

# 计算机网络

## 实验七 IPv6隧道

---

谢瑞桃

[xie@szu.edu.cn](mailto:xie@szu.edu.cn)

[rtxie.github.io](https://github.com/rtxie)

计算机与软件学院  
深圳大学



# 实验七 IPv6隧道

## ■ 实验目的

- 学习安装与使用华为eNSP网络仿真软件
- 理解IPv6 over IPv4的原理
- 掌握IPv6 over IPv4手工隧道的配置方法
- 掌握OSPF路由的配置方法
- 掌握IPv6静态路由的配置方法

## ■ 实验环境

- Windows系统
- eNSP网络仿真软件



# 实验内容

1. 建立三路由拓扑
2. OSPF路由配置
3. 创建虚接口
4. 创建IPv6虚接口
5. 创建IPv6 over IPv4隧道
6. 配置IPv6静态路由



# 实验任务要求

- 请参考本讲义学习eNSP软件的使用方法
- 理解IPv6 over IPv4的原理
- 掌握IPv6 over IPv4手工隧道配置的方法
- 掌握OSPF路由配置的方法
- 掌握IPv6静态路由的配置方法
- 掌握用eNSP仿真小型网络的技能
- 依照步骤完成实验内容1—6
- 对实验结果截图
- 撰写实验报告



# 实验报告撰写要求

- 使用教务处制作的实验报告模板
- 注意按进度填写实验时间和实验报告提交时间
- 填写模板中的每一部分
- 填写实验步骤时，做到条理清晰
- 注意截图清晰、美观
- 对于实验操作返回结果的解释为加分项，解释地越详细越好
- 禁止抄袭实验报告，抄袭的实验报告都记为0分

