

项目 3 PPP 协议

3.1 项目介绍

3.1.1 项目目标

- 1. 理解 PPP 协议的基本概念。
- 2. 掌握 PPP 协议的工作过程与认证方法。
- 3. 掌握华为路由器 PPP 协议配置的基本方法与命令。
- 4. 提升观察和分析网络行为的能力。

3.1.2 网络拓扑

本项目基于的网络拓扑如图 1.1 所示。

3.1.3 任务列表

任务一 默认配置下的 PPP 协议运行分析

- (1) 在 eNSP 中搭建图 1.1 所示网络，要求 R1 和 R2 通过串口相连，确认接口已封装 PPP 协议。然后，在 R1 与 R2 的互连链路上开启抓包功能。
- (2) 利用 display interface 命令查看互连接口信息，同时观察抓取到的数据包。将结果分别按表 3.1 和表 3.2 的要求进行记录。

表 3.1 接口（假设 Serial 0/0/0）信息表

参 数	R1	R2
Serial0/0/0 current state		
Line protocol current state		
Internet protocol processing		
LCP		
IPCP		

表 3.2 抓包结果表

包 序 号	PPP 帧封装信息的类别	PPP 帧封装信息的细分类别	Identifier 值	选 项

(3) 根据 PPP 协议运行的知识和抓包结果, 分析此刻 PPP 协议处于哪个阶段。解释抓包结果和查看路由器互连接口的结果。

任务二 PPP 协议认证过程分析

(1) 保持抓包。在 R1 上开启密码认证协议 (Password Authentication Protocol, PAP) 认证 (R1 作为认证方)。保持 R2 为前一状态, 即不进行任何认证配置。完成后, 查看互连接口信息, 观察新抓取到的数据包, 将结果分别按表 3.1 和表 3.2 的要求进行记录。

(2) 根据 PPP 协议运行的知识和抓包结果, 分析此刻 PPP 协议处于哪个阶段。解释抓包结果和查看路由器互连接口的结果。

(3) 保持抓包。在 R2 上配置被认证方的 PAP 认证, 提供正确的账号和密码。猜测 PPP 协议将进入哪个阶段, 并进行验证。

任务三 PPP 协议网络控制协议 (NCP) 协商过程分析

(1) 保持抓包。给 R1 的互连接口配置 IP 地址 (建议: 192.168.10.1/24)。完成后, 观察路由器互连接口信息的变化和抓包结果的变化, 并解释这个变化。

(2) 保持抓包。给 R2 的互连接口配置 IP 地址 (建议: 192.168.10.2/24)。完成后, 观察路由器互连接口信息的变化和抓包结果的变化, 并解释这个变化。

任务四 验证 PPP 链路的网络连通性并观察流量传输

任务进行到这里, 已经完成了认证配置和网络协议的配置, 此时双方应已具备网络连通性。利用 ping 命令进行测试, 并通过抓包观察 ping 报文如何通过 PPP 链路。

3.2 知识准备

3.2.1 PPP 协议概述

点对点协议 (Point-to-Point Protocol, PPP) 是应用于点对点信道的数据链路层协议, 该类信道上的数据传输本身是简单的——因为是专线连接, 一方想发送就发送; 并且因为信道的另一端就是接收方, 协议不涉及编址。但实际上, PPP 协议并不简单, 它包含 3 个组成部分:

