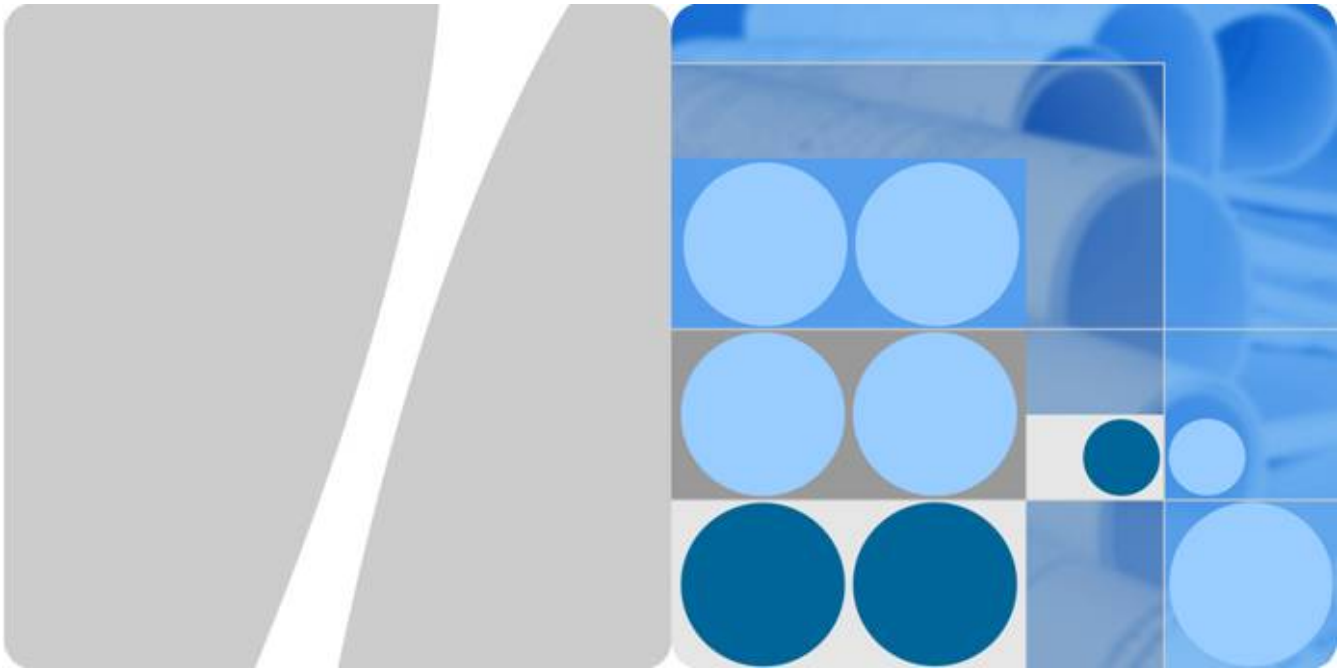


Quidway Eudemon 8080E/8160E 防火墙
V100R003

配置指南 接口及网络协议分册

文档版本 04
发布日期 2011-01-05

华为技术有限公司

HUAWEI

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址： http://www.huawei.com

客户服务邮箱： support@huawei.com

客户服务电话： 0755-28560000 4008302118

客户服务传真： 0755-28560111

前 言

读者对象

本文档介绍了 Eudemon 8080E/8160E 的接口及网络协议配置，包括接口基本参数、以太网接口、POS 接口、AUX 接口、Trunk 接口、逻辑接口、VLAN、PPP 和 PPP 验证、HDLC。

本文档提供了 Eudemon 8080E/8160E 接口及网络协议的配置方法和配置举例。

本文档主要适用于以下工程师：

- 技术支持工程师
- 维护工程师
- 网络工程师
- 网络管理员
- 网络维护工程师

目 录

插图目录

表格目录

1 配置接口基本参数

关于本章

介绍接口的分类及接口基本参数的配置方法。

1.1 简介

Eudemon 的接口是 Eudemon 与网络中的其它设备交换数据并相互作用的部分，分为物理接口和逻辑接口两类。

1.1.1 物理接口

物理接口是物理上存在的接口，分以下两种：

- 局域网 LAN（Local Area Network）接口，如千兆以太网接口。
- 广域网 WAN（Wide Area Network）接口，如 POS 接口。

1.1.2 逻辑接口

逻辑接口是物理上不存在，通过配置才能建立的逻辑意义上的接口。包括子接口、虚拟接口模板、Tunnel 接口、Loopback 接口、Null 接口等。

1.2 配置接口基本参数

介绍创建接口、配置接口描述信息以及打开和关闭接口的配置方法。

1.2.1 建立配置任务

应用环境

为了便于对接口进行配置和维护，在 Eudemon 中支持接口视图，与接口有关的各种命令都必须在相应的接口视图下使用。

前置任务

在开始配置接口之前，需在 Eudemon 上正确安装接口板卡。

数据准备

在配置接口之前，需准备以下数据：

- 需配置接口的接口类型和接口编号
- 接口的描述信息

1.2.2 创建接口

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type* { *interface-number* | *interface-number.subinterface-number* }，进入指定的接口视图。

其中，*interface-type* 为接口类型；*interface-number* 为接口编号。物理接口的接口号由槽号/卡号/端口号组合表示，大部分逻辑接口的接口号由一个数字表示，但是虚拟以太网接口号表示方法与物理接口相同。组合编号规则请参见表 1-1。

表 1-1 Eudemon 8080E/8160E 接口的编号规则

产品	接口的编号规则
槽位号	对于 Eudemon 8080E，接口线路板的槽位号从 1 开始计数，其计数范围是 1 ～ 8。排列顺序为正对前面板从左至右递增（面板上有相应的标志）。 对于 Eudemon 8160E，接口线路板的槽位号从 1 开始计数，其计数范围是 1 ～ 16。排列顺序为正对前面板从左至右，从上到下递增（面板上有相应的标志）。
卡号	业务接口卡号从 0 开始计数，按照从左到右、从上到下递增。若单板没有业务接口卡，则该卡号为 0。
端口号	端口号从 0 开始计数，按照从左到右、从上到下递增。

----结束

1.2.3 配置接口描述信息

背景信息

Eudemon 对于接口有一个接口描述，用来帮助识别和记忆接口的用途，以便管理。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type* { *interface-number* | *interface-number.subinterface-number* }，进入指定的接口视图。

步骤 3 执行命令 **description** *interface-description*，配置接口的描述信息。

----结束

1.2.4 启动接口

背景信息

当接口的业务配置完成后，需要启动接口，使配置的业务加载到接口上。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type* { *interface-number* | *interface-number.subinterface-number* }，进入接口视图。

步骤 3 执行命令 **undo shutdown**，启动接口。

缺省情况下，接口处于启动状态。

----结束

1.2.5 关闭接口

背景信息

关闭接口时，请注意以下情况：

- 某物理接口闲置，没有连接电缆时，请使用 **shutdown** 命令关闭该接口，以防止由于干扰导致接口异常。
- 执行 **restart** 命令，相当于连续执行 **shutdown** 和 **undo shutdown** 命令。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type* { *interface-number* | *interface-number.subinterface-number* }，进入指定接口视图。

步骤 3 执行命令 **shutdown**，关闭接口。

----结束

1.2.6 检查配置结果

检查配置接口参数配置结果相关操作，请参见**表 1-2**。

表 1-2 检查接口参数配置结果相关操作

操作	命令
查看接口配置信息	**display current-configuration** [**controller** | **interface** [*interface-type interface-number*] | **configuration** [*configuration-type*]] [| { **begin** | **exclude** | **include** } *regular-expression*]
查看接口当前运行状态和统计信息	**display interface** [*interface-type* [*interface-number*]] [| { **begin** | **exclude** | **include** } *regular-expression*]
查看接口当前的简要配置信息	**display interface brief** [| { **begin** | **include** | **exclude** } *regular-expression*]
查看接口 IP 的简要配置信息	**display ip interface brief**

2 配置以太网接口

关于本章

介绍以太网接口的分类及配置方法。

2.1 简介

简单介绍以太网接口的基本概念和类型。

局域网主要有以太网、令牌环网等类型。其中以太网以其灵活、简单、易于实现的特点，成为当今最重要的一种局域网组网技术。

Eudemon 支持 10M/100M/1000M 自适应以太网电接口、千兆以太网光接口、千兆/百兆光电接口、万兆以太网光接口（LAN）和万兆以太网光接口（WAN）。

以太网电接口有半双工和全双工两种工作方式。它具有自动协商模式，可以与其他网络设备协商确定工作方式和速率，自动选择最合适的工作方式和速率，从而简化系统的配置和管理。

说明

本章主要介绍 1000M 以太网接口的配置。

2.2 配置以太网接口

介绍如何配置以太网接口。

2.2.1 建立配置任务

应用环境

Eudemon 通过以太网承载报文时，需要配置以太网接口。

前置任务

无

数据准备

在配置以太网接口之前，需准备以下数据：

- Eudemon 以太网接口编号
- 以太网接口的 IP 地址和子网掩码

- ARP 映射项的 IP 地址和以太网 MAC（Medium Access Control）地址
- 以太网接口最大传输单元

说明

- 除了以太网接口的 IP 地址和静态 ARP 映射项外，其他配置项均有缺省值。当改变这些配置项的缺省值时，修改后的值要与对端以太网接口保持一致。
- 千兆以太网接口的固定配置为：全双工模式工作方式、PKTFMT_ETHNT_2 帧格式、1000Mbit/s 速率。
- 万兆以太网接口的固定配置为：全双工模式工作方式、PKTFMT_ETHNT_2 帧格式、10Gbit/s 速率。

2.2.2 配置接口 IP 地址

背景信息

一般情况下，一个接口只需配置一个主 IP 地址，但在有些特殊情况下需要配置从 IP 地址。

例如：Eudemon 通过接口连接了一个物理网络，但该物理网络的计算机分别属于 2 个不同的 C 类网络，为了使 Eudemon 与物理网络中的所有计算机通信，就需要在该接口上配置一个主 IP 地址和一个从 IP 地址，主从 IP 地址应属于两个不同的 C 类型子网。配置时第二个及以后的 IP 地址需要使用关键字 **sub**。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type* { *interface-number* | *interface-number.subinterface-number* }，进入指定以太网接口视图。

步骤 3 执行命令 **ip address** *ip-address* { *mask* | *mask-length* } [**sub**]，配置以太网接口的 IP 地址。

----结束

2.2.3 配置接口 MTU

背景信息

说明

使用 **mtu** 命令改变接口最大传输单元 MTU 后，需要重启接口以保证配置的 MTU 生效。可以先执行 **shutdown** 命令将接口关闭，再执行 **undo shutdown** 命令将接口重启；也可以在接口视图下执行 **restart** 命令重启接口。

执行 **shutdown** 和 **undo shutdown** 命令之间的间隔必须大于 15 秒钟。

由于 QoS（Quality of Service）队列长度有限，如果 MTU 太小而报文尺寸较大，可能会造成分片过多，报文被 QoS 队列丢弃。为避免这种情况，可适当增大 MTU 值。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type* { *interface-number* | *interface-number.subinterface-number* }，进入指定以太网接口视图。

步骤 3 执行命令 **mtu** *mtu*，配置以太网接口的 MTU。

以太网接口的 MTU 取值范围为 46 ～ 9600，缺省值为 1500。单位为字节。

----结束

2.2.4 配置接口速率

背景信息

千兆以太网电接口支持速率为 10Mbit/s、100Mbit/s、1000Mbit/s 或自适应方式。

说明

- 千兆以太网电接口的缺省速率为自动协商模式，用户也可强制更改速率，但应确保与连接对端相同。
- 对于千兆以太网电接口，工作速率为 1000Mbit/s 与半双工模式是互斥的，不能同时配置。
- 只需对千兆以太网电接口进行配置，光接口不需配置。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type* { *interface-number* | *interface-number.subinterface-number* }，进入指定接口视图。

步骤 3 执行命令 **speed** { **10** | **100** | **1000** | **auto** }，选择以太网接口的工作速率。

----结束

2.2.5 配置接口自协商方式

背景信息

当以太网两端配置的双工模式不一致时，可能会无法通信。通过设置为自协商模式，接口自动与对端协商双工模式。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type* { *interface-number* | *interface-number.subinterface-number* }，进入指定接口视图。

步骤 3 执行命令 **negotiation auto**，配置以太网接口采用自协商方式。

----结束

2.2.6 配置接口环回方式

背景信息

环回用于对接口本身进行测试。接口正常工作时，应禁止环回。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface***interface-typeinterface-number*，进入指定的以太网接口视图。

步骤 3 执行命令 **loopback** { **local** | **remote** }，配置接口环回方式。

----结束

2.2.7 检查配置结果

检查配置以太网接口相关操作，请参见表 2-1。

表 2-1 检查以太网接口配置结果相关操作

操作	命令
查看指定以太网接口和控制器接口的配置信息	**display current-configuration** [**controller** | **interface** [*interface-type interface-number*] | **configuration** [*configuration-type*]] [| { **begin** | **exclude** | **include** } *regular-expression*]
查看指定以太网接口的运行状态和统计信息	**display interface** [*interface-type* [*interface-number*]] [| { **begin** | **exclude** | **include** } *regular-expression*]
查看接口当前的简要配置信息	**display interface brief** [| { **begin** | **include** | **exclude** } *regular-expression*]
查看接口 IP 的简要配置信息	**display ip interface brief**

2.3 配置举例

介绍以太网接口配置举例。

2.3.1 配置以太网接口举例

2.3.1 配置以太网接口举例

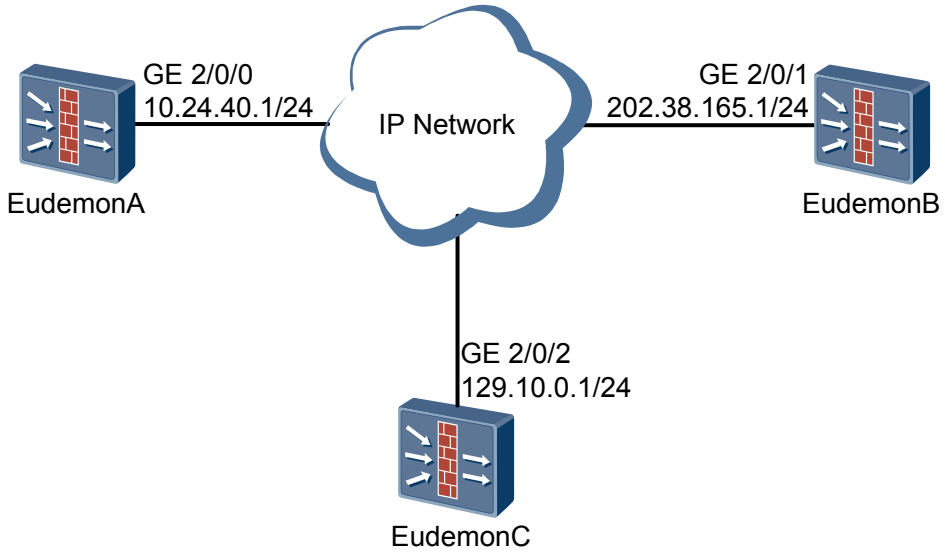
组网需求

说明

在举例中，只涉及以太网相关的配置命令。

如图 2-1 所示，要求配置 EudemonA、EudemonB 和 EudemonC 的以太网口，使 3 台 Eudemon 通过 IP 网络互连。

图 2-1 以太网接口配置组网图



配置思路

配置思路如下：

1. 配置以太网接口的 IP 地址、接口说明。
2. 测试配置是否正确。

数据准备

为完成此配置例，需准备如下的数据：

- 每个 Eudemon 接口的 IP 地址
- 每个 Eudemon 接口的描述信息

操作步骤

步骤 1 配置 EudemonA。

进入系统视图。

```
<EudemonA> system-view
```

进入安全区域。

```
[EudemonA] firewall zone trust
```

配置 GigabitEthernet 2/0/0 加入 Trust 安全区域。

```
[EudemonA-zone-trust] add interface GigabitEthernet 2/0/0
```