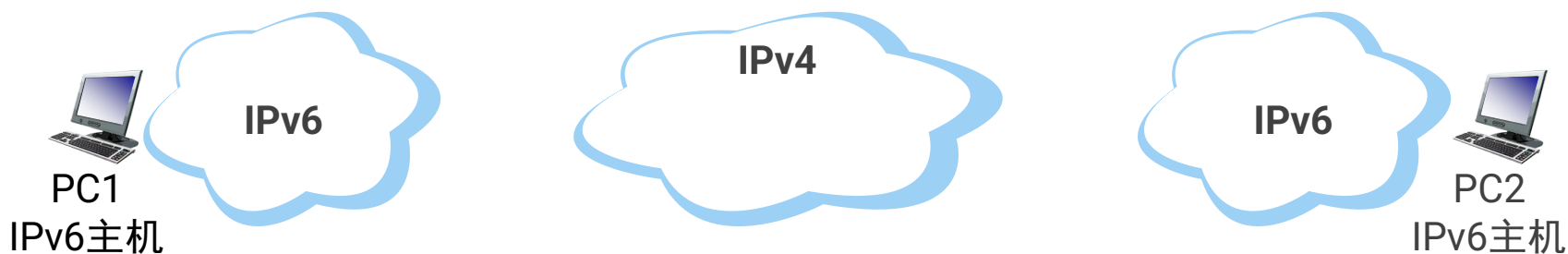


# 实验准备

1. IPv6 over IPv4隧道技术
2. IPv6 over IPv4手动隧道
3. 实验介绍
4. eNSP介绍
5. 关于eNSP的使用

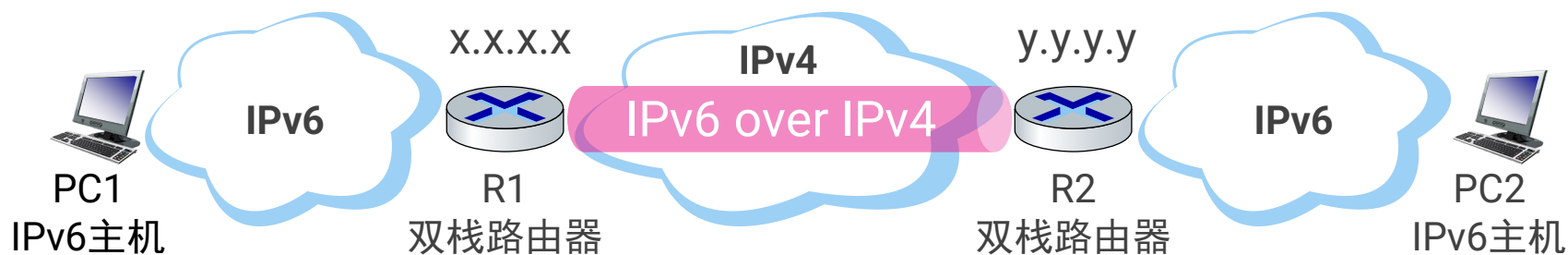
# 1. IPv6 over IPv4 隧道技术

- IPv6逐步过渡中，两个IPv6网络之间只有IPv4网络，两个IPv6网络能否利用已有的IPv4网络实现互联呢？



# 1. IPv6 over IPv4 隧道技术

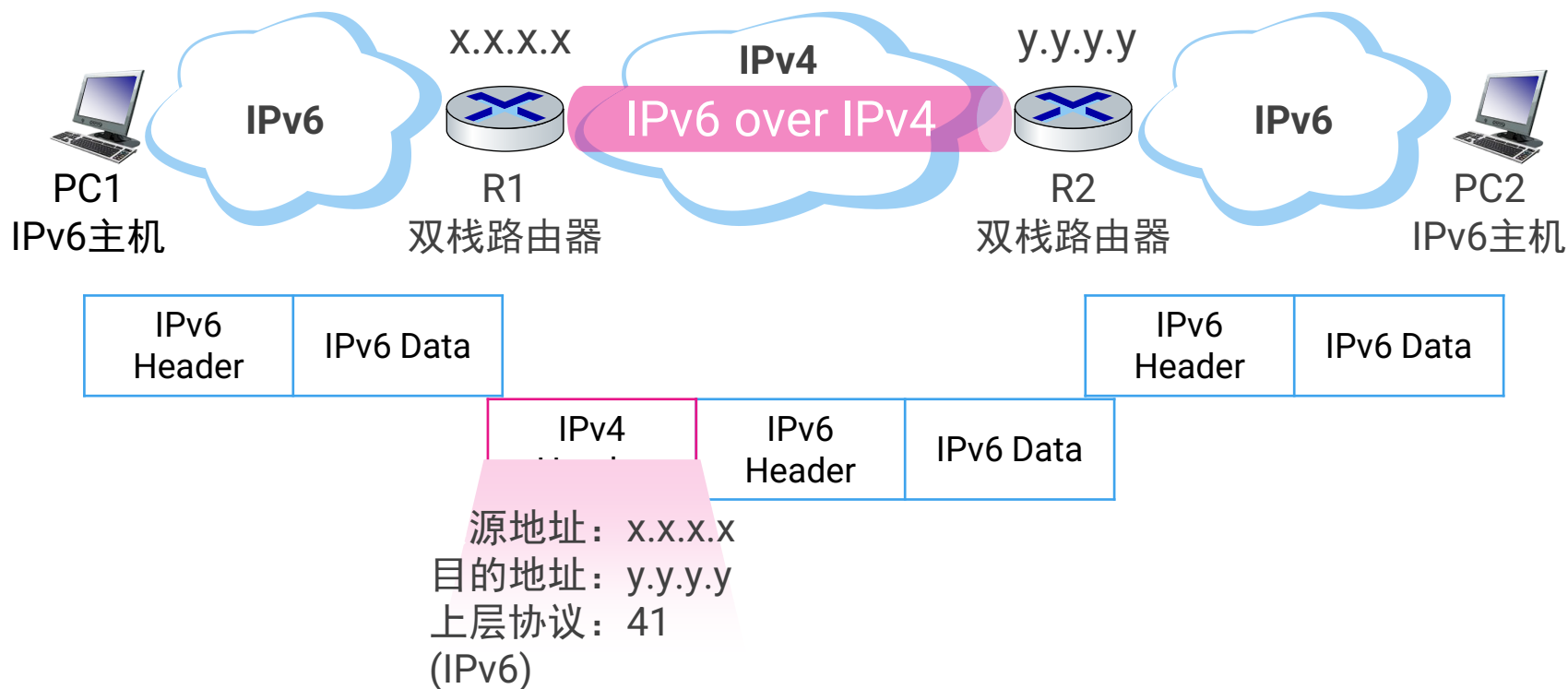
- 隧道技术
- 将IPv6数据报封装成IPv4数据报传输





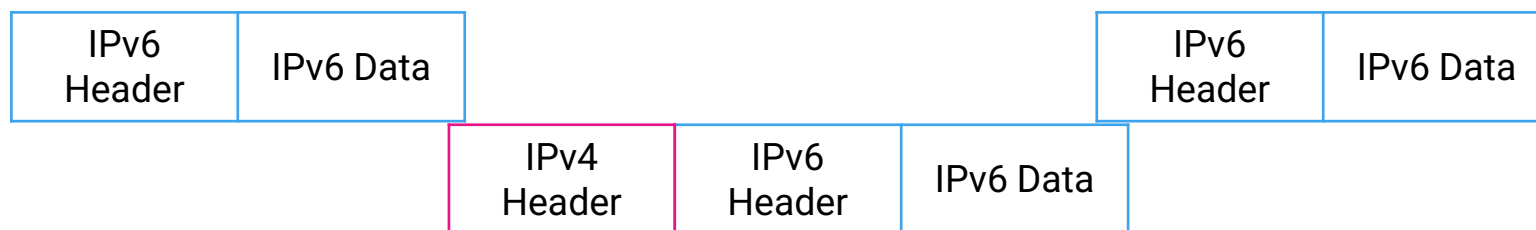
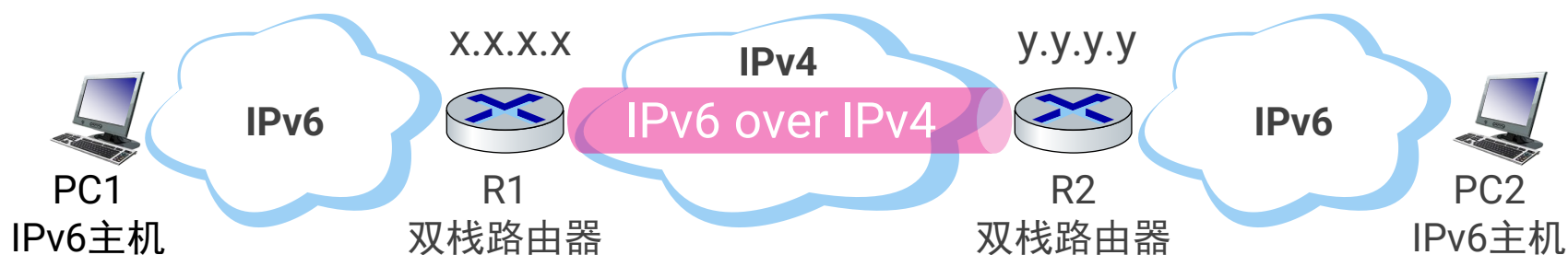
# 1. IPv6 over IPv4 隧道技术

