

Cisco IOS中的虛擬訪問PPP功能

目錄

[簡介](#)

[開始之前](#)

[慣例](#)

[必要條件](#)

[採用元件](#)

[IPS簽名提示](#)

[虛擬接入介面概述](#)

[虛擬存取介面的應用程式](#)

[多重連結PPP](#)

[L2F](#)

[VPDN](#)

簡介

本檔案介紹Cisco IOS®中虛擬存取PPP應用程式的整體架構。有關特定功能的詳細資訊，請參閱辭彙表結尾列出的文檔。

開始之前

慣例

如需文件慣例的詳細資訊，請參閱[思科技術提示慣例](#)。

必要條件

本文件沒有特定先決條件。

採用元件

本文件所述內容不限於特定軟體和硬體版本。

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您在即時網路中工作，請確保在使用任何命令之前瞭解其潛在影響。

IPS簽名提示

下列術語將出現在本文檔中。

- **訪問服務器**：Cisco Access Server平台，包括用於提供遠端訪問的ISDN和非同步介面。
- **L2F**:第2層轉送通訊協定 (實驗草案RFC)。這是用於多機箱MP和虛擬專用網路(VPN)的基礎鏈路級技術。
- **Link**:由系統提供的連線點。它可以是專用硬體介面 (例如非同步介面) 或多通道硬體介面 (例如PRI或BRI) 上的通道。
- **MP**:多重連結PPP通訊協定 (請參閱RFC 1717)。
- **多機箱MP**:MP + SGBP + L2F + Vtemplate。
- **PPP**:點對點通訊協定 (請參閱RFC 1331)。
- **Rotary Group**:分配給撥出或接收呼叫的一組物理介面。該組就像一個池，任何鏈路都可用於從該池撥出或接收呼叫。
- **SGBP**:堆疊群組投標通訊協定
- **堆疊組**:將配置為作為一個組運行並支援不同系統上具有連結的MP捆綁包的兩個或多個系統的集合。
- **VPDN**:虛擬專用撥接網路。將PPP鏈路從網際網路服務提供商(ISP)轉發到家庭網關。
- **Vtemplate**:虛擬模板介面。

註：有關本文檔中引用的RFC的資訊，請參閱[Cisco IOS版本11.2中支援的RFC](#) (產品公告) ；或[獲取直接到InterNIC的連結的RFC和其他標準檔案](#)。

虛擬接入介面概述

在Cisco IOS版本11.2F中，Cisco支援以下撥號存取功能：VPDN、多機箱多重連結、VP、使用虛擬存取的通訊協定轉譯和PPP/ATM。這些功能使用虛擬介面在其目標電腦上傳輸PPP。

虛擬訪問介面是Cisco IOS介面，與串列介面等物理介面一樣。串列介面配置位於串列介面配置中。

```
#config
  int s0
  ip unnumbered e0
  encaps ppp
  :
```

物理介面具有靜態、固定的配置。但是，虛擬訪問介面是按需動態建立的 (本文檔下一部分將討論各種用途)。當不再需要它們時，它們也會被釋放。因此，**必須通過其他方式**錨定虛擬接入介面的配置源。

虛擬存取獲取其組態的各種方法是透過位於驗證伺服器上的**虛擬範本**介面和/或RADIUS和TACAC+記錄。後一種方法稱為**每使用者虛擬配置檔案**。由於虛擬訪問介面可以使用全域性虛擬模板進行配置，因此各種使用者的虛擬訪問介面可以從一個虛擬模板介面繼承相同的配置。例如，網路管理員可以選擇為系統的所有Virtual Access使用者定義一個公用PPP身份驗證方法(CHAP)。對於**特定每個使用者**定製的配置，網路管理員可以定義特定於虛擬配置檔案中的使用者的介面配置 (如PAP身份驗證)。簡而言之，虛擬接入介面可用的通用到特定配置方案允許網路管理員定製對所有使用者通用和/或針對使用者單獨定製的介面配置。

