分层 VPLS 模型中 PW 的创建和维护

刘 峰,朱恒晔,崔子筠
(同济大学 计算机系,上海 200092)

摘 要:随着 VPN 技术的发展和广泛应用,L3 VPN 技术和 L2 VPN 技术逐渐流行起来。基于 L2 VPN 的 VPLS 技术在传输速度和灵活性上的优势更为明显。为了提高 VPLS 的扩展性,文中提出了分层 VPLS 模型。通过对分层 VPLS 模型的原理的分析,使用消息通信机制有效地解决了分层 VPLS 的核心问题:PW 的创建和维护。

关键词:虚拟专用以太网服务;分层 VPLS 模型;多协议标签交换;标签分配协议;PW 标签
中图分类号:TP393.03 **文献标识码**:A **文章编号**:1005-3751(2006)02-0103-03

Establishment and Maintenance of PW in Hierarchical VPLS Model

LIU Feng, ZHU Heng-ye, CUI Zi-jun
(Computer Science Department, Tongji University, Shanghai 200092, China)

Abstract: With the development and widespread application of the VPN technique, VPLS based on L2VPN technique becomes popular. The advantage of VPLS is its speed of transmission and flexibility and the disadvantage of it is lack of scalability. In order to improve its scalability, a hierarchical VPLS model is introduced. Aimed at the character of the hierarchical VPLS model, solve the problem of the establishment and maintenance of PW in the hierarchical VPLS model.

Key words: VPLS; hierarchical VPLS model; MPLS; LDP; PW label

0 引 言

虚拟专用以太网服务(Virtual Private LAN Service, VPLS)是一种能够在核心 IP/MPLS 网络上单个桥接域中进行多站点链接的 L2 VPN 技术。VPLS 内的所有客户站点无论具体位于何处,都看似处于同一个局域网中。由于 VPLS 使用以太网接口进行客户交接,简化了局域网/广域网的边界,能够使服务的提供快速、灵活地进行[1]。由于 VPLS 解决方案要求参与 VPLS 服务的所有 PE 路由器之间通过 MPLS 隧道技术建立 LSP 全连接,对于每个 VPLS 服务来说,PE 路由器之间需要建立 $n*(n-1)$ 条 PW(Pseudowire)[2]。其缺点:

(1)PE 设备极大的信令开支。

(2)PE 路由器需要为所有的属于同一个 VPLS 实例的 PW 复制包,扩展性不好。

文中以 VPLS 的解决方案为基础,研究了分层 VPLS 模型,其优势是:消除对所有参与设备之间全网状 PW 的需要,提高 VPLS 的可扩展性。用 MTU-s 或路由器代替一部分的 PE-rs,节省了开支。主备自动切换有效地保障了错误的自动恢复,提高可维护性。分层 VPLS 模型中的核心部分 PW 的创建和维护还没有制定统一的标准。以下将从分层 VPLS 模型的原理进行分析,并用消息通信机制实现了分层 VPLS 模型中 PW 的创建和维护。

收稿日期:2005-05-02

作者简介:刘 峰(1981—),男,湖南岳阳人,硕士研究生,研究方向为网络通信与信息管理;崔子筠,研究员,同济大学计算中心主任,研究方向为网络技术与网络应用。

1 基本术语

术语 1 CE(customer edge)。指接入公网边缘设备(PE)的 VPLS 用户路由器。

术语 2 PE(provider edge)。与 VPLS 用户路由器对接的公网边缘三层交换机。在分层模型中,PE 分成 3 种类型:具有桥接功能的 MTU-s(PE-s)、路由器 PE-r 和三层交换机 PE-rs。PE-rs 与 MTU-s 连接的一侧命名为用户侧;PE-rs 与 PE-rs 连接的一侧命名为网络侧。

术语 3 VPLS 实例具有与 VPN 类似的概念。一个 VPLS 实例表示需要通过 IP/MPLS 网互相通信的多组私网。这些私网通常属于同一单位。

术语 4 PW 是建立在一对 PE 之间的虚连接,有可能多个 PW 连接共享一个 PSN 隧道[3]。它的主要作用体现在 PW 标签上。考虑入口 PE1 需要传送二层的 PDU 到出口 PE2,中间要穿过 PSN 隧道,VPLS 的功能就是将 PDU 打上 PW 标签,然后打上隧道标签,传到 PE2。PE2 凭借 PW 标签就知道此 PDU 应传送到哪个 VPLS 实例。