- 隧道技术
- 将IPv6数据报封装成IPv4数据报传输



## 2. IPv6 over IPv4 手动隧道

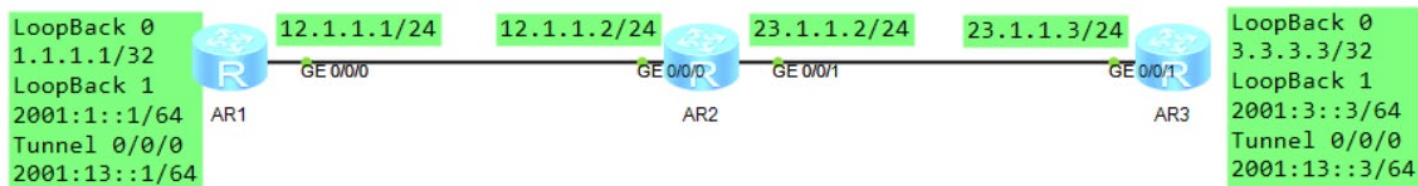
- 在隧道两端的边界路由设备上通过人工配置而创建的。
- 需要静态指定隧道的源IPv4地址和目的IPv4地址。
- 隧道两端的主机和路由设备均需要支持IPv4和IPv6协议栈。



源地址: x.x.x.x  
目的地址: y.y.y.y  
上层协议: 41 (IPv6)

# 3. 实验介绍

- 本实验要求实现IPv6 over IPv4的手工隧道配置。实验过程将包括：
  - 1) 搭建IPv4的网络
  - 2) 配置IPv4的路由
  - 3) 配置IPv6 over IPv4隧道
  - 4) 配置虚连接以测试隧道





## 4. eNSP介绍

- eNSP(Enterprise Network Simulation Platform)是一款由华为提供的免费的、可扩展的、图形化的网络设备仿真平台。
- 主要对企业网路由器、交换机、WLAN等设备进行软件仿真，支持大型网络模拟。
- 让你有机会在没有真实设备的情况下也能够开展实验测试，学习网络技术。



# 5. 关于eNSP的使用

- eNSP软件提供了软件使用帮助。
- 打开软件以后，点击右上角的帮助图标即可打开。
- “墙裂”建议同学们在实验之前浏览一下帮助里的一下内容（只需要十分钟）：

## 主界面和菜单

引导界面概述

主界面概述

主菜单

工具栏

网络设备区

设备接口区

## 快速入门

阅读本章有助于用户快速上手

新建拓扑

数据实验



## 5. 关于eNSP的使用

- 与前两次实验一样，所有指令都需在系统视图下操作。
- 双击路由器进入的是用户视图，请执行system-view后进入系统视图。
- 输入指令的时候，按Tab键可以补全。
- 加入新设备或设备连线之后，**一定要点击“开启设备”**。
- 在添加设备时，按ESC键即可退出。
- **保存设备的配置信息：在用户视图<Huawei>下输入save指令，并选择y选项。**

# 5. 关于eNSP的使用

- 如何解决AR设备启动失败问题
- 请按照帮助中提供的方法解决
- 注意，关闭防火墙以后重启主机。



The screenshot shows the eNSP Help window with the following content:

### 如何解决AR/WLAN设备启动失败问题

#### 现象描述

1. 启动AR设备之后，设备命令行无法接收输入，在长时间等待后一直输出“####”。
2. 启动AR/WLAN设备时，提示“...错误代码40...”。

#### 处理步骤

- 检查虚拟网卡设置。

1. 检查安装eNSP的PC上是否存在名为“VirtualBox Host-Only Network”的虚拟网卡。
  - 如果不存在，请[添加虚拟网卡](#)。
  - 如果存在，请[检查虚拟网卡的状态](#)。
  - 如果虚拟网卡的名称为“VirtualBox Host-Only Ethernet Adapter #2”，可尝试点击eNSP主界面“菜单-工具-注册设备”重新注册设备。如果还是出现这样的错误，请[检查虚拟网卡的状态](#)。
2. 检查虚拟网卡的状态。

