ICMPv6和NDP

HUAWEI

前言

- 在IPv4中，ICMP允许主机或设备报告差错情况。ICMP报文作为IP报文的数据部分，再封装上IP报文头部，组成完整的IP报文发送出去。常用的Ping、Tracert等应用都是基于ICMP实现的。
- IPv6定义了ICMPv6（Internet Control Message Protocol for IPv6），除了提供类似IPv4中ICMP的功能外，还有诸多扩展应用。
- 邻居发现协议（Neighbor Discovery Protocol，NDP）便是基于ICMPv6实现的，作为IPv6的关键协议，NDP提供了如IPv6前缀发现、重复地址检测、地址解析、重定向等功能。
- 本课程详细介绍ICMPv6和NDP。

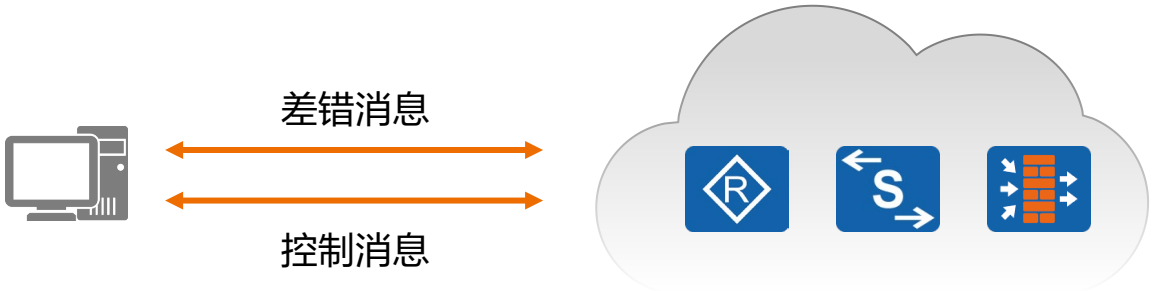
课程目标

- 学完本课程后，您将能够：
  - 描述ICMPv6的功能
  - 描述ICMPv6的报文格式
  - 描述ICMPv6的报文类型
  - 描述NDP的功能与技术原理

目录

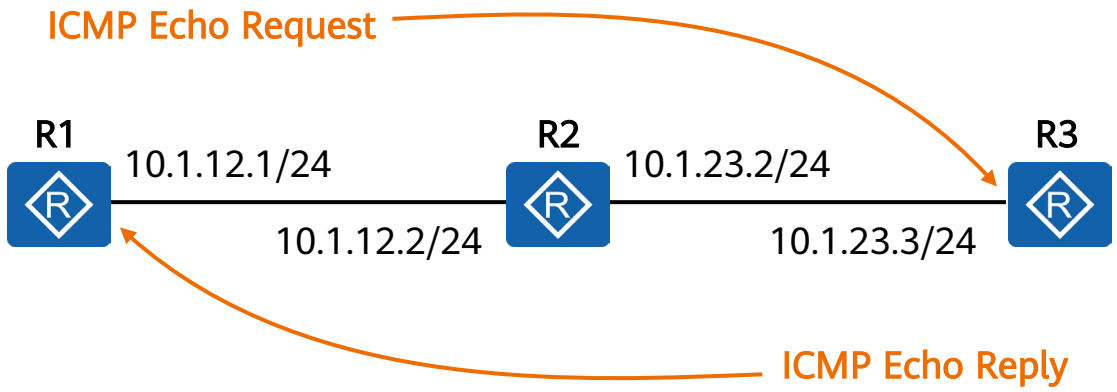
回顾一下ICMP

- 在IPv4中，Internet控制消息协议ICMP（Internet Control Message Protocol）是IPv4协议的重要辅助协议。
- ICMP用来在网络设备间传递各种差错和控制信息，对于收集各种网络信息、诊断和排除各种网络故障等方面起着至关重要的作用。



基于ICMP的重要应用：Ping

ICMP Echo消息常用于诊断源和目的地之间的网络连通性，同时还可以提供其他信息，如报文往返时间等。



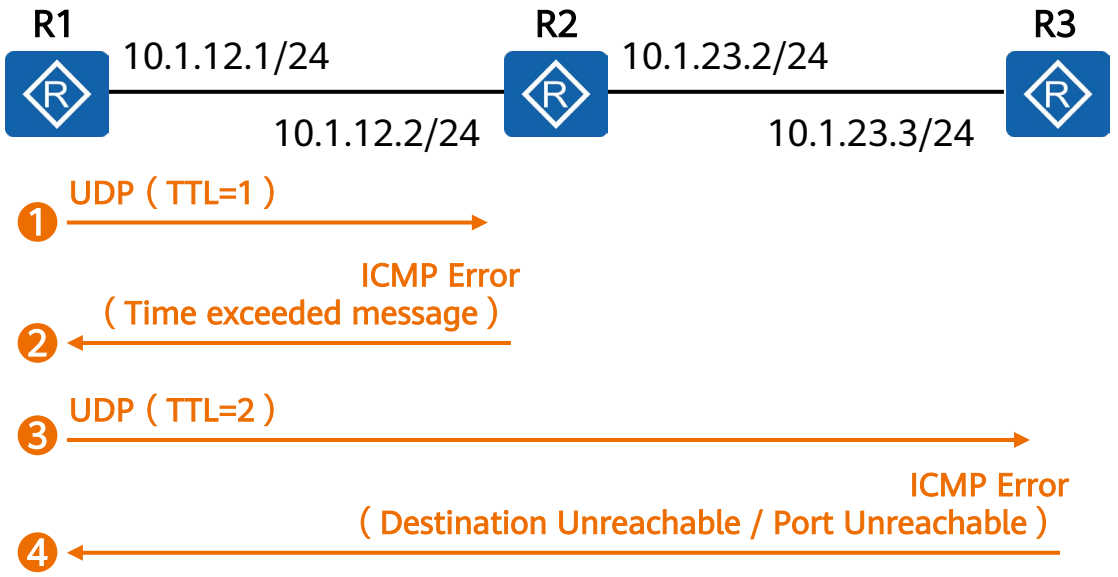
应用：Ping

Ping是一个广为人知的应用程序，该应用基于ICMP协议，常用于探测到达目的节点的网络可达性。

```
[R1] ping 10.1.23.3
  PING 10.1.23.3: 56  data bytes, press CTRL_C to break
    Reply from 10.1.23.3: bytes=56 Sequence=1 ttl=254 time=40 ms
    Reply from 10.1.23.3: bytes=56 Sequence=2 ttl=254 time=20 ms
    Reply from 10.1.23.3: bytes=56 Sequence=3 ttl=254 time=20 ms
    Reply from 10.1.23.3: bytes=56 Sequence=4 ttl=254 time=30 ms
    Reply from 10.1.23.3: bytes=56 Sequence=5 ttl=254 time=30 ms

  --- 10.1.23.3 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 20/28/40 ms
```

基于ICMP的重要应用：Tracert

ICMP定义了各种错误消息，用于诊断网络连接性问题；根据这些错误消息，设备可以判断出网络问题的原因。



应用：Tracert

Tracert基于IP报文头中的TTL值来逐跳跟踪报文的转发路径。Tracert是检测网络丢包和时延的有效手段，也可帮助管理员发现网络中的路由环路。

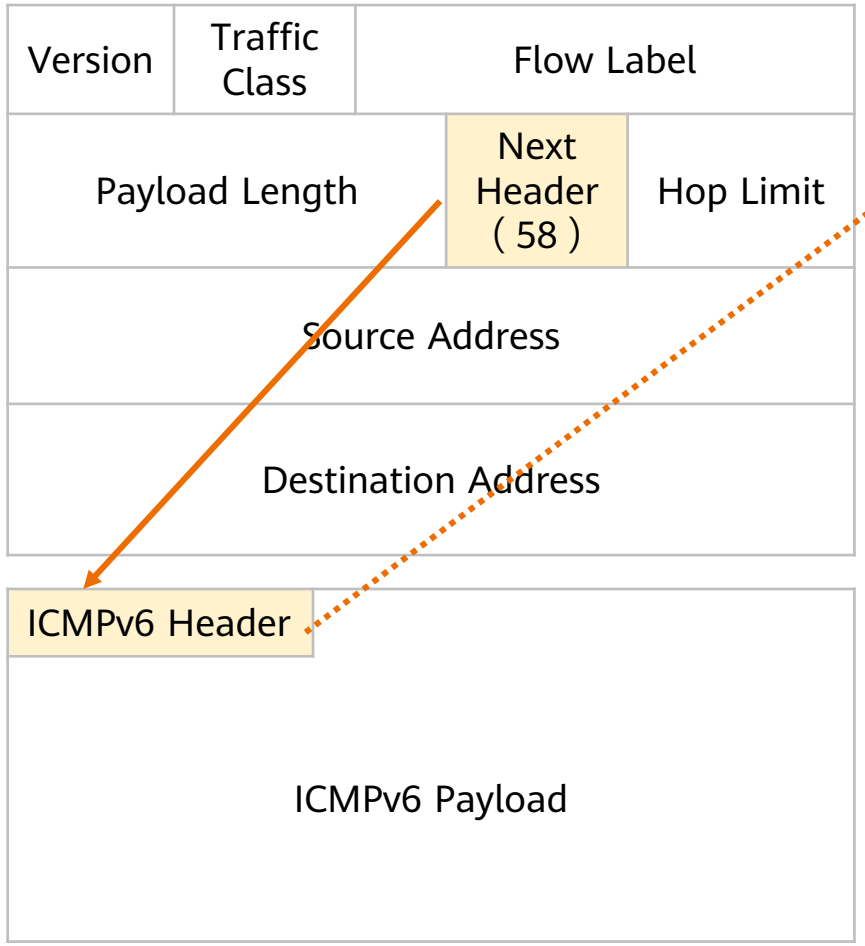
```
[R1] tracert 10.1.23.3

 traceroute to  10.1.23.3(10.1.23.3), max hops: 30 ,packet
length: 40,press CTRL_C to break

 1 10.1.12.2 30 ms  20 ms  20 ms

 2 10.1.23.3 20 ms  20 ms  20 ms
```

ICMPv6概述

- ICMPv6（Internet Control Message Protocol for the IPv6）是IPv6的基础协议之一。
- 在IPv4中，Internet控制报文协议ICMP（Internet Control Message Protocol）向源节点报告关于向目的地传输IP数据包过程中的错误和信息。它为诊断、信息和管理目的定义了一些消息，如：目的不可达、数据包超长、超时、回应请求和回应应答等。
- 在IPv6中，ICMPv6除了提供ICMPv4常用的功能之外，还是其它一些功能的基础，如邻居发现、无状态地址配置（包括重复地址检测）、PMTU发现等。
- ICMPv6的协议号（即IPv6报文中的Next Header字段的值）为58。

目录

ICMPv6报文格式



ICMPv6 Header

Type	Code	Checksum

消息类型	Type	名称	Code
差错消息	1	目的不可达	0 无到达目标设备的路由
			1 因管理原因禁止访问
			2 未指定
			3 目的IP地址不可达
			4 目的端口不可达
	2	数据包过长	0
	3	超时	0 在传输中超越了跳数限制
			1 分片重组超时
	4	参数错误	0 错误的报头字段
			1 无法识别的NextHeader值
			2 扩展头中出现未知的选项
信息消息	128	Echo Request	0
	129	Echo Reply	0