2 分层 VPLS 模型

一座大楼中有多个 VPLS 用户,服务供应商就必须提

供连接多个用户的中间设备。这种设备属于公网的一部分并且它将连接一个或多个 PE。有 3 种类型的中间设备可以满足要求:MTU－s;PE－r;PE－rs。分层 VPLS 模型的功能分为两部分:①VPLS 的基本功能[4];②中间设备与 PE－rs 之间的协同工作。

2.1 MTU－s 与 PE－rs 协同工作模型

MTU－s 有聚集和 2 层桥接功能,不能象 PE－rs 一样提供 VPLS 功能。它必须连接到具有 VPLS 功能的设备上。如图 1 所示,MTU－s 与核心网上的一个 PE－rs 建立了连接。同时,所有的 PE－rs 设备建立了全网状连接。对于一个 VPLS 服务来说,总有一条 PW 建立在 MTU－s 和 PE－rs 之间。其中,PW2 是冗余连接。当 PW1 无法建立时,MTU－s 就会进行主备切换以保障错误恢复。在 MTU－s 和 PE－rs 之间建立 PW 连接需要使用诸如 Q－in－Q[5]之类的封装技术。

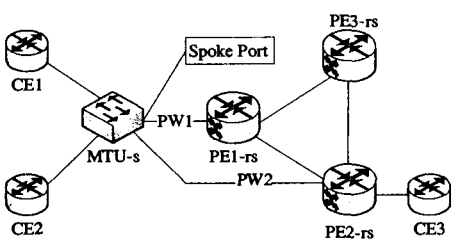

图 1 MTU－s 与 PE－rs 协同工作

考虑一个包要从 CE1 发往 CE3。核心网上的 PE－rs 已经建立好了连接,每个 PE－rs 分配了 PW 标签给其他 PE－rs,同时也已经接收到了其他 PE－rs 分配给它的 PW 标签。假设 PE2－rs 分配了 PW 标签 2001 给 PE1－rs。而 PE1－rs 与 MTU－s 之间也已经用 Q－in－Q 交换了 PW 标签。

数据传输的过程:包从 CE1 发往 MTU－s,MTU－s 根据 VLAN 标记和包所进入的物理端口,立即判断它是属于哪个 VPLS 实例。然后找出 VPLS 实例的对应的 FIB(forwarding information base)表。最后根据包中封装的目的 MAC 地址搜索 FIB 表,找到所对应的 PW 标签。如果找不到目的 MAC 地址对应的 PW 标签,MTU－s 向属于此实例的所有端口以及虚拟 Spoke 端口多播。否则,打上 PW 标签,利用 Q－in－Q 建立的 2 层隧道传输给 PE1－rs。PE1－rs 根据包所带的 PW 标签,马上判断出此包所属的 VPLS 实例,然后根据目的 MAC 地址查找此 VPLS 实例的 FIB 表,如果找到 PW 标签 2001,PE1－rs 把 2001 压入数据包并打上隧道标签,传送给 PE2－rs。否则,PE1－rs 将数据包打上各自的 PW 标签和隧道标签,多播给除发送数据给它的 MTU－s 以外的其它对端 PE。

2.2 PE－r 与 PE－rs 协同工作模型

路由器 PE－r 首先与 PE－rs 设备建立点到点的隧道 LSP。PE－r 需要为每一个享用 VPLS 服务的端口建立一条 PW。这条 PW 的起点是 PE－r 上的物理端口,终点是属于同一个 VPLS 服务的桥接模型。

数据传输过程与第一种模型基本一样。由于 PE－r 需要为一个 VPLS 服务内的每一个端口建立 PW,而 MTU－s 只需要为属于此 VPLS 实例的所有端口建立一条 PW,用 PE－r 做中间设备增加了更多的开支[6,7](如图 2 所示)。