(1) 一个基本的串行链路传输控制协议, 为链路两端的设备提供基本的帧传输能力。

(2) 一个建立、配置和测试数据链路连接的链路控制协议 (Link Control Protocol, LCP), 支持通信双方协商链路的一些选项、实施认证等。

(3) 一套网络控制协议 (Network Control Protocol, NCP), 其中的每一个协议支持一种网络层协议, 用于协商网络层地址等参数。

LCP 和 NCP 的存在使 PPP 协议能够适应多种类型链路、支持多种网络层协议, 成为互联网上应用最广泛的数据链路层协议。

3.2.2 PPP 协议的工作过程

如图 3.1 所示，PPP 协议运行可能经历以下 6 个阶段。

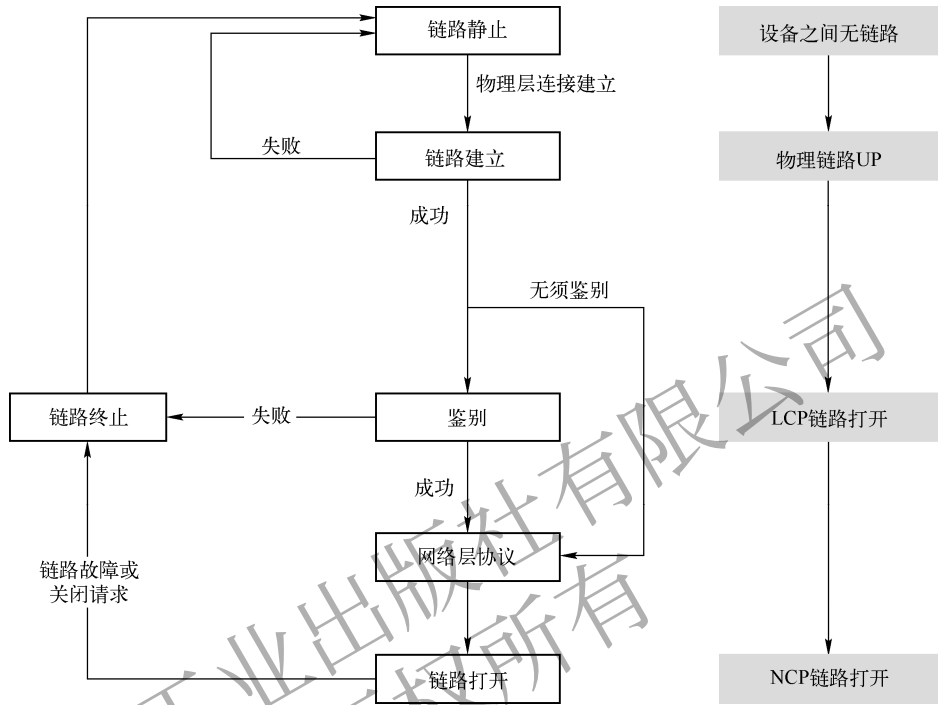


图 3.1 PPP 协议的阶段及状态转移图

(1) **链路静止**：这是 PPP 会话的起点和终点，对应设备之间无链路（接口还没有连接有效的对端设备）的状态。

(2) **链路建立**：当 PPP 协议检测到**物理链路 UP**，即进入此阶段，进行 LCP 协商，也就是关于链路参数的协商。存在有效的物理链路并不代表就可以实现数据传输。这就好比在生活中，就算道路建好了，若道路两端通行规则未统一，比如一端要求左行，而另一端要求右行，车辆仍无法正常通行。LCP 协商成功即进入“**LCP 链路打开**”（LCP opened）状态，此时数据链路层为通信做好了准备；否则回到“**链路静止**”阶段。

(3) **鉴别**：该阶段发生在“**LCP 链路打开**”且 PPP 配置了认证时。鉴别是一方对另一方身份的确认，默认情况下 PPP 链路不要求认证。鉴别成功不改变“**LCP 链路打开**”状态，但流程进入“**网络层协议**”阶段；失败的话则进入“**链路终止**”阶段。

(4) **网络层协议**：进行 NCP 协商，即关于网络层配置的协商。实际的通信建立在网络层上，NCP 协商成功即意味着链路两端的网络层为通信做好了准备，将对应的状态称为“**NCP 链路打开**”（NCP opened）。

通过上述阶段，PPP 按照自下而上的顺序，为通信两端依次准备好了物理层、数据链路层和网络层，此时即可进行数据传输。图 3.1 右侧的灰色框描述了链路功能逐步增强的过

程。需要注意的是，上层功能建立在下层功能的基础上，因此“NCP 链路打开”意味着作为它基础的“LCP 链路打开”“物理链路 UP”同时成立。

(5) **链路打开**：可以进行数据传输的阶段。PPP 协议的官方文档（RFC 1661）中并不包含此阶段，但是显式画出此阶段有助于理解 PPP 会话。

(6) **链路终止**：释放所有的资源，将 PPP 会话带回到“链路静止”阶段。

3.2.3 LCP 协商与 NCP 协商

1. LCP 协商

如图 3.2 所示，链路两端互相发送 LCP 配置请求（Configuration Request）分组来向另一方请求自己需要的 LCP 选项。接收方视自身情况，可以发送以下 3 种分组作为响应。

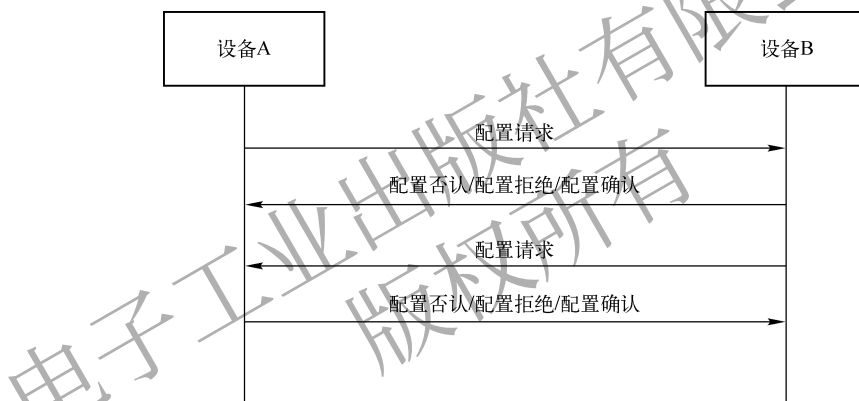


图 3.2 LCP 协商过程

(1) **配置确认（Configuration Ack）**：若完全支持对端的 LCP 选项，则回应此分组，分组中必须携带收到的 LCP 配置请求分组中的全部选项。

(2) **配置否认（Configuration Nak）**：若支持对端的协商选项，但不认可这些选项的具体取值，则回应此分组。该分组仅携带自己不认可的选项，并填上自己期望的内容。如：对端 MTU 值为 1500，而自己期望 MTU 值为 1492，则在配置否认分组中填上自己的期望值 1492。

(3) **配置拒绝（Configuration Reject）**：若不支持对端的某些参数，则发送此分组并携带那些本端不支持的配置参数。

如图 3.3 所示，LCP 分组封装在 PPP 帧中传输，此时 PPP 帧的协议字段值为 0xC021。LCP 分组自身由 4 个字段组成，各字段的解释如下。

- **代码（Code）**：1 字节，用于标识 LCP 分组类型。例如：0x01 表示配置请求，0x02 表示配置确认。

- 标识 (Identifier): 1 字节, 用来匹配请求和响应, 即对请求的响应与请求具有相同的标识。
- 长度 (Length): 2 字节, 用于标识该 LCP 分组的总的字节长度。
- 数据 (Data): 长度可变, 承载各种参数用于协商配置选项, 每个参数都由 Type、Length 和 Value 组成。