查看名为“VirtualBox Host-Only Network”的虚拟网卡是否被禁用。

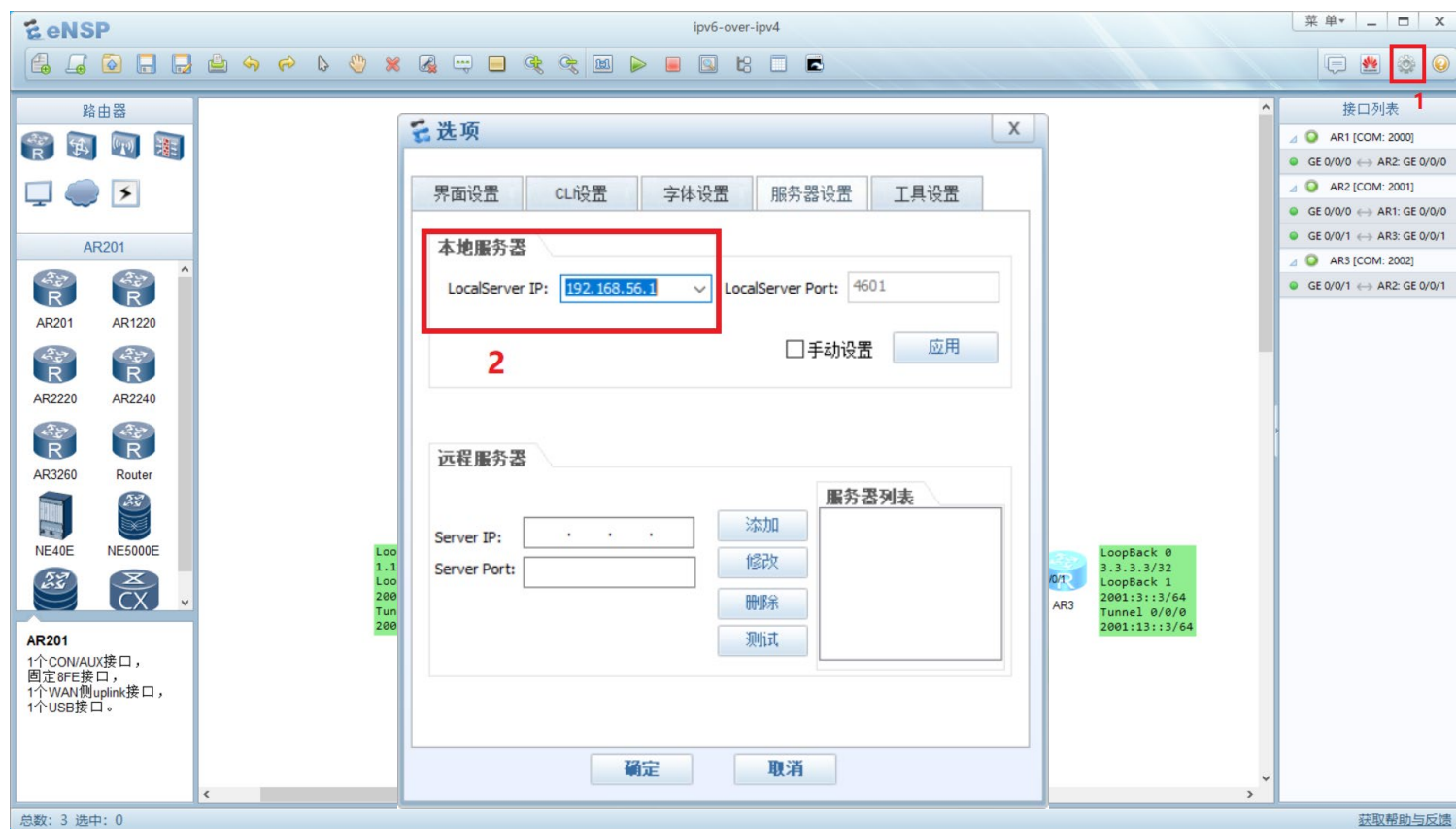


如果被禁用，请启用这个虚拟网卡。若重启设备仍然失败，请[检查虚拟网卡的参数设置](#)。

3. 检查虚拟网卡的参数设置。

# 5. 关于eNSP的使用

- 如何解决AR设备启动失败问题
- 如果还不行，按下图设置eNSP选项。







# 实验内容

1. 建立三路由拓扑
2. OSPF路由配置
3. 创建虚接口
4. 创建IPv6虚接口
5. 创建IPv6 over IPv4隧道
6. 配置IPv6静态路由



# 说明

- 所有命令的详细说明请参考华为官方出版的命令参考（2019-03-05发布的03版本），已上传在BB。
- 本讲义内的大部分命令后面都有一串带括号的数字，例如(9.4.70)，这指的是命令参考中的章节号。