退回系统视图。

```
[EudemonA-zone-trust] quit
```

进入接口视图。

```
[EudemonA] interface GigabitEthernet 2/0/0
```

配置 GigabitEthernet 2/0/0 的描述信息。

```
[EudemonA-Gigabitethernet2/0/0] description toFWB_G2/0/1&FWC_G2/0/2
```

配置 GigabitEthernet 2/0/0 的 IP 地址。

```
[EudemonA-Gigabitethernet2/0/0] ip address 10.24.40.1 255.255.255.0
```

退回系统视图。

```
[EudemonA-Gigabitethernet2/0/0] quit
```

配置 ACL 规则。

```
[EudemonA] acl 3005
[EudemonA-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonA-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonA] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonA-interzone-local-trust] packet-filter 3005 inbound
[EudemonA-interzone-local-trust] packet-filter 3005 outbound
```

步骤 2 配置 EudemonB。

进入系统视图。

```
<EudemonB> system-view
```

进入安全区域。

```
[EudemonB] firewall zone trust
```

配置 GigabitEthernet 2/0/1 加入 Trust 安全区域。

```
[EudemonB-zone-trust] add interface GigabitEthernet 2/0/1
```

退回系统视图。

```
[EudemonB-zone-trust] quit
```

进入接口视图。

```
[EudemonB] interface GigabitEthernet 2/0/1
```

配置 GigabitEthernet 2/0/1 的描述信息。

```
[EudemonB-Gigabitethernet2/0/1] description toFWA_G2/0/0&FWC_G2/0/2
```

配置 GigabitEthernet 2/0/1 的 IP 地址。

```
[EudemonB-Gigabitethernet2/0/1] ip address 202.38.165.1 255.255.255.0
```

退回系统视图。

```
[EudemonB-Gigabitethernet2/0/1] quit
```

配置 ACL 规则。

```
[EudemonB] acl 3005
[EudemonB-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonB-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonB] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonB-interzone-local-trust] packet-filter 3005 inbound
[EudemonB-interzone-local-trust] packet-filter 3005 outbound
```

步骤 3 配置 EudemonC。

进入系统视图。

```
<EudemonC> system-view
```

进入安全区域。

```
[EudemonC] firewall zone trust
```

配置 GigabitEthernet 2/0/2 加入 Trust 安全区域。

```
[EudemonC-zone-trust] add interface GigabitEthernet 2/0/2
```

退回系统视图。

```
[EudemonC-zone-trust] quit
```

进入接口视图。

```
[EudemonC] interface GigabitEthernet 2/0/2
```

配置 GigabitEthernet 2/0/2 的描述信息。

```
[EudemonC-Gigabitethernet2/0/2] description toFWA_G2/0/0&FWB_G2/0/1
```

配置 GigabitEthernet 2/0/2 的 IP 地址。

```
[EudemonC-Gigabitethernet2/0/2] ip address 129.10.0.1 255.255.255.0
```

退回系统视图。

```
[EudemonC-Gigabitethernet2/0/2] quit
```

配置 ACL 规则。

```
[EudemonC] acl 3005
[EudemonC-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonC-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonC] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonC-interzone-local-trust] packet-filter 3005 inbound
[EudemonC-interzone-local-trust] packet-filter 3005 outbound
```

步骤 4 验证配置结果。

配置完成之后，可在任一 Eudemon 上采用如下测试方法判别以太网接口是否正常：

- 在业务数据量较小时从 PC 机 **ping**Eudemon 的以太网接口（PC 机与 Eudemon 位于同一局域网内），观察是否能够正确返回全部报文。如果以太网接口正常，则应正确返回全部报文。
- 通过命令 **display interface** 查看连接双方（Eudemon）的统计信息，具体显示信息解释请参见《*Quidway Eudemon 8080E/8160E 命令参考*》。

----结束

3 配置 POS 接口

关于本章

介绍 POS 接口的配置方法。

3.1 简介

简单介绍 POS 接口的基本概念。

3.1.1 SONET/SDH

POS 接口以 SONET（Synchronous Optical Network）/SDH（Synchronous Digital Hierarchy）为物理层协议，支持在城域网及广域网中传送分组数据（如 IP 分组）。

SONET 是 ANSI 定义的同步传输体制。SONET 能够把各种数字化业务信号作为净负荷在光纤线路上传输。由于是同步信号，因此 SONET 可以方便地实现多路信号的复用。

SDH 是 CCITT（现在的 ITU-T）定义的，使用 SONET 速率的一个子集。由于是同步信号，因此 SDH 可以方便地实现多路信号的复用。

SONET 和 SDH 的常见速率如下表所示。常见的几种速率是 4 倍速率的等级关系。为方便起见，常使用括号中的近似值表达它们的速率。

表 3-1 SONET 和 SDH 常见速率对应关系表

SONET		SDH	
电信号	光信号	光信号	速率（Mbit/s）
STS-1	OC-1	-	51.840
STS-3	OC-3	STM-1	155.520（155）
STS-9	OC-9	STM-3	466.560
STS-12	OC-12	STM-4	622.080（622）
STS-18	OC-18	STM-6	933.120
STS-24	OC-24	STM-8	1244.160
STS-36	OC-36	STM-12	1866.240
STS-48	OC-48	STM-16	2488.320（2.5Gbit/s）
STS-96	OC-96	STM-32	4876.640
STS-192	OC-192	STM-64	9953.280（10Gbit/s）

Eudemon 8080E/8160E 提供了 8 口的 155Mbit/s 接口；4 口的 622Mbit/s 接口；4 口的 2.5Gbit/s 接口；1 口的 10Gbit/s 接口。

3.1.2 POS

POS 接口基本的协议体系如下：

- 物理层使用 SONET 协议。
- 链路层使用 HDLC（High-level Data Link Control）或者 PPP 协议。
- 网络层使用 IP 协议。

通过以上协议体系，POS 接口提供了一种高速、可靠、点到点的数据连接。

3.1.3 参考信息

如果想更详细了解 SDH 信息，请参考表 3-2。

表 3-2 参考信息

文档编号	描述
G.707	Network node interface for the synchronous digital hierarchy
G.708	Synchronous digital hierarchy (SDH) network to network interface (NNI)
G.783	Characteristics of synchronous digital hierarchy (SDH)
G.957	Optical interfaces for equipments and systems relating to the synchronous digital hierarchy (SDH)

3.2 配置 POS 接口

介绍如何配置 POS 接口。

3.2.1 建立配置任务

应用环境

当通过 SONET/SDH 光接口承载分组数据时，需要对 POS 接口进行配置。

前置任务

无

数据准备

在配置 POS 接口之前，需要准备以下数据：

- POS 接口编号
- 开销字节 C2、J0 和 J1 值
- POS 接口的 MTU 值

3.2.2 配置接口链路层协议

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface pos** *interface-number*，进入指定的 POS 接口视图。

步骤 3 执行命令 **link-protocol** { **hdlc** | **ppp** }，配置 POS 接口的链路层协议。

----结束

3.2.3 配置接口时钟模式

背景信息

POS 接口工作时需要选择工作的时钟模式。配置 POS 接口的时钟模式时，请注意以下情况：

- 支持两种时钟模式：
  - 主时钟模式，使用内部时钟信号。
  - 从时钟模式，使用线路提供的时钟信号。
- 当两个 POS 接口直接相连或通过 WDM（Wavelength Division Multiplexing）相连时，应配置一端使用主时钟模式，另一端使用从时钟模式。
- 当 POS 接口与交换设备连接时，交换设备为 DCE（Data Circuit-terminating Equipment），使用内部时钟信号，Eudemon 的 POS 接口为 DTE（Data Terminal Equipment），时钟设为从时钟模式。
- 缺省情况下，POS 接口的时钟模式为主时钟模式（master）。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface pos** *interface-number*，进入指定的 POS 接口视图。

步骤 3 执行命令 **clock** { **master** | **slave** }，配置 POS 接口的时钟模式。

----结束

3.2.4 配置接口开销字节

背景信息

SONET/SDH 提供丰富的开销字节，用以提供不同层次的监控功能。

配置 POS 接口的开销字节请注意以下情况：

- 信号标记字节 **C2** 属于高阶通道开销（Higher-Order Path Overhead）字节，用于指示虚拟容器 VC（Virtual Container）帧的复接结构和信息净负荷的性质。
- 再生段踪迹字节 **J0** 属于段开销字节（Section Overhead），用于检测两个端口之间的连接在段层次上的连续性。
- 通道踪迹字节 **J1**（Higher-Order VC-N path trace byte），用于检测两个端口处于持续连接状态。收、发端的 **C2**、**J0**、**J1** 要一致，否则会产生告警。
- 对于 POS 接口，**C2** 缺省值为 22（0x16），**J0** 和 **J1** 的缺省值为“NetEngine”。POS 接口收、发端的 **C2** 配置要一致，否则配置不成功，接口状态为 Down。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface pos** *interface-number*，进入指定的 POS 接口视图。

步骤 3 执行命令 **flag** { **c2** *c2-value* | { **j0** | **j1** } { **1byte-mode** *1byte-value* | **16byte-mode** *16byte-text* | **64byte-or-null-mode** [*64byte-text*] | **peer**} }，配置 POS 接口的开销字节。

----结束

3.2.5 配置接口帧格式

背景信息

POS 接口支持两种帧格式：SDH 格式和 SONET 格式。缺省情况下，POS 接口的帧格式为 SDH。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface pos** *interface-number*，进入指定的 POS 接口视图。

步骤 3 执行命令 **frame-format** { **sdh** | **sonet** }，配置 POS 接口的帧格式。

----结束

3.2.6 配置接口加扰功能

背景信息

POS 接口支持对载荷数据的加扰功能，以避免出现过多连续的 1 或 0，便于接收端提取线路时钟信号。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface pos** *interface-number*，进入指定的 POS 接口视图。

步骤 3 执行命令 **scramble**，开启 POS 接口的载荷加扰功能。

缺省情况下，启动 POS 接口对载荷数据的加扰功能。

----结束

3.2.7 配置接口 CRC 校验字长度

背景信息

POS 接口支持两种 CRC 校验字长度：16 比特和 32 比特。

缺省情况下，CRC 校验字长度为 32 比特。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface pos** *interface-number*，进入指定的 POS 接口视图。

步骤 3 执行命令 **crc** { **16** | **32** }，配置 POS 接口的 CRC 校验字长度。

----结束

3.2.8 配置接口 MTU

背景信息

说明

使用 **mtu** 命令改变接口最大传输单元 MTU 后，需要重启接口以保证配置的 MTU 生效。可以先执行 **shutdown** 命令将接口关闭，再执行 **undo shutdown** 命令将接口重启；也可以在接口视图下执行 **restart** 命令重启接口。

执行 **shutdown** 和 **undo shutdown** 命令之间的间隔必须大于 15 秒钟。

由于 QoS（Quality of Service）队列长度有限，如果 MTU 太小而报文尺寸较大，可能会造成分片过多，报文被 QoS 队列丢弃。为避免这种情况，可适当增大 MTU 值。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface pos** *interface-number*，进入指定的 POS 接口视图。

步骤 3 执行命令 **mtu** *mtu*，配置 POS 接口的 MTU。

POS 接口的 MTU 取值范围为 46 ～ 9600 字节，缺省值为 4470 字节。

----结束

3.2.9 配置接口环回方式

背景信息

环回主要用于一些特殊功能的测试，正常工作时接口禁止环回。

说明

POS 接口不能同时进行 local 环回和 remote 环回。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface pos** *interface-number*，进入指定的 POS 接口视图。

步骤 3 执行命令 **loopback** { **local** | **remote** }，配置 POS 接口的环回方式。

----结束

3.2.10 检查配置结果

检查配置 POS 接口相关操作如表 3-3 所示。

表 3-3 检查 POS 接口配置相关操作

操作	命令
查看 POS 接口和控制器接口的配置信息	**display current-configuration** [**controller** | **interface** [*interface-type interface-number*] | **configuration** [*configuration-type*]] [| { **begin** | **exclude** | **include** } *regular-expression*]
查看 POS 接口的配置和状态信息	**display interface pos** [*interface-number*] [| { **begin** | **exclude** | **include** } *regular-expression*]
查看 POS 接口的简要信息	**display interface brief** [| { **begin** | **include** | **exclude** } *regular-expression*]
查看 POS 接口物理状态	**display pos interface pos** *interface-number*

3.3 配置举例

介绍 POS 接口配置举例。

3.3.1 配置 POS 接口举例

3.3.1 配置 POS 接口举例

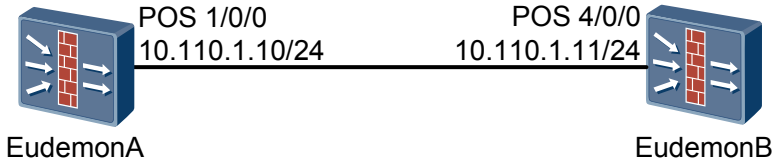
组网需求

说明

配置举例中，只涉及 POS 接口相关的配置命令。

如图 3-1 所示，EudemonA 和 EudemonB 的 POS 接口直接连接，链路层协议为 PPP，请对 Eudemon 进行配置。

图 3-1 Eudemon 通过 POS 接口直接连接组网图



配置思路

配置思路如下：

1. 配置 POS 接口的 IP 地址。
2. 配置链路层协议。
3. 测试配置结果，确认两台 Eudemon 能够 Ping 通。

数据准备

为完成此配置，需准备两端 POS 接口的 IP 地址。

操作步骤

步骤 1 配置 EudemonA。

进入系统视图。

```
<EudemonA> system-view
```

进入 Trust 安全区域视图。

```
[EudemonA] firewall zone trust
```

配置 POS 1/0/0 加入 Trust 安全区域。

```
[EudemonA-zone-trust] add interface pos 1/0/0
```

退回系统视图。

```
[EudemonA-zone-trust] quit
```

进入接口视图。

```
[EudemonA] interface pos 1/0/0
```

配置接口 POS 1/0/0 的 IP 地址。

```
[EudemonA-Pos1/0/0] ip address 10.110.1.10 255.255.255.0
```

配置 POS 1/0/0 接口的链路层协议。

```
[EudemonA-Pos1/0/0] link-protocol ppp
```

关闭 POS 1/0/0 接口。

```
[EudemonA-Pos1/0/0] shutdown
```

启用 POS 1/0/0 接口。

```
[EudemonA-Pos1/0/0] undo shutdown
```

退回系统视图。

```
[EudemonA-Pos1/0/0] quit
```

配置 ACL 规则。

```
[EudemonA] acl 3005
[EudemonA-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonA-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonA] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonA-interzone-local-trust] packet-filter 3005 inbound
[EudemonA-interzone-local-trust] packet-filter 3005 outbound
```

步骤 2 配置 EudemonB。

进入系统视图。

```
<EudemonB> system-view
```

进入安全区域。

```
[EudemonB] firewall zone trust
```

配置 POS 4/0/0 加入 Trust 安全区域。

```
[EudemonB-zone-trust] add interface pos 4/0/0
```

退回系统视图。

```
[EudemonB-zone-trust] quit
```

进入接口视图。

```
[EudemonB] interface pos 4/0/0
```

配置接口 POS 4/0/0 的 IP 地址。

```
[EudemonB-Pos4/0/0] ip address 10.110.1.11 255.255.255.0
```