ICMPv6报文类型

ICMPv6报文分为两类：差错报文和信息报文。

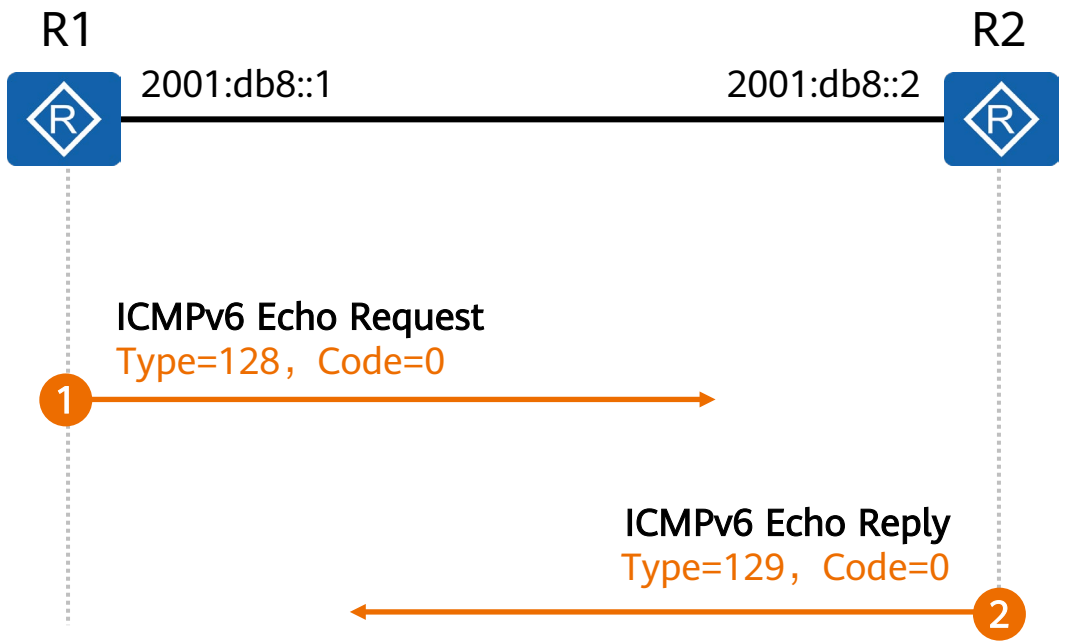
差错报文（Error Message）	信息报文（Informational Message）
• Type字段最高bit为0，即ICMPv6 Type=[0，127]。 • 差错消息用于报告在转发IPv6数据包过程中出现的错误，如目的不可达、超时等等。	• Type字段最高bit为1，即ICMPv6 Type=[128，255]。 • 信息报文可以用来实现同一链路上节点间的通信和子网内的组播成员管理等。

ICMPv6报文类型（续）

报文类型	TYPE	名称	Code
差错报文	1	目的不可达	0 没有到达目的网络的路由
			1 因管理原因禁止访问
			2 未指定
			3 目的地址不可达
			4 目的端口不可达
	2	数据包过长	0
	3	超时	0 跳数到0
			1 分片重组超时
	4	参数错误	0 IPv6基本头或扩展头的某个字段有错误
			1 IPv6基本头或扩展头的NextHeader值不可识别
			2 扩展头中出现未知的选项
信息报文	128	Echo request	0
	129	Echo reply	0

- ICMPv6错误报文用于报告在转发IPv6数据包过程中出现的错误。
- ICMPv6信息报文提供诊断功能和附加的主机功能，例如多播侦听发现和邻居发现。
- **还有一些其他报文，为NDP而定义，后续介绍。**

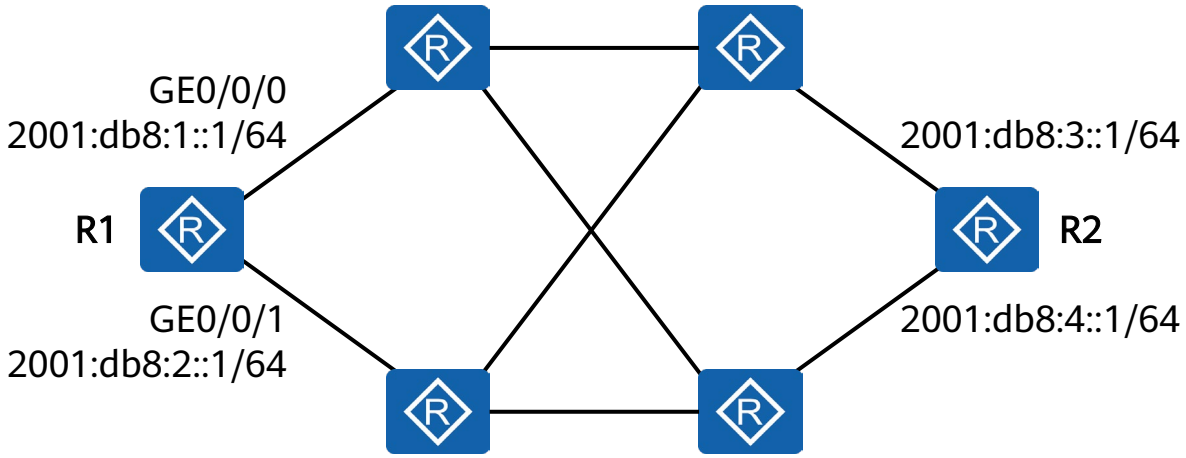
ICMPv6信息报文应用 - Ping



- **Echo Request**：用于发送到目标节点，以使目标节点立即发回一个Echo Reply应答报文。Echo Request报文的Type字段值为128，Code字段的值为0。
- **Echo Reply**：当收到一个Echo Request报文时，ICMPv6会用Echo Reply报文响应。Echo Reply报文的Type字段的值为129，Code字段的值为0。

ICMPv6信息报文应用 – Ping命令在华为设备上的执行

- **ping ipv6**命令用来检查指定的IPv6地址是否可达，并输出相应的统计信息。



<R1> **ping ipv6** ?

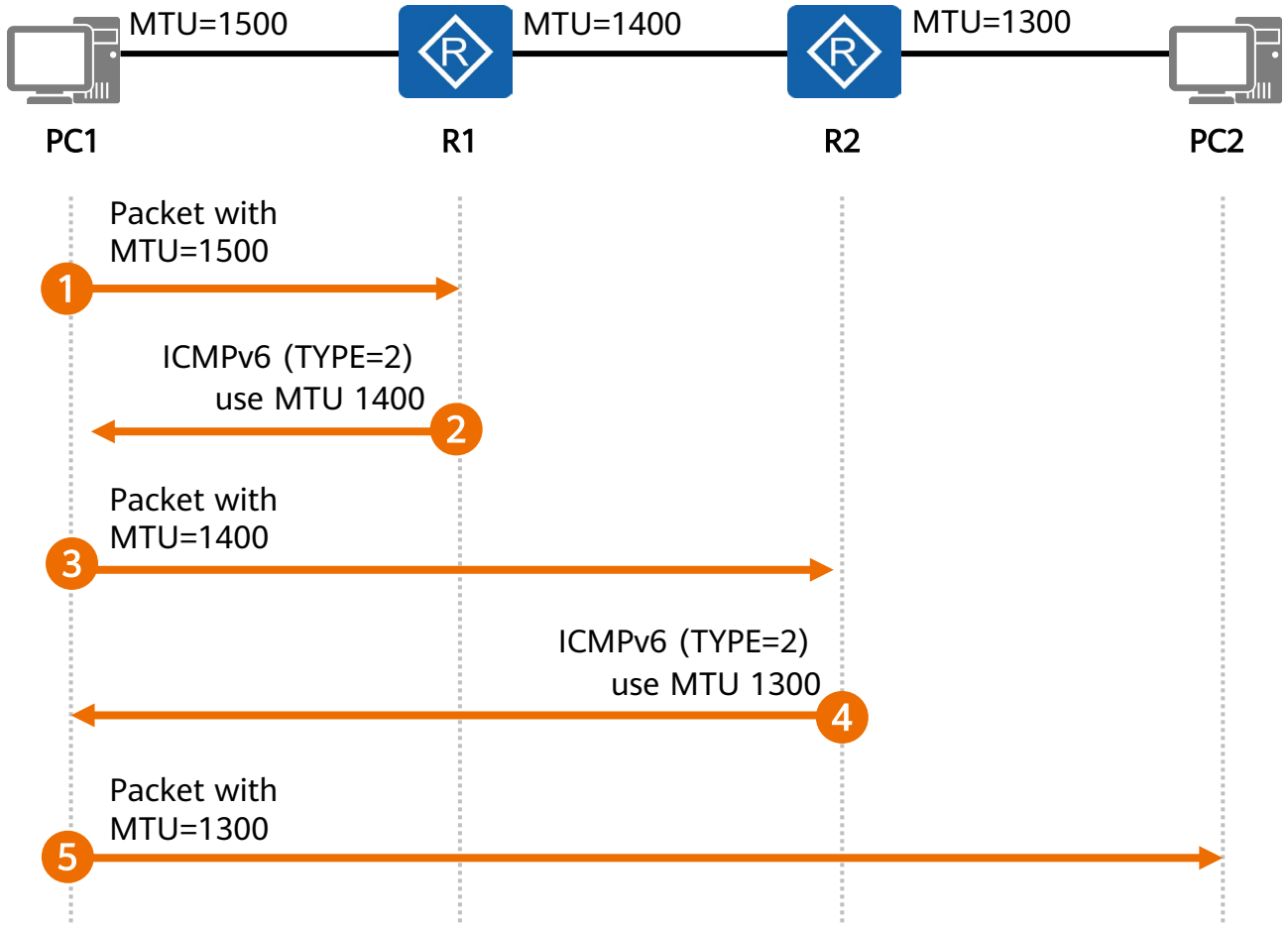
- **-a** *source-ipv6-address*
  指定发送ICMPv6 Echo Request报文的源IPv6地址。
- **-c** *count*
  指定发送ICMPv6 Echo Request报文次数，缺省为5次。
- **-s** *packetsize*
  指定ICMPv6 Echo Request报文长度（不包括IPv6基本头部和ICMPv6报文头部），缺省为56字节。
- **-t** *timeout*
  指定发送完ICMPv6 ECHO-REQUEST后，等待ICMPv6 ECHO-REPLY的超时时间，缺省为2秒。
- **-i** *interface-type interface-number*
  指定发送ICMP Echo Request报文的出接口。