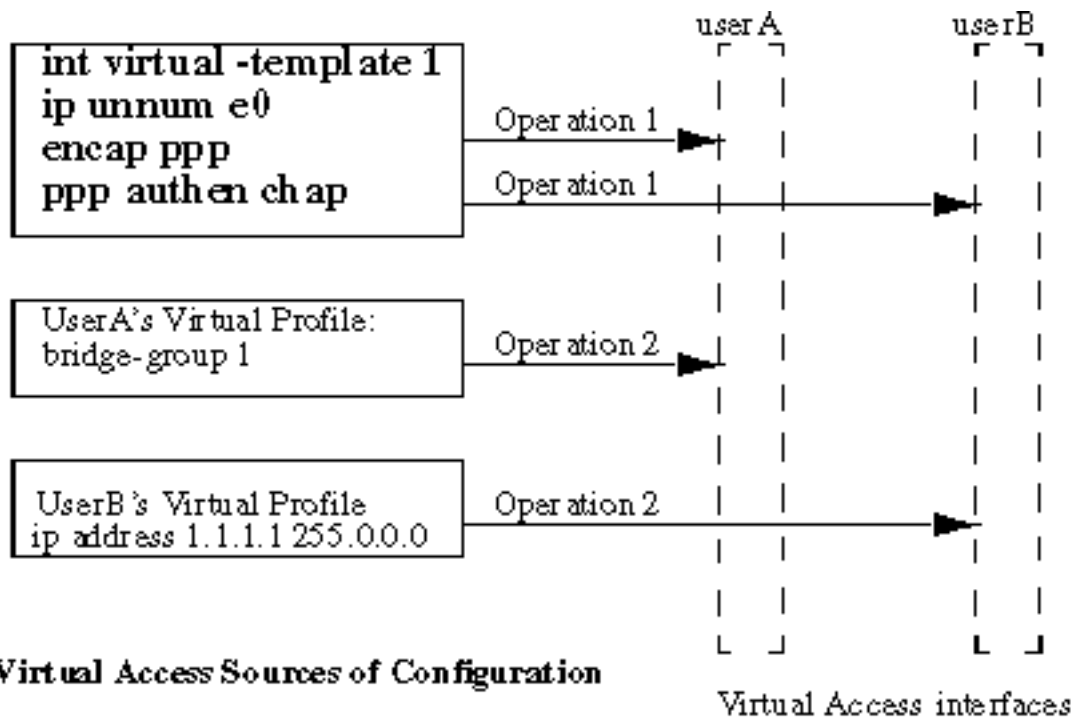


Figure 1. Virtual Access Sources of Configuration

上面圖1顯示了使用者A和使用者B的兩個虛擬訪問介面。操作1表示將介面配置從**全域性虛擬模板**介面應用到兩個虛擬訪問介面。操作2表示從不同的虛擬配置檔案對兩個**虛擬**訪問介面應用每個使用者介面配置。

虛擬存取介面的應用程式

本節介紹Cisco IOS使用虛擬訪問介面的各種方式。

您會注意到每個應用程式的重複主題 — 它們允許特定於應用程式（操作1）的通用虛擬模板。然後對每個使用者應用每個使用者的虛擬配置檔案（操作2）

多重連結PPP

多重連結PPP使用虛擬存取介面作為套件組合介面，來重組透過個別連結接收之封包，並對透過個別連結傳送之封包進行分段。套件組合介面會從特定於多重連結PPP的虛擬模板中獲取其組態。如果網路管理員選擇啟用虛擬設定檔，則每個使用者名稱的虛擬設定檔介面組態會套用到該使用者的套件組合介面。