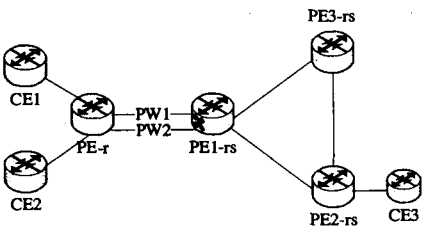

图 2 PE－r 与 PE－rs 协同工作

3 PW 标签的分配和维护

PW 的创建和维护实际上就是 PW 标签的分配和维护。VPLS 利用 LDP(Martini 方式)协议分配和维护 PW 标签,当隧道建立好且 LDP 会话建立起来后,本端 PE 收到了对端 PE 分配给自己的 PW 标签,同时对端 PE 也收到了本端 PE 的 PW 标签,PW 才创建起来。PW 的创建与建立 PW 的命令有关。同时也和以下几类消息的发送和处理相关:标签映射消息(LdpMappingMsg),标签回收消息(LdpWithdrawMsg),标签释放消息(LdpReleaseMsg),标签请求消息(LdpRequestMsg)。PW 的维护一般要处理隧道建立和断开,会话建立和断开及接口 UP 和接口 DOWN 事件。

3.1 PW 的创建过程中命令和消息的处理算法

消息通信算法中主要用到两个重要的数据结构:LocalLdpPW 结构和 RemoteLdpPW 结构。LocalLdpPW 结构中主要包括本端 PE 分给对端 PE 的 PW 标签、本端的 VPLS 的接口指针、隧道指针、封装类型、对端 PE 的 IP 地址,同时还有一个指向 RemoteLdpPW 结构的指针。而 RemoteLdpPW 结构中主要包括对端 PE 分给本端 PE 的 PW 标签及封装类型、对端 IP 地址,同时也有一个指向 LocalLdpPW 结构的指针。当本端 PE 和对端 PE 都发送过标签映射消息给对方,也就是说,两端 PE 都已经收到对方分配给自己的 PW 标签后,LocalLdpPW 结构和 RemoteLdpPW 结构就会对应起来。PW 也就建立起来。当 PE 收到一个来自对端 PE 的包后,凭借包里的入 PW 标签和 LocalLdpPW 结构与 RemoteLdpPW 结构的对应关系就能找到相应的出 PW 标签,并把出 PW 标签和数据包封装的源 MAC 地址对应起来,写入 FIB 表。

(1)分层 VPLS 在网络上运行,必须输入建立和绑定 PW 连接的命令。PW 连接命令的处理过程归结为:

输入:PW id、封装类型、对端 IP 地址、PW 标签

输出:空

Step1 检验所接收的命令消息的正确性。

Step2 如果 PW 已经建立好,返回。

Step3 把 PW id、封装类型、对端 IP 地址、PW 标签填入 LocalLdpPW 结构。

Step4 把此 LocalLdpPW 结构指针与相关接口关联。

Step5 如果隧道建立好,把 LocalLdpPW 指针与相关隧道关联。

Step6 把 LocalLdpPW 指针与对应的会话关联。

Step7 如果相关隧道和会话都已建立好,就发送 LdpMappingMsg 给对端 PE。否则,返回。

Step8 如果对端发送过 LdpMappingMsg 到本端,建立本端和对端之间的 PW。否则,发送 LdpRequestMsg。

Step9 下发数据给底层,返回。

(2)标签映射消息会将本端 PE 分配给对端 PE 的 PW 标签及其它配置消息送给对端。发送标签映射消息的条件是:隧道建立且会话建立。因此建立 PW 连接的命令、隧道建立事件、接口建立事件或者会话建立事件都有可能触发 PE 发送标签映射消息。处理标签映射消息的算法归结为:

输入:对端分配过来的 PW 标签、PW id、封装类型、对端 IP 地址等信息

输出:空

检验所接收的消息的正确性。

Step1 如果曾经收到过对端的 LdpMappingMsg,返回。

Step2 把对端分配过来的标签及其它信息存入新分配的 RemoteLdpPW 结构中。

Step3 如果本端 PE 还未输入建立连接的命令,返回。

Step4 否则为本端与对端建立一条 PW。

Step5 当标签映射消息是从 MTU－s 发给 PE－rs 时,PE－rs 多播 LdpMappingMsg 给所有对端的 PE－rs 和其它与之相连的 MTU－s。