……

ICMPv6信息报文应用 – Ping命令在华为设备上的执行（续）

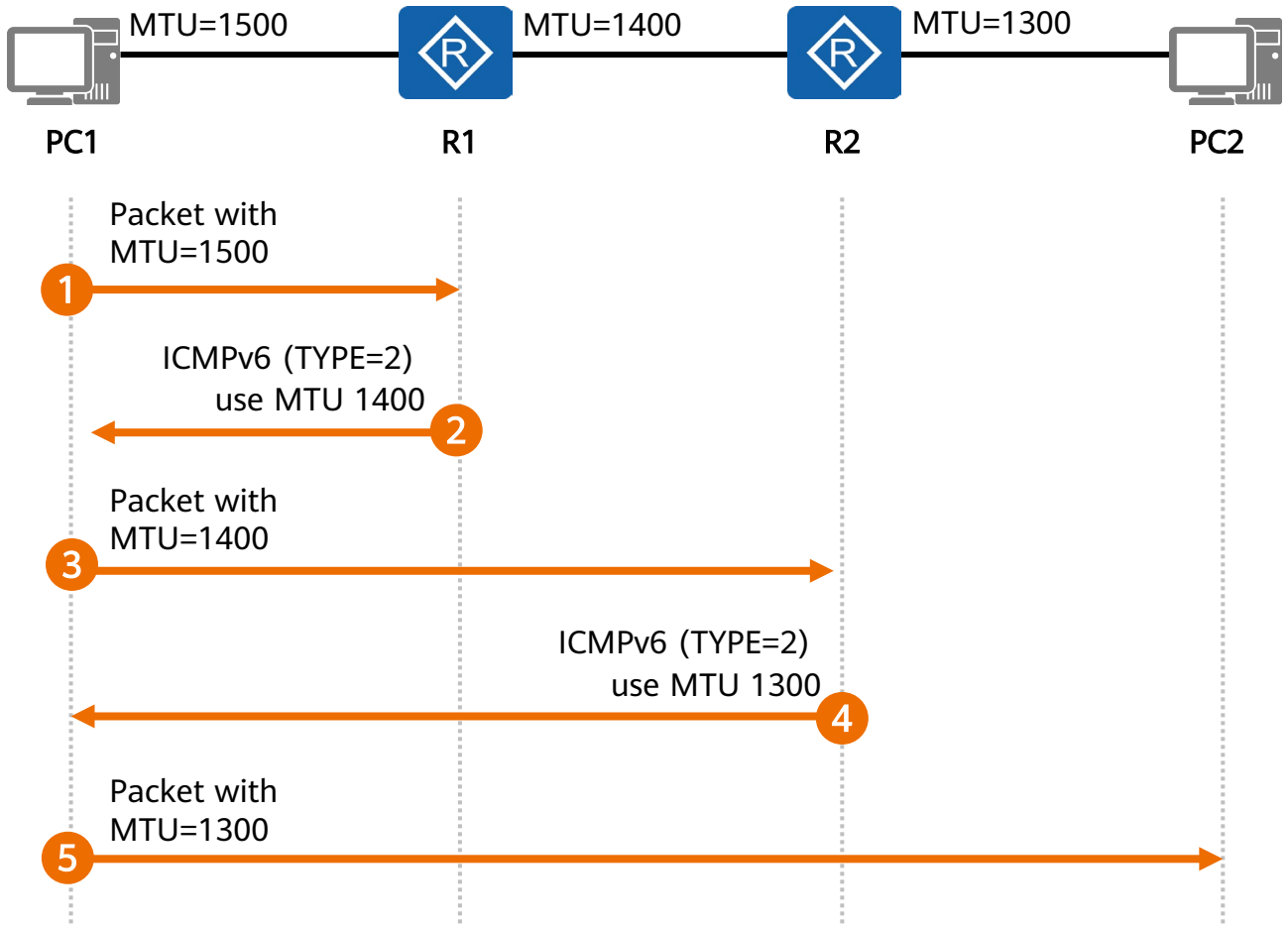
- 使用**ping ipv6**时，主要有以下几种场景：
  - 检查本机协议栈。执行**ping ipv6** <IPv6环回地址>，可以检查本机TCP/IP协议栈是否正常。
  - 在IPv6网络中检测目的IPv6主机是否可达。 执行**ping ipv6** *host*，向对端发送ICMPv6 ECHO-REQUEST报文，如果能够收到对端应答（reply），则可以判定对端路由可达。
  - 网络环境较差时，通过**ping ipv6 -c** *count* **-t** *timeout host*命令可以检测本端到对端设备间的网络质量。通过分析显示结果中的丢包率和平均时延，可以评估网络质量。对于可靠性较差的网络，建议发包次数（-c）和超时时间（-t）取较大值，这样可以更加准确的得到检测信息。

ICMPv6差错报文应用：Path MTU发现概述



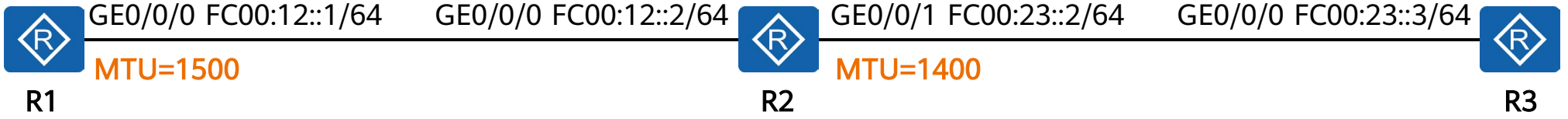
- 在IPv6中，为了减少中间转发设备的处理压力，中间转发设备不对IPv6报文进行分片，报文的分片将在源节点进行。
- Path MTU是路径上的最小接口MTU。
- PMTUD（Path MTU发现机制）的主要目的是发现路径上的MTU，当数据包被从源转发到目的地的过程中避免分段。
- PMTUD是通过ICMPv6的Packet Too Big报文来完成的。

ICMPv6差错报文应用：Path MTU发现详解



1. 首先PC1用1500字节作为MTU向PC2发送IPv6数据包。
2. R1意识到数据包过大，出接口MTU为1400字节，于是回复一个ICMPv6（Type=2）报文给PC1，指定MTU值为1400字节。
3. 然后，PC1开始使用1400作为MTU发送IPv6数据。
4. 数据包到达R2后，R2意识到出站接口MTU为1300字节，于是发送一个ICMPv6（Type=2）报文给PC1，指定MTU值为1300字节。
5. PC1开始使用1300作为MTU发送IPv6数据。

ICMPv6差错报文应用：Path MTU发现机制验证



R2的配置修改如下：

```
[R2] interface GigabitEthernet 0/0/1
[R2-GigabitEthernet0/0/1] ipv6 mtu 1400
```

```
<R2> display ipv6 interface GigabitEthernet 0/0/1
......
IPv6 is enabled, link-local address is FE80::2E0:FCFF:FEB3:4691
  Global unicast address(es):
    FC00:23::2, subnet is FC00:23::/64
  Joined group address(es):
    FF02::1:FF00:2
    FF02::2
    FF02::1
    FF02::1:FFB3:4691
  MTU is 1400 bytes
......
```

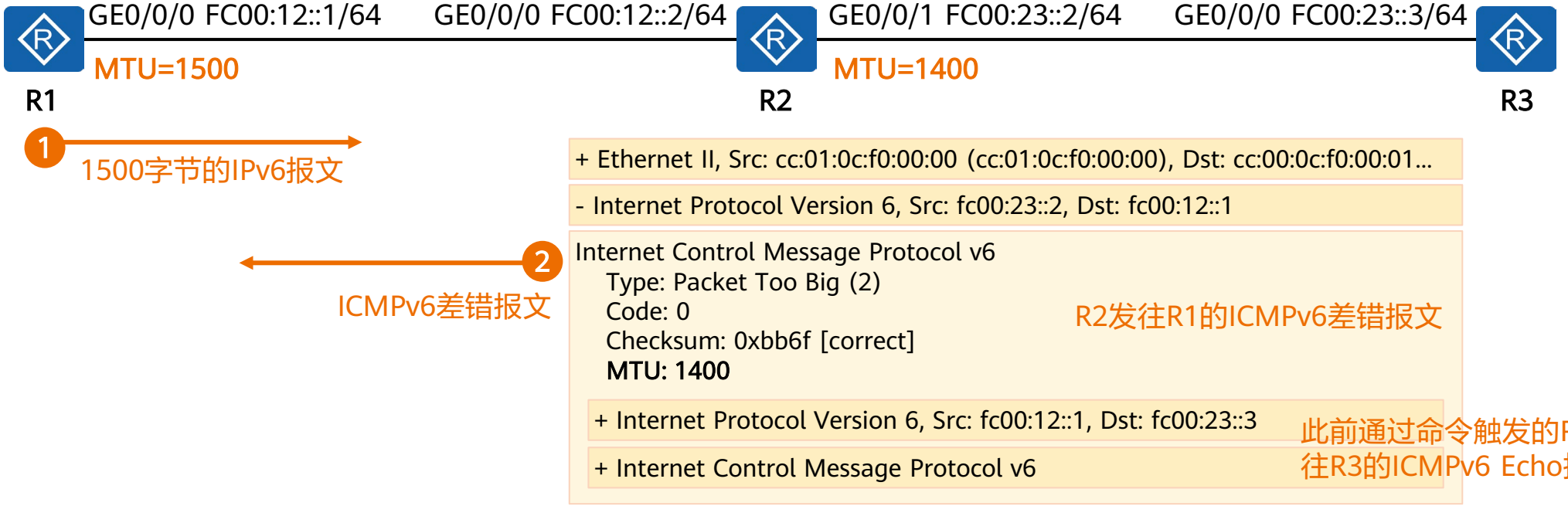
修改R2接口MTU值后，让R1产生一个1500字节的IPv6报文：

```
<R1> ping ipv6 -s 1452 FC00:23::3
```

执行上述命令时产生的ICMP request报文载荷为1452字节，加上8字节的ICMPv6头部，以及40字节的IPv6基本头部，正好1500字节。

R1将该ICMPv6 Echo request报文发送给R2。

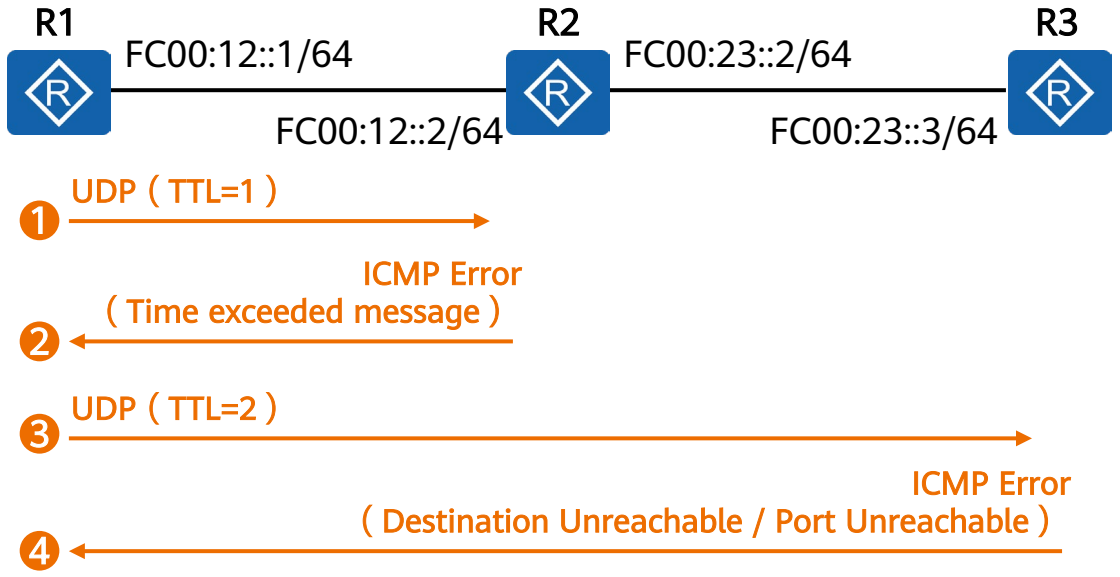
ICMPv6差错报文应用：Path MTU发现机制验证（续）



3

```
<R1> display ipv6 pathmtu all
IPv6 Destination Address        ZoneID    PathMTU        LifeTime(M)          Type       FF
FC00:23::3                      0         1400           7                    Dynamic    No
-------------------------------------------------------------------------------------------
Total: 1      Dynamic: 1      Static: 0
```