# 1. 建立三路由器拓扑

- 1) 选用AR1220型号路由器。
- 2) 选择Copper型号线，即以太网线。
- 3) 按照下图所示的接口建立拓扑。
- 4) 为了便于分析，建议按照下图重命名路由器。

- `[Huawei]sysname R1`



# 1. 建立三路由器拓扑

## 5) 为每个路由器配置IPv4地址。

- 双击一个路由器就会弹出命令行界面，操作方法与实验5和6相同，举例如下。
- `[R1]interface GigabitEthernet 0/0/0`
- `[R1-GigabitEthernet0/0/0]ip address 12.1.1.1 255.255.255.0`



# 1. 建立三路由器拓扑

- 6) 用R1 ping R2的12.1.1.2，观察是否能通？想想为什么？
- 7) 用R1 ping R2的23.1.1.2，观察是否能通？想想为什么？
- 8) 用R1 ping R3的地址，观察是否能通？想想为什么？

- R1和R3应该是ping不通的，因为它们分属于两个不同的网络，同理，R1也ping不通R2的23.1.1.2。



# 1. 建立三路由器拓扑

9) 在R2执行 `ping 12.1.1.1`，观察是否能通？

10) 在R2执行 `ping 23.1.1.3`，观察是否能通？

- 都显示能ping通，这是不是和前一页的结果矛盾呢？
- 其实是不矛盾的。我们接下来分析一下原因。先看看ping指令的参数说明（20.16.1）。



# Ping指令的参数说明

## 命令格式

```
ping [ ip ] [ -a source-ip-address | -c count | -d | -h ttl-value | { -nextthop nexthop-ip-address |  
-i interface-type interface-number } * | -m time | -n | -name | -p pattern | -q | -r | { -s  
packet-size | -range [ min min-size | max max-size | step step-size ] * } | -system-time | -t  
timeout | -tos tos-value | -v | -vpn-instance vpn-instance-name | [ -f | ignore-mtu ] ] * host  
[ ip-forwarding ]
```

## 参数说明

参数	参数说明	取值
<b>ip</b>	使用IPv4协议。不指定该参数时，也表示使用IPv4协议。	-
<b>-a source-ip-address</b>	指定发送ICMP ECHO-REQUEST报文的源IP地址。  如果不指定源IP地址，将采用出接口的IP地址作为ICMP ECHO-REQUEST报文发送的源地址。	点分十进制形式。只能为A、B、C类合法地址。
<b>host</b>	目的主机的域名或IP地址。	字符串形式主机名，不支持空格，区分大小写，长度范围是1~255，当输入的字符串两端使用双引号时，可在字符串中输入空格。或者合法的点分十进制IPv4地址。

# 1. 建立三路由器拓扑

- 我们知道ping指令生成ICMP ECHO-REQUEST报文，并用IP协议封装。当我们在ping指令中不指定该报文的源IP地址时，按照ping指令的参数说明（20.16.1），IP源地址将采用出接口的IP地址。出接口也就是host目的地址的转发接口。
- 其实[R2]ping 12.1.1.1时，ICMP ECHO-REQUEST报文的源IP地址用的是12.1.1.2，因为路由表中网络12.1.1.0/24的出接口是GigabitEthernet 0/0/0。

11) 执行[R2]display ip routing看一下路由表中是否有这样的条目？





# 1. 建立三路由器拓扑

- 为了进一步验证这一点，我们可以用Wireshark抓包看一下[R2]ping 12.1.1.1产生的分组。

12) 选中AR2，右键→数据抓包，分别开启GE 0/0/0和GE 0/0/1两个接口的数据抓包，就会弹出两个Wireshark窗口。然后，重新执行[R2]ping 12.1.1.1，看看哪边接口的Wireshark抓包界面出现了ICMP协议分组？源地址是不是12.1.1.2呢？



# 1. 建立三路由器拓扑

13) 同理，重新执行 `[R2]ping 23.1.1.3`，看看哪边接口的 Wireshark 抓包界面出现了 ICMP 协议分组？源地址是不是 23.1.1.2 呢？



# 1. 建立三路由器拓扑

- 为了实现用R2的23.1.1.2 ping R1的12.1.1.1，需要用参数 -a 指定源IP地址

14) 执行 `[R2]ping -a 23.1.1.2 12.1.1.1`，观察是否能ping通，以及Wireshark里对应的ICMP分组。

15) 执行 `[R2]ping -a 12.1.1.2 23.1.1.3`，观察是否能ping通，以及Wireshark里对应的ICMP分组。

- 为了实现它们之间的互通，我们应该来配置路由。
- 请在我们开始配置之前，用下述指令打印路由表来看看。
- `[R1]display ip routing-table`



## 2. OSPF路由配置

1) ospf命令用来创建并运行OSPF进程。(9.4.70)

- `[R1]ospf 2`

2) 请先打印OSPF 2的路由表看看。(9.4.30)