配置 POS 4/0/0 接口的链路层协议。

```
[EudemonB-Pos4/0/0] link-protocol ppp
```

关闭 POS 4/0/0 接口。

```
[EudemonB-Pos4/0/0] shutdown
```

启用 POS 4/0/0 接口。

```
[EudemonB-Pos4/0/0] undo shutdown
```

退回系统视图。

```
[EudemonB-Pos4/0/0] quit
```

配置 ACL 规则。

```
[EudemonB] acl 3005
[EudemonB-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonB-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonB] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonB-interzone-local-trust] packet-filter 3005 inbound
[EudemonB-interzone-local-trust] packet-filter 3005 outbound
```

步骤 3 验证配置结果。

可以通过 **display interface pos** 查看 POS 接口连通状态，用 **ping** 命令检查网络是否配通。

```
<EudemonA> display interface pos
Pos1/0/0 current state : UP
Line protocol current state : UP
Description : Pos1/0/0 Interface
The Maximum Transmit Unit is 4470 bytes, Hold timer is 10(sec)
Internet Address is 10.110.1.10/24
Link layer protocol is PPP
LCP Opened, IPCP Opened, MPLSCP Opened
BW:10G, Distance:2km,WaveLength:1310nm
Tx Fiber:SingleMode,Rx Fiber:SingleMode
Physical layer is Packet Over SDH
Scramble enabled, clock master, CRC-32, loopback: none
Flag J0 "NetEngine       "
Flag J1 "NetEngine       "
Flag C2 0x16
    SDH alarm:
        section layer:  none
        line    layer:  none
        path    layer:  none
    SDH error:
        section layer:  B1 9456
        line    layer:  B2 1812562  REI 16777215
        path    layer:  B3 10027
Statistics last cleared:never
    Last 5 minutes input rate: 208 bits/sec, 0 Packets/sec
    Last 5 minutes output rate: 104 bits/sec, 54096023676567563 Packets/sec
    Input: 27203 packets, 2025267 bytes
    Input error: 39 shortpacket, 0 longpacket, 21 CRC, 1 lostpacket
    Output: 14683 packets, 1046504 bytes
    Output error: 0 lostpackets
<EudemonA> ping 10.110.1.11
  PING 10.110.1.11: 56  data bytes, press CTRL_C to break
    Reply from 10.110.1.11: bytes=56 Sequence=1 ttl=255 time=1 ms
    Reply from 10.110.1.11: bytes=56 Sequence=2 ttl=255 time=1 ms
```

```
    Reply from 10.110.1.11: bytes=56 Sequence=3 ttl=255 time=1 ms
    Reply from 10.110.1.11: bytes=56 Sequence=4 ttl=255 time=1 ms
    Reply from 10.110.1.11: bytes=56 Sequence=5 ttl=255 time=1 ms

  --- 10.110.1.11 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 1/1/1 ms
```

----结束

4 配置 AUX 接口

关于本章

介绍 AUX 接口的配置方法。

4.1 简介

简单介绍 AUX 口的基本概念。

AUX 接口也称为辅助接口（auxiliary），或备份接口，只支持异步方式，可用作异步串口，通过外接 Modem 实现远程配置的功能。Eudemon 提供了对 AUX 接口的命令行配置功能。

说明

AUX 接口不能作为业务接口，不能进行业务配置。

4.2 配置 AUX 接口

介绍如何配置 AUX 接口。

4.2.1 建立配置任务

应用环境

当通过 AUX 接口外接 Modem 实现远程配置，或作为辅助接口对 Eudemon 进行配置时，需要对 AUX 接口进行配置。

前置任务

无

数据准备

在配置 AUX 接口前，需要准备以下数据：

- AUX 接口 MTU
- AUX 接口的备份接口编号

4.2.2 配置接口链路层协议

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface aux** *interface-number*，进入 AUX 接口视图。

步骤 3 执行命令 **link-protocol ppp**，配置 AUX 接口的链路层协议。

----结束

4.2.3 配置接口 MTU

背景信息

说明

使用 **mtu** 命令改变接口最大传输单元 MTU 后，需要重启接口以保证配置的 MTU 生效。可以先执行 **shutdown** 命令将接口关闭，再执行 **undo shutdown** 命令将接口重启；也可以在接口视图下执行 **restart** 命令重启接口。

执行 **shutdown** 和 **undo shutdown** 命令之间的间隔必须大于 15 秒钟。

由于 QoS（Quality of Service）队列长度有限，如果 MTU 太小而报文尺寸较大，可能会造成分片过多，报文被 QoS 队列丢弃。为避免这种情况，可适当增大 MTU 值。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface aux** *interface-number*，进入 AUX 接口视图。

步骤 3 执行命令 **mtu** *mtu*，配置 AUX 接口的 MTU。

AUX 接口的 MTU 取值范围为 128 ～ 1500KB，缺省值为 1500KB。

----结束

4.2.4 检查配置结果

检查配置 AUX 接口相关操作如表 4-1 所示。

表 4-1 检查 AUX 接口配置结果相关操作

操作	命令
查看 AUX 接口和控制器接口的配置信息	**display current-configuration** [**controller** | **interface** [*interface-type interface-number*] | **configuration** [*configuration-type*]] [| { **begin** | **exclude** | **include** } *regular-expression*]

操作	命令
查看 AUX 接口的配置和状态信息	**display interface aux** [*interface-number*] [| { **begin** | **exclude** | **include** } *regular-expression*]

4.3 配置举例

介绍 AUX 接口配置举例。

4.3.1 通过 AUX 口接入设备
介绍通过 AUX 口搭建配置环境的过程。

4.3.1 通过 AUX 口接入设备

介绍通过 AUX 口搭建配置环境的过程。

前提条件

通过 AUX 口搭建配置环境前，需要完成以下任务：

- 准备好 PC 终端（含串口和 RS-232 电缆）。
- 准备好 PC 终端仿真程序（如 Windows XP 的超级终端）。
- 准备好 Modem。
- Eudemon 上电，且运行正常。

背景信息

Eudemon 满足以下条件时，才能通过 AUX 口搭建配置环境：

- Eudemon 不是第一次上电。
- 用户已经正确配置了 Eudemon 的 AUX 口，包括支持 Modem 拨号连接、用户账号和登录验证方式。

通过 AUX 口搭建配置环境的连接如图 4-1 所示，PC 终端的串口和 Eudemon 的 AUX 口通过 Modem 拨号连接。

图 4-1 通过 AUX 口搭建配置环境

操作步骤

步骤 1 建立物理连接。

1. 在 PC 侧通过串口外挂 Modem 并连接网络。
2. 在 Eudemon 侧通过 AUX 口外挂 Modem 并连接网络。

步骤 2 配置 Modem 拨号连接功能。

1. 通过 Console 口进入 Eudemon 用户视图。

   或者，通过 Telnet 方式进入 Eudemon 用户视图。

   具体操作可参见《*Quidway Eudemon 8080E/8160E 配置指南 基础配置分册*》。

2. 配置 Eudemon 支持 Modem 拨号连接功能和用户信息。

   以下面的情况为例进行配置：配置 Eudemon 支持的远程用户，通过 Modem 接入到 Eudemon 的 AUX0 口，并且配置用户接口的验证方式为 AAA，用户名为 user1，用户级别为 level 3，密码为 Password1，密码的存储方式为密文方式。

```
<Eudemon> system-view
[Eudemon] aaa
[Eudemon-aaa] local-user user1 password cipher Password1
[Eudemon-aaa] local-user user1 service-type terminal
[Eudemon-aaa] local-user user1 level 3
[Eudemon-aaa] quit
[Eudemon] user-interface aux 0
[Eudemon-ui-aux0] authentication-mode aaa
[Eudemon-ui-aux0] modem both
```

步骤 3 通过 AUX 登录 Eudemon。

1. 为 PC 的串口（如 COM1）添加 Modem，具体操作请参见 PC 操作系统（如 Windows XP）的相关文档。
2. 在 PC 上运行终端仿真程序（以 Windows XP 的超级终端为例），选择“开始 > 所有程序 > 附件 > 通讯 > 超级终端”，显示“连接描述”对话框。
3. 在“名称”中输入 PC 与 Eudemon 的连接名称，例如**COMM1**；并在“图标”中选择任一图标，如图 4-2 所示。

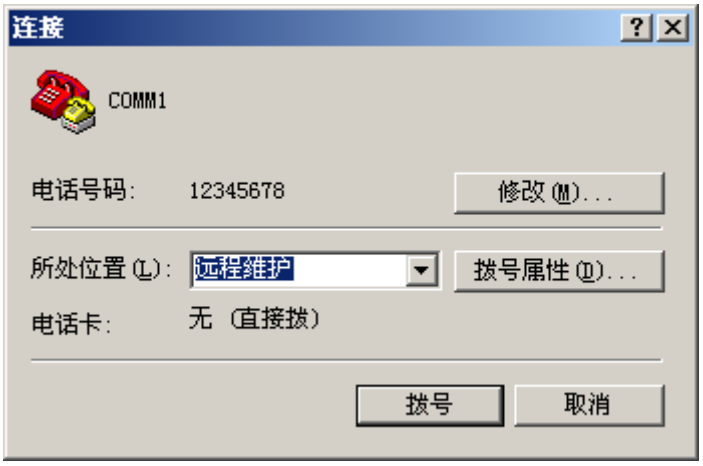
图 4-2 “连接描述”对话框（通过 AUX 口登录）

4. 单击“确定”，显示“连接到”对话框。
5. 输入待拨电话的详细信息，如图 4-3 所示。

图 4-3 “连接到”对话框（通过 AUX 口登录）

6. 单击“确定”，显示“连接”对话框。
7. 在“所处位置”中选择**远程维护**，如图 4-4 所示。

图 4-4 “连接”对话框（通过 AUX 口登录）



8. 单击“拨号”。
9. 在 PC 仿真终端上，单击“Enter”，通过 Eudemon 配置的认证方式后，即可进入用户视图，具体如下：
   - 如果 Eudemon 的认证方式为 AAA，需要输入正确的用户名和口令后，在终端上显示命令行提示符<Eudemon>。
   - 如果 Eudemon 的认证方式为 Password，需要输入正确的口令后，在终端上显示命令行提示符<Eudemon>。

- 如果 Eudemon 的认证方式为 None，则在终端上直接显示命令提示符 <Eudemon>。

----结束

5 配置 Trunk 接口

关于本章

介绍 Trunk 接口的配置方法。

5.1 简介

介绍 Trunk 接口的基本概念和类型。

5.1.1 Trunk 接口简介

Trunk 接口分为 Eth-Trunk 和 IP-Trunk 两种，前者只能由以太网接口构成，后者只能由 POS 接口构成。

表 5-1 Eudemon 8080E/8160E 支持的 Trunk 接口规格

项目	数量
每台 Eudemon 8080E/8160E 最多可创建的 Trunk 接口（包括 Eth-Trunk 和 IP-Trunk）数量	64
每个 Trunk 接口最多可包含的物理链路数量	16
每个 Eth-Trunk 接口最多可配置的子接口数量（IP-Trunk 不支持子接口）	1024

Trunk 接口实现下述特性：

- 支持配置 IP 地址。
- 支持二层转发（仅 Eth-Trunk 接口支持）、三层转发。
- 支持基于物理端口及逻辑端口的 QoS。
- 支持热插拔。

说明

- 如果 Trunk 组里存在某一物理接口，此物理接口所属的接口板在热插拔时，Trunk 组里的成员会自动相应增加或者减少。

5.1.2 负载分担

在一个 Trunk 内，通过对各成员链路配置不同的权重，可以实现流量负载分担。

负载分担分为逐流负载分担和逐包负载分担。

- 逐流负载分担是指当报文的源 IP 地址、目的 IP 地址都相同时，这些报文从同一条成员链路上通过。

  逐流负载分担能保证包的顺序，但不能保证带宽利用率。
- 逐包负载分担是指不区分数据流，而是以报文为单位，将流量分担到不同的成员链路上进行传输。

逐包负载分担能保证带宽利用率，但不能保证包的顺序。

5.2 配置 Eth-Trunk 接口

介绍如何配置 Eth-Trunk 接口。

5.2.1 建立配置任务

应用环境

有时为了提高链路的通信能力，需要将多个以太网端口捆绑为一个 Eth-Trunk 接口，Eth-Trunk 接口的总带宽是各成员带宽之和，通过这种方式，可以增加接口的带宽。

通过 Eth-Trunk 接口可以实现负载分担。Eth-Trunk 接口将流量分流到不同的链路上，最后到达统一目的地。这样可以避免流量都走同一条路径造成的流量阻塞。

Eth-Trunk 接口还可以提高链路的可靠性。在 Eth-Trunk 接口中，如果某个成员端口状态为 Down，流量还能依靠其他的端口进行传输。

前置任务

无

数据准备

在创建 Trunk 接口之前，需准备以下数据：

- Eth-Trunk ID
- Eth-Trunk 口成员端口的类型和编号
- Eth-Trunk 的 IP 地址

5.2.2 创建 Eth-Trunk 接口

背景信息

说明

- 一个物理接口只能加入到一个 Trunk 接口，如果需要加入其他 Trunk 接口，必须先退出原来的 Trunk 接口。
- Trunk 接口不能嵌套，即成员端口不能是 Eth-Trunk 或 IP-Trunk。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface eth-trunk** *trunk-id*，创建 Eth-Trunk。

步骤 3 根据 Eth-Trunk 接口的工作模式，选择执行如下 2 个配置之一。

- 如果是路由模式

  执行命令 **ip address** *ip-address* { *mask* | *mask-length* } [**sub**]，配置 Eth-Trunk 接口的 IP 地址。

  缺省情况下，Eth-Trunk 接口工作在路由模式。

- 如果是透明模式

  执行命令 **portswitch**，配置 Eth-Trunk 接口工作在透明模式下。

  当配置 Eth-Trunk 接口工作在透明模式下后，执行命令 **undo portswitch**，可以恢复接口工作在路由模式下。

步骤 4 执行命令 **quit**，退回系统视图。

步骤 5 执行命令 **interface** *interface-type interface-number*，进入要捆绑到此 Eth-Trunk 的以太接口的接口视图。

步骤 6 执行命令 **eth-trunk** *trunk-id*，将当前接口加入 Eth-Trunk。

Eth-Trunk 的工作模式不影响成员接口的加入。成员接口既可以加入透明模式下 Eth-Trunk，也可以加入路由模式下 Eth-Trunk。

说明

接口加入或退出 Eth-Trunk 后，需要对接口先执行 **shutdown** 命令，再执行 **undo shutdown** 命令将接口重启，以保证配置生效。

执行 **shutdown** 和 **undo shutdown** 命令之间的间隔必须大于 15 秒钟。

----结束

5.2.3 配置影响 Eth-Trunk 状态的 Up 链路下限阈值

背景信息

说明

为保证报文转发正常，建议在同一条 Trunk 链路两端的 Trunk 接口上配置相同的下限阈值。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface eth-trunk** *trunk-id*，进入 Eth-Trunk 接口视图。

步骤 3 执行命令 **least active-linknumber** *link-number*，设置 Trunk 接口中处于 Up 状态的成员端口的下限阈值。

缺省情况下，下限阈值是 1。即只要有一个成员端口保持 Up 状态，该 Eth-Trunk 接口就是 Up 状态。

----结束

5.2.4 配置接口的散列算法

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface eth-trunk** *trunk-id*，进入 Eth-Trunk 接口视图。

步骤 3 执行命令 **load-balance** { **ip** | **packet-all** }，配置 Eth-Trunk 接口的散列依据。

缺省情况下，Eth-Trunk 接口根据 IP 地址进行散列。

- 基于 IP 的散列算法能保证包顺序，但不能保证带宽利用率。
- 基于包的散列算法能保证带宽利用率，但不能保证包的顺序。

----结束

5.2.5 配置 Eth-Trunk 成员链路负载分担权重

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type* { *interface-number* | *interface-number.subinterface-number* }，进入 Eth-Trunk 成员接口视图。