ICMPv6差错报文应用：Tracert




```
<R1> tracert ipv6 FC00:23::3

 traceroute to fc00:23::3  30 hops max,60 bytes packet

 1 FC00:12::2 20 ms  20 ms  30 ms

 2 FC00:23::3 30 ms  40 ms  20 ms
```

应用：Tracert

Tracert基于IP报文头中的TTL值来逐跳跟踪报文的转发路径。Tracert是检测网络丢包和时延的有效手段，也可帮助管理员发现网络中的路由环路。

ICMPv6差错报文应用：Tracert（观察相关报文：1/4）

Source	Destination	Protocol	Infor
fc00:12::1	fc00:23::3	UDP	30037 -> 33434
fc00:12::2	fc00:12::1	ICMPv6	Time Exceeded (Hop limit exceeded in transit)
fc00:12::1	fc00:23::3	UDP	30037 -> 33435
fc00:12::2	fc00:12::1	ICMPv6	Time Exceeded (Hop limit exceeded in transit)
fc00:12::1	fc00:23::3	UDP	30037 -> 33436
fc00:12::2	fc00:12::1	ICMPv6	Time Exceeded (Hop limit exceeded in transit)
fc00:12::1	fc00:23::3	UDP	30037 -> 33437
fc00:23::3	fc00:12::1	ICMPv6	Destination Unreachable (Port unreachable)
fc00:12::1	fc00:23::3	UDP	30037 -> 33438
fc00:23::3	fc00:12::1	ICMPv6	Destination Unreachable (Port unreachable)
...	...	...	...

- Internet Protocol Version 6, Src: fc00:12::1, Dst: fc00:23::3

```
  0110 .... = Version: 6
> .... 0000 0000 .... .... .... .... .... = Traffic Class: 0x00 (DSCP: CS0, ECN: Not-ECT)
  .... 0000 0000 0000 0000 0000 = Flow Label: 0x00000
  Payload Length: 20
  Next Header: UDP (17)
  Hop Limit: 1
  Source Address: fc00:12::1
  Destination Address: fc00:23::3
```

+ User Datagram Protocol, Src Port:30037, Dst Port: 33434

R1首先发送去往目标地址FC00:23::3的UDP（UDP目的端口为特殊端口，端口不会被具体的应用所使用）报文，Hop Limit为1。

ICMPv6差错报文应用：Tracert（观察相关报文：2/4）

Source	Destination	Protocol	Infor
fc00:12::1	fc00:23::3	UDP	30037 -> 33434
fc00:12::2	fc00:12::1	ICMPv6	Time Exceeded (Hop limit exceeded in transit)
fc00:12::1	fc00:23::3	UDP	30037 -> 33435
fc00:12::2	fc00:12::1	ICMPv6	Time Exceeded (Hop limit exceeded in transit)
...	...	...	...

- Internet Protocol Version 6, Src: fc00:12::2, Dst: fc00:12::1

// 此处省略部分字段
Next Header: ICMPv6(58)
Hop Limit: 64
Source Address: fc00:12::2
Destination Address: fc00:12::1

- Internet Control Message Protocol v6

Type: Time Exceeded (3)
Code: 0 (Hop limit exceeded in transit)
checksum: 0x9368 [Correct]
Reserved: 00000000

Type=3，Code=0，即错误消息，且错误为TTL超时

+ Internet Protocol Version 6, Src: fc00:12::1, Dst: fc00:23::3

+ User Datagram Protocol, Src Port: 30037, Dst Port: 33434

R1此前发往R3的IPv6报文被封装在此处发回R1

+ Data

R2收到该报文后将Hop Limit字段值减1后发现值已为0，因此立即向R1发送ICMPv6错误消息，告知报文的生存时间截止，这个错误消息的源地址为R2的接口地址。此时R1便可获得第一跳设备的地址。

ICMPv6差错报文应用：Tracert（观察相关报文：3/4）

Source	Destination	Protocol	Infor
fc00:12::1	fc00:23::3	UDP	30037 -> 33434
fc00:12::2	fc00:12::1	ICMPv6	Time Exceeded (Hop limit exceeded in transit)
fc00:12::1	fc00:23::3	UDP	30037 -> 33435
fc00:12::2	fc00:12::1	ICMPv6	Time Exceeded (Hop limit exceeded in transit)
fc00:12::1	fc00:23::3	UDP	30037 -> 33436
fc00:12::2	fc00:12::1	ICMPv6	Time Exceeded (Hop limit exceeded in transit)
fc00:12::1	fc00:23::3	UDP	30037 -> 33437
fc00:23::3	fc00:12::1	ICMPv6	Destination Unreachable (Port unreachable)
fc00:12::1	fc00:23::3	UDP	30037 -> 33438
fc00:23::3	fc00:12::1	ICMPv6	Destination Unreachable (Port unreachable)
...	...	...	...

- Internet Protocol Version 6, Src: fc00:12::1, Dst: fc00:23::3

```
  0110 .... = Version: 6
> .... 0000 0000 .... .... .... .... .... = Traffic Class: 0x00 (DSCP: CS0, ECN: Not-ECT)
  .... 0000 0000 0000 0000 0000 = Flow Label: 0x00000
  Payload Length: 20
  Next Header: UDP (17)
  Hop Limit: 2
  Source Address: fc00:12::1
  Destination Address: fc00:23::3
```

+ User Datagram Protocol, Src Port:30037, Dst Port: 33437

R1继续发送Hop Limit为2的UDP报文。

ICMPv6差错报文应用：Tracert（观察相关报文：4/4）

Source	Destination	Protocol	Infor
...	...	...	...
fc00:12::1	fc00:23::3	UDP	30037 -> 33437
fc00:23::3	fc00:12::1	ICMPv6	Destination Unreachable (Port unreachable)
fc00:12::1	fc00:23::3	UDP	30037 -> 33438
fc00:23::3	fc00:12::1	ICMPv6	Destination Unreachable (Port unreachable)
...	...	...	...

- Internet Protocol Version 6, Src: fc00:23::3, Dst: fc00:12::1

// 此处省略部分字段
Next Header: ICMPv6(58)
Hop Limit: 64
Source Address: fc00:23::3
Destination Address: fc00:12::1

- Internet Control Message Protocol v6

Type: Destination Unreachable (1)
Code: 4 (Port Unreachable)
checksum: 0x9552 [Correct]
Reserved: 00000000

Type=1，Code=4，即错误消息，且错误为目的端口不可达

+ Internet Protocol Version 6, Src: fc00:12::1, Dst: fc00:23::3

+ User Datagram Protocol, Src Port: 30037, Dst Port: 33437

+ Data

该UDP报文到达目的节点R3后，R3发现目的UDP端口在本地并未侦听，因此回复ICMPv6报错消息。此时R1便知道报文已达终点。

目录

NDP概述

- 邻居发现协议NDP（Neighbor Discovery Protocol）是IPv6协议体系中一个重要的基础协议。
- 邻居发现协议替代了IPv4的ARP（Address Resolution Protocol）和ICMP路由器发现（Router Discovery），它定义了使用ICMPv6报文实现地址解析，邻居不可达性检测，重复地址检测，路由器发现，重定向以及ND代理等功能。

NDP的主要功能

NDP		
	路由器发现	发现链路上的路由器，获得路由器通告的信息。
	无状态地址自动配置	通过路由器通告的地址前缀，终端自动生成IPv6地址。
	重复地址检测	获得地址后进行地址重复检测，确保地址不存在冲突。
	地址解析	请求目的网络地址对应的数据链路层地址，类似IPv4的ARP。
	邻居状态跟踪	发现链路上的邻居并跟踪邻居状态。
	前缀重编址	路由器对所通告的地址前缀进行灵活设置，实现网络重编址。
	重定向	告知其他设备，到达目标网络的更优下一跳。

NDP报文类型及功能

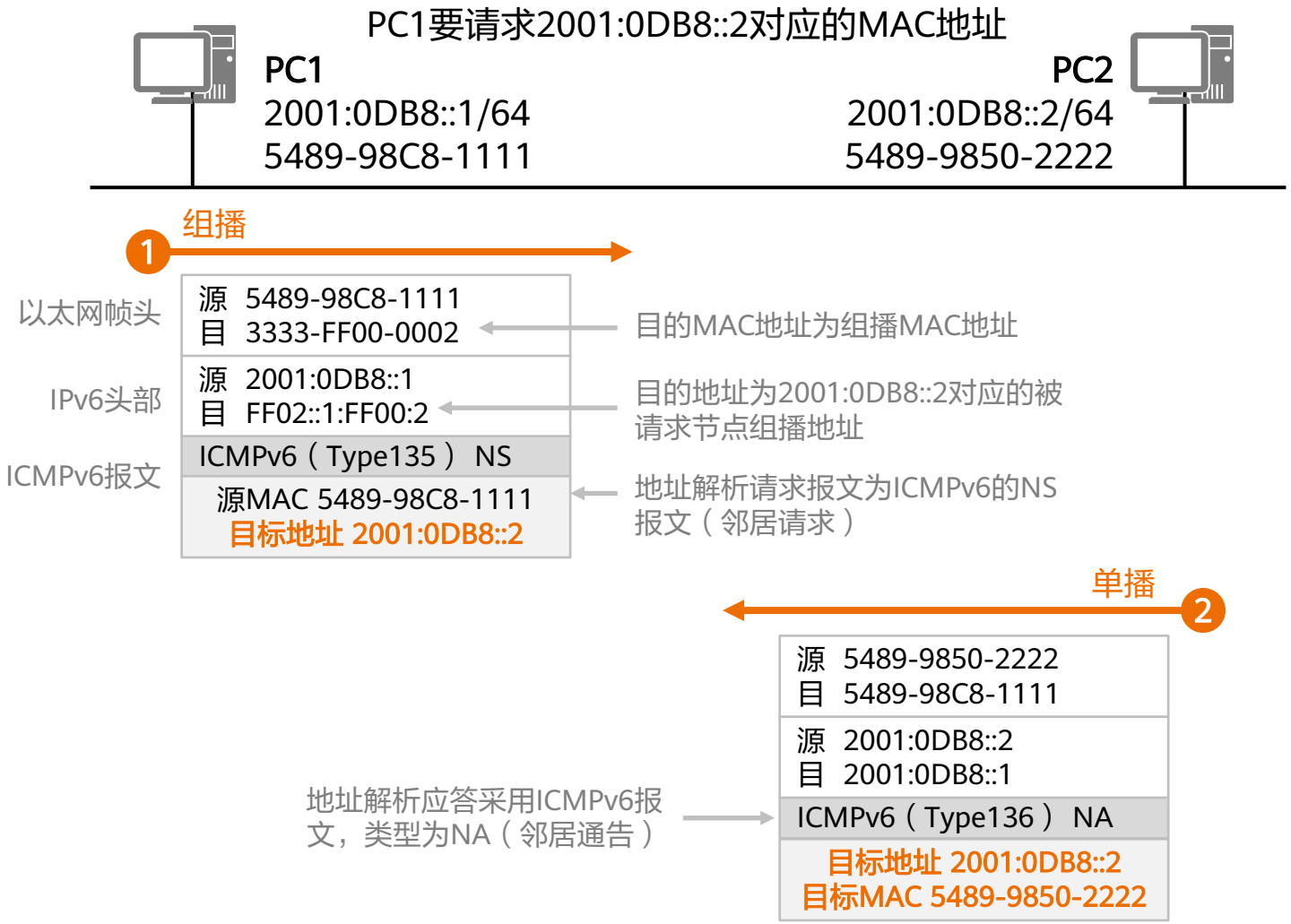
NDP使用以下几种ICMPv6报文：

- RS（Router Solicitation）：路由器请求报文
- RA（Router Advertisement）：路由器通告报文
- NS（Neighbor Solicitation）：邻居请求报文
- NA（Neighbor Advertisement）：邻居通告报文
- Redirect：重定向报文