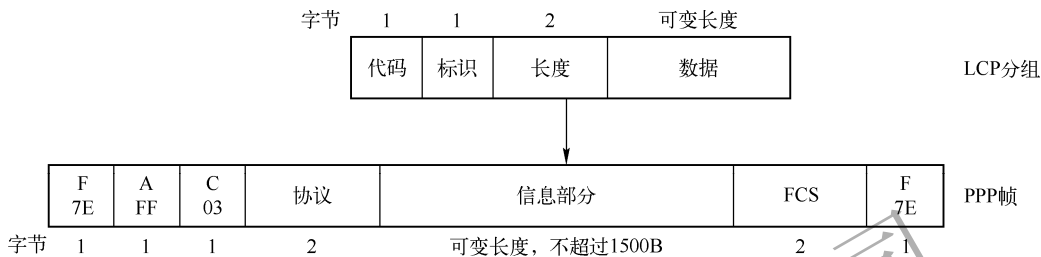


图 3.3 LCP 分组格式及封装

2. NCP 协商

最为常见的 NCP 协议是 IP 控制协议 (Internet Protocol Control Protocol, IPCP), 用于网络层协议为 IP 的情形, 用来协商 PPP 链路两端的 IP 参数, 如 IP 地址等。NCP 协商流程与 LCP 协商流程类似, 两端互相发送 NCP 配置请求分组并做出相应回应 (配置确认/否认/拒绝)。协商完毕后, 说明两端的网络层参数已经准备好, 可以进行数据通信了。

3.2.4 PPP 协议认证

1. PAP 认证

如图 3.4 所示, PAP 认证为两次握手认证。

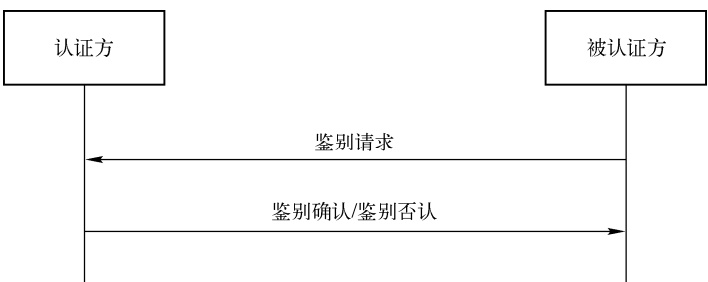


图 3.4 PAP 两次握手认证过程

第 1 步, 被认证方将配置的用户名和密码信息通过 PAP 鉴别请求 (Authenticate-Request) 分组以明文方式发送给认证方。

第 2 步，认证方收到被认证方发送的用户名和密码信息后，根据本地配置的用户名和密码数据库检查用户名和密码信息是否匹配，如果匹配，则返回 PAP 鉴别确认（Authenticate-Ack）分组，表示认证成功；否则，返回 PAP 鉴别否认（Authenticate-Nak）分组，表示认证失败。

2. CHAP 认证

如图 3.5 所示，挑战握手认证协议（Challenge Handshake Authentication Protocol，CHAP）认证需要交互 3 个分组，这些分组均含有标识字段，同一认证过程的分组具有相同的标识。



图 3.5 CHAP 三次握手认证过程

首先，认证方发送一个挑战（Challenge）分组给被认证方，分组中含有标识和一个随机产生的挑战字符串。

被认证方按照公式 $MD5\{\text{标识}+\text{密码}+\text{挑战字符串}\}$ 进行一次加密运算，“+”是字符串连接的意思，然后将此摘要信息、对应的标识及端口上配置的 CHAP 用户名一起封装在响应（Response）分组中，发回给认证方。

认证方接收到响应分组后，根据其中的用户名在本地查找对应的密码。找到密码后，采用与被认证方相同的公式进行运算，然后将运算得到的摘要信息与响应分组中封装的摘要信息进行比较。如果相同，则认证方发送一个 CHAP 成功（Success）分组给被认证方，表示认证成功；反之，则发送一个 CHAP 失败（Failure）分组，表示认证失败。

采用 CHAP 认证方式时，被认证方的密码是经过加密后才进行传输的，这样就极大地提高了安全性。

3.2.5 华为路由器 PPP 协议的基本配置命令

（1）启用 PPP 协议，在接口视图下执行。

```
link-protocol ppp
```

（2）认证方创建本地用户，规定用户账号、密码，并指定用户服务类型为 PPP。

```
aaa
```

```
local-user 用户名 password cipher 密码
local-user 用户名 service-type ppp
```

(3) 认证方开启认证，指明认证方法（pap 或 chap，二选一）。

```
ppp authentication-mode pap/chap
```

(4) 认证方式为 PAP 时，被认证方使用该命令配置账号和密码。

```
ppp pap local-user 用户名 password cipher 密码
```

(5) 认证方式为 CHAP 时，被认证方使用以下命令配置账号和密码。

```
ppp chap user 用户名
ppp chap password cipher 密码
```

