进 OSPFv3

OR1 的配置

```
interface Loopback0
  ipv6 address 2001::1/64
  ipv6 enable
  ipv6 ospf 1 area 1
```

OR2 的配置

```
interface Loopback0
```

```
ipv6 address 2002::1/64
ipv6 enable
ipv6 ospf 1 area 2
```

完成后，OR1 及 OR2 都能学习到对方的 Loopback 接口路由，但是却无法 ping 通。

查看 OR1 的路由表：

```
OI   2002::1/128 [110/64]
       via FE80::5C10:8CFF:FEE0:FE89, Serial0/0
```

发现去往 OR2 loopback 接口的路由，下一跳是 FE80::5C10:8CFF:FEE0:FE89，也就是 OR2 接口的链路本地地址，原来 Ipv6 环境下，一个接口往往具有多个 Ipv6 地址，而 OSPF 邻居关系的维护又以稳定为前提，每个接口都必备的链路本地地址，就成了建立邻居关系最好的一句，那么为什么 ping 不通呢？正是由于这是个帧中继的环境，FE80::5C10:8CFF:FEE0:FE89 这个 IPV6 地址，OR1 上并没有做映射，OR2 上同理，因此分别添上各自对端的链路本地地址的帧中继映射即可。

如 OR1，OR2 同理

```
frame-relay map ipv6 FE80::5C10:8CFF:FEE0:FE89 102 broadcast
```

8 本文档已阅参考读物

- 《Cisco IPv6 网络实现技术》