功能 \ ICMPv6报文	RS 133	RA 134	NS 135	NA 136	重定向 137
地址解析			●	●	
路由器发现	●	●			
前缀重编址	●	●			
重复地址检测			●	●	
重定向					●

NDP实现的主要功能与所使用的ICMPv6报文（表格中报文后的数字表示ICMPv6的Type字段值）

地址解析



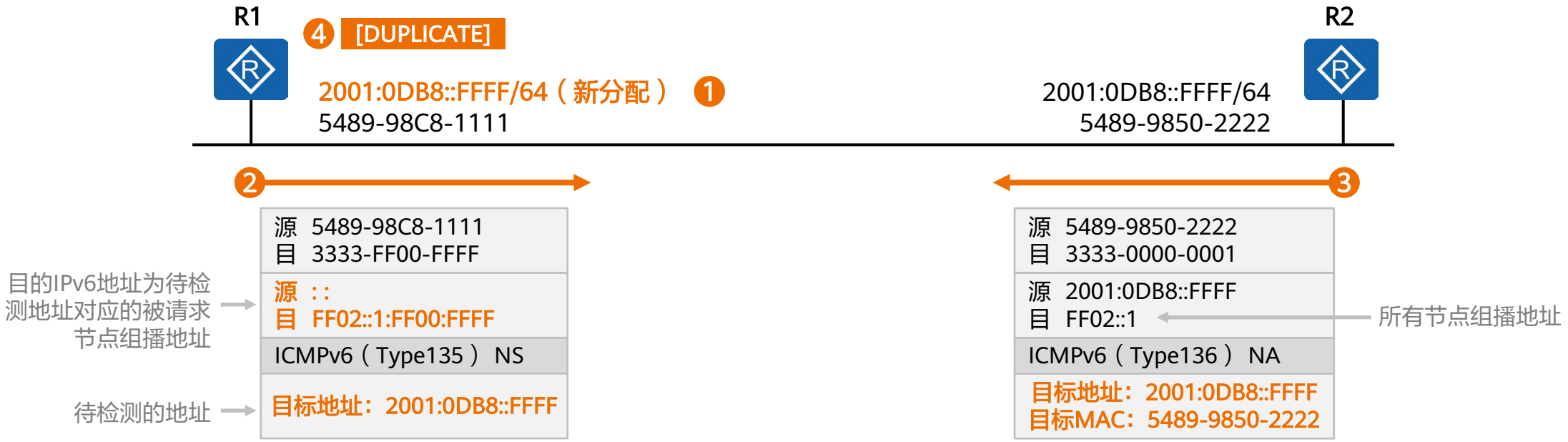
- 在IPv4中，当主机需要和目标主机通信时，必须先通过ARP协议获得目的主机的链路层地址。
- 在IPv6中，同样需要从IP地址解析到链路层地址的功能，NDP实现了这个功能。

地址解析

- 在IPv4中，ARP报文是直接封装在以太网报文中，以太网协议类型为0x0806，普遍观点认为ARP定位为第2.5层的协议。
- NDP基于ICMPv6实现，数据帧头部中的以太网协议类型为0x86DD，即IPv6报文，IPv6下一个报头字段值为58，表示ICMPv6报文，由于NDP使用的报文均为ICMPv6报文，一般来说，其被看作第3层的协议。
- 在三层完成地址解析，主要带来以下几个好处：
  - 地址解析在三层完成，不同的二层介质可以采用相同的地址解析协议。
  - 可以使用三层的安全机制避免地址解析攻击。
  - 使用组播方式发送请求报文，减少了设备性能压力。

重复地址检测

- 重复地址检测（Duplicate Address Detect，DAD）是指接口在使用某个单播IPv6地址之前，需要先探测是否有其它的节点使用了该地址，从而确保该地址未被其他节点占用。
- 接口在启用任何一个单播IPv6地址前都需要先进行DAD，包括Link-Local地址。



重复地址检测（续）

- 一个IPv6单播地址在分配给一个接口之后且通过重复地址检测之前称为试验地址（Tentative Address）。此时该接口不能使用这个试验地址进行单播通信，但是仍然会加入两个组播组：ALL-NODES组播组和试验地址所对应的Solicited-Node组播组。
- IPv6重复地址检测技术和IPv4中的免费ARP类似：节点向试验地址所对应的Solicited-Node组播组发送NS报文。NS报文中目标地址即为该试验地址。如果收到某个其他站点回应的NA报文，就证明该地址已被网络上使用，节点将不能使用该试验地址通讯。

IPv6邻居表

- IPv6邻居表是设备维护的一张体现网络中其他IPv6邻居节点信息的表。
- IPv6邻居表中缓存了IPv6地址与MAC地址的映射关系，在华为网络设备上可以通过**display ipv6 neighbors**命令来查看IPv6邻居表。

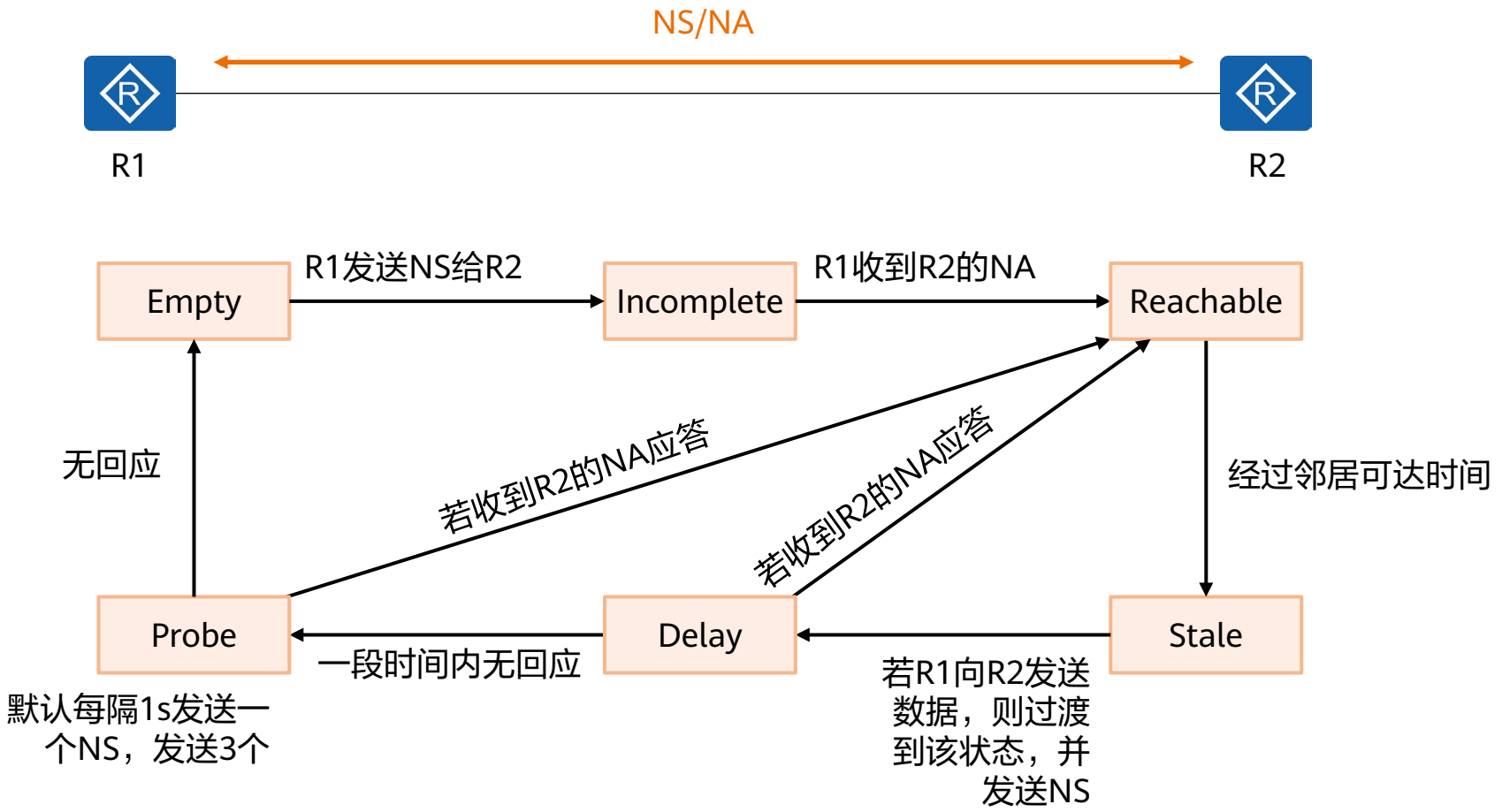
```
<Huawei> display ipv6 neighbors
------------------------------------------------------------------------------
IPv6 Address     : 2001:DB8::2
Link-layer       : 00e0-fc23-26e3                State : REACH
Interface        : GE0/0/0                       Age   : 0
VLAN             : -                             CEVLAN: -
VPN name         :                               Is Router: TRUE
Secure FLAG      : UN-SECURE

IPv6 Address     : FE80::2E0:FCFF:FE23:26E3
Link-layer       : 00e0-fc23-26e3                State : REACH
Interface        : GE0/0/0                       Age   : 0
VLAN             : -                             CEVLAN: -
VPN name         :                               Is Router: TRUE
Secure FLAG      : UN-SECURE

------------------------------------------------------------------------------
Total: 2        Dynamic: 2        Static: 0
```

IPv6邻居状态

- 通过邻居或到达邻居的通信，会因各种原因而中断，包括硬件故障等。如果目的地失效，则恢复是不可能的，通信失败；如果路径失效，则恢复是可能的。 因此节点需要维护一张邻居表，每个邻居都有相应的状态，状态之间可以迁移。
- 邻居状态有5种，分别是：未完成（Incomplete）、可达（Reachable）、陈旧（Stale）、延迟（Delay）、探查（Probe）。