3.3 实践指导

3.3.1 任务一 默认配置下的 PPP 协议运行分析

(1) 按照图 1.1，在 eNSP 工作区域中布置路由器并通过串口连接，得到图 3.6 所示的仿真网络。

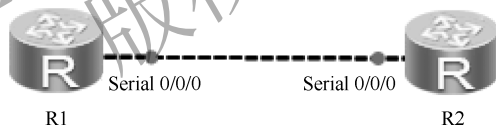


图 3.6 项目 3 的网络拓扑实现图

确认互连接口为 UP，且启用 PPP 协议。

```
#确认接口开启 PPP 协议，以 R1 为例。
[R1]interface Serial0/0/0      #进入接口视图
[R1-Serial0/0/0]display this  #检查接口上的配置，若出现以下结果，说明接口已经
绑定 PPP 协议
#
interface Serial0/0/0
link-protocol ppp
#
```

(2) 查看路由器互连接口信息，抓取经过 R1 与 R2 间互连链路的数据包。

① 查看互连接口的详情，以 R1 为例，查看命令及结果如图 3.7 所示。

```

[R1]dis interface Serial 0/0/0
Serial0/0/0 current state : UP
Line protocol current state : UP
Last line protocol up time : 2025-10-11 14:23:11 UTC-08:00
Description:
Route Port,The Maximum Transmit Unit is 1500, Hold timer is 10(sec)
Internet protocol processing : disabled
Link layer protocol is PPP
LCP opened
Last physical up time : 2025-10-11 14:23:09 UTC-08:00
Last physical down time : 2025-10-11 14:23:06 UTC-08:00
Current system time: 2025-10-11 14:23:24-08:00Interface is V35
  Last 300 seconds input rate 0 bytes/sec, 0 packets/sec
  Last 300 seconds output rate 0 bytes/sec, 0 packets/sec
  Input: 60 bytes, 4 Packets
  Output: 60 bytes, 4 Packets
  Input bandwidth utilization :    0%
  Output bandwidth utilization :    0%

```

图 3.7 默认配置下查看 R1 互连接口的结果

② 提取路由器互连接口的关键信息，并记录在表 3.3 中。

表 3.3 默认配置下路由器互连接口信息

项 目	R1	R2
Serial0/0/0 current state	UP	UP
Line protocol current state	UP	UP
Internet protocol processing	disabled	disabled
LCP	opened	opened
IPCP	无	无

③ 启动 Wireshark，设置在 R1 的互连接口进行抓包，此即为经过 R1 与 R2 间互连链路的数据包。抓包结果如图 3.8 所示。

Apply a display filter ... <Ctrl-/>					
No.	time	Source	Destination	Protocol	Info
1	0.000000			PPP LCP Echo Request	
2	0.000000			PPP LCP Echo Reply	
3	0.031000			PPP LCP Echo Request	
4	0.046000			PPP LCP Echo Reply	
5	10.015000			PPP LCP Echo Request	
6	10.015000			PPP LCP Echo Reply	
7	10.031000			PPP LCP Echo Request	
8	10.046000			PPP LCP Echo Reply	
< []					
> Frame 1: 12 bytes on wire (96 bits), 12 bytes captured (96 bits) on interface 0					
> Point-to-Point Protocol					
▼ PPP Link Control Protocol					
Code: Echo Request (9)					
Identifier: 1 (0x01)					
Length: 8					
Magic Number: 0x000043e5					
0000	ff 03 c0 21 09 01 00 08	00 00 43 e5		...	!....C.

图 3.8 默认配置下 R1 互连接口的抓包结果

④ 查看捕获的数据包，提取关键信息并记录在表 3.4 中。

表 3.4 默认配置下 R1 互连接口的抓包关键信息

包 序 号	PPP 帧封装信息的类别	PPP 帧封装信息的细分类别	Identifier 值	选 项
1	LCP	Echo Request	1	无
2	LCP	Echo Reply	1	无
3	LCP	Echo Request	1	无
4	LCP	Echo Reply	1	无
...	...	...	...	...

(3) 分析。

为了明确 PPP 协议当前所处的阶段，我们首先回顾一下链路两端进行过什么样的配置。此时路由器刚刚启动，未进行任何配置上的修改，链路两端的 PPP 设置均为默认值。这意味着链路两端 PPP 支持的 LCP 选项及选项值相同，LCP 协商理应顺利通过。又因为 PPP 默认不使用认证，因此鉴别阶段直接跳过。由于两端接口尚未进行 IP 地址配置，NCP 协商不会发生。综合以上分析，此时 PPP 会话处于“LCP 链路打开”，但尚未进行“网络层协议”协商的状态。

图 3.8 显示，此时互连接口上往来的均为 PPP 帧，帧中封装的是 LCP 回送请求（Echo Request）分组和回送回答（Echo Reply）分组。这两种分组仅在“LCP 链路打开”状态下才出现，目的是测试和维持与对端的连接。它们的出现同时也验证了前面关于 PPP 会话处于“LCP 链路打开”状态的结论是正确的。

要详细解释抓包结果，就需要查看每个 LCP 分组的细节。经查看，捕获分组表现出循环模式——每 4 个连续分组构成一个片段，其中包含 2 个回送请求分组和 2 个回送回答分组，这些分组的标识值相同。在随后的片段中，标识值加 1，这种模式不断循环。表 3.4 展示了其中一个片段的分组详情。这种循环模式的背后原因是，每端 PPP 在定时器控制下不断发送回送请求分组，每次将标识值加 1；接收方返回同样标识值的回送回答分组。由于在 R1 互连接口上捕获的是两端发送的叠加，所以表现为 2 个回送请求分组加 2 个回送回答分组的循环。

路由器接口信息理应反映接口上 PPP 协议的状态。由表 3.3 可见，此时 R1 和 R2 的互连接口主要参数值是一致的：物理接口是 UP，链路层协议 PPP 是 UP，LCP 为 opened，未见 IPCP 参数。结合 PPP 协议的当前阶段，可得出这些参数的含义如下：