- `[R1-ospf-2]display ospf 2 routing`

3) area命令用来创建OSPF区域，并进入OSPF区域视图。(9.4.2)

- `[R1-ospf-2]area 0`



## 2. OSPF路由配置

4) network命令用来指定运行OSPF协议的接口和接口所属的区域。(9.4.66)

- `[R1-ospf-2-area-0.0.0.0]network 12.1.1.0 0.0.0.255`

- 含义：指定运行OSPF协议的接口，主IP地址位于网段12.1.1.0/24，接口所在的Area ID为0。
- 注意：0.0.0.255是将网络地址12.1.1.0的掩码反转的结果（0变1,1变0），表示掩码长度是24位。



## 2. OSPF路由配置

5) 接下来，按照上述方法配置R2和R3。

- `[R2]ospf 2`
- `[R2-ospf-2]area 0`
- `[R2-ospf-2-area-0.0.0.0]network 12.1.1.0 0.0.0.255`
- `[R2-ospf-2-area-0.0.0.0]network 23.1.1.0 0.0.0.255`
- `[R3]ospf 2`
- `[R3-ospf-2]area 0`
- `[R3-ospf-2-area-0.0.0.0]network 23.1.1.0 0.0.0.255`



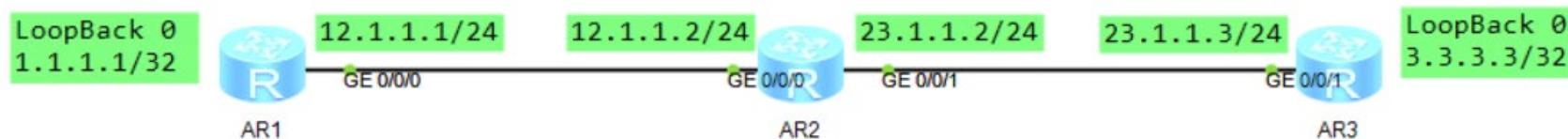
## 2. OSPF路由配置

- 6) 用指令 `[R1]display ip routing-table` 查看三个路由器上路由表，有没有变化？
- 7) 用下面的指令分别查看三个路由器上OSPF2的路由表，  
`display ospf 2 routing`
- 8) 再次执行ping指令，看能不能通，并在Wireshark里观察对应的ICMP协议分组。
  - 用R1 ping R2的23.1.1.2
  - 用R1 ping R3的地址
  - 在R2执行 `ping -a 23.1.1.2 12.1.1.1`
  - 在R2执行 `ping -a 12.1.1.2 23.1.1.3`



### 3. 创建虚接口

- 为了创建隧道，我们在边界路由器（R1和R3）处创建虚拟接口，即LoopBack类型的接口。
- 此类接口状态永远是UP，所以非常适合做隧道的源地址。
- 该类型的接口也经常被用于管理路由器。
- 此类接口的掩码经常设为32位，因为没有连接子网的需求。





### 3. 创建虚接口

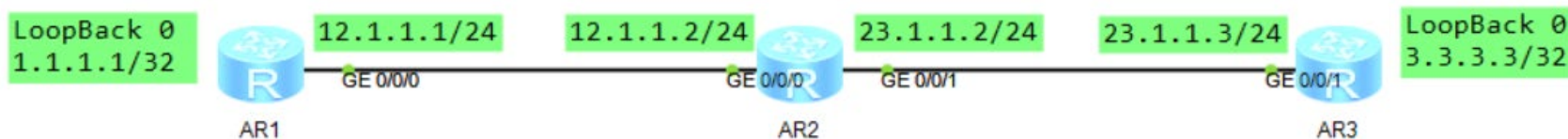
1) 按照图示，创建LoopBack虚接口并分配IP地址。

- `[R1] interface LoopBack 0`

- `[R1-LoopBack0] ip address 1.1.1.1 255.255.255.255`

2) 对R3做类似的操作。

3) 在R1 ping R3的LoopBack 0接口，能不能通呢？



### 3. 创建虚接口

4) 应该不通，为了互通，我们需要配置OSPF。

- `[R1]ospf 2`
- `[R1-ospf-2]area 0`
- `[R1-ospf-2-area-0.0.0.0]network 1.1.1.1 0.0.0.0`

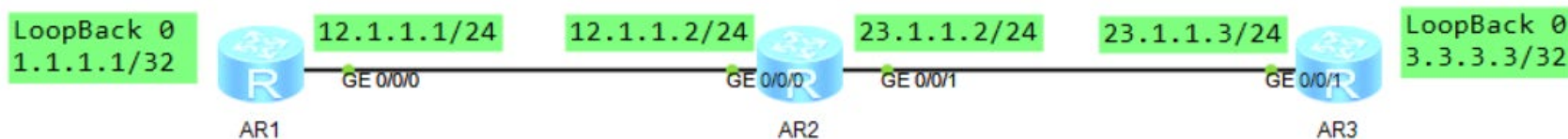
5) 试试 `[R1]ping -a 1.1.1.1 3.3.3.3` 是否能通，Wireshark 有对应的ICMP分组吗？（注意用-a指定源IP地址）