状态	描述
Incomplete	邻居不可达。正在进行地址解析，邻居的数据链路层地址未探测到，如果解析成功，则进入Reachable状态
Reachable	邻居可达。表示在规定时间（邻居可达时间，缺省为30秒）内邻居可达。如果超过规定时间该表项没有被使用，则进入Stale状态
Stale	邻居是否可达未知。表明该表项在规定时间（邻居可达时间）内没有被使用。此时除非有发送到邻居的报文，否则不对邻居是否可达进行探测
Delay	邻居是否可达未知。已向邻居发送报文，如果在指定时间内没有收到响应，则进入Probe状态
Probe	邻居是否可达未知。已向邻居发送NS报文，探测邻居是否可达。在规定时间内收到NA报文回复，则进入Reachable状态；否则进入Incomplete状态

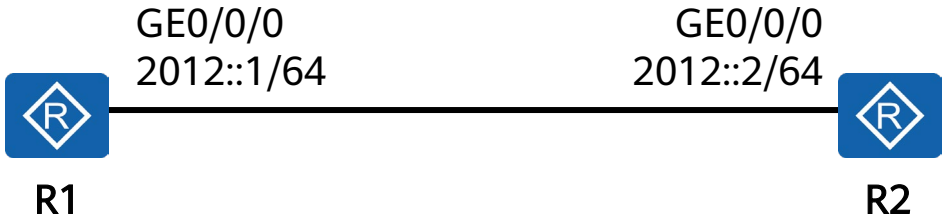
IPv6邻居状态迁移



以上为R1的邻居表中关于R2的状态

IPv6邻居状态迁移过程（1/5）

环境描述



在R1上打开IPv6 ND的Debug开关：

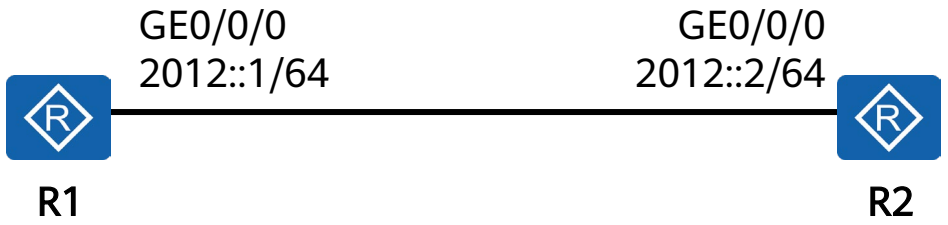
```
debugging ipv6 nd
terminal monitor
terminal debugging
```

测试过程

1. 在R1上ping ipv6 2012::2，这个动作会触发R1发送NS去请求R2的数据链路层信息。R1将在其IPv6邻居表中为2012::2创建一个表项，在收到R2的NA答复后，将该表项的状态从Incomplete切换到Reachable。
2. R1及R2之间的通信闲置30s后，该表项的状态从Reachable切换为Stale。
3. 此时当R1再ping 2012::2时，该表项的状态从Stale切换为Delay，并发送NS给R2，一段固定时间后（5s），状态从Delay切换到Probe，在此期间，如果收到R2的NA，则切换到Reachable。
4. 在R2的GE0/0/0接口上部署ACL，过滤所有IPv6报文来模拟R2故障的情况。当表项的状态处于Stale时，如果R1发送数据给R2，那么会将表项的状态切换为Delay，并发送NS，一段时间后切换为Probe，并间隔1s发送NS，此时R2已经无法应答，最终表项状态变为Empty并被删除。

IPv6邻居状态迁移过程（2/5）

① Incomplete -> Reachable



在R1上ping ipv6 2012::2成功后，R1的IPv6邻居表如下。从右边Debug信息可以看到Incomplete到Reachable的邻居状态变化过程。

```
<R1>display ipv6 neighbors
------------------------------------------------------------------------
IPv6 Address : 2012::2
Link-layer   : 00e0-fca5-51aa          State : REACH
Interface    : GE0/0/0                 Age   : 0
VLAN         : -                       CEVLAN: -
VPN name     :                         Is Router: TRUE
Secure FLAG  : UN-SECURE
```

R1的Debug信息

Jul 13 2018 00:06:32.673.1-08:00 R1 ND/7/debug_ipv6 ND: Adding NB Entry: 2012::2 on GigabitEthernet0/0/0

Jul 13 2018 00:06:32.673.2-08:00 R1 ND/7/debug_ipv6 ND: Address Resolution started for 2012::2 on GigabitEthernet0/0/0

Jul 13 2018 00:06:32.673.3-08:00 R1 ND/7/debug_ipv6 ND: Sending NS to FF02::1:FF00:2, on the interface GigabitEthernet0/0/0

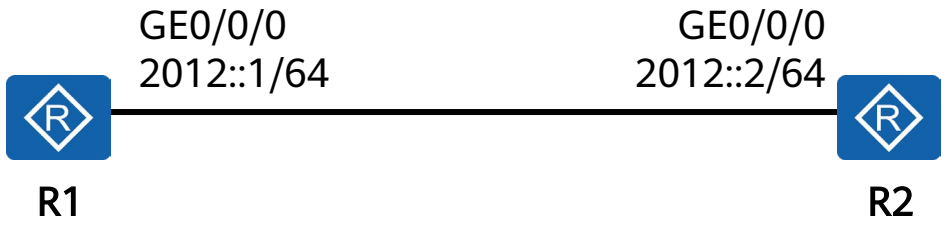
Jul 13 2018 00:06:32.683.1-08:00 R1 ND/7/debug_ipv6 ND: Received NA from 2012::2, on the interface GigabitEthernet0/0/0

Jul 13 2018 00:06:32.683.3-08:00 R1 ND/7/debug_ipv6 ND: Neighbor cache update 0000-0000-0000 -> 00e0-fca5-51aa : 2012::2 on GigabitEthernet0/0/0

Jul 13 2018 00:06:32.683.4-08:00 R1 ND/7/debug_ipv6 ND: INCOMPLETE->REACHABLE : 2012::2 on GigabitEthernet0/0/0

IPv6邻居状态迁移过程（3/5）

② Reachable -> Stale



R1及R2之间的通信闲置30s后，从Debug信息可以看到表项从Reachable到Stale的邻居状态变化过程。

此时R1的IPv6邻居表如下：

```
<R1>display ipv6 neighbors
------------------------------------------------------------------------------
IPv6 Address : 2012::2
Link-layer   : 00e0-fca5-51aa        State : STALE
Interface    : GE0/0/0               Age   : 0
VLAN         : -                     CEVLAN: -
VPN name     :                       Is Router: TRUE
Secure FLAG  : UN-SECURE
```

R1的Debug信息

Jul 13 2018 00:06:32.683.4-08:00 R1 ND/7/debug_ipv6 ND:
INCOMPLETE->REACHABLE : 2012::2 on GigabitEthernet0/0/0

.....

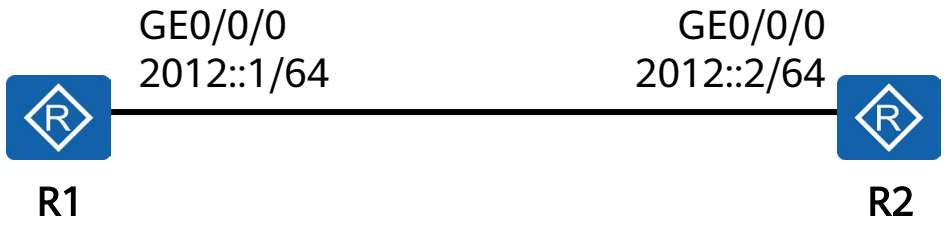
Jul 13 2018 00:07:03.673.1-08:00 R1 ND/7/debug_ipv6 ND:
REACHABLE->STALE : 2012::2 on GigabitEthernet0/0/0

.....

闲置30s

IPv6邻居状态迁移过程（4/5）

③ Stale -> Delay -> Probe -> Reachable



当R1的IPv6邻居表中关于R2的表项的状态为Stale时，如果R1发送数据给R2（例如ping ipv6 2012::2），可以观察到如右边所示的Debug信息。此时R1的邻居表如下：

```
<R1>display ipv6 neighbors
------------------------------------------------------------------------------
IPv6 Address : 2012::2
Link-layer   : 00e0-fca5-51aa        State : REACH
Interface    : GE0/0/0                       Age   : 0
VLAN         : -                             CEVLAN: -
VPN name     :                               Is Router:
TRUE
Secure FLAG  : UN-SECURE
```

R1的Debug信息

Jul 13 2018 00:07:54.363.1-08:00 R1 ND/7/debug_ipv6 ND: STALE->DELAY : 2012::2 on GigabitEthernet0/0/0

Jul 13 2018 00:07:59.703.1-08:00 R1 ND/7/debug_ipv6 ND: DELAY->PROBE : 2012::2 on GigabitEthernet0/0/0

Jul 13 2018 00:07:59.703.2-08:00 R1 ND/7/debug_ipv6 ND: Sending NS to 2012::2, on the interface GigabitEthernet0/0/0

Jul 13 2018 00:07:59.723.1-08:00 R1 ND/7/debug_ipv6 ND: Received NA from 2012::2, on the interface GigabitEthernet0/0/0

Jul 13 2018 00:07:59.723.3-08:00 R1 ND/7/debug_ipv6 ND: PROBE->REACHABLE : 2012::2 on GigabitEthernet0/0/0

IPv6邻居状态迁移过程（5/5）

④ Stale -> Delay -> Probe -> Empty

当R1的IPv6邻居表中，关于R2的表项的状态为Stale时，R1 ping ipv6 2012::2，可以观察到如右边所示的Debug信息。以下显示的是处于Probe状态的表项：

```
<R1>display ipv6 neighbors
------------------------------------------------------------------------------
IPv6 Address : 2012::2
Link-layer   : 00e0-fca5-51aa          State : PROBE
Interface    : GE0/0/0                 Age   : 0
VLAN         : -                       CEVLAN: -
VPN name     :                         Is Router: TRUE
Secure FLAG  : UN-SECURE
```

R1的Debug信息

Jul 13 2018 00:10:01.883.1-08:00 R1 ND/7/debug_ipv6 ND: STALE->DELAY : 2012::2 on GigabitEthernet0/0/0

Jul 13 2018 00:10:07.723.1-08:00 R1 ND/7/debug_ipv6 ND: DELAY->PROBE : 2012::2 on GigabitEthernet0/0/0

Jul 13 2018 00:10:07.723.2-08:00 R1 ND/7/debug_ipv6 ND: Sending NS to 2012::2, on the interface GigabitEthernet0/0/0

Jul 13 2018 00:10:08.723.1-08:00 R1 ND/7/debug_ipv6 ND: Sending NS to 2012::2, on the interface GigabitEthernet0/0/0

Jul 13 2018 00:10:09.723.1-08:00 R1 ND/7/debug_ipv6 ND: Sending NS to 2012::2, on the interface GigabitEthernet0/0/0

Jul 13 2018 00:10:10.723.1-08:00 R1 ND/7/debug_ipv6 ND: Address Resolution Failed for 2012::2 on GigabitEthernet0/0/0