- 物理接口：描述物理接口的状态。UP 表示物理接口已经建立连接，并且处于可以正常收发信号的状态；DOWN 则反之，表示不可用。
- LCP：描述链路建立的状态。“LCP opened”翻译成中文是“链路打开”的意思，表示链路已建立，处于可通行数据的状态。
- 链路层协议：描述链路层协议的状态。UP 表示激活可用，DOWN 则表示不可用。
- IPCP：描述网络层协议的状态。在双方都没有配置 IP 地址的情况下，IPCP 不出现。

3.3.2 任务二 PPP 协议认证过程分析

(1) 按照任务要求，在 R1 上开启 PAP 认证（作为认证方）。

```
#配置认证方的 PAP 认证
[R1]aaa
[R1-aaa]local-user R2 password cipher huawei #创建一个本地账户“R2”，密码为“huawei”
[R1-aaa]local-user R2 service-type ppp #设置用户“R2”为 ppp 用户
[R1-aaa]quit
[R1]interface Serial0/0/0 #进入接口视图
#开启认证，要求对端使用 PAP 认证
[R1-Serial0/0/0]ppp authentication-mode pap
#必须重启接口，认证配置才能生效
[R1-Serial0/0/0]restart
```

① 查看并记录 R1 的互连接口信息，得到表 3.5。

表 3.5 R1 配置 PAP 认证（作为认证方）后路由器互连接口信息

项 目	R1	R2
Serial0/0/0 current state	UP	UP
Line protocol current state	DOWN	DOWN
Internet protocol processing	disabled	disabled
LCP	closed	stopped
IPCP	无	无

② 查看并记录新抓取的数据包。抓包截图如图 3.9 所示，按表 3.2 的要求提取数据包关键信息，结果如表 3.6 所示。

Apply a display filter ... <Ctrl-/>				
No.	time	Source	Destination	Protocol Info
82	214.781000			PPP LCP Configuration Request
83	214.828000			PPP LCP Configuration Request
84	214.828000			PPP LCP Configuration Reject
85	214.828000			PPP LCP Configuration Ack
86	214.828000			PPP LCP Configuration Request
87	214.843000			PPP LCP Configuration Ack
88	214.843000			PPP LCP Termination Request
89	214.843000			PPP LCP Termination Ack
90	244.859000			PPP LCP Configuration Request
91	244.890000			PPP LCP Configuration Request
92	244.890000			PPP LCP Configuration Reject
93	244.890000			PPP LCP Configuration Ack

> Frame 82: 22 bytes on wire (176 bits), 22 bytes captured (176 bits) on interface 0				
> Point-to-Point Protocol				
▼ PPP Link Control Protocol				
Code: Configuration Request (1)				
Identifier: 8 (0x08)				
Length: 18				
> Options: (14 bytes), Maximum Receive Unit, Authentication Protocol, Magic Number				

图 3.9 R1 配置 PAP 认证后 R1 互连接口的抓包结果

表 3.6 R1 配置 PAP 认证后 R1 互连接口的抓包关键信息（使用相对序号）

包 序 号	PPP 帧封装信息的类别	PPP 帧封装信息的细分类别	标 识	选 项
1	LCP	Configuration Request	8	MRU, AU Protocol, Magic Number
2	LCP	Configuration Request	5	MRU, Magic Number
3	LCP	Configuration Reject	8	AU Protocol
4	LCP	Configuration Ack	5	MRU, Magic Number
5	LCP	Configuration Request	9	MRU, Magic Number
6	LCP	Configuration Ack	9	MRU, Magic Number
7	LCP	Termination Request	10	无
8	LCP	Termination Ack	10	无
...	...	...	...	...

（2）分析此刻 PPP 协议处于哪个状态。解释抓包结果，解释接口信息。

此时 R1 配置了 PAP 认证，而 R2 没有配置，两端的认证配置不同，链路肯定无法停留在“LCP 链路打开”的状态了。根据图 3.1 的知识，在协商失败的情况下，PPP 协议应该回到“链路静止”阶段。

然而，通过图 3.9 所示互连接口上此时捕获的 PPP 帧，可以看到 PPP 协议并非静止不动，相反其不断交互 LCP 分组。从 Wireshark 数据包列表面板的“Info”列来看，交互过程表现为 8 个连续 LCP 分组的重复：前 6 个是 LCP 配置分组，后 2 个是 LCP 终止分组。结合在一起，可以得出 PPP 协议此刻的状态：**因链路两端的认证配置不同，LCP 配置协商失败后，通过“链路终止”操作，回到“链路静止”阶段。然而，由于物理接口是 UP，PPP 协议立即又进入“链路建立”阶段，尝试建立链路。**总之，此时 PPP 协议处于不断尝试建立链路但始终失败的状态。

表 3.6 记录了抓包结果中的一个循环片段的信息，表中的包序号使用相对序号。结合 R1 要求认证、R2 尚未配置认证的背景，以及标识变化的规律，该片段隐含了图 3.10 所示的 LCP 协商过程。