步骤 3 执行命令 **distribute-weight** *weight-value* ，配置成员链路的负载分担权重。

对于一个 Eth-Trunk 接口，其所有成员接口权重之和不能大于 16。Eth-Trunk 接口根据各成员链路的权重等信息进行散列，实施负载分担。在一个 Eth-Trunk 接口中，某成员接口的权重值占所有成员接口权重之和的比例越大，该成员链路承担的负载就越大。

缺省情况下，成员接口权重为 1。

----结束

5.2.6 配置 Eth-Trunk 接口的 MTU

背景信息

说明

使用 **mtu** 命令改变接口最大传输单元 MTU 后，需要重启接口以保证配置的 MTU 生效。可以先执行 **shutdown** 命令将接口关闭，再执行 **undo shutdown** 命令将接口重启；也可以在接口视图下执行 **restart** 命令重启接口。

执行 **shutdown** 和 **undo shutdown** 命令之间的间隔必须大于 15 秒钟。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface eth-trunk** *trunk-id*，进入 Eth-Trunk 接口视图。

步骤 3 执行命令 **mtu** *mtu*，配置 Eth-Trunk 接口的 MTU。

Eth-Trunk 接口的 MTU 取值范围为 46 ～ 9600 字节，缺省值为 1500 字节。

----结束

5.2.7 配置 Eth-Trunk 接口 MAC 地址

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface eth-trunk** *trunk-id*，进入 Eth-Trunk 接口视图。

步骤 3 执行命令 **mac-address** *mac-address*，配置 Eth-Trunk 接口的 MAC 地址。

说明

只有当 Eth-Trunk 接口工作在路由模式时，才能使用 mac-address 命令。

如果在 Eth-Trunk 或者其成员口上有 NAT 或 NAT Server，需要对接口先执行 shutdown 命令，再执行 undo shutdown 命令将接口重启，以保证配置生效。shutdown 和 undo shutdown 命令之间的间隔必须大于 15 秒钟。

接口的 MAC 地址必须唯一，请配置不与其他接口冲突的 MAC 地址。

该节为可选配置，缺省情况下，与其他设备通信时会借用系统的 MAC 地址。

----结束

5.2.8 检查配置结果

检查 Eth-Trunk 接口配置结果相关操作如表 5-2 所示。

表 5-2 检查 Eth-Trunk 接口配置结果相关操作

操作	命令
查看 Eth-Trunk 接口的状态信息	**display interface eth-trunk** [*trunk-id*] [| { **begin** | **exclude** | **include** } [*regular-expression*]
查看 Eth-Trunk 成员端口信息	**display trunkmembership eth-trunk***trunk-id*
查看接口的转发表	**display trunkfwdtbl eth-trunk***trunk-id*

5.3 配置 IP-Trunk 接口

介绍如何配置 IP-Trunk 接口。

5.3.1 建立配置任务

应用环境

为了提高链路的通信能力，需要将多个 POS 接口捆绑为一个 IP-Trunk 接口，IP-Trunk 接口的总带宽是各成员带宽之和，通过这种方式，可以增加接口的带宽。

通过 IP-Trunk 接口可以实现负载分担。IP-Trunk 接口将流量分流到不同的链路上，最后到达统一目的地。这样可以避免流量都走同一条路径造成的流量阻塞。

IP-Trunk 接口还可以提高链路的可靠性。在 IP-Trunk 接口中，如某个成员端口状态为 Down，流量还能依靠其他的端口进行传输。

前置任务

无

数据准备

在创建 Trunk 接口之前，需准备以下数据：

- IP-Trunk ID
- IP-Trunk 口成员端口的类型和编号
- IP-Trunk 的 IP 地址
- IP-Trunk 接口中处于 Up 状态的成员链路的下限阈值

5.3.2 创建 IP-Trunk

背景信息

说明

- 一个物理接口只能加入到一个 Trunk 接口，如果需要加入其他 Trunk 接口，必须先退出原来的 Trunk 接口。
- Trunk 接口不能嵌套，即成员端口不能是 Eth-Trunk 或 IP-Trunk。
- 加入 IP-Trunk 的接口，链路层协议必须为 HDLC。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface ip-trunk** *trunk-id*，创建 IP-Trunk。

步骤 3 执行命令 **ip address** *ip-address* { *mask* | *mask-length* } [**sub**]，配置 IP-Trunk 接口 IP 地址。

步骤 4 执行命令 **quit**，退回系统视图。

步骤 5 执行命令 **interface pos***interface-number*，进入要捆绑到此 IP-Trunk 的 POS 口的接口视图。

步骤 6 执行命令 **link-protocol hdlc**，将接口的链路层协议配置为 HDLC。

步骤 7 执行命令 **ip-trunk** *trunk-id*，将当前接口加入 IP-Trunk。

----结束

5.3.3 配置 IP-Trunk 成员接口下限阈值

背景信息

缺省情况下，下限阈值是 1，即只要有一个成员端口保持 Up 状态，该 IP-Trunk 接口就是 Up 状态。

说明

为保证转发正常，建议在同一条 Trunk 链路两端的 Trunk 接口上配置相同的下限阈值。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface ip-trunk** *trunk-id*，进入 IP-Trunk 接口视图。

步骤 3 执行命令 **least active-linknumber** *link-number*，设置 Trunk 接口中处于 Up 状态的成员端口的下限阈值。

----结束

5.3.4 配置接口的散列算法

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface ip-trunk** *trunk-id*，进入 Eth-Trunk 接口视图。

步骤 3 执行命令 **load-balance** { **ip** | **packet-all** }，配置 IP-Trunk 接口的散列依据。

缺省情况下，IP-Trunk 接口根据 IP 地址进行散列。

- 基于 IP 的散列算法能保证包顺序，但不能保证带宽利用率。

- 基于包的散列算法能保证带宽利用率，但不能保证包的顺序。

----结束

5.3.5 配置 IP-Trunk 成员链路负载分担权重

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type* { *interface-number* | *interface-number.subinterface-number* }，进入 IP-Trunk 成员接口视图。。

步骤 3 执行命令 **distribute-weight** *weight-value* ，配置成员链路的负载分担权重。

对于一个 IP-Trunk 接口，其所有成员接口权重之和不能大于 16。IP-Trunk 接口根据各成员链路的权重等信息进行散列，实施负载分担。在一个 IP-Trunk 接口中，某成员接口的权重值占所有成员接口权重之和的比例越大，该成员链路承担的负载就越大。

缺省情况下，成员接口权重为 1。

----结束

5.3.6 配置 IP-Trunk 接口的 MTU

背景信息

说明

使用 **mtu** 命令改变接口最大传输单元 MTU 后，需要重启接口以保证配置的 MTU 生效。可以先执行 **shutdown** 命令将接口关闭，再执行 **undo shutdown** 命令将接口重启；也可以在接口视图下执行 **restart** 命令重启接口。

执行 **shutdown** 和 **undo shutdown** 命令之间的间隔必须大于 15 秒钟。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface ip-trunk** *trunk-id*，进入 IP-Trunk 接口视图。

步骤 3 执行命令 **mtu** *mtu*，配置 IP-Trunk 接口的 MTU。

IP-Trunk 接口的 MTU 取值范围为 46 ～ 9600 字节，缺省值为 4470 字节。

----结束

5.3.7 检查配置结果

检查 IP-Trunk 接口配置结果相关操作如表 5-3 所示。

表 5-3 检查 IP-Trunk 接口配置结果相关操作

操作	命令
查看 IP-Trunk 接口的状态信息	**display interface ip-trunk** [*trunk-id*] [| { **begin** | **exclude** | **include** } [*regular-expression*]
查看 IP-Trunk 成员端口信息	**display trunkmembership ip-trunk***trunk-id*
查看接口的转发表	**display trunkfwdtbl ip-trunk***trunk-id*

5.4 配置举例

介绍 Eth-Trunk 和 IP-Trunk 接口配置举例。

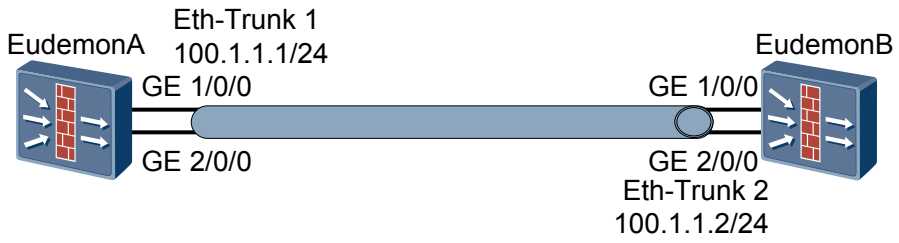
5.4.1 配置 Eth-Trunk 接口举例

介绍 Eth-Trunk 配置举例。

组网需求

如图 5-1 所示，在 EudemonA 与 EudemonB 之间创建 Eth-Trunk，将两个 GigabitEthernet 接口捆绑成一个 Eth-Trunk。

图 5-1 Eth-Trunk 组网图



配置思路

说明

以太网接口不能配置 IP 地址或加入安全域，否则不能将以太网接口加入到 Eth-Trunk 接口。

1. 在 EudemonA 上创建 Eth-Trunk 1，配置 Eth-Trunk 1 的 IP 地址。将 GigabitEthernet 1/0/0 和 GigabitEthernet 2/0/0 加入到 Eth-Trunk 1。

2. 在 EudemonB 上创建 Eth-Trunk 2，配置 Eth-Trunk 2 的 IP 地址。将 GigabitEthernet 1/0/0 和 GigabitEthernet 2/0/0 加入到 Eth-Trunk 2。
3. 检查配置结果。

数据准备

为完成此配置例，需准备如下的数据：

1. EudemonA 侧的 Eth-Trunk 使用 IP 地址 100.1.1.1/24。
2. EudemonB 侧的 Eth-Trunk 使用 IP 地址 100.1.1.2/24。

操作步骤

步骤 1 配置 EudemonA。

进入系统视图。

```
<EudemonA> system-view
```

创建 Eth-Trunk 1，并进入 Trunk 接口视图。

```
[EudemonA] interface eth-trunk 1
```

配置 Eth-Trunk 1 的 IP 地址。

```
[EudemonA-Eth-Trunk1] ip address 100.1.1.1 24
```

退回到系统视图。

```
[EudemonA-Eth-Trunk1] quit
```

进入 GigabitEthernet 1/0/0 接口视图。

```
[EudemonA] interface Gigabitethernet 1/0/0
```

将 GigabitEthernet 1/0/0 加入到 Eth-Trunk 1 中。

```
[EudemonA-GigabitEthernet1/0/0] eth-trunk 1
```

退回到系统视图。

```
[EudemonA-GigabitEthernet1/0/0] quit
```

进入 GigabitEthernet 2/0/0 接口视图。

```
[EudemonA] interface Gigabitethernet 2/0/0
```

将 GigabitEthernet 2/0/0 加入到 Eth-Trunk 1 中。

```
[EudemonA-GigabitEthernet2/0/0] eth-trunk 1
```

退回到系统视图。

```
[EudemonA-GigabitEthernet2/0/0] quit
```

进入 Trust 安全区域。

```
[EudemonA] firewall zone trust
```

添加 Eth-Trunk 1 隶属于 Trust 区域。

```
[EudemonA-zone-trust] add interface eth-trunk 1
```

退回系统视图。

```
[EudemonA-zone-trust] quit
```

配置 ACL 规则。

```
[EudemonA] acl 3005
[EudemonA-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonA-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonA] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonA-interzone-local-trust] packet-filter 3005 inbound
[EudemonA-interzone-local-trust] packet-filter 3005 outbound
```

步骤 2 配置 EudemonB。

进入系统视图。

```
<EudemonB> system-view
```

创建 Eth-Trunk 2，并进入 Trunk 接口视图。

```
[EudemonB] interface eth-trunk 2
```

配置 Eth-Trunk 2 的 IP 地址。

```
[EudemonB-Eth-Trunk2] ip address 100.1.1.2 24
```

退回到系统视图。

```
[EudemonB-Eth-Trunk2] quit
```

进入 GigabitEthernet 1/0/0 接口视图。

```
[EudemonB] interface Gigabitethernet 1/0/0
```

将 GigabitEthernet 1/0/0 加入到 Eth-Trunk 2 中。

```
[EudemonB-GigabitEthernet1/0/0] eth-trunk 2
```

退回到系统视图。

```
[EudemonB-GigabitEthernet1/0/0] quit
```

进入 GigabitEthernet 2/0/0 接口视图。

```
[EudemonB] interface Gigabitethernet 2/0/0
```

将 GigabitEthernet 2/0/0 加入到 Eth-Trunk 2 中。

```
[EudemonB-GigabitEthernet2/0/0] eth-trunk 2
```

退回到系统视图。

```
EudemonB-GigabitEthernet2/0/0] quit
```

进入 Trust 安全区域。

```
[EudemonB] firewall zone trust
```

添加 Eth-Trunk 2 隶属于 Trust 区域。

```
[EudemonB-zone-trust] add interface eth-trunk 2
```

退回系统视图。

```
[EudemonB-zone-trust] quit
```

配置 ACL 规则。

```
[EudemonB] acl 3005
[EudemonB-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonB-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonB] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonB-interzone-local-trust] packet-filter 3005 inbound
[EudemonB-interzone-local-trust] packet-filter 3005 outbound
```

步骤 3 检查配置结果。

在 EudemonA 或 EudemonB 上执行 **display interface eth-trunk** 命令，可以看到接口状态为 Up。

以 EudemonA 的显示为例。

```
<EudemonA> display interface Eth-Trunk 1
Eth-Trunk1 current state : UP
Line protocol current state : UP
Description : HUAWEI, Eudemon Series, Eth-Trunk1 Interface, Route Port
Hash arithmatic : According to IP
The Maximum Transmit Unit is 1500 bytes
Internet Address is 100.1.1.1/24
IP Sending Frames' Format is PKTFMT_ETHNT_2, Hardware address is 00e0-8509-9722
Physical is ETH_TRUNK
    5 minutes input rate 0 bytes/sec, 0 packets/sec
    5 minutes output rate 0 bytes/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 drops
    0 packets output, 0 bytes, 0 drops
```

EudemonA 和 EudemonB 的 Eth-Trunk 接口能够互相 Ping 通。

```
<EudemonA> ping -a 100.1.1.1 100.1.1.2
  PING 100.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 100.1.1.2: bytes=56 Sequence=1 ttl=255 time=31 ms
    Reply from 100.1.1.2: bytes=56 Sequence=2 ttl=255 time=31 ms
    Reply from 100.1.1.2: bytes=56 Sequence=3 ttl=255 time=62 ms
    Reply from 100.1.1.2: bytes=56 Sequence=4 ttl=255 time=62 ms
    Reply from 100.1.1.2: bytes=56 Sequence=5 ttl=255 time=62 ms

  --- 100.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 31/49/62 ms
```

----结束

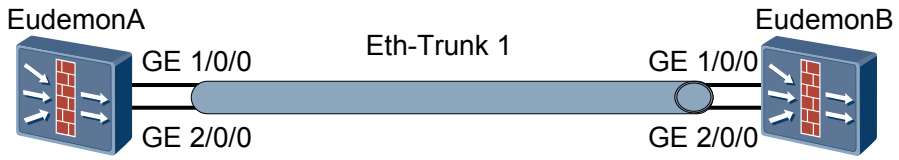
5.4.2 配置透明模式下 Eth-Trunk 端口允许 VLAN 通过示例

组网需求

如图 5-2 所示，EudemonA 和 EudemonB 之间通过 Eth-Trunk1 连接，Eth-Trunk1 是二层端口。

将 Eth-Trunk1 配置为 Trunk 类型接口，允许 EudemonA 和 EudemonB 之间 VLAN 2 ～ 10 的帧通过。

图 5-2 配置二层 Eth-Trunk 端口允许 VLAN 通过组网图



配置思路

说明

以太网接口不能配置 IP 地址或加入安全域，否则不能将以太网接口加入到 Eth-Trunk 接口。

采用如下的思路配置二层 Eth-Trunk 端口允许 VLAN 通过：

1. 创建 Eth-Trunk 接口。
2. 将 Eth-Trunk 接口转为二层端口。
3. 将 Eth-Trunk 端口配置为 Trunk 类型的端口，并允许 VLAN 2 ～ 10 的帧通过。
4. 将成员口接入 Eth-Trunk 端口中。