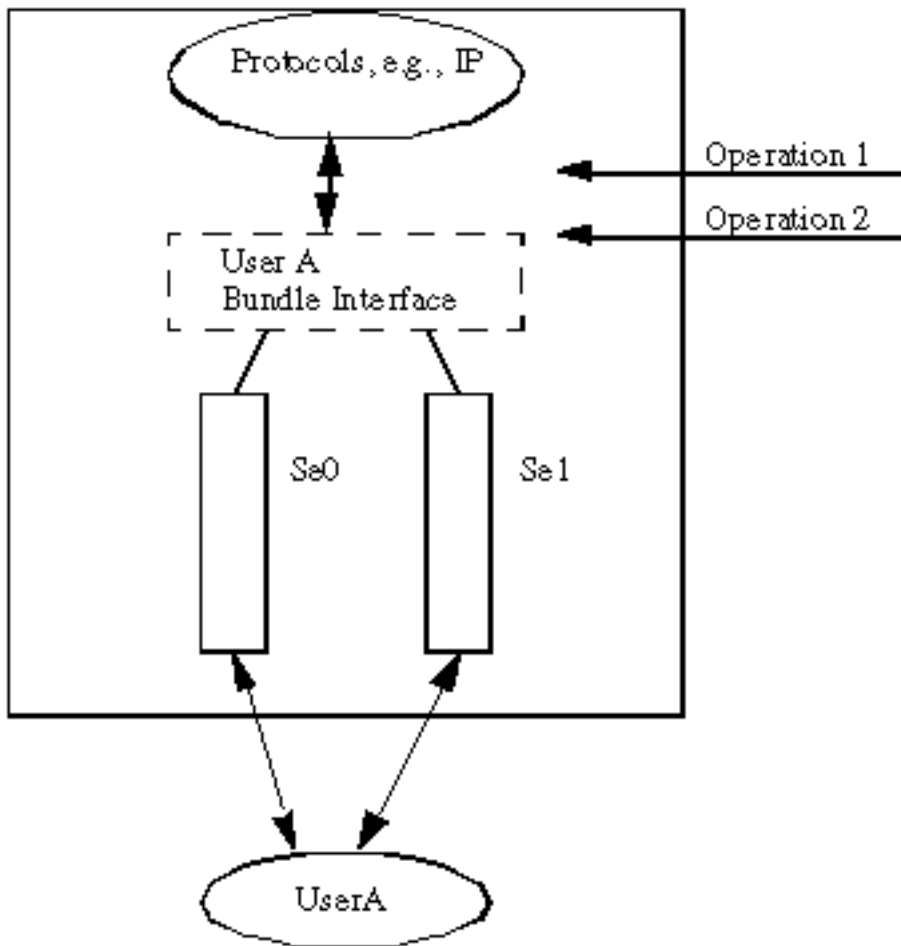


Figure 2. Multilink PPP Bundle Interface

圖2描述了串列介面的多鏈路PPP的使用。由於沒有Dialer介面，虛擬模板介面由以下項定義：

```

multilink virtual-template 1

  int virtual-template 1
  ip unnum e0
  encaps ppp
  ppp chap authen