图 3.10 R1 配置 PAP 认证后抓包结果反映的 LCP 协商过程

首先，LCP 两端相互发送 LCP 配置请求分组，R1 发送分组携带选项“MRU, AU Protocol, Magic Number”（序号 1），R2 发送分组携带选项“MRU, Magic Number”（序号 2）。然后，R2 回应了配置拒绝（序号 3）分组，拒绝了 R1 的请求，并且注明不被接受的选项是“AU Protocol”；而 R1 则对来自 R2 的配置请求给予了确认（序号 4）。被拒绝的一端，即 R1，剔除了不被支持的“AU Protocol”选项后，再次发送配置请求（序号 5）分组，这次得到了 R2 的确认（序号 6）。最后，R1 因必要的选项得不到支持，发送 LCP 终止请求（序号 7）分组，这次得到了 R2 的确认（序号 8），双方回到“链路静止”阶段。

再看表 3.5 记录的路由器互连接口信息，此时：

- 双方的物理接口状态都为 UP。
- 链路层协议都为 DOWN。
- R1 的 LCP 值为 closed，R2 的 LCP 值为 stopped。

根据前面的分析，此时物理接口为 UP，但因为链路两端关于认证的协商失败，链路处于不通的状态，链路层协议失去支撑自然不可用，因此 DOWN 表示不可用。根据 R1 和 R2 在 LCP 协商中表现出的不同，R1 是要求得不到支持主动终止链路的一方，而 R2 是被动终止。这说明在链路不通的情况下，接口信息对主动终止链路的一方赋予的 LCP 值为“closed”，而对被动终止链路的一方赋予的 LCP 值为“stopped”。

细心的读者可能已经注意到：在图 3.1 中，LCP 协商失败的箭头直接指向“链路静止”阶段，而本任务反映华为 eNSP 中的 PPP 实现在 LCP 协商失败时进入“链路终止”阶段，这种不一致可能导致不同实现之间产生通信问题。

（3）保持抓包，在 R2 上配置被认证方的 PAP 认证，提供正确的用户名和密码。

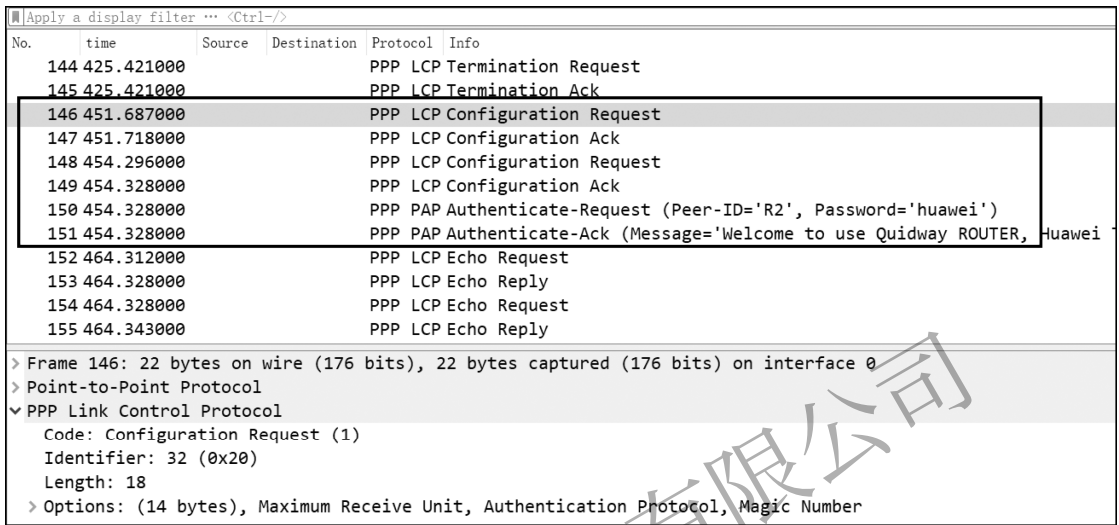
```
#配置被认证方的 PAP 认证
[R2]interface Serial0/0/0
[R2-Serial0/0/0]ppp pap local-user R2 password cipher huawei
[R2-Serial0/0/0]restart
```

理论上，当 R2 也配置了 PAP 认证，这时链路两端 PPP 在“AU Protocol”选项上就不存在冲突，LCP 协商成功，PPP 会话进入“LCP 链路打开”状态。在随后的“鉴别”阶段，由于 R2 提供的用户名和密码符合认证方的要求，因此“鉴别”也成功，LCP 链路保持打开状态。

查看路由器接口信息，可以发现与任务一中的路由器接口信息即表 3.3 所示完全一致，双方的 LCP 值为“opened”，链路层协议为“UP”，无 IPCP 参数。该接口信息能够反映当前 PPP 会话处于“LCP 链路打开”状态，验证了理论分析中对链路最终状态的判断。

由图 3.11 所示的抓包结果可见：R2 配置 PAP 认证后，链路两端 PPP 发送的 LCP 配置请求分组这次都得到了确认（146~149 号），LCP 协商成功。随后被认证方发送 PAP 鉴别请求分组（150 号），此分组中携带明文的用户名和密码；认证方返回 PAP 鉴别确认分组（151 号），鉴别通过。从 152 号包开始，链路两端开始单纯发送 LCP 回送请求分组和回送

回答分组，这是仅在“LCP 链路打开”状态下才会出现的心跳分组，它同样为 PPP 会话处于“LCP 链路打开”状态提供了证据。



No.	time	Source	Destination	Protocol	Info
144	425.421000			PPP LCP	Termination Request
145	425.421000			PPP LCP	Termination Ack
146	451.687000			PPP LCP	Configuration Request
147	451.718000			PPP LCP	Configuration Ack
148	454.296000			PPP LCP	Configuration Request
149	454.328000			PPP LCP	Configuration Ack
150	454.328000			PPP PAP	Authenticate-Request (Peer-ID='R2', Password='huawei')
151	454.328000			PPP PAP	Authenticate-Ack (Message='Welcome to use Quidway ROUTER, Huawei')
152	464.312000			PPP LCP	Echo Request
153	464.328000			PPP LCP	Echo Reply
154	464.328000			PPP LCP	Echo Request
155	464.343000			PPP LCP	Echo Reply