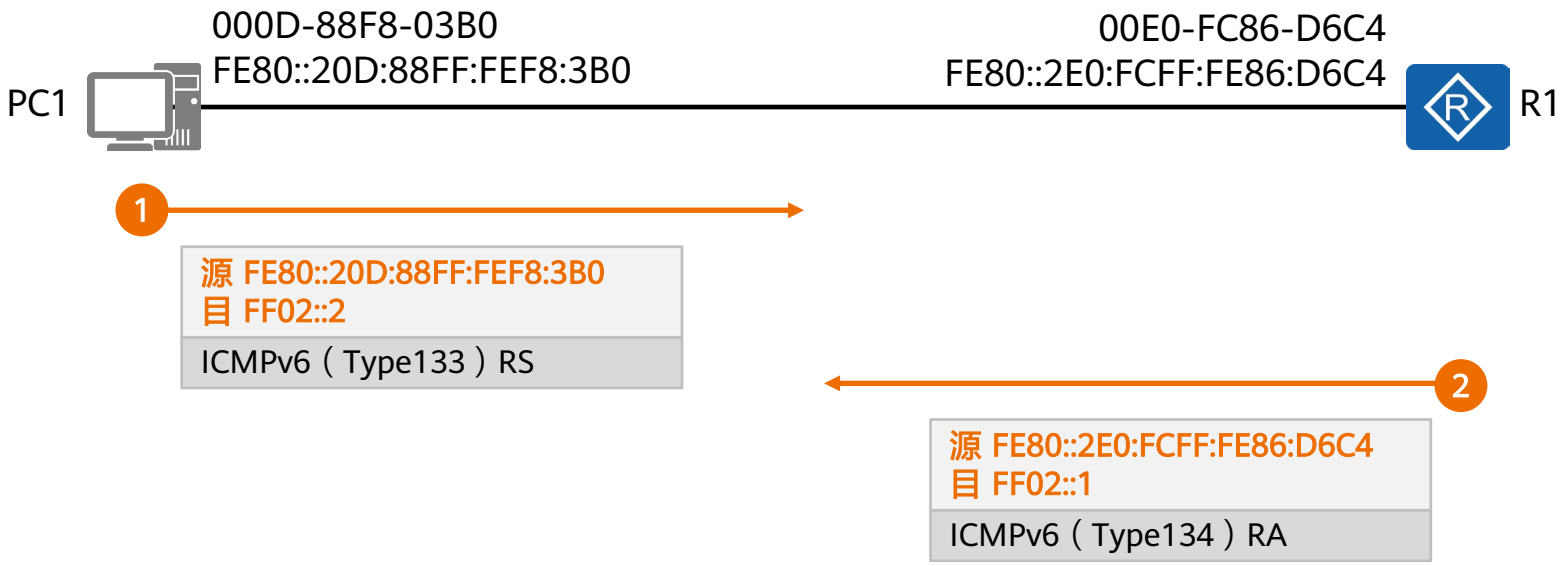
Jul 13 2018 00:10:10.723.2-08:00 R1 ND/7/debug_ipv6 ND: Deleting NB Entry: 2012::2 on GigabitEthernet0/0/0

路由器发现

- 路由器发现功能用来发现与本地链路相连的设备，并获取与地址自动配置相关的前缀和其他配置参数。
- IPv6支持无状态地址自动配置，即主机通过某种机制获取网络前缀信息，然后主机自己生成地址的接口标识部分。
- 路由器发现功能是IPv6地址自动配置功能的基础，主要通过以下两种报文实现：
  - **路由器通告RA（Router Advertisement）报文**：每台设备为了让二层网络上的主机和设备知道自己的存在，通常会周期性发送RA报文，RA报文中可携带IPv6网络前缀信息，及其他一些标志位信息。RA报文的Type字段值为134。
  - **路由器请求RS（Router Solicitation）报文**：很多情况下主机接入网络后希望尽快获取网络前缀，此时可以立刻发送RS报文，网络上的设备将回应RA报文。RS报文的Type字段值为133。

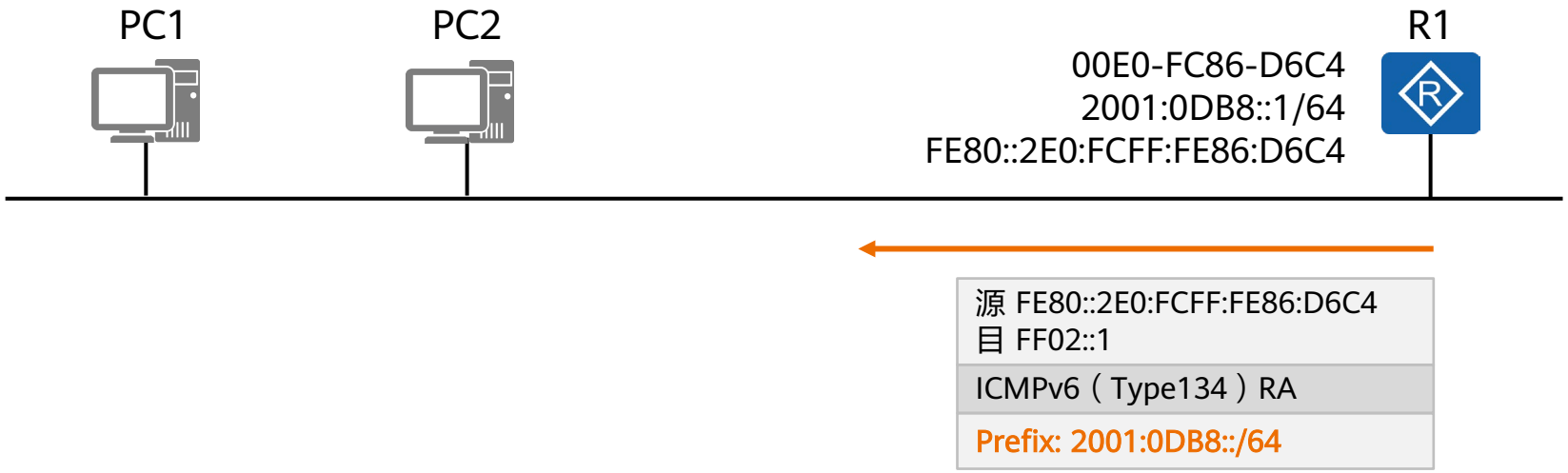
路由器发现流程 - 主机请求触发

当主机启动时，主机会向本地链路范围内所有的路由器发送RS报文，触发路由器响应RA报文。主机发现本地链路上的路由器后，自动配置缺省路由器，建立缺省路由表、前缀列表和设置其它的配置参数。



路由器发现流程 - 路由器周期性发送

- 路由器周期性发送RA报文，RA发送间隔是一个有范围的随机值，缺省的最大时间间隔是600秒，最小时间间隔是200秒（以S5700交换机为例）。
- 对于定期发送的RA报文，其地址有如下要求：
  - 源IPv6地址：必须是发送接口的链路本地地址。
  - 目的IPv6地址：FF02::1。

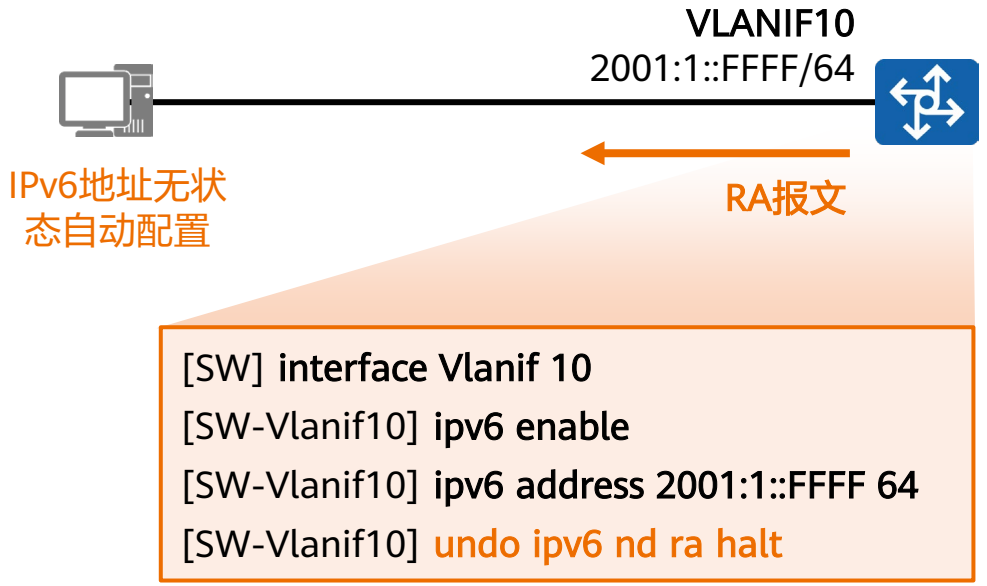


路由器发现与无状态地址自动配置概述

SLAAC的基本概念

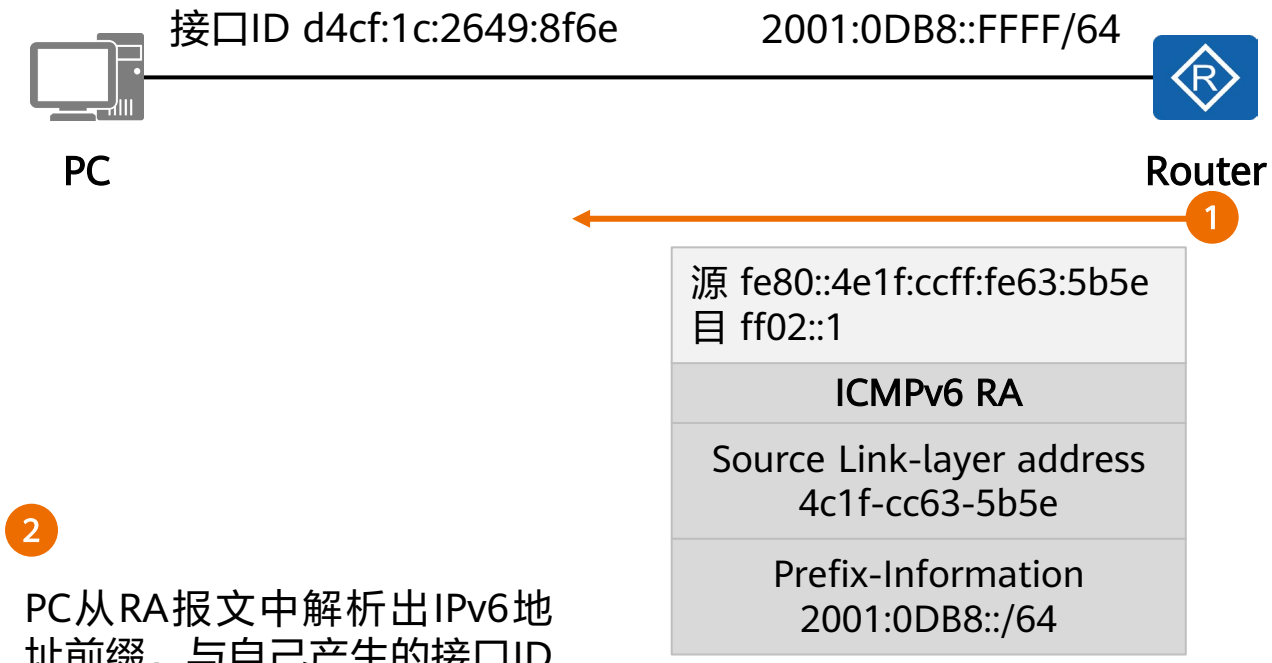
- IPv6地址无状态自动配置（StateLess Address Auto Configuration，SLAAC）在RFC2462（后被RFC4862废弃）中定义，是IPv6最有吸引力的新特性之一。
- 实现即插即用，终端的IPv6地址无需手工配置。
- 网络中无需部署额外的应用服务器（例如DHCP）。