```

然後，可選的每使用者名稱虛擬配置檔案配置將應用到捆綁包介面。涉及撥號器介面時，套件組合介面是被動介面 — 無需虛擬範本介面。

例如，下圖3描述了配置為支援多鏈路PPP的PRI se0:23。

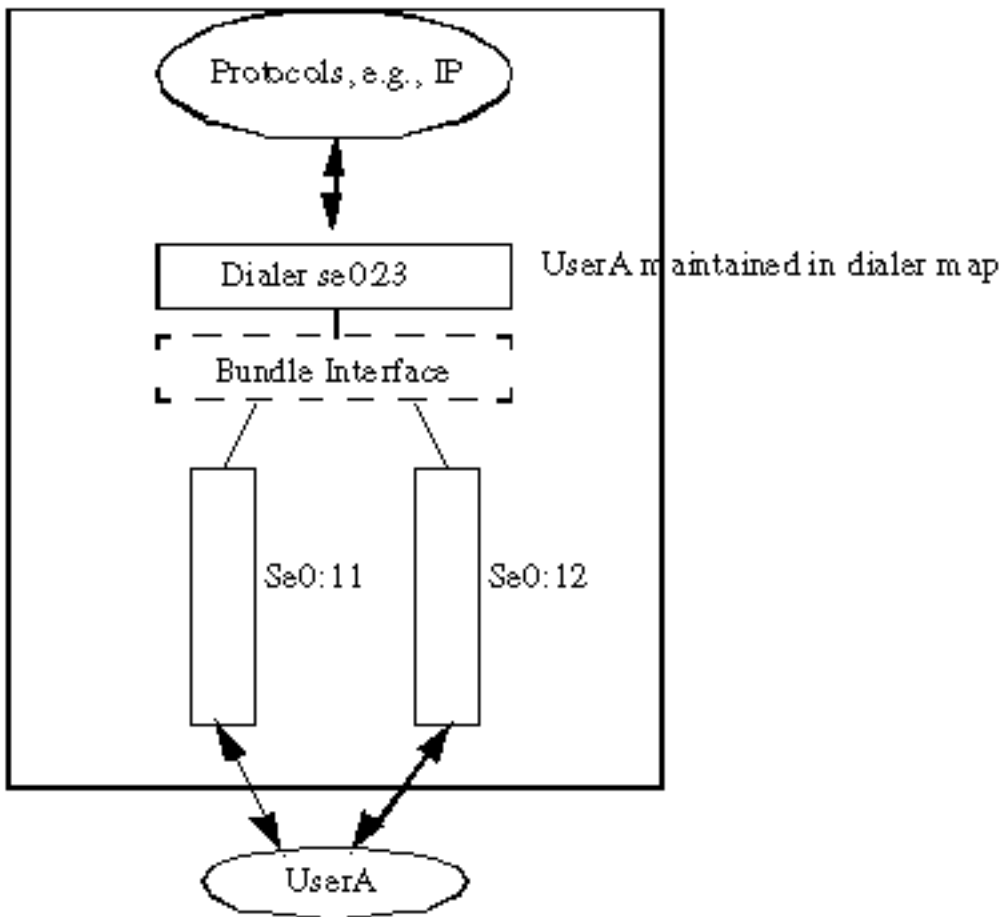


Figure 3. Multilink PPP Interface (Passive)

請注意，如果已啟用虛擬配置檔案，則方案將還原圖2中所示的方案。也就是說，如果在撥號器介面上收到傳入呼叫並啟用了虛擬配置檔案，則配置源將不再來自撥號器。相反，捆綁包介面（見圖2）是所有協定將讀取或寫入的「活動」介面。配置的來源首先是虛擬模板介面，然後是特定使用者的虛擬配置檔案。

L2F

鏈路級第2層轉發（或L2F）允許在遠端目標上終止PPP。通常，如果不使用L2F，PPP是在撥入的客戶端和應答傳入呼叫的NAS之間進行的。使用L2F時，PPP被投影到目標節點。就客戶端而言，它「認為」它通過PPP連線到目的節點。實際上，NAS將成為簡單的PPP幀轉發器。在L2F術語中，目標節點稱為Home-Gateway。