> Frame 146: 22 bytes on wire (176 bits), 22 bytes captured (176 bits) on interface 0
> Point-to-Point Protocol
▼ PPP Link Control Protocol
Code: Configuration Request (1)
Identifier: 32 (0x20)
Length: 18
> Options: (14 bytes), Maximum Receive Unit, Authentication Protocol, Magic Number

图 3.11 R2 配置 PAP 认证后 R1 互连接口的抓包结果

这里有个细节需要澄清：图 3.1（遵照 PPP 标准 RFC 1661）强调，LCP 协商成功便进入“LCP 链路打开”状态，但是无论从接口信息还是抓包结果来看，目前都只观察到鉴别成功后 LCP 处于打开状态。根据图 3.1，鉴别不成功则 PPP 进入“链路终止”阶段，这意味着即使“鉴别”阶段之前就进入了“LCP 链路打开”状态，也是转瞬即逝。因此，“LCP 链路打开”状态只会稳定出现在鉴别成功（无鉴别视同鉴别成功）之后。

3.3.3 任务三 PPP 协议 NCP 协商过程分析

（1）按照任务要求，首先给 R1 的互连接口配置 IP 地址。

```
[R1]interface Serial0/0/0 #进入接口视图
[R1-Serial0/0/0]ip address 192.168.10.1 24 #配置接口 IP 地址
```

如图 3.12 所示，给 R1 配置 IP 地址后，R1 的互连接口除了继续捕获到两端 PPP 为维持链路发送的 LCP 回送请求分组和回送回答分组外，还捕获到一对周期性出现的新分组，即 168 号的 IPCP 配置请求分组和 169 号的 LCP 协议拒绝（Protocol Reject）分组。前者来自 IPCP 协议，携带 IP 地址 192.168.10.1，肯定是由 R1 的互连接口发送。这表明 R1 互连接口拥有 IP 地址的事实，将触发 IPCP 协议启动，该协议发出携带自身 IP 地址的 IPCP 配置请求分组。但是链路另一端，即 R2 的互连接口，因为尚未配置 IP 地址，IPCP 协议未得到运行，所以到达的 IPCP 配置请求分组缺乏适配协议，由 LCP 回应协议拒绝分组。这意味着 R1 发起的以建立 IPCP 链路为目标的协商失败。抓包结果中重复出现这一对分组，说明一旦配置了 IP 地址，IPCP 启动，它便会不断尝试网络层协商。

Apply a display filter ... <Ctrl-/>					
No.	time	Source	Destination	Protocol	Info
166	494.328000			PPP LCP	Echo Reply
167	494.359000			PPP LCP	Echo Reply
168	494.593000			PPP IPCP	Configuration Request
169	494.625000			PPP LCP	Protocol Reject
170	504.328000			PPP LCP	Echo Request
171	504.328000			PPP LCP	Echo Request

> Frame 168: 14 bytes on wire (112 bits), 14 bytes captured (112 bits) on interface 0					
> Point-to-Point Protocol					
▼ PPP IP Control Protocol					
Code: Configuration Request (1)					
Identifier: 1 (0x01)					
Length: 10					
▼ Options: (6 bytes), IP address					
▼ IP address: 192.168.10.1					
Type: IP address (3)					
Length: 6					
IP Address: 192.168.10.1					

0000	ff 03 80 21 01 01 00 0a 03 06 c0 a8 0a 01	...!....
------	---	----------------

图 3.12 R1 配置 IP 地址后其互连接口的抓包结果

此时 R1 的互连接口信息如图 3.13 所示。与配置 IP 地址前相比，R1 的接口信息多出了 IPCP，值为“stopped”。结合抓包结果，可将 IPCP 的出现对应 IPCP 协议开始运行，其值“stopped”表示被动停止。

```
[R1]dis interface Serial 0/0/0
Serial0/0/0 current state : UP
Line protocol current state : UP
Last line protocol up time : 2025-10-11 14:34:13 UTC-08:00
Description:
Route Port, The Maximum Transmit Unit is 1500, Hold timer is 10(sec)
Internet Address is 192.168.10.1/24
Link layer protocol is PPP
LCP opened, IPCP stopped
Last physical up time : 2025-10-11 14:23:09 UTC-08:00
Last physical down time : 2025-10-11 14:23:06 UTC-08:00
Current system time: 2025-10-11 14:37:18-08:00
Interface is V35
Last 300 seconds input rate 2 bytes/sec, 0 packets/sec
Last 300 seconds output rate 2 bytes/sec, 0 packets/sec
Input: 2032 bytes, 151 Packets
Output: 1926 bytes, 151 Packets
Input bandwidth utilization : 0.02%
Output bandwidth utilization : 0.02%
```