- 应该不能，因此此时路由器R1和R2还不知道怎么转发目的地址是3.3.3.3的分组。



### 3. 创建虚接口

- 6) 试试 `[R3]ping -a 3.3.3.3 1.1.1.1` 是否能通，Wireshark 里有对应的ICMP分组吗？（注意用-a指定源IP地址）
- 不通，Wireshark里只有请求分组，但是没有响应分组，因为路由器此时知道怎么转发目的地址是1.1.1.1的分组，但是不知道怎么转发目的地址是3.3.3.3的分组。打印路由表验证这一点。



### 3. 创建虚接口

- 7) 用类似的方法配置R3的OSPF路由。
- 8) 再次试试两个LoopBack接口之间能否ping通？并观察Wireshark里对应的ICMP分组。（注意用-a指定源IP地址）
- 9) 打印各个路由器的路由表，以及OSPF 2的路由表看看和之前有什么变化。
  - `display ip routing-table`
  - `display ospf 2 routing`



## 4. 创建IPv6虚接口

- 我们创建IPv6虚接口是为了测试R1和R3之间IPv6的连通性，下面以R1为例介绍操作。
- 1) ipv6命令用来使能设备转发IPv6单播报文，包括本地IPv6报文的发送与接收。(8.8.20)
    - `[R1] ipv6`
  - 2) 创建虚接口LoopBack 1
    - `[R1] interface LoopBack 1`
  - 3) 在接口上使能IPv6功能。(8.8.28)
    - `[R1-LoopBack1] ipv6 enable`
  - 4) 配置接口的全球单播地址。(8.8.21)
    - `[R1-LoopBack1] ipv6 address 2001:1::1 64`



## 4. 创建IPv6虚接口

5) 用下面的指令看看接口配置对吗？

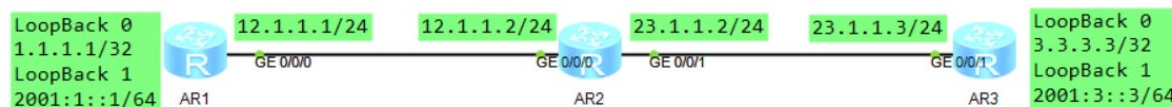
- `[R1-LoopBack1] display ipv6 interface`

6) 用同样的方法配置R3的LoopBack 1。

7) R1能ping通自己的LoopBack1吗？不通的话，说明刚刚的IPv6配置一定有问题。

- **注意**IPv6地址之间ping指令为(20.16.2): `ping ipv6 2001:1::1`

8) R1能ping通R3的LoopBack1吗？为什么？



# 5. 创建IPv6 over IPv4隧道

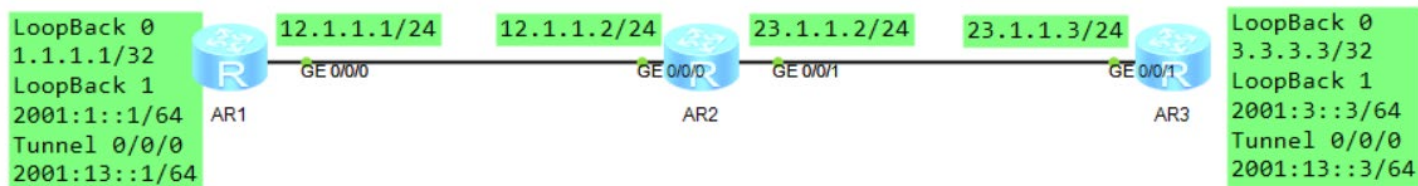
- R1应该ping不通R3的IPv6地址，这就是为什么我们要建IPv6 over IPv4隧道，下面以R1为例介绍操作。

## 1) 创建并进入Tunnel接口视图。(8.11.2)

- `[R1] interface tunnel 0/0/0`

## 2) 配置IPv6地址。

- `[R1-Tunnel0/0/0] ipv6 enable`
- `[R1-Tunnel0/0/0] ipv6 address 2001:13::1 64`



# 5. 创建IPv6 over IPv4隧道

## 3) 配置Tunnel接口的隧道协议。(8.11.4)

- `[R1-Tunnel0/0/0] tunnel-protocol ipv6-ipv4`

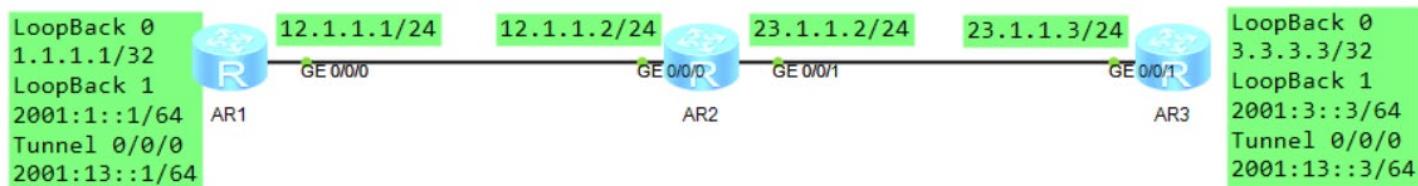
## 4) 配置Tunnel源地址或源接口。(8.11.3)