在华为交换机上激活RA通告



接口视图下执行**undo ipv6 nd ra halt**，使能系统发布RA报文功能（周期性发送）。缺省情况下，发布RA报文功能处于未使能状态。

路由器发现与无状态地址自动配置过程




```
Internet Control Message Protocol v6
  Type: 134 (Router advertisement)
  Code: 0
  Checksum: 0x4a68 [correct]
  Cur hop limit: 64
⊟ Flags: 0x00
    0... .... = Not managed
    .0.. .... = Not other
    ..0. .... = Not Home Agent
    ...0 0... = Router preference: Medium
    .... .0.. = Not Proxied
  Router lifetime: 1800
  Reachable time: 0
  Retrans timer: 0
⊞ ICMPv6 Option (Source link-layer address)
⊞ ICMPv6 Option (MTU)
⊞ ICMPv6 Option (Prefix information)
```

2

PC从RA报文中解析出IPv6地址前缀，与自己产生的接口ID一起生产IPv6单播地址

2001:0DB8::d4cf:1c:2649:8f6e/64

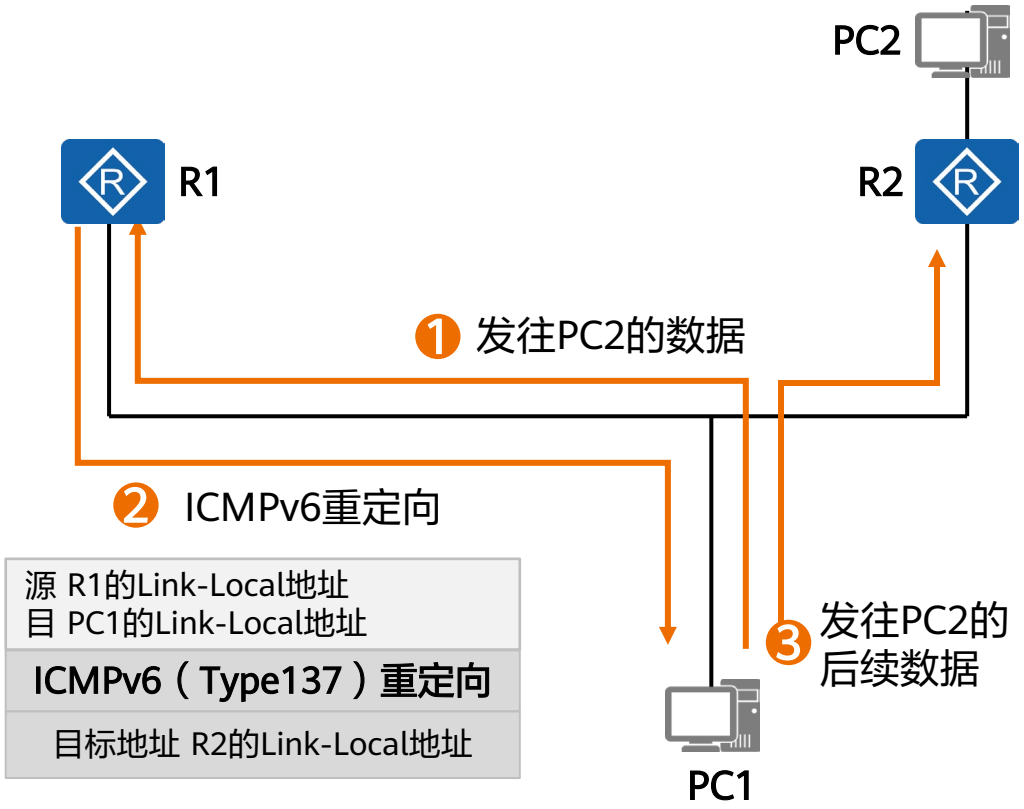
64bit Prefix　　64bit Interface ID

路由器发现与无状态地址自动配置过程详解

- IPv6主机无状态自动配置过程：
  1. 主机网卡激活IPv6后，根据接口标识产生链路本地地址。
  2. 主机对链路本地地址进行DAD。
  3. 如地址冲突，则停止地址自动配置过程，此时需要手工配置地址。
  4. 如不冲突，则链路本地地址生效，节点具备本地链路通信能力。
  5. 主机会以链路本地地址发送RS报文。
  6. IPv6路由器收到RS后，以RA回应，在该RA中，包含路由器的MAC地址、IPv6单播前缀信息等（RA不一定必须以RS为触发，即使没有收到RS，路由器也可以发送RA）。
  7. 主机根据RA报文中的前缀信息和接口标识得到IPv6地址。
  8. 主机对该地址进行DAD，检测通过后启用该地址。

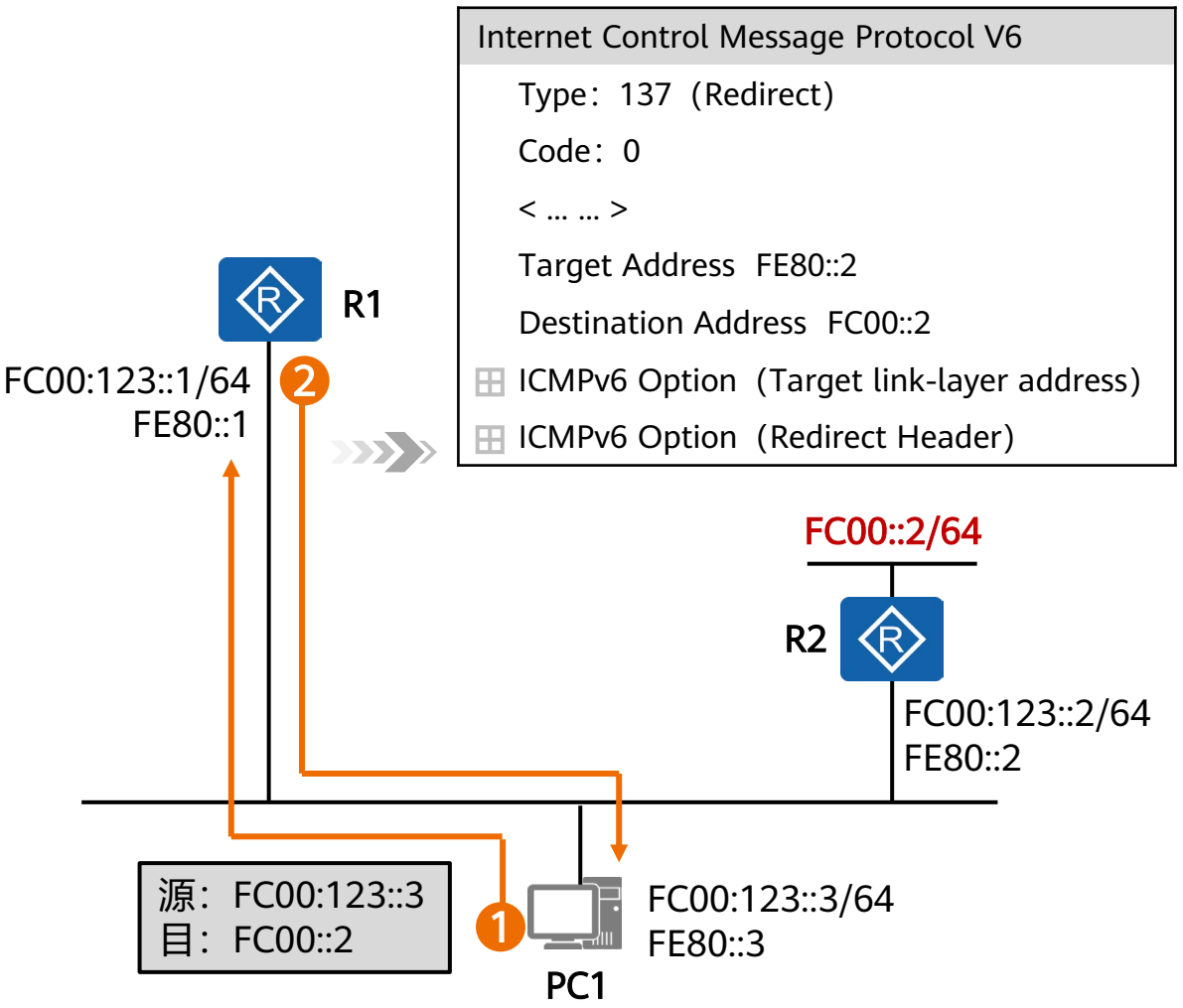
重定向

- 当网关设备发现报文从其它网关设备转发更好，它就会发送重定向报文告知报文的发送者，让报文发送者选择另一个网关设备。重定向报文也承载在ICMPv6报文中，其Type字段值为137，报文中会携带更好的路径下一跳地址和需要重定向转发的报文的目的地址等信息。



1. PC1需要和PC2通信，PC1的默认网关设备是R1，当PC1发送报文给PC2时报文会被送到R1。
2. R1接收到PC1发送的报文以后会发现PC1直接发送给R2更好，它将发送一个重定向报文给PC1，其中报文中更好的路径下一跳地址为R2。
3. PC1接收到了重定向报文之后，会在默认路由表中添加一个主机路由，以后发往PC2的报文就直接发送给R2。

重定向的实现



当设备收到一个报文后，只有在如下情况下会向报文发送者发送重定向报文：

- 报文的目的地址不是一个组播地址。
- 报文并非通过路由转发给设备。
- 经过路由计算后，路由的下一跳出接口是接收报文的接口。
- 设备发现报文的最佳下一跳IP地址和报文的源IP地址处于同一网段。
- 设备检查报文的源地址，发现自身的邻居表项中有用该地址作为全球单播地址或链路本地地址的邻居存在。

思考题

- （多选题）ICMPv6报文类型分为哪几大类?
  - A. 差错报文
  - B. 信息报文
  - C. 其他报文
  - D. 参数报文
- （多选题） IPv6地址解析通过以下哪种报文实现?
  - A. RS
  - B. RA
  - C. NS
  - D. NA

课程总结

- ICMPv6是IPv6的基础协议之一，具有差错报文和信息报文两种，用于IPv6节点报告报文处理过程中的错误和信息。其中，差错报文用于报告在转发IPv6数据包过程中出现的错误，如常见的目的不可达、超时等等；信息报文则可以实现路由器发现，重复地址检测，组播成员管理等等。
- NDP是5个ICMPv6信息报文的“打包”，Type133、134为RS、RA报文，可以实现路由器发现，实现主机的网关发现，地址的自动配置；Type135、136为NS、NA报文，可以实现邻居链路层地址解析，重复地址检测；Type137为重定向报文，可以实现重定向。

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home and organization for a fully connected, intelligent world.

HUAWEI