数据准备

为完成此配置例，需准备如下的数据。

- EudemonA 侧 Eth-Trunk 接口的成员口 GE1/0/0 和 GE2/0/0。
- EudemonB 侧 Eth-Trunk 接口的成员口 GE1/0/0 和 GE2/0/0。

操作步骤

步骤 1 配置 EudemonA。

进入系统视图。

```
<EudemonA> system-view
```

创建 Eth-Trunk 1，并进入 Trunk 接口视图。

```
[EudemonA] interface eth-trunk 1
```

配置 Eth-Trunk 1 工作在透明模式。

```
[EudemonA-Eth-Trunk1] portswitch
```

设置 Trunk 端口 Eth-Trunk1 允许通过的 VLAN 为 2 ～ 10。

```
[EudemonA-Eth-Trunk1] port trunk allow-pass vlan 2 to 10
```

说明

端口设置允许通过的 VLAN 后，则变为 802.1Q 中定义的 Trunk 类型端口。当 Trunk 端口取消所有允许通过的 VLAN 后，则变为 Access 端口。

退回到系统视图。

```
[EudemonA-Eth-Trunk1] quit
```

进入 GigabitEthernet 1/0/0 接口视图。

```
[EudemonA] interface Gigabitethernet 1/0/0
```

将 GigabitEthernet 1/0/0 加入到 Eth-Trunk 1 中。

```
[EudemonA-GigabitEthernet1/0/0] eth-trunk 1
```

退回到系统视图。

```
[EudemonA-GigabitEthernet1/0/0] quit
```

进入 GigabitEthernet 2/0/0 接口视图。

```
[EudemonA] interface Gigabitethernet 2/0/0
```

将 GigabitEthernet 2/0/0 加入到 Eth-Trunk 1 中。

```
[EudemonA-GigabitEthernet2/0/0] eth-trunk 1
```

退回到系统视图。

```
[EudemonA-GigabitEthernet2/0/0] quit
```

进入 Trust 安全区域。

```
[EudemonA] firewall zone trust
```

添加 Eth-Trunk 1 隶属于 Trust 区域。

```
[EudemonA-zone-trust] add interface eth-trunk 1
```

退回系统视图。

```
[EudemonA-zone-trust] quit
```

配置 ACL 规则。

```
[EudemonA] acl 3005
[EudemonA-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonA-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonA] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonA-interzone-local-trust] packet-filter 3005 inbound
[EudemonA-interzone-local-trust] packet-filter 3005 outbound
```

步骤 2 配置 EudemonB

配置与 EudemonA 相同，具体过程省略。

步骤 3 验证配置结果

查看 Eth-Trunk1 的状态，以 EudemonA 为例：

```
<EudemonA> display trunkmembership eth-trunk 1
Trunk ID: 1
used status: VALID
TYPE: ethernet
Working Mode : Normal
Working State: Normal
```

```
Number Of Ports in Trunk = 2
Number Of UP Ports in Trunk = 2
operate status: up
Interface  GigabitEthernet1/0/0, valid, selected, operate up, weight=1,
standby interface NULL
Interface  GigabitEthernet2/0/0, valid, selected, operate up, weight=1,
standby interface NULL
```

用 **display vlan interface** 命令查看 Eth-Trunk 端口上可以通过的 VLAN 信息。以 EudemonA 为例：

```
<EudemonA> display vlan interface Eth-Trunk 1
 Eth-Trunk1 1
 Eth-Trunk1 2
 Eth-Trunk1 3
 Eth-Trunk1 4
 Eth-Trunk1 5
 Eth-Trunk1 6
 Eth-Trunk1 7
 Eth-Trunk1 8
 Eth-Trunk1 9
 Eth-Trunk1 10
```

由显示信息可以看出 Eth-Trunk 1 允许 VLAN ID 为 1 ～ 10 的帧通过。

说明

二层接口缺省加入 VLAN 1，可执行命令 **port default vlan** 改变接口的缺省 VLAN ID。

----结束

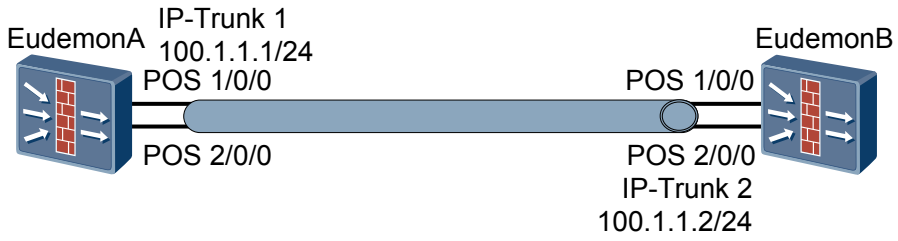
5.4.3 配置 IP-Trunk 接口举例

介绍 IP-Trunk 配置举例。

组网需求

如图 5-3 所示，在 EudemonA 与 EudemonB 之间建立 IP-Trunk。将两个 POS 接口捆绑成一个 IP-Trunk。

图 5-3 IP-Trunk 组网图



配置思路

说明

POS 接口不能配置 IP 地址或加入安全域，否则不能将 POS 接口加入到 IP-Trunk 接口。

1. 在 EudemonA 上创建 IP-Trunk 1，配置 IP-Trunk 的地址。将 POS 1/0/0 和 POS 2/0/0 封装格式配置为 HDLC，并加入到 IP-Trunk 1。

2. 在 EudemonB 上创建 IP-Trunk 2，配置 IP-Trunk 的地址。将 POS 1/0/0 和 POS 2/0/0 封装格式配置为 HDLC，并加入到 IP-Trunk 2。
3. 检查配置结果。

数据准备

为完成此配置例，需准备如下的数据：

1. EudemonA 侧的 IP-Trunk 使用 IP 地址 100.1.1.1/24。
2. EudemonB 侧的 IP-Trunk 使用 IP 地址 100.1.1.2/24。

操作步骤

步骤 1 配置 EudemonA。

进入系统视图。

```
<EudemonA> system-view
```

创建 IP-Trunk 接口，并进入 Trunk 接口视图。

```
[EudemonA] interface ip-trunk 1
```

配置 IP-Trunk 1 的 IP 地址。

```
[EudemonA-IP-Trunk1] ip address 100.1.1.1 255.255.255.0
```

退回到系统视图。

```
[EudemonA-IP-Trunk1] quit
```

进入 POS 1/0/0 接口视图。

```
[EudemonA] interface pos 1/0/0
```

将 POS 1/0/0 封装格式配置成 HDLC。

```
[EudemonA-Pos1/0/0] link-protocol hdlc
```

关闭接口。

```
[EudemonA-Pos1/0/0] shutdown
```

开启接口。

```
[EudemonA-Pos1/0/0] undo shutdown
```

将 POS 1/0/0 接口加入到 IP-Trunk 1 中。

```
[EudemonA-Pos1/0/0] ip-trunk 1
```

退回到系统视图。

```
[EudemonA-Pos1/0/0] quit
```

进入 POS 2/0/0 接口视图。

```
[EudemonA] interface pos 2/0/0
```

将 POS 2/0/0 接口封装成 HDLC。

```
[EudemonA-Pos2/0/0] link-protocol hdlc
```

关闭接口。

```
[EudemonA-Pos2/0/0] shutdown
```

开启接口。

```
[EudemonA-Pos2/0/0] undo shutdown
```

将 POS 2/0/0 接口加入到 IP-Trunk 1

```
[EudemonA-Pos2/0/0] ip-trunk 1
```

退回到系统视图。

```
[EudemonA-Pos2/0/0] quit
```

进入 Trust 安全区域。

```
[EudemonA] firewall zone trust
```

配置 IP-Trunk 1 隶属于 Trust 区域。

```
[EudemonA-zone-trust] add interface ip-trunk 1
```

退回系统视图。

```
[EudemonA-zone-trust] quit
```

配置 ACL 规则。

```
[EudemonA] acl 3005
[EudemonA-acl-adv-3005] rule permit icmp
```

退回系统视图。

```
[EudemonA-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonA] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonA-interzone-local-trust] packet-filter 3005 inbound
[EudemonA-interzone-local-trust] packet-filter 3005 outbound
```

步骤 2 配置 EudemonB。

进入系统视图。

```
<EudemonB> system-view
```

创建 IP-Trunk 2，并进入 Trunk 接口视图。

```
[EudemonB] interface ip-trunk 1
```

配置 IP-Trunk 2 的 IP 地址。

```
[EudemonB-IP-Trunk1] ip address 100.1.1.2 255.255.255.0
```

退回到系统视图。

```
[EudemonB-IP-Trunk1] quit
```

进入 POS 1/0/0 接口视图。

```
[EudemonB] interface pos 1/0/0
```

将 POS 1/0/0 封装格式配置成 HDLC。

```
[EudemonB-Pos1/0/0] link-protocol hdlc
```

关闭接口。

```
[EudemonB-Pos1/0/0] shutdown
```

开启接口。

```
[EudemonB-Pos1/0/0] undo shutdown
```

将 POS 1/0/0 接口加入到 IP-Trunk 2 中。

```
[EudemonB-Pos1/0/0] ip-trunk 2
```

退回到系统视图。

```
[EudemonB-Pos1/0/0] quit
```

进入 POS 2/0/0 接口视图。

```
[EudemonB] interface pos 2/0/0
```

将 POS 2/0/0 接口封装成 HDLC。

```
[EudemonB-Pos2/0/0] link-protocol hdlc
```

关闭接口。

```
[EudemonB-Pos2/0/0] shutdown
```

开启接口。

```
[EudemonB-Pos2/0/0] undo shutdown
```

将 POS2/0/0 接口加入到 IP-Trunk 2

```
[EudemonB-Pos2/0/0] ip-trunk 2
```

退回到系统视图。

```
[EudemonB-Pos2/0/0] quit
```

进入 Trust 安全区域。

```
[EudemonB] firewall zone trust
```

配置 IP-Trunk 2 隶属于 Trust 区域。

```
[EudemonB-zone-trust] add interface ip-trunk 2
```

退回系统视图。

```
[EudemonB-zone-trust] quit
```

配置 ACL 规则。

```
[EudemonB] acl 3005
[EudemonB-acl-adv-3005] rule permit icmp
```

退回系统视图。

```
[EudemonB-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonB] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonB-interzone-local-trust] packet-filter 3005 inbound
[EudemonB-interzone-local-trust] packet-filter 3005 outbound
```

步骤 3 检查配置结果。

在 EudemonA 或 EudemonB 上执行 **display interface ip-trunk** 命令，可以看到接口状态为 Up。

以 EudemonA 的显示为例。

```
<EudemonA> display interface ip-trunk 1
Ip-Trunk1 current state : UP
Line protocol current state : UP
Description : HUAWEI, Eudemon Series, Ip-Trunk1 Interface, Route Port
Hash arithmatic : According to IP
The Maximum Transmit Unit is 4470 bytes
Internet Address is 100.1.1.1/24
Link layer protocol is HDLC
Physical is IP_TRUNK
    5 minutes input rate 0 bytes/sec, 0 packets/sec
    5 minutes output rate 0 bytes/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 drops
    0 packets output, 0 bytes, 0 drops
```

EudemonA 和 EudemonB 的 IP-Trunk 接口能够互相 Ping 通。

```
<EudemonA> ping -a 100.1.1.1 100.1.1.2
  PING 100.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 100.1.1.2: bytes=56 Sequence=1 ttl=255 time=62 ms
    Reply from 100.1.1.2: bytes=56 Sequence=2 ttl=255 time=62 ms
    Reply from 100.1.1.2: bytes=56 Sequence=3 ttl=255 time=62 ms
    Reply from 100.1.1.2: bytes=56 Sequence=4 ttl=255 time=62 ms
    Reply from 100.1.1.2: bytes=56 Sequence=5 ttl=255 time=62 ms

  --- 100.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 62/62/62 ms
```

----结束

6 配置逻辑接口

关于本章

介绍逻辑接口的基本概念、分类和配置方法。

6.1 简介
介绍各种逻辑接口。

6.2 配置子接口
介绍子接口的创建方法。

6.3 配置虚拟接口模板
介绍虚拟接口模板的基本配置。

6.4 配置 Tunnel 接口
介绍 Tunnel 接口的基本配置。

6.5 配置 Loopback 接口及 IP 相关选项
介绍 Loopback 接口的基本配置。

6.6 配置 Null 接口
介绍 Null 接口的基本配置。

6.1 简介

介绍各种逻辑接口。

6.1.1 概述

介绍逻辑接口的概念。

逻辑接口指能够实现数据交换功能但物理上不存在，需要通过配置建立的接口，包括子接口、虚拟接口模板、Loopback 接口、Tunnel 接口以及 Null 接口等。

6.1.2 子接口

介绍子接口的概念。

子接口主要是在一条物理链路上面，逻辑的区分不同的流量，提供在一个物理接口上支持多个逻辑接口的功能。这些逻辑接口在工作时，共用物理接口的物理配置参数，但具有各自的链路层和网络层配置参数。

Eudemon 支持在千兆以太网接口上配置多个子接口，为用户组网提供灵活性。

6.1.3 虚拟接口模板

介绍虚拟接口模板的概念。

虚拟接口模板（Virtual-Template）是用于配置虚拟访问接口的模板，主要应用于 VPN（Virtual Private Network）。

在 VPN 会话连接建立之后，需要创建虚拟访问接口用于和对端交换数据。此时，系统将按照用户的配置，选择虚拟接口模板，根据该模板的配置参数动态地创建虚拟访问接口。

虚拟接口模板要保证在虚拟访问接口创建之前已经建立，在虚拟访问接口关闭之后才可以删除。虚拟接口模板在链路层只支持 PPP（Point to Point Protocol）协议，在网络层只支持 IP 协议。

虚拟访问接口将在需要的时候由系统自动创建，并使用相应虚拟接口模板的参数进行工作，虚拟访问接口会由于底层链路断开或用户干预而被删除。在打开 PPP 的 Debugging 开关时，虚拟访问接口 Up 与 Down 的状态变换可以从终端用户屏幕的输出信息中看到。

6.1.4 Tunnel 接口

介绍 Tunnel 接口的概念。

Tunnel 接口是 GRE 等隧道使用的一种虚拟逻辑接口。

6.1.5 Loopback 接口

介绍 Loopback 接口的概念。

Loopback 是一种虚拟接口。TCP/IP 协议规定，127.0.0.0 网段的地址属于环回地址，使用这类地址的接口属于环回接口。系统在启动时自动创建一个使用环回地址 127.0.0.1 的接口，用来接收所有发送给本机的数据包。

有些应用（如配置 SNA 的 localpeer）需要在不影响物理接口配置的情况下，配置一个带有指定 IP 地址的本地接口,要求该本地接口的 IP 地址为 32 位掩码（节约 IP 地址），并且能够被路由协议发布出去。