Step6 当标签映射消息从对端 PE－rs 发送给本端 PE－rs 时,PE－rs 向与它相连的所有 MTU－s 广播 LdpMappingMsg。

Step7 下发数据给底层,返回。

(3)当命令强行断开 PW 或者隧道断开,接口断开或会话中断时,必须发送标签回收消息回收本端 PE 分配过去的旧标签,同时把与隧道、接口或者会话相关的所有 PW 断开。为此引进两个计数器 NetCount 和 UserNetCount。如果 PE 已经接收到 X 个 MTU－s 分配给它的标签,Y 个 PE－rs 分配给它的标签,同时有 Z 个属于同一实例的 CE 与之相连,那么 NetCount = $X + Z$;表示有多少个与此 PE 连接的设备需要网络侧的 PW 连接;UserNetCount = $X + Y + Z$ 表示有多少个与此 PE 连接的设备既需要用户侧又需要网络侧的 PW 连接。处理标签回收消息的算法归结为:

输入:PW id、封装类型、对端 IP 地址等信息

输出:空

Step1 检验所接收的消息的正确性。

Step2 如果对端 PE 没有发送标签给本端,错误返回。

Step3 如果 PW 未建立起来,删除相应的 RemoteLdpPW 结构。释放对端分配过来的标签,发送标签释放消息,返回。

Step4 断开相应的 PW,删除相应的 RemoteLdpPW 结构,并通知底层 PW 连接状态改变。

Step5 计算 NetCount 和 UserNetCount。

Step6 如果 LdpWithDrawMsg 来自 MTU－s 且 UserNetCount 的值为 1,PE 发送标签 LdpWithDrawMsg 回收所有分配给对端 PE 的标签。

Step7 如果消息来自 PE－rs 且 NetCount 的值为 1,本端 PE 发送 LdpWithDrawMsg 回收自己分配给 MTU－s 设备的标签。

Step8 返回。

(4)当本端 PE 收到与对端 PE 建立 PW 连接的命令后,本端 PE 会发送标签映射消息给相应的对端 PE。如果对端 PE 没有发送标签映射消息过来,它可以发送标签请求消息给对端 PE。标签请求消息的处理过程描述为:

输入:PW id、封装类型、对端 IP 地址等信息

输出:空

Step1 检验所接收的消息的正确性。

Step2 如果本端 PE 还没有为对端 PE 分配标签,错误返回。

Step3 如果 LSP 隧道没有建立好或者会话没有建立好,错误返回。

Step4 如果对端 PE 已经发送了 LdpMappingMsg 给本端 PE,建立本端与对端的 PW 连接。

Step5 发送 LdpMappingMsg。

Step6 下发数据给底层。

Step7 返回。

当本端 PE 处理对端 PE 的标签回收消息之后,本端 PE 会发送标签释放消息通知对端。标签释放消息的处理过程很简单,主要是去除本端 PE 为对端分配的标签,重新分配一个新标签给相应的对端 PE。

3.2 PW 的维护

PW 的维护主要处理隧道建立和断开,会话建立和断开及接口 UP 和接口 DOWN 事件。处理过程比较简单。多个 PW 可以共用一个隧道,当隧道断开,必须去除 PW 和隧道之间的关系。而当隧道建立时,如果本端 PE 满足发送标签映射消息的条件,便发送标签映射消息。如果对端 PE 也发送了 PW 标签给本端,则建立 PW,然后处理 PW 与隧道的关联问题。处理会话和接口事件的方法与其类似。

4 结束语

文中分析了分层 VPLS 的工作原理,并通过对标签映射消息、标签回收消息、标签释放消息、标签请求消息及隧道、会话、接口事件的分析和处理,有效地实现了分层 VPLS 模型中的核心部分:PW 的创建及维护。并给出了具体实现方案。相对于 VPLS 来说,分层 VPLS 实例内的

(下转第 109 页)

```
…
    <process:If-Then-Else rdf:ID="If1">
     <process:ifCondition>
      <expr:Condition rdf:ID="HaveAirOrNo"></expr:
Condition>
     </process:ifCondition>
     <process:then>
      <process:Perform rdf:ID="IfTrue">
       <process:process>
        <process:AtomicProcess rdf:ID="AccountService"/
>
       </process:process>
      </process:Perform>
     </process:then>
     <process:else>
      <process:Perform rdf:ID="IfFalse">
       <process:process rdf:resource="#ExceptionService"/
>
      </process:Perform>
     </process:else>
    </process:If-Then-Else>
   </process:ControlConstructList>
   <process:Perform rdf:ID="perform1">
    <process:process>
     <process:AtomicProcess rdf:ID="AirSupplyService"/>
    </process:process>
   </process:Perform>
  </process:ControlConstructList>
</rdf:RDF>
```