图 3.13 R1 配置 IP 地址后其互连接口信息

(2) 参照 (1)，给 R2 的互连接口配置 IP 地址为 192.168.10.2/24。

此时抓包结果如图 3.14 所示，在源源不断出现的 LCP 回送请求分组和回送回答分组中，出现了 2 个 IPCP 配置请求分组、2 个 IPCP 配置确认分组。这是因为 R2 的互连接口配置 IP 地址后，R2 上的 IPCP 协议也开始运行，并发送 IPCP 配置请求分组。这时互连链路上抓到的是链路两端发送的 IPCP 配置请求分组，并且对端收到后均返回 IPCP 配置确认分组。由此，IPCP 协商成功，IPCP 链路打开。

此时 R1、R2 的接口信息均包含 IPCP 参数，值为“opened”。其与抓包结果相互验证。

Apply a display filter ... <Ctrl-/>					
No.	time	Source	Destination	Protocol	Info
200	564.406000			PPP LCP	Echo Reply
201	564.406000			PPP LCP	Echo Reply
202	566.593000			PPP IPCP	Configuration Request
203	566.609000			PPP IPCP	Configuration Request
204	566.609000			PPP IPCP	Configuration Ack
205	566.625000			PPP IPCP	Configuration Ack
206	574.375000			PPP LCP	Echo Request
207	574.375000			PPP LCP	Echo Reply
Frame 202: 14 bytes on wire (112 bits), 14 bytes captured (112 bits) on interface 0					
Point-to-Point Protocol					
PPP IP Control Protocol					
Code: Configuration Request (1)					
Identifier: 1 (0x01)					
Length: 10					
Options: (6 bytes), IP address					
IP address: 192.168.10.2					
Type: IP address (3)					
Length: 6					
IP Address: 192.168.10.2					
0000 ff 03 80 21 01 01 00 0a 03 06 c0 a8 0a 02 ...!....					

图 3.14 R2 配置 IP 地址后互连链路的抓包结果

3.3.4 任务四 验证 PPP 链路的网络连通性并观察流量传输

至此，PPP 会话已经成功通过“链路建立”“鉴别”和“网络层协议”阶段，处于 LCP 链路和 NCP 链路均为打开的状态。根据 PPP 状态机的知识，此时来到“链路打开”阶段，可以进行数据传输。

图 3.15 是在 R1 上执行 ping 192.168.10.2（R2 的互连接口地址）命令时，在 R1 的互连接口捕获到的数据包。在一系列 LCP 回送请求分组和回送回答分组中，出现了 ping 命令生成的 ICMP 报文。这意味着 R1 发送的 ICMP 回送请求报文能够传到对端 R2，而 R2 的 ICMP 回送回答报文也能够回传到 R1，R1 和 R2 能够正常通信。图 3.15 中，Wireshark 数据包细节面板展示的是 238 号包的封装结构，可见 ICMP 报文被封装到 IPv4 数据报中，然后又被封装到 PPP 帧中，最终通过串行链路传输。

Apply a display filter ... <Ctrl-/>					
No.	time	Source	Destination	Protocol	Info
236	644.437000			PPP LCP	Echo Request
237	644.453000			PPP LCP	Echo Reply
238	650.875000	192.168.10.1	192.168.10.2	ICMP	Echo (ping) request id=0xcdab, seq=256/1, t
239	650.875000	192.168.10.2	192.168.10.1	ICMP	Echo (ping) reply id=0xcdab, seq=256/1, t
240	651.421000	192.168.10.1	192.168.10.2	ICMP	Echo (ping) request id=0xcdab, seq=512/2, t
241	651.437000	192.168.10.2	192.168.10.1	ICMP	Echo (ping) reply id=0xcdab, seq=512/2, t
242	651.937000	192.168.10.1	192.168.10.2	ICMP	Echo (ping) request id=0xcdab, seq=768/3, t
243	651.937000	192.168.10.2	192.168.10.1	ICMP	Echo (ping) reply id=0xcdab, seq=768/3, t
244	652.453000	192.168.10.1	192.168.10.2	ICMP	Echo (ping) request id=0xcdab, seq=1024/4, t
245	652.484000	192.168.10.2	192.168.10.1	ICMP	Echo (ping) reply id=0xcdab, seq=1024/4, t
246	652.953000	192.168.10.1	192.168.10.2	ICMP	Echo (ping) request id=0xcdab, seq=1280/5, t
247	652.968000	192.168.10.2	192.168.10.1	ICMP	Echo (ping) reply id=0xcdab, seq=1280/5, t
248	654.406000			PPP LCP	Echo Request
249	654.437000			PPP LCP	Echo Reply
Frame 238: 88 bytes on wire (704 bits), 88 bytes captured (704 bits) on interface 0					
Point-to-Point Protocol					
Internet Protocol Version 4, Src: 192.168.10.1, Dst: 192.168.10.2					
Internet Control Message Protocol					

图 3.15 ping 命令执行后 R1 互连接口的抓包结果

拓展思考

1. 配置并抓取 PPP 的 CHAP 认证分组，观察并分析它与 PAP 认证在分组交互和安全性方面的差异。
2. 假设 R2 在配置 PAP 认证时，输入了错误的用户名或密码，猜一猜两个路由器的互连接口 PPP 相关参数值是什么？链路两端的交互分组有什么不同？请实际验证。
3. 项目 1 中介绍，同一个网络的接口地址要求网络 ID 相同、主机 ID 不同。假设给 R1 的互连接口配置地址为 192.168.10.1/24，给 R2 的互连接口配置地址为 192.168.20.2/24，即两者网络 ID 不同。请问，PPP 协议能够成功通过“网络层协议”阶段并实现 R1 和 R2 的网络连通吗？试一试，并分析结果。

电子工业出版社有限公司
版权所有