在家庭網關上，虛擬接入介面用於終止PPP鏈路。同樣，虛擬模板用作配置源。如果定義了虛擬配置檔案，則每個使用者介面配置將應用於虛擬訪問介面。

L2F隧道當前通過UDP/IP傳播。

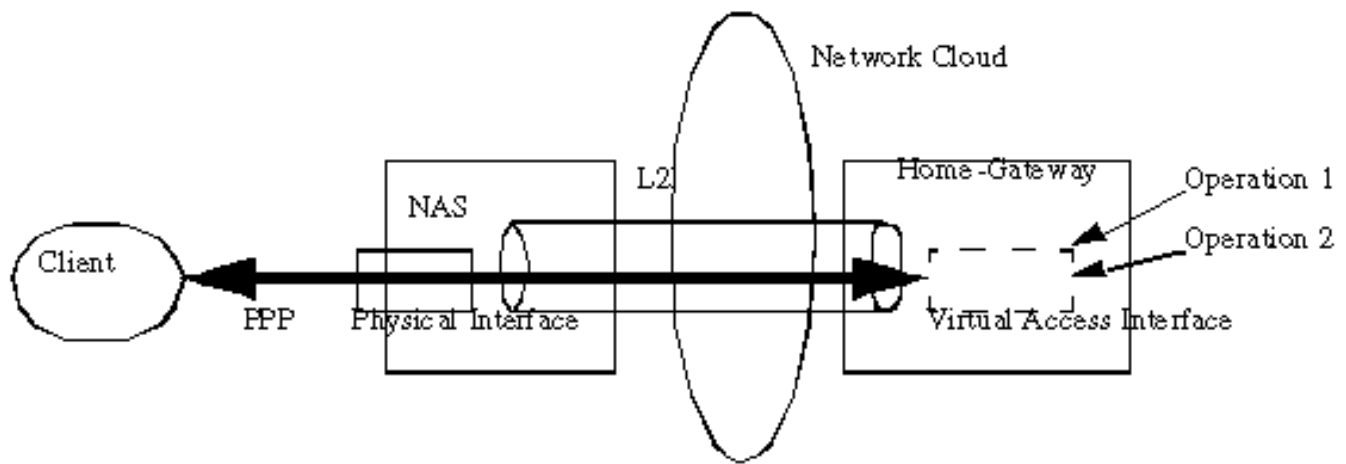


Figure 4. Client PPP to the Home-Gateway via a L2F Tunnel

L2F通道技術目前用於兩種Cisco IOS 11.2功能：VPDN(虛擬專用撥接網路)和多機箱多重連結PPP(MMP)。

VPDN

VPDN允許私人網路從使用者端直接跨越到所選的家庭閘道。例如，HP的移動使用者（例如銷售人員）希望能夠隨時隨地連線到選擇的HP Home-Gateway。HP將簽約支援PDN的ISP。這些ISP的配置方式是：如果jsmith@hp.com撥入任何ISP提供的號碼，NAS將自動轉發到HP家庭網關。因此，ISP無需管理HP使用者的IP地址、路由以及與HP使用者群關聯的其他功能。ISP HP的管理已簡化為HP家庭網關的IP連線問題。

NAS:isp

```
vpdn outgoing hp.com isp ip 1.1.1.2
```

Home-Gateway:hp-gateway

```
int virtual-template 1
 ip unnum e0
 encaps ppp
 ppp chap authen

vpdn incoming isp hp-gateway virtual-template 1
```

多機箱

PPP Multilink為使用者提供按需提供的額外頻寬，並能夠通過由多個鏈路組成的邏輯管道（捆綁）拆分和重新組合資料包。這減少了慢速WAN鏈路上的傳輸延遲，並且還提供了增加最大接收單位的方法。單個訪問伺服器環境支援多鏈路。

例如，ISP希望方便地為跨多個接入伺服器的多個PRI分配一個旋轉編號，該編號可擴展且靈活以滿足其業務需求。

使用多機箱多鏈路，來自同一客戶端的多條多鏈路鏈路可能終止於不同的接入伺服器。雖然同一捆綁包中的各個MP鏈路實際上可能終止於不同的訪問伺服器，但就MP客戶端而言，它就像終止於單

個訪問伺服器一樣。當元件與VPDN的元件進行比較時，多機箱差異僅在於使用額外的StackGroup投標協定(SGBP)來促進多鏈路捆綁包的投標和仲裁。在通過SGBP確定堆疊組勝出者的目標IP地址後，多機箱使用L2F從NAS投射到另一個NAS（其中一個是堆疊組勝出者）。