利用 Loopback 接口永远 Up 的特性，还可以用其 IP 地址做 Router ID、LSR ID、或被 Tunnel 隧道借用等。

6.1.6 Null 接口

介绍 Null 接口的概念。

Null 接口类似于一些操作系统中支持的空设备（null devices），任何送到该接口的网络数据报文都会被丢弃。

Null 接口主要应用在路由选择中。在配置了 Null 接口的前提下，路由选择时如果匹配不到路由，就会把报文送到该接口。

6.2 配置子接口

介绍子接口的创建方法。

前提条件

在配置以太网子接口之前，需准备以下数据：

- 以太网的接口编号
- 子接口的接口编号

背景信息

Eudemon 整机最多支持 8192 个子接口，每块接口板最多可创建 2048 个子接口、每个 Eth-trunk 接口最多可创建 1024 个子接口。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number.subinterface-number*，创建子接口并进入子接口视图。

subinterface-number 是子接口的编号。

说明

主控板的管理接口 GigabitEthernet 0/0/0 不能配置子接口。

----结束

6.3 配置虚拟接口模板

介绍虚拟接口模板的基本配置。

前提条件

在配置虚拟接口模板之前，需准备虚拟接口模板的编号。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface virtual-template** *vt-number*，创建虚拟接口模板。

----结束

6.4 配置 Tunnel 接口

介绍 Tunnel 接口的基本配置。

前提条件

在配置 Tunnel 接口之前，需准备以下数据：

- Tunnel 接口的编号
- Tunnel 接口的 IP 地址

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface tunnel** *interface-number*，创建 Tunnel 接口。

步骤 3 执行命令 **tunnel-protocol** { **gre** | **ipsec** | **ipv6-ipv4** [**6to4** | **auto-tunnel** | **isatap**] | **ipv4-ipv6** | **none** }，配置隧道的封装模式。

缺省情况下，Tunnel 接口封装协议为 GRE。对于一条隧道，两端的 Tunnel 接口应使用相同的封装模式。

步骤 4 （可选）执行命令 **mtu** *mtu-value*，配置接口的 MTU。

执行该步骤后需要先对接口执行 **shutdown** 命令，再执行 **undo shutdown** 命令将接口重启，以保证设置的 MTU 生效。执行 **shutdown** 和 **undo shutdown** 命令之间的间隔必须大于 15 秒钟。

步骤 5 执行命令 **ip address** *ip-address* { *mask* | *mask-length* } [**sub**]，配置 Tunnel 接口的 IP 地址。

步骤 6 执行命令 **source** { *interface-type interface-number* | *source-ip-address* }，配置 Tunnel 接口的源端地址。

Tunnel 接口的源端地址可以选择接口名称或 IP 地址两种形式。当选择接口名称时，取值为 GigabitEthernet、POS、Eth-Trunk 和 IP-Trunk。

步骤 7 执行命令 **destination** *dest-ip-address*，配置 Tunnel 接口的目的端地址。

Tunnel 接口的目的端地址不能与源端地址相同。

----结束

6.5 配置 Loopback 接口及 IP 相关选项

介绍 Loopback 接口的基本配置。

前提条件

在配置 Loopback 接口之前，需准备以下数据：

- Loopback 接口的接口编号
- Loopback 接口的 IP 地址

背景信息

用户可以创建或删除自己的 Loopback 接口。Loopback 接口一旦被创建，将一直保持 Up 状态，直到被删除。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface loopback** *number*，创建 Loopback 接口。

interface-number 取值范围是 0 ～ 1023。

步骤 3 执行命令 **ip address** *ip-address* { *mask* | *mask-length* } [**sub**]，配置 Loopback 接口的 IP 地址。

步骤 4 执行命令 **ip forward-broadcast** [**acl** *acl-number*]，配置允许 Loopback 接口转发广播报文。

acl*acl-number* 表示对广播报文应用该 ACL 规则号对应的过滤条件，根据过滤结果决定是否转发该定向广播报文。*acl-number* 取值范围为 2000 ～ 3999，其中 2000 ～ 2999 是基本 ACL 规则，3000 ～ 3999 是高级 ACL 规则。

----结束

6.6 配置 Null 接口

介绍 Null 接口的基本配置。

背景信息

Null 接口永远处于 Up 状态，但不能转发数据包，也不能配置 IP 地址或封装其他协议。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface null 0**，进入 Null 0 接口视图。

----结束

7 配置 VLAN

关于本章

介绍 VLAN 的配置方法。

7.1 VLAN 简介
介绍 VLAN 的引入、原理和优点。

7.2 配置路由模式下的 VLAN
介绍路由模式下 VLAN 的配置方法。

7.3 配置透明模式下的 VLAN
介绍透明模式下 VLAN 的配置方法。

7.1 VLAN 简介

介绍 VLAN 的引入、原理和优点。

7.1.1 LAN 互联存在的问题

介绍 VLAN 技术的引入。

以太网是一种基于 CSMA/CD（Carrier Sense Multiple Access/Collision Detect）的共享通讯介质的数据网络通讯技术，共享介质上的各节点轮流使用介质传送帧，同一时刻只能有一个主机发送，其他主机只能接收。

当多个主机通过双绞线连接到集线器 HUB（星型结构），或者通过同轴电缆串连（总线型结构）时，所有互联在共享物理介质上的主机形成一个物理上的冲突域（Collision Domain），一般看作一个局域网的网段（LAN segmentation）。根据上面说明的以太网基本原理，可以看出用 HUB 作 LAN 互联的问题是：当主机数目较多时将导致冲突严重、性能显著下降甚至使网络不可用。

解决上述问题的办法是使用透明网桥（Transparent Bridge）或者交换机（LAN Switch）作 LAN 互联。交换机通过接收到的数据帧的源 MAC 地址建立起 MAC-PORT 映射表，对于收到的单播数据帧，读取帧头的目的 MAC 地址，在 MAC-PORT 映射表中查找，如果能够查找到映射项，则把帧向对应的端口发送；如果找不到，就向所有端口发送。这样，冲突域被交换机隔离在各自的端口，而不会扩展到其他端口。交换机并不改变以太帧的源地址和目的地址，而只是转发到适当的网段（LAN segmentation），是一种透明设备。

交换机虽然解决了使用 HUB 带来冲突（Collision）严重的问题，但仍然不能隔离广播。实际上，所有用交换机互联起来的主机（可能包括多个交换机）是在一个广播域（Broadcast Domain），对于目的 MAC 地址为全 F（0xffffffffffff）的广播报文，例如 ARP 请求报文，交换机会向所有的端口转发。在主机较多的情况下，会造成广播风暴，导致整个网络的性能下降。

为了解决用交换机做 LAN 互联无法限制广播的问题，出现了 VLAN（Virtual Local Area Network）技术。

7.1.2 VLAN 概述

介绍 VLAN 的原理和优点。

VLAN 即虚拟局域网，是一种实现虚拟工作组的技术，方法是将局域网内的设备的划分成一个个逻辑的而不是物理网段。

VLAN 将一个物理的 LAN 在逻辑上划分成多个广播域（多个 VLAN），VLAN 内的主机间通信就和在一个 LAN 内一样，而 VLAN 间则不能直接互通。这样，广播报文被限制在一个 VLAN 内。

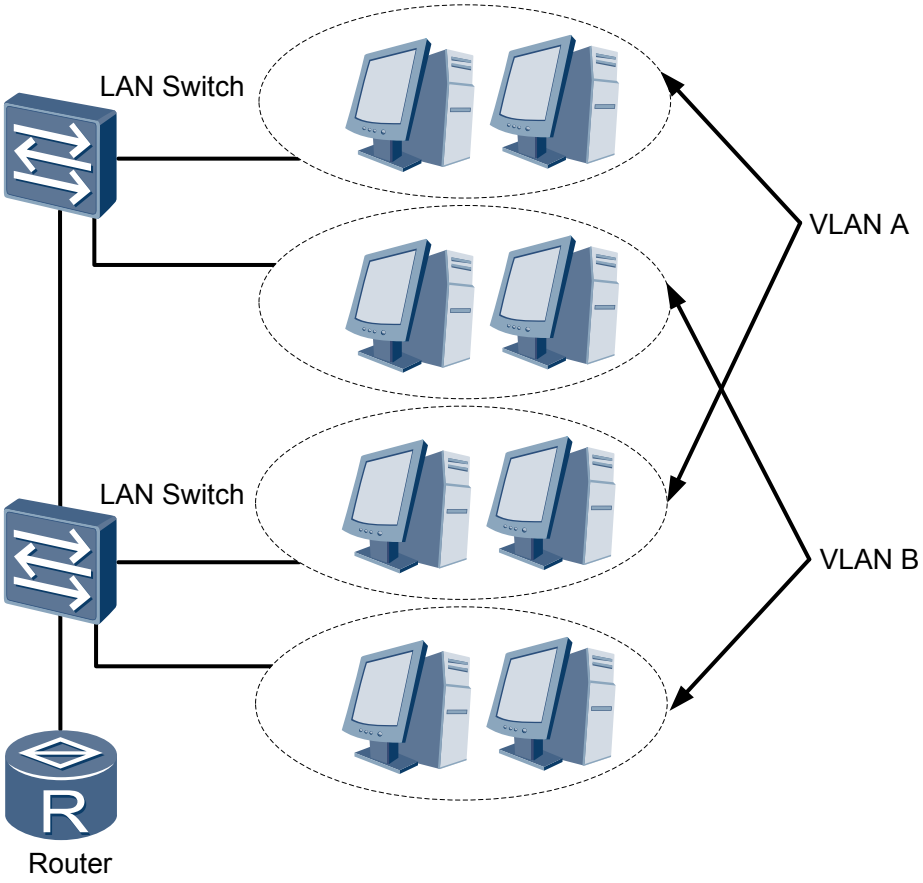
除了划分广播域，VLAN 还可以满足更复杂的网络应用。

例如，一个写字楼租给不同的企业客户，如果这些企业客户都建立各自独立的 LAN，企业的网络投资成本将很高；如果各用户共用写字楼已有的 LAN，又会导致企业信息安全无法保证。

采用 VLAN，可以实现各企业客户共享 LAN 设施，同时保证各自的网络信息安全。

如图 7-1 所示。

图 7-1 VLAN 的典型应用示意图



VLAN 的组成不受物理位置的限制，一个 VLAN 可以在一个交换机内，也可以跨越交换机，甚至可以跨越路由器和 Eudemon。

VLAN 的划分方法有很多，可以基于端口、基于 MAC 地址、基于协议类型、基于 IP 地址映射、基于组播、基于策略等，目前通用的划分方法是基于端口的 VLAN。

说明

本手册中的 VLAN，如果没有特别说明，都是指基于端口的 VLAN。

使用 VLAN 具有如下好处：

- 限制广播报文（广播风暴），节省带宽，提高了网络处理能力。

  广播域限制在一个 VLAN 内，如果该交换机是二层交换机，则广播报文不会从一个 VLAN 直接发送到另外一个 VLAN。
- 增强 LAN 的安全性。

VLAN 间不能直接通信，即一个 VLAN 内的用户和其他 VLAN 内的用户不能直接互访，如果要访问需要通过路由器或三层交换机等三层设备。

- 组成虚拟工作组。

  用 VLAN 可以划分不同的用户到不同的工作组，当用户工作组改变时，不需要改变物理位置。在实际使用中，一般都是同一工作组的用户在一起协作，异地的情况比较少。

在交换机上，一般的端口只能属于一个 VLAN，只能识别和发送本 VLAN 的报文，但当 VLAN 跨越交换机时，就需要交换机间的端口（链路）能够同时识别和发送多个 VLAN 的报文。同样的问题也存在于支持 VLAN 的交换机和路由器之间，这样的链路称为 Trunk。

Trunk 链路的意义如下：

- 中继，把 VLAN 报文透明传输到互联的交换机或路由器，使 VLAN 得到扩展。
- 干线，一条链路上透明传输多个 VLAN。

实现 Trunk 的协议常见的有 IEEE 802.1Q（简称 dot1q），是 IEEE 的标准协议，它在原以太报文的源地址字段后增加一个 4 字节的 VLAN TAG，达到识别 VLAN 的目的。

VLAN 之间不能直接互通，为了实现 VLAN 之间的互通，必须使用支持 VLAN 路由器或三层交换机连接各个 VLAN，一般这种互通是三层（IP 层）的互通。

7.2 配置路由模式下的 VLAN

介绍路由模式下 VLAN 的配置方法。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type* { *interface-number* | *interface-number.subinterface-number* }，创建子接口并进入子接口视图。

Eudemon 工作于路由模式下时，只有以太网子接口和千兆以太网子接口下能够配置 VLAN。

步骤 3 执行命令 **ip address** *ip-address* { *mask* | *mask-length* } [**sub**]，配置子接口的 IP 地址。

子接口的 IP 地址和主接口的 IP 地址可以在同一主网段上，但其子网掩码不能相同。

步骤 4 执行命令 **vlan-type dot1q** *vlan-id*，配置子接口的封装类型及关联的 VLAN ID。

Eudemon 的一个子接口只能配置一个 VLAN ID。为了保证 VLAN 的连通性，两端的子接口关联的 VLAN ID 必须相同。

步骤 5 执行如下命令，检查配置结果。

- 执行命令 **display interface** [*interface-type* [*interface-number*]] [| { **begin** | **exclude** | **include** } *regular-expression*]，查看指定接口的状态信息。
- 执行命令 **display vlan statistics** { **vid** *vlan-id* | **interface** *interface-type interface-number.subinterface-number* } *，查看指定 VLAN 或子接口的 VLAN 报文收发统计信息。

----结束

7.3 配置透明模式下的 VLAN

介绍透明模式下 VLAN 的配置方法。

背景信息

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface***interface-typeinterface-number*，进入以太网接口视图。

Eudemon 只有千兆以太网接口支持透明模式。此处需要进入千兆以太网接口视图。

步骤 3 执行命令 **portswitch**，将接口切换到透明模式。

步骤 4 执行命令 **port default vlan** *vlan-id*，设置端口所属的缺省 VLAN。

在 VLAN 视图下配置 **port interface** *interface-type interface-number* 命令的效果与 **port default vlan** *vlan-id* 的效果相同。

步骤 5 执行命令 **port trunk allow-pass vlan** { *vlan-id1* [**to** *vlan-id2*] } & <1-10>，配置 Trunk 端口允许通过的 VLAN。

端口设置允许通过的 VLAN 后，则变为 Trunk 端口。当 Trunk 端口取消所有允许通过的 VLAN 后，则变为 Access 端口。

步骤 6 执行命令 **interface vlanif** *vlan-id*，创建 VLAN 接口。

创建 VLAN 接口时，相关联的 VLAN 必须已经存在。

步骤 7 执行命令 **ip address** *ip-address* { *mask* | *mask-length* } [**sub**]，配置 VLAN 接口的 IP 地址。

不同 VLAN 接口的 IP 地址应该在不同的网段，这样不同 VLAN 的用户之间才具有可达的路由。Eudemon 在透明模式下最多可创建 4094 个 VLAN 接口。

步骤 8 执行命令 **display interface** [*interface-type* [*interface-number*]] [| { **begin** | **exclude** | **include** } *regular-expression*]，查看指定接口的状态和统计信息。

----结束

8 配置 PPP 和配置 PPP 验证

关于本章

介绍 PPP 的基本概念以及 PPP 验证的配置方法。

8.1 PPP 简介
简单介绍 PPP 的基本概念和原理。

8.2 配置 PPP
介绍接口封装 PPP 协议及 PPP 可选参数的配置。

8.3 配置 PPP 验证
介绍 Eudemon 支持的 PAP 和 CHAP 两种 PPP 验证方式的配置。

8.4 调试 PPP
介绍调试 PPP 的方法。

8.5 清除 IPHC 压缩信息
介绍清除 IPHC 压缩信息的方法。

8.6 配置举例
介绍配置 PPP，并分别采用 PAP 和 CHAP 验证的配置举例。

8.1 PPP 简介

简单介绍 PPP 的基本概念和原理。

概述

PPP（Point-to-Point Protocol）定义了一整套的协议，其中包括：

- 链路控制协议 LCP（Link Control Protocol）
  用于建立、拆除和监控数据链路。
- 网络层控制协议 NCP（Network Control Protocol）
  用于协商数据链路上传输的数据包的格式与类型。
- 验证协议 PAP（Password Authentication Protocol），CHAP（Challenge Handshake Authentication Protocol），MSCHAP（Microsoft CHAP）v1 和 MSCHAPv2
  用于验证网络安全。