- `[R1-Tunnel0/0/0] source LoopBack 0`

## 5) 指定Tunnel接口的目的地址。(8.11.1)

- `[R1-Tunnel0/0/0] destination 3.3.3.3`

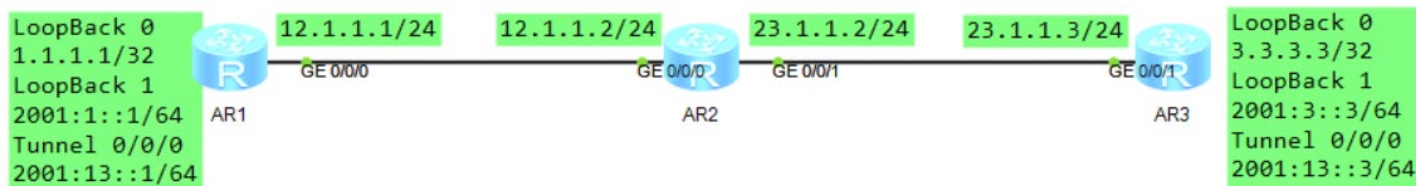
## 6) 按照类似的方法配置R3的Tunnel0/0/0。





## 5. 创建IPv6 over IPv4隧道

- 7) R1能ping通R3的Tunnel0/0/0吗？若不能，说明隧道不通，配置有问题。
- 8) 观察Wireshark抓包，回答执行 `[R1]ping ipv6 2001:13::3` 时，ICMPv6请求报文的源地址是多少？
- 9) R1能ping通R3的LoopBack1吗？为什么？



## 6. 配置IPv6静态路由

- R1应该ping不通R3的LoopBack1接口，因为路由器不知道怎么转发去往R3 LoopBack1的分组。这就需要配置IPv6的路由。

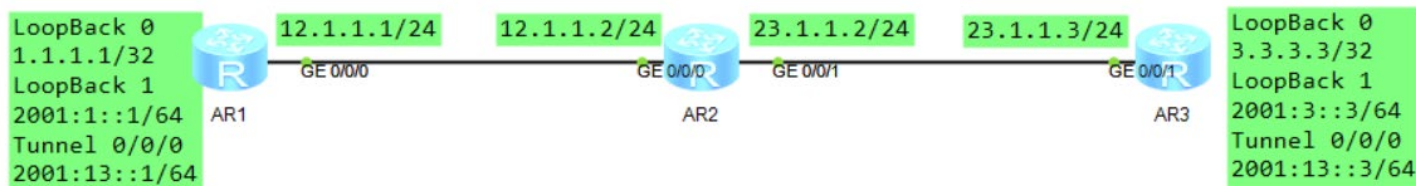
### 1) 配置静态路由。(9.1.7)

- 网络2001:3::，前缀长度为64，通过接口Tunnel0/0/0转发。

- `[R1] ipv6 route-static 2001:3:: 64 Tunnel0/0/0`

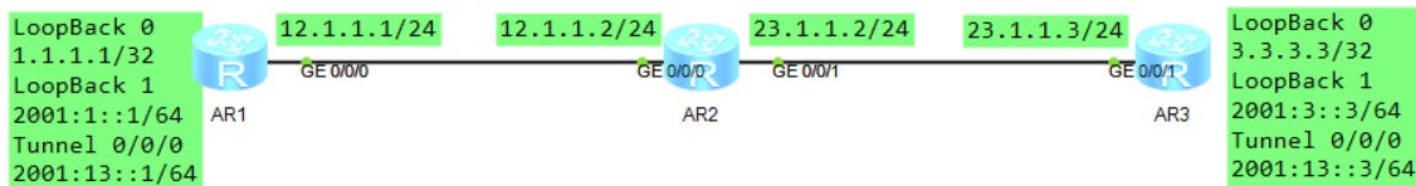
### 2) 再次试试R1的LoopBack1能ping通R3的LoopBack1吗？ (注意使用-a指定ICMPv6ECHO请求报文的源ipv6地址)

- `[R1] ping ipv6 -a 2001:1::1 2001:3::3`



## 6. 配置IPv6静态路由

- 3) 试试R3的LoopBack1能ping通R1的LoopBack1吗？  
(注意使用-a指定ICMPv6 ECHO请求报文的源ipv6地址)
- `[R3]ping ipv6 -a 2001:3::3 2001:1::1`
- 4) 应该不能，需要用类似的方法配置R3的静态路由。
- 5) 如果R1和R3的LoopBack1之间能相互ping通，说明实验成功。
- 6) 可以用下面的指令查看IPv6的路由表。
- `[R1] display ipv6 routing`







恭喜你已完成实验