International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Y.1314

(10/2005)

SERIES Y: GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS

Internet protocol aspects – Transport

Virtual private network functional decomposition

ITU-T Recommendation Y.1314

ITU-T Y-SERIES RECOMMENDATIONS

GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS

GLOBAL INFORMATION INFRASTRUCTURE	
General	Y.100–Y.199
Services, applications and middleware	Y.200–Y.299
Network aspects	Y.300–Y.399
Interfaces and protocols	Y.400–Y.499
Numbering, addressing and naming	Y.500–Y.599
Operation, administration and maintenance	Y.600–Y.699
Security	Y.700–Y.799
Performances	Y.800–Y.899
INTERNET PROTOCOL ASPECTS	
General	Y.1000–Y.1099
Services and applications	Y.1100–Y.1199
Architecture, access, network capabilities and resource management	Y.1200–Y.1299
Transport	**Y.1300–Y.1399**
Interworking	Y.1400–Y.1499
Quality of service and network performance	Y.1500–Y.1599
Signalling	Y.1600–Y.1699
Operation, administration and maintenance	Y.1700–Y.1799
Charging	Y.1800–Y.1899
NEXT GENERATION NETWORKS	
Frameworks and functional architecture models	Y.2000–Y.2099
Quality of Service and performance	Y.2100–Y.2199
Service aspects: Service capabilities and service architecture	Y.2200–Y.2249
Service aspects: Interoperability of services and networks in NGN	Y.2250–Y.2299
Numbering, naming and addressing	Y.2300–Y.2399
Network management	Y.2400–Y.2499
Network control architectures and protocols	Y.2500–Y.2599
Security	Y.2700–Y.2799
Generalized mobility	Y.2800–Y.2899

For further details, please refer to the list of ITU-T Recommendations.

ITU-T Recommendation Y.1314

Virtual private network functional decomposition

Summary

This Recommendation describes the set of functions required to establish, operate and maintain client/server and peer level Virtual Private Networks (VPNs). The network functionality is described from a network level viewpoint, taking into account the VPN network layered structure, client characteristic information, client/server associations, networking topology and layer network functionality.

The functional models are described using the modelling methodology described in ITU-T Recs G.805 and G.809. The modelling methodology employed is network technology-independent and therefore the functional models and associated functions described apply to all VPN layer network technologies.

Source

ITU-T Recommendation Y.1314 was approved on 14 October 2005 by ITU-T Study Group 13 (2005-2008) under the ITU-T Recommendation A.8 procedure.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

© ITU 2006

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

CONTENTS

ITU-T Recommendation Y.1314

Virtual private network functional decomposition

1 Scope

This Recommendation describes the set of functions required to establish, operate and maintain client/server and peer level Virtual Private Networks (VPNs). The network functionality is described from a network level viewpoint, taking into account the VPN network layered structure, client characteristic information, client/server associations, networking topology and layer network functionality. The functional models are described using the network technology-independent modelling methodology described in ITU-T Recs G.805 and G.809.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

– ITU-T Recommendation G.805 (2000), *Generic functional architecture of transport networks*.

– ITU-T Recommendation G.809 (2003), *Functional architecture of connectionless layer networks*.

– ITU-T Recommendation G.8010/Y.1306 (2004), *Architecture of Ethernet layer networks*.

– ITU-T Recommendation Y.1311 (2002), *Network-Based VPNs – Generic architecture and service requirements*.

3 Definitions

This Recommendation uses the following terms defined in ITU-T Rec. G.805:

3.1 access point

3.2 access group

3.3 adapted information

3.4 characteristic information

3.5 client/server relationship

3.6 connection

3.7 connection point

3.8 layer network

3.9 link

3.10 link connection

3.11 matrix

3.12 network

3.13 network connection

3.14 port

3.15 reference point

3.16 subnetwork

3.17 subnetwork connection

3.18 termination connection point

3.19 trail

3.20 trail termination

3.21 transport

3.22 transport entity

3.23 transport processing function

3.24 unidirectional connection

3.25 unidirectional trail

This Recommendation uses the following terms defined in ITU-T Rec. G.809:

3.26 access point

3.27 access group

3.28 adapted information

3.29 characteristic information

3.30 client/server relationship

3.31 connectionless trail

3.32 flow

3.33 flow domain

3.34 flow domain flow

3.35 flow point

3.36 flow point pool

3.37 flow termination

3.38 flow termination sink

3.39 flow termination source

3.40 layer network

3.41 link flow

3.42 network

3.43 network flow

3.44 port

3.45 reference point

3.46 traffic unit

3.47 transport

3.48 transport entity

3.49 transport processing function

3.50 termination flow point

This Recommendation uses the following term defined in ITU-T Rec. G.8010/Y.1306:

3.51 flow domain fragment

This Recommendation uses the following terms defined in ITU-T Rec. Y.1311:

3.52 Layer 1 VPN

3.53 Layer 2 VPN

3.54 Layer 3 VPN

This Recommendation defines the following terms:

3.55 VPN client layer network: A topological component in a client/server VPN that represents the set of access points of the same type associated for the purpose of transferring VPN client layer characteristic information.

3.56 VPN server layer network: A topological component in a client/server VPN that represents the set of access points of the same type associated for the purpose of transferring adapted VPN client layer information.

3.57 VPN peer layer network: A topological component that represents the set of access points of the same type associated for the purpose of transferring VPN peer layer characteristic information.

4 Abbreviations and Acronyms

This Recommendation uses the following abbreviations and acronyms:

AAA	Authentication, Authorization and Accounting
AAL	ATM Adaptation Layer
AG	Access Group
AI	Adapted Information
AIS	Alarm Indication Signal
AP	Access Point
ASON	Automatically Switched Optical Network
ATM	Asynchronous Transfer Mode
BFD	Bidirectional Forwarding Detection
BGP	Border Gateway Protocol
CAC	Connection Admission Control
CBR	Constant Bit Rate
CC	Connectivity Check
CE	Customer Edge
CI	Characteristic Information
CL-PS	Connectionless Packet-Switched
CO-CS	Connection-Orientated Circuit-Switched
CO-PS	Connection-Orientated Packet-Switched

CP	Connection point
CV	Connectivity Verification
DHCP	Dynamic Host Configuration Protocol
DLCI	Data Link Connection Identifier
DSCP	Differentiated Services Code Point
DWDM	Dense Wave Division Multiplexing
EBGP	External Border Gateway Protocol
E-LMI	External LMI
ES	End System
FDF	Flow Domain Flow
FDFr	Flow Domain Fragment
FDI	Forward Defect Indication
FP	Flow Point
FPP	Flow Point Pool
FR	Frame Relay
FT	Flow Termination
FTP	Flow Termination Point
GRE	Generic Routing Encapsulation
IGP	Interior Gateway Protocol
IKE	Internet Key Exchange
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
ISIS	Intermediate System to Intermediate System
L2TP	Layer 2 Tunnelling Protocol
LDP	Label Distribution Protocol
LF	Link Flow
LMI	Local Management Interface
LOC	Loss Of Continuity
LOS	Loss Of Signal
LSP	Label Switched Path
MAC	Media Access Control
MP2P	Multipoint-to-Point
MP-BGP	Multi-Protocol BGP
MPLS	Multi-Protocol Label Switching
MTU	Maximum Transmission Unit
NE	Network Entity
NF	Network Flow

NMS	Network Management System
NSAP	Network Service Access Point
OAM	Operations, Administration and Maintenance
OOB	Out Of Band
OSI	Open Systems Interconnection
OSPF	Open Shortest Path First
OSS	Operational Support System
P	Provider (Node)
P2P	Point-to-Point
P2MP	Point-to-Multipoint
PCR	Peak Cell Rate
PE	Provider Edge
PM	Performance Monitoring
PNNI	Private Network-to-Network Interface
PHP	Penultimate Hop Popping
PM	Performance Monitoring
PW	Pseudo Wire
QoS	Quality of Service
RADIUS	Remote Authentication Dial In User Service
RIP	Routing Information Protocol
RPR	Resilient Packet Ring
RMON	Remote MONitoring
RSVP-TE	Resource ReserVation Protocol (with) Traffic Engineering (extensions)
SCR	Sustained Cell Rate
SDH	Synchronous Digital Hierarchy
SES	Severely Errored Second
SLA	Service Level Agreement
SNC	SubNetwork Connection
SNMP	Simple Network Management Protocol
SONET	Synchronous Optical NETwork
SPVC	Switched Permanent Virtual Circuit
SSL	Secure Socket Layer
STP	Spanning Tree Protocol
SVC	Switched Virtual Circuit
TCP	Termination Connection Point
TDM	Time Division Multiplexing
TFP	Termination Flow Point

TTL	Time-To-Live
TTSI	Trail Termination Source Identifier
UNI	User-to-Network Interface
VC	Virtual Circuit/Channel
VCCV	Virtual Circuit Connectivity Verification
VCI	Virtual Channel Identifier
VLAN	Virtual Local Area Network
VPI	Virtual Path Identifier
VPN	Virtual Private Network
WDM	Wavelength Division Multiplexing

5 Client/server VPNs

Client/server VPNs have a two-layer hierarchy in which a VPN server layer network is used to support one or more VPN client layer networks.

ITU-T Rec. Y.1311 describes