验证方式

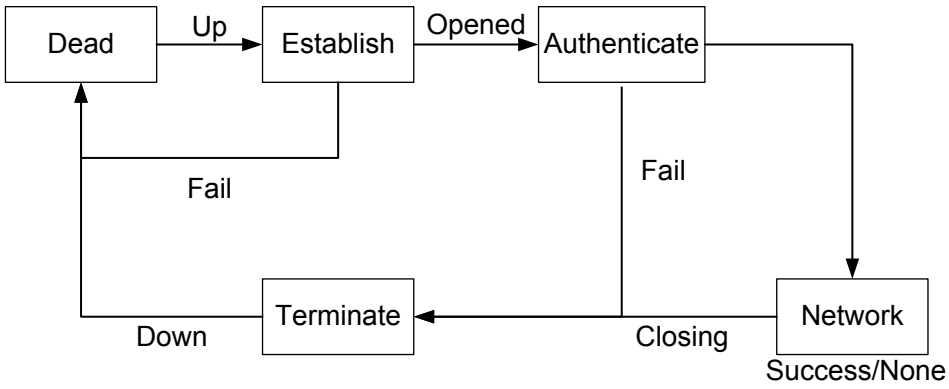
Eudemon 支持 PPP 的 PAP 和 CHAP 两种验证方式。

- PAP 验证
  PAP 验证为两次握手验证，口令为明文，PAP 验证的过程如下。
  1. 被验证方发送用户名和口令到验证方。
  2. 验证方根据本端用户表查看用户名和口令是否正确，然后返回不同的响应（Acknowledge 或者 Not Acknowledge）。
- CHAP 验证
  CHAP 验证为三次握手验证，口令为密文（密钥），CHAP 验证的过程如下。
  1. 验证方把随机产生的“质询（Challenge）”报文和本端主机名一起发送给被验证方。
  2. 被验证方收到报文后，根据验证方的用户名在本地用户列表中查找本地口令。根据查找到的口令和质询报文，通过 MD5 算法进行计算得出一个数值，并将计算得出的数值和自己的主机名发回验证方（Response）。
  3. 验证方收到 Response 后，根据其中携带的被验证方主机名，在本端用户表中查找被验证方口令字，找到匹配项后，利用质询报文和被验证方口令字，通过 MD5 算法进行计算得出一个数值，根据此数值与收到的 Response 的结果进行比较，然后返回不同的响应（接受或拒绝）。

工作流程

PPP 运行过程如图 8-1 所示。

图 8-1 PPP 运行流程图



PPP 运行流程的解释如下。

1. 开始建立 PPP 链路时，先进入到 Establish 阶段。
2. 在 Establish 阶段 PPP 链路进行 LCP 协商。协商内容包括工作方式、验证方式和最大接收单元等。LCP 在协商成功后进入 Opened 状态，表示底层链路已经建立。
3. 如果配置了验证（远端验证本地或者本地验证远端）就进入 Authenticate 阶段，开始 CHAP 或 PAP 验证。
4. 如果验证失败进入 Terminate 阶段，拆除链路，LCP 状态转为 Down。如果验证成功就进入 Network 协商阶段（NCP），此时 LCP 状态仍为 Opened，而 NCP 状态从 Initial 转到 Request。
5. NCP 协商支持 IPCP、OSCICP 等协商。IPCP 协商主要包括双方的 IP 地址。通过 NCP 协商来选择和配置一个网络层协议。只有相应的网络层协议协商成功后，该网络层协议才可以通过这条 PPP 链路发送报文。
6. PPP 链路将一直保持通信，直至有明确的 LCP 或 NCP 帧，或发生了某些外部事件（如用户干预）关闭这条链路。

参考信息

如果想更详细了解 PPP 协议信息，请参考表 8-1。

表 8-1 参考信息

文档编号	描述
RFC1661	The Point-to-Point Protocol (PPP)
RFC1334	PPP Authentication Protocols
RFC1994	PPP Challenge Handshake Authentication Protocol (CHAP)

8.2 配置 PPP

介绍接口封装 PPP 协议及 PPP 可选参数的配置。

前提条件

在配置 PPP 链路层协议之前，需完成 POS 接口或 AUX 接口物理属性的配置。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type* { *interface-number* | *interface-number.subinterface-number* }，进入接口视图。

步骤 3 执行命令 **link-protocol ppp**，配置接口封装的链路层协议为 PPP。

步骤 4 （可选）执行命令 **ppp callback** { **client** | **server** }，配置接口为回呼客户端或呼服务器端。

只有 AUX 接口支持该项配置。

步骤 5 执行命令 **ppp callback ntstring** *dial-string*，配置回呼 Eudemon 时所需要的拨号串。

步骤 6 （可选）执行命令 **ppp timer negotiate** *seconds*，配置协商超时时间间隔。

在 PPP 协商过程中，如果在配置的协商时间间隔内，没有收到对端的应答报文，PPP 将会重发前一次发送的报文。

该命令执行的前提是已经配置 PPP 验证方式。

步骤 7 （可选）执行命令 **timer hold** *seconds*,设置轮询时间间隔。

轮询时间间隔的范围为 0 ～ 32767，单位为秒，轮询时间间隔缺省值为 10 秒。

步骤 8 （可选）执行命令 **ppp lqc** *close-percentage* [*resume-percentage*]，配置 PPP 链路质量检测功能。

PPP 链路质量监测功能实时监测链路质量。当链路质量低于禁用链路质量百分比时，链路会被禁用；当链路质量恢复到恢复链路质量百分比时，链路会被自动重新启用。为防止链路在禁用和恢复之间反复振荡，重新启用链路时需要有一定的时间延迟。

只有 AUX 接口支持 PPP 链路质量检测，POS 接口不支持该项配置。

----结束

8.3 配置 PPP 验证

介绍 Eudemon 支持的 PAP 和 CHAP 两种 PPP 验证方式的配置。

8.3.1 配置 PAP 验证方式
介绍 PAP 验证的配置。

8.3.2 配置 CHAP 验证方式
介绍 CHAP 验证的配置。

8.3.1 配置 PAP 验证方式

介绍 PAP 验证的配置。

背景信息

PAP 认证方式使用明文口令。被验证方的用户名和密码可以通过 AAA 模式加入验证方的本地用户列表，也可以通过 RADIUS 服务器进行。

PAP 的单向认证方式是指通信双方，只有一方作为验证方，而另一方作为被验证方。双向认证也就是双方都既作为验证方，同时也作为被验证方。

操作步骤

- 配置验证方
  1. 在用户视图下执行命令 **system-view**，进入系统视图。
  2. 执行命令 **aaa**，进入 AAA 视图。
  3. 执行命令 **local-user** *user-name* **password** { **simple** | **cipher** } *password*，将被验证方的用户名和密码加入本地用户列表。
  4. 执行命令 **quit**，退出到系统视图。
  5. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  6. 执行命令 **ppp authentication-mode pap** [**call-in**]，配置本地以 PAP 方式验证对端。
  7. 执行命令 **restart**，重启接口。

     说明

     如果更改用户名或密码，必须在相应接口视图下执行 **restart** 或 **shutdown**、**undo shutdown** 命令才能使配置生效。
- 配置被验证方
  1. 在用户视图下执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface***interface-typeinterface-number*，进入接口视图。
  3. 执行命令 **ppp pap local-user** *user-name* **password** { **cipher** | **simple** } *password*，配置本地被对端以 PAP 方式验证时，本地发送 PAP 用户名和口令。
  4. 执行命令 **restart**，重启接口。

     说明

     如果更改用户名或密码，必须在相应接口视图下执行 **restart** 或 **shutdown**、**undo shutdown** 命令才能使配置生效。

----结束

8.3.2 配置 CHAP 验证方式

介绍 CHAP 验证的配置。

背景信息

CHAP 认证方式使用密文口令。被验证方的用户名和密码可以通过 AAA 模式加入验证方的本地用户列表，也可以通过 RADIUS 服务器进行。

CHAP 的单向认证方式是指通信双方，只有一方作为验证方，而另一方作为被验证方。双向认证也就是双方都既作为验证方，同时也作为被验证方。

说明

对于 CHAP 验证，当验证方配置用户名时，验证方和被验证方必须配置相同的密码。

操作步骤

- 配置验证方
  1. 在用户视图下执行命令 **system-view**，进入系统视图。
  2. 执行命令 **aaa**，进入 AAA 视图。
  3. 执行命令 **local-user** *user-name* **password** { **simple** | **cipher** } *password*，将对端用户名和密码加入本地用户列表。
  4. 执行命令 **quit**，退出到系统视图。
  5. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  6. 执行命令 **ppp authentication-mode chap** [**pap**] [**call-in**]，配置本地以 CHAP 方式验证对端。

     ppp authentication-mode chap pap 命令用来在 LCP 协商时优先进行 CHAP 协商，若对方不支持 CHAP，将会进行 PAP 协商。如果被验证方对这两种方式都不支持，协商将无法通过。这条命令并不表示在一次 PPP 协商过程同时协商两种验证方式。
  7. （可选）执行命令 **ppp chap user** *user-name*，配置本地用户名。
  8. 执行命令 **restart**，重启接口。

     说明

     如果更改用户名或密码，必须在相应接口视图下执行 **restart** 或 **shutdown**、**undo shutdown** 命令才能使配置生效。
- 配置被验证方
  1. 在用户视图下执行命令 **system-view**，进入系统视图。
  2. 执行命令 **aaa**，进入 AAA 视图。
  3. 执行命令 **local-user** *user-name* **password** { **simple** | **cipher** } *password*，将对端用户名和密码加入本地用户列表。
  4. 执行命令 **quit**，退出到系统视图。
  5. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
  6. （可选）执行命令 **ppp chap user** *user-name*，配置本地用户名。
  7. 执行命令 **restart**，重启接口。

     说明

     如果更改用户名或密码，必须在相应接口视图下执行 **restart** 或 **shutdown**、**undo shutdown** 命令才能使配置生效。

----结束

8.4 调试 PPP

介绍调试 PPP 的方法。

注意

启动调试开关将影响系统的性能。调试完毕后，请执行 **undo debugging all** 命令关闭调试开关。

在 PPP 运行出现故障时，请在用户视图下执行下面的 **debugging** 命令进行调试，查看调试信息，定位故障原因。

调试 PPP 相关操作如表 8-2 所示。

表 8-2 调试 PPP 相关操作

操作	命令
打开所有 PPP 调试开关	**debugging ppp all** [**verbose**] [**interface** *interface-type interface-number*]
打开 PPP 各种控制协议调试开关	**debugging ppp** { **chap** | **ipcp** | **osicp** | **pap** | **lcp** } { **all** | **error** | **event** | **packet** [**verbose**] | **state** } [**interface** *interface-type interface-number*]
打开 PPP 各种报文调试开关	**debugging ppp** { **cbcp** | **ip** | **lqc** | **osi-npdu** | **vjcomp** } **packet** [**verbose**] [**interface** *interface-type interface-number*]
打开 PPP 核心事件调试开关	**debugging ppp core event** [**interface** *interface-type interface-number*]
打开 PPP 的 IPHC 压缩调试开关	**debugging ppp compression iphc** { **rtp** | **tcp** }

8.5 清除 IPHC 压缩信息

介绍清除 IPHC 压缩信息的方法。

注意

清除 PPP 的 IPHC（IP Header Compression）压缩后，以前的压缩信息将无法恢复，请务必仔细确认。

IP 报文头压缩协议 IPHC（IP Header Compression）是一种主机-主机协议，用于在 IP 网络上压缩语音、视频等实时多媒体业务。为了减少有效带宽的消耗，可以在链路上使用 IP 报文头压缩功能，压缩 TCP/IP 和 IP/UDT/RTP 报文头。

在确认需要清除 PPP 的 IPHC 压缩信息后，请在用户视图下执行下面的 **reset** 命令。

清除 PPP 的 IPHC 压缩相关操作如表 8-3 所示。

表 8-3 清除 PPP 的 IPHC 压缩相关操作

操作	命令
清除 PPP 的 IPHC 压缩	**reset ppp compression iphc** [**interface** *interface-type interface-number*]

8.6 配置举例

介绍配置 PPP，并分别采用 PAP 和 CHAP 验证的配置举例。

8.6.1 配置 PAP 单向验证举例
介绍配置 PPP 采用 PAP 进行单向验证的示例。

8.6.2 配置 CHAP 单向验证举例
介绍配置 PPP 采用 CHAP 进行单向验证的示例。

8.6.3 配置 CHAP 双向验证举例
介绍配置 PPP 采用 CHAP 进行双向验证的示例。

8.6.1 配置 PAP 单向验证举例

介绍配置 PPP 采用 PAP 进行单向验证的示例。

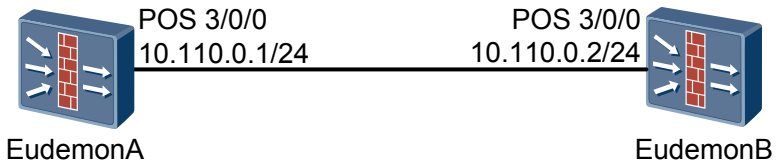
组网需求

说明

配置举例中只涉及 PPP 相关的配置命令。

如图 8-2 所示，EudemonA 和 EudemonB 之间用接口 POS 3/0/0 互连，要求 EudemonA 用 PAP 方式验证 EudemonB。

图 8-2 PAP 验证示例组网图



配置思路

配置思路如下：

1. 配置两台 Eudemon 的链路层封装协议为 PPP。
2. 配置 EudemonA 用 PAP 方式验证 EudemonB。
3. 在 EudemonB 上配置向验证方发送的用户名和明码分别为 userb、PpWwBb123。

数据规划

为完成此配置例，需准备如下的数据：

- EudemonA 和 EudemonB 的 IP 地址。
- EudemonB 在 EudemonA 上的用户名和密码。

操作步骤

步骤 1 配置 EudemonA。

将 EudemonB 的用户名和密码加入 EudemonA 的本地用户列表。

```
<EudemonA> system-view
[EudemonA] aaa
[EudemonA-aaa] local-user userb password simple PpWwBb123
[EudemonA-aaa] quit
```

配置 Pos3/0/0 的 IP 地址并封装链路层协议为 PPP。

```
[EudemonA] interface pos 3/0/0
[EudemonA-Pos3/0/0] ip address 10.110.0.1 255.255.255.0
[EudemonA-Pos3/0/0] link-protocol ppp
```

配置本地以 PAP 方式验证对端。

```
[EudemonA-Pos3/0/0] ppp authentication-mode pap
```

重启接口。

```
[EudemonA-Pos3/0/0] shutdown
[EudemonA-Pos3/0/0] undo shutdown
[EudemonA] quit
```

配置 POS 3/0/0 加入 Trust 安全区域。

```
[EudemonA] firewall zone trust
[EudemonA-zone-trust] add interface pos 3/0/0
[EudemonA-zone-trust] quit
```

配置 ACL 规则。

```
[EudemonA] acl 3005
[EudemonA-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonA-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonA] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonA-interzone-local-trust] packet-filter 3005 inbound
[EudemonA-interzone-local-trust] packet-filter 3005 outbound
```

步骤 2 配置 EudemonB。

配置接口 POS 3/0/0 的 IP 地址及封装链路协议为 PPP。

```
<EudemonB> system-view
[EudemonB] interface pos 3/0/0
[EudemonB-Pos3/0/0] ip address 10.110.0.2 255.255.255.0
[EudemonB-Pos3/0/0] link-protocol ppp
```