例如，在堆疊組上呼叫兩個NAS的stackq:nasa和nasb。

nasa:

```
username stackq password hello
multilink virtual-template 1

int virtual-template 1
ip unnum e0
encap ppp
ppp authen chap

sgbp stack stackq
sgbp member nasb 1.1.1.2
```

nasb:

```
username stackq password hello
multilink virtual-template 1

int virtual-template 1
ip unnum e0
encap ppp
ppp authen chap

sgbp stack stackq
sgbp member nasb 1.1.1.2
```

通訊協定轉譯

協定轉換允許網關上的PPP封裝流量（如X.25/TCP）作為虛擬訪問介面終止（兩步轉換）。通過一步轉換也支援虛擬訪問介面。

兩步協定轉換示例：

```
int virtual-template 1
ip unnum e0
encap ppp
ppp authen chap

vty-async virtual-template 1
```

一步式通訊協定轉譯範例：

```
int virtual-template 1
ip unnum e0
encap ppp
```

```
ppp authen chap

translate tcp 1.1.1.1 virtual-template 1
```

使用ATM的PPP

此功能支援在根據思科(StrataCom)幀轉發封裝格式化資料時在路由器ATM介面上終止多個PPP連線。PPP協定在路由器上終止，就像是從典型的PPP串列介面接收一樣。每個PPP連線將封裝在單獨的ATM VC中。使用其他封裝型別的VC也可在同一介面上配置。

```
interface Virtual-Template1
 ip unnumbered e0/0
 ppp authentication chap

interface ATM2/0.2 point-to-point
 atm pvc 34 34 34 aal5ppp virtual-template 1
```

虛擬設定檔

虛擬配置檔案是一個獨特的PPP應用程式，為撥入路由器的使用者定義並應用每個使用者的配置資訊。虛擬配置檔案允許應用使用者特定的配置資訊，而不管用於撥入呼叫的介質是什麼。虛擬配置檔案的配置資訊可以來自虛擬介面模板、儲存在AAA伺服器上的每個使用者配置資訊，或同時來自兩者，具體取決於路由器和AAA伺服器的配置方式。虛擬配置檔案的應用可以在單盒環境中、VPDN家庭網關或多機箱環境中。

要將虛擬模板定義為虛擬配置檔案的配置源，請執行以下操作：

```
virtual-profile virtual-template 1
 int virtual-template 1
 ip unnum e0
 encaps ppp
 ppp authen chap
 :
```

將AAA定義為虛擬配置檔案的配置源：

```
virtual-profile aaa
```

在本例中，系統管理員決定過濾通告給John的路由，並將訪問清單應用於Rick的撥入連線。當John或Rick通過介面S1或BRI 0撥入並進行身份驗證時，會建立虛擬配置檔案：路由過濾器應用於John，訪問清單應用於Rick。

使用者John和Rick的AAA配置：

```
john Password = ``welcome''
 User-Service-Type = Framed-User,
 Framed-Protocol = PPP,
 cisco-avpair = ``ip:rte-fltr-out#0=router igrp 60'',
 cisco-avpair = ``ip:rte-fltr-out#3=deny 171.0.0.0 0.255.255.255'',
 cisco-avpair = ``ip:rte-fltr-out#4=deny 172.0.0.0 0.255.255.255'',
 cisco-avpair = ``ip:rte-fltr-out#5=permit any''
 rick Password = ``emoclew''
```



```
User-Service-Type = Framed-User,  
Framed-Protocol = PPP,  
  cisco-avpair = ``ip:inacl#3=permit ip any any precedence immediate'',  
  cisco-avpair = ``ip:inacl#4=deny igmp 0.0.1.2 255.255.0.0 any'',  
  cisco-avpair = ``ip:outacl#2=permit ip any any precedence immediate'',  
  cisco-avpair = ``ip:outacl#3=deny igmp 0.0.9.10 255.255.0.0 any''
```

簡而言之，AAA **cisco-avpairs** 包含要應用於特定使用者的 Cisco IOS per-interface 命令。