(2)业务过程的执行。

根据 AirOnLine 系统中机票销售业务过程描述，在服务注册中心中查找匹配 Web 服务，并选择调用最合适的服务。其部分执行代码如下：

```
public RunAirOnline(){.
  reader=OWLSFactory.createOWLSReader();
  exec=OWLSFactory.createExecutionEngine();
…
public void runAirSupply() throws Exception {
  service=reader.read(URI.create("http://localhost:8080/test/
AirOnline.owl"));
  process=service.getProcess();
  emptyvalues=OWLSFactory.createValueMap();
…
  values=exec.execute(process, values);
…
  }
}
```

4　结束语

文中设计了一种基于语义 Web 服务的业务过程集成模型，并给出了一个基于该模型的系统实例的设计和实现。语义 Web 服务技术不仅使服务之间能够理解相互交互的内容，而且为业务过程建模提供了工具，把 Web 服务和业务过程有机地结合起来实现企业间的过程层集成。这样不仅支持跨企业的业务合作，而且使在本领域内有共同的语义基础，增强了企业间业务集成能力和协调能力，使过程层集成朝着自动化的方向发展。

参考文献：

[1] 吕庆中，韩燕波，麦中凡. Web 服务环境中的业务过程建模语言比较框架[J]. 计算机工程与应用，2003，23：7－9.

[2] 崔　航，马殿富，王　勇，等. 组合 Web 服务建模工具的设计与实现[J]. 北京航空航天大学学报，2003，29(10)：943－945.

[3] Kavantzas N，Burdett D，Ritzinger G. Web Services Choreography Description Language Version 1.0. W3C Working Draft [M]. [s.l.]：World Wide Web Consortium，2004.

[4] 宋　炜，张　铭. 语义网简明教程[M]. 北京：高等教育出版社，2004.

[5] Martin D. OWL－S：Semantic Markup for Web Services [EB/OL]. http://www.daml.org/services/owl－s/1.1/owl－s/owl－s.html，2004－08.

[6] 谢小轩，张　浩，夏敬华，等. 企业应用集成综述[J]. 计算机工程与应用 2002，22：1－4.

(上接第 105 页)

传输速率基本在 10M～1G 之间，而且分层 VPLS 消除了 PW 的全网状连接，从而明显减少了信令开支和数据复制管理费用，其可扩展性能好。

参考文献：

[1] 吴　昱、胡　钧. VPLS——二层虚拟专用网业务的最佳选择[N]. 通讯世界，2004－05－04(10)：35－37.

[2] Andersson L，Rosen E C. Framework for Layer 2 Virtual Private Networks [DB/OL]. http://bgp.potaroo.net/ietf/ids/draft－ietf－l2vpn－l2－framework－05.txt，2004－03.

[3] Rosen E，Viswanathan A. Multiprotocol Label Switching. Architecture. RFC3031 [DB/OL]. Http://www.ietf.org/rfc/rfc3031.txt，2001.

[4] Lasserre M，Kompella V. Virtual Private LAN Services over MPLS [DB/OL]. http://bgp.potaroo.net/ietf/ids/draft－ietf－l2vpn－vpls－ldp－05.txt，2004－04.

[5] Cisco System Working Group. IEEE 802.1Q－in－Q VLAN Tag Termination [DB/OL]. http://www.cisco.com/en/US/products/sw/iosswrel/ps5207/products_feature_guide09186a00801f0f4a.html，2004－03.

[6] Martini L，Rosen E C. Pseudowire Setup and Maintenance using LDP [DB/OL]. http://www.ietf.org/internet－drafts/draft－ietf－pwe3－control－protocol－14.txt，2004－09.

[7] Boscher C，Cheval P. LDP State Machine. RFC3215 [DB/OL]. Http://www.ietf.org/rfc/rfc3215.txt，2002－01.