配置 EudemonB 向验证方 EudemonA 发送的用户名和密码。

```
[EudemonB-Pos3/0/0] ppp pap local-user userb password simple PpWwBb123
[EudemonB-Pos3/0/0] shutdown
[EudemonB-Pos3/0/0] undo shutdown
[EudemonB] quit
```

配置 POS 3/0/0 加入 Trust 安全区域。

```
[EudemonB] firewall zone trust
[EudemonB-zone-trust] add interface pos 3/0/0
[EudemonB-zone-trust] quit
```

配置 ACL 规则。

```
[EudemonB] acl 3005
[EudemonB-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonB-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonB] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonB-interzone-local-trust] packet-filter 3005 inbound
[EudemonB-interzone-local-trust] packet-filter 3005 outbound
```

----结束

结果验证

1. 在 EudemonA 上执行 **display interface** 命令，可以看到 LCP 状态为 opened。

```
<EudemonA> display interface pos3/0/0
Pos3/0/0 current state : UP
Line protocol current state : UP
Description : Huawei, Eudemon Series, Pos3/0/0 Interface
The Maximum Transmit Unit is 4470 bytes, Hold timer is 10(sec)
Internet Address is 10.110.0.1/24
Link layer protocol is PPP
LCP opened, IPCP opened
Physical layer is POS over STM-1
Scramble enabled, crc 32, clock slave, loopback not set
Port 0 :
 Hardware is POS155
 VendorName    : AGILENT
 Compliance    : OC3-SM-INTER REACH
 PartNumber    : HFCT-5760T
 Mode          : SingleMode
 LaserwaveLen  : 1310
 Length for 9/125um :    15000m
Output queue : (Urgent queue : Size/Length/Discards)  0/50/0
Output queue : (Protocol queue : Size/Length/Discards) 0/1000/0
Output queue : (FIFO queuing : Size/Length/Discards)  0/256/0
    SDH alarm:
        section layer:  none
        line    layer:  none
        path    layer:  none
        C2(Rx): 0x16  C2(Tx): 0x16
    SDH error:
        section layer:  B1 0
        line    layer:  B2 0  M1 0
        path    layer:  B3 0  G1 0
  Traffic statistics:
    Last 5 minutes input rate 6 bytes/sec, 0 packets/sec
    Last 5 minutes output rate 17 bytes/sec, 0 packets/sec
    Input: 2247 packets, 143808 bytes
           0 errors, 0 CRC, 0 packets too long
    Output:5637 packets, 360768 bytes, 0 underruns
           0 CRC, 0 aborted sequences, 0 packets too long
```

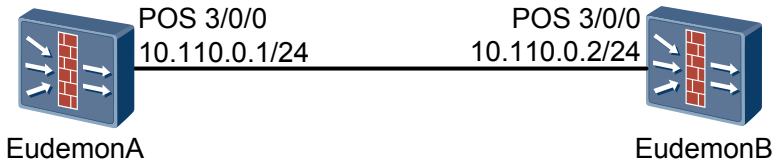
8.6.2 配置 CHAP 单向验证举例

介绍配置 PPP 采用 CHAP 进行单向验证的示例。

组网需求

如图 8-3 所示，要求 EudemonA 用 CHAP 方式验证 EudemonB。

图 8-3 CHAP 单向验证示例组网图



配置思路

配置思路如下：

1. 配置 EudemonA 和 EudemonB 的链路层封装协议为 PPP。
2. 配置 EudemonA 用 CHAP 方式验证 EudemonB。
3. 在 EudemonB 上配置向验证端发送的用户名和密码分别为 userb、PpWwBb123。

数据规划

为完成此配置例，需准备如下的数据：

- EudemonA 和 EudemonB 的 IP 地址分别为 10.110.0.1/24 和 10.110.0.2/24。
- EudemonB 在 EudemonA 上的用户名/密码为 userb/PpWwBb123。

操作步骤

步骤 1 配置 EudemonA。

将 EudemonB 的用户名和密码加入 EudemonA 的本地用户列表。

```
<EudemonA> system-view
[EudemonA] aaa
[EudemonA-aaa] local-user userb password simple PpWwBb123
[EudemonA-aaa] quit
```

配置 Pos3/0/0 的 IP 地址并封装链路层协议为 PPP。

```
[EudemonA] interface pos 3/0/0
[EudemonA-Pos3/0/0] ip address 10.110.0.1 255.255.255.0
[EudemonA-Pos3/0/0] link-protocol ppp
```

配置本地以 CHAP 方式验证对端。

```
[EudemonA-Pos3/0/0] ppp authentication-mode chap
```

配置 EudemonA 的本地用户名。

```
[EudemonA-Pos3/0/0] ppp chap user usera
```

重启接口。

```
[EudemonA-Pos3/0/0] shutdown
[EudemonA-Pos3/0/0] undo shutdown
[EudemonA] quit
```

配置 POS 3/0/0 加入 Trust 安全区域。

```
[EudemonA] firewall zone trust
[EudemonA-zone-trust] add interface pos 3/0/0
[EudemonA-zone-trust] quit
```

配置 ACL 规则。

```
[EudemonA] acl 3005
[EudemonA-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonA-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonA] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonA-interzone-local-trust] packet-filter 3005 inbound
[EudemonA-interzone-local-trust] packet-filter 3005 outbound
```

步骤 2 配置 EudemonB。

将 EudemonA 的用户名和本地密码加入 EudemonB 的本地用户列表。

```
<EudemonB> system-view
[EudemonB] aaa
[EudemonB-aaa] local-user usera password simple PpWwBb123
[EudemonB-aaa] quit
```

配置 Pos3/0/0 的 IP 地址并封装链路层协议为 PPP。

```
[EudemonB] interface pos 3/0/0
[EudemonB-Pos3/0/0] ip address 10.110.0.2 255.255.255.0
[EudemonB-Pos3/0/0] link-protocol ppp
```

配置本地被对端以 CHAP 方式验证。

```
[EudemonB-Pos3/0/0] ppp chap user userb
[EudemonB-Pos3/0/0] shutdown
[EudemonB-Pos3/0/0] undo shutdown
[EudemonB] quit
```

配置 POS 3/0/0 加入 Trust 安全区域。

```
[EudemonB] firewall zone trust
[EudemonB-zone-trust] add interface pos 3/0/0
[EudemonB-zone-trust] quit
```

配置 ACL 规则。

```
[EudemonB] acl 3005
[EudemonB-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonB-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonB] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonB-interzone-local-trust] packet-filter 3005 inbound
```

```
[EudemonB-interzone-local-trust] packet-filter 3005 outbound
```

----结束

结果验证

1. 在 EudemonA 上执行 **display interface** 命令，可以看到 LCP 状态为 opened。

```
<EudemonA> display interface pos 3/0/0Pos3/0/0 current state : UP
Line protocol current state : UP
Description : Huawei, Eudemon Series, Pos3/0/0 Interface
The Maximum Transmit Unit is 1500 bytes, Hold timer is 10(sec)
Internet Address is 10.110.0.1/24
Link layer protocol is PPP
LCP opened, IPCP opened
Output queue : (Urgent queue : Size/Length/Discards)  0/50/0
Output queue : (Protocol queue : Size/Length/Discards) 0/1000/0
Output queue : (FIFO queuing : Size/Length/Discards)  0/75/0
Interface is V35
    96 packets input,  1546 bytes
    97 packets output,  1568 bytes
```

2. 在 EudemonB 上执行 **display interface** 命令，可以看到 LCP 状态为 opened。

```
<EudemonB> display interface pos 3/0/0Pos3/0/0 current state : UP
Line protocol current state : UP
Description : Huawei, Eudemon Series, Pos3/0/0 Interface
The Maximum Transmit Unit is 1500 bytes, Hold timer is 10(sec)
Internet Address is 10.110.0.2/24
Link layer protocol is PPP
LCP opened, IPCP opened
Output queue : (Urgent queue : Size/Length/Discards)  0/50/0
Output queue : (Protocol queue : Size/Length/Discards) 0/1000/0
Output queue : (FIFO queuing : Size/Length/Discards)  0/75/0
Interface is V35
    96 packets input,  1546 bytes
    97 packets output,  1568 bytes
```

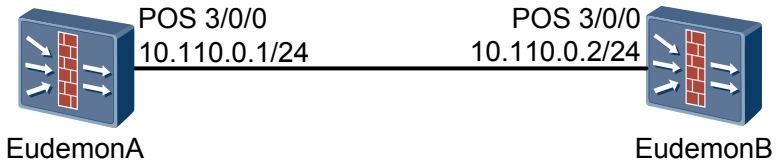
8.6.3 配置 CHAP 双向验证举例

介绍配置 PPP 采用 CHAP 进行双向验证的示例。

组网需求

如图 8-4 所示，要求 EudemonA 与 EudemonB 采用 CHAP 方式进行双向验证。

图 8-4 CHAP 双向验证示例组网图



配置思路

配置思路如下：

1. 配置 EudemonA 和 EudemonB 的本地用户列表。
2. 在 EudemonA 和 EudemonB 的接口下创建本地用户名。

3. 在 EudemonA 和 EudemonB 的接口下使能 CHAP 验证方式。

数据规划

为完成此配置，需准备如下的数据：

- EudemonA 和 EudemonB 的用户名分别是 usera 和 userb。
- EudemonA 和 EudemonB 的验证密码都是 Hello123。
- EudemonA 接口的 IP 地址是 10.110.0.1/24。
- EudemonB 接口的 IP 地址是 10.110.0.2/24。

操作步骤

步骤 1 配置 EudemonA。

将 EudemonB 的用户名和密码加入 EudemonA 的本地用户列表。

```
<EudemonA> system-view
[EudemonA] aaa
[EudemonA-aaa] local-user userb password simple Hello123
[EudemonA-aaa] quit
```

配置 Pos3/0/0 的 IP 地址并封装链路层协议为 PPP。

```
[EudemonA] interface pos 3/0/0
[EudemonA-Pos3/0/0] ip address 10.110.0.1 255.255.255.0
[EudemonA-Pos3/0/0] link-protocol ppp
```

配置 EudemonA 的本地被对方验证的用户名。

```
[EudemonA-Pos3/0/0] ppp chap user usera
```

配置采用 CHAP 方式验证 EudemonB。

```
[EudemonA-Pos3/0/0] ppp authentication-mode chap
[EudemonA-Pos3/0/0] shutdown
[EudemonA-Pos3/0/0] undo shutdown
[EudemonA] quit
```

配置 POS 3/0/0 加入 Trust 安全区域。

```
[EudemonA] firewall zone trust
[EudemonA-zone-trust] add interface pos 3/0/0
[EudemonA-zone-trust] quit
```

配置 ACL 规则。

```
[EudemonA] acl 3005
[EudemonA-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonA-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonA] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonA-interzone-local-trust] packet-filter 3005 inbound
[EudemonA-interzone-local-trust] packet-filter 3005 outbound
```

步骤 2 配置 EudemonB。

将 EudemonA 的用户名和本地密码加入 EudemonB 的本地用户列表。

```
<EudemonB> system-view
[EudemonB] aaa
[EudemonB-aaa] local-user usera password simple Hello123
[EudemonB-aaa] quit
```

配置 Pos3/0/0 的 IP 地址并封装链路层协议为 PPP。

```
[EudemonB] interface pos 3/0/0
[EudemonB-Pos3/0/0] ip address 10.110.0.2 255.255.255.0
[EudemonB-Pos3/0/0] link-protocol ppp
```

配置 EudemonB 的本地被对方验证的用户名。

```
[EudemonB-Pos3/0/0] ppp chap user userb
```

配置采用 CHAP 方式验证 EudemonA。

```
[EudemonB-Pos3/0/0] ppp authentication-mode chap
[EudemonB-Pos3/0/0] shutdown
[EudemonB-Pos3/0/0] undo shutdown
[EudemonB] quit
```

配置 POS 3/0/0 加入 Trust 安全区域。

```
[EudemonB] firewall zone trust
[EudemonB-zone-trust] add interface pos 3/0/0
[EudemonB-zone-trust] quit
```

配置 ACL 规则。

```
[EudemonB] acl 3005
[EudemonB-acl-adv-3005] rule permit ip
```

退回系统视图。

```
[EudemonB-acl-adv-3005] quit
```

进入 Trust 和 Local 域间视图。

```
[EudemonB] firewall interzone trust local
```

配置域间包过滤规则。

```
[EudemonB-interzone-local-trust] packet-filter 3005 inbound
[EudemonB-interzone-local-trust] packet-filter 3005 outbound
```

----结束

结果验证

1. 在 EudemonA 上执行 **display interface** 命令，可以看到 LCP 状态为 opened。

```
<EudemonA> display interface pos 3/0/0Pos3/0/0 current state : UP
Line protocol current state : UP
Description : Huawei, Eudemon Series, Pos3/0/0 Interface
The Maximum Transmit Unit is 1500 bytes, Hold timer is 10(sec)
Internet Address is 10.110.0.1/24
Link layer protocol is PPP
LCP opened, IPCP opened
Output queue : (Urgent queue : Size/Length/Discards)  0/50/0
Output queue : (Protocol queue : Size/Length/Discards) 0/1000/0
Output queue : (FIFO queuing : Size/Length/Discards)  0/75/0
Interface is V35
    96 packets input,  1546 bytes
    97 packets output,  1568 bytes
```

2. 在 EudemonB 上执行 **display interface** 命令，可以看到 LCP 状态为 opened。

```
<EudemonB> display interface pos 3/0/0Pos3/0/0 current state : UP
Line protocol current state : UP
Description : Huawei, Eudemon Series, Pos3/0/0 Interface
The Maximum Transmit Unit is 1500 bytes, Hold timer is 10(sec)
Internet Address is 10.110.0.2/24
Link layer protocol is PPP
LCP opened, IPCP opened
Output queue : (Urgent queue : Size/Length/Discards)  0/50/0
Output queue : (Protocol queue : Size/Length/Discards) 0/1000/0
Output queue : (FIFO queuing : Size/Length/Discards)  0/75/0
Interface is V35
    96 packets input,  1546 bytes
    97 packets output,  1568 bytes
```

9 配置 HDLC

介绍 HDLC 的基本概念和配置方法。

前提条件

在配置 HDLC 协议之前，需配置 Eudemon 同步串口的物理属性。

背景信息

HDLC（High level Data Link Control procedure）最大的特点是不需要规定数据必须是字符集，对任何一种比特流，均可以实现透明的传输。标准 HDLC 协议族中的协议运行于同步串行线路之上，如 DDN（Digital Data Network）。

HDLC 的地址字段和控制字段均是 8 比特，用来实现 HDLC 协议的各种控制信息，并标识是否是数据。

当需要建立一种面向比特的链路层协议进行同步传输时，可以采用 HDLC 协议。

操作步骤

步骤 1 在用户视图下执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface***inteface-type interface-number*，进入 POS 接口视图。

步骤 3 执行命令 **link-protocol hdlc**，配置接口封装的链路层协议为 HDLC。

步骤 4 执行命令，配置接口的 IP 地址或 IP 地址借用。

请根据实际组网情况进行配置 IP 地址。

- 执行命令 **ip address***ip-address* { *mask* | *mask-length* } [**sub**]，配置接口的 IP 地址。接口封装 HDLC 协议时，IP 地址必须与对端接口的 IP 地址在同一网段。
- 执行命令 **ip address unnumbered interface***inteface-type interface-number*，配置 IP 地址借用。

步骤 5（可选）执行命令 **timer hold***seconds*，设置轮询时间间隔。

轮询时间可使用缺省设置，也可根据网络实际情况进行调整。如果网络的延迟比较大，或拥塞程度较高，可以适当加大轮询时间间隔，以减少网络震荡的发生。轮询时间设置为 0 将不再进行轮询。

步骤 6 在任意视图下执行命令 **display interface** [*interface-type* [*interface-number*]] [| { **begin** | **exclude** | **include** } *regular-expression*]，查看接口状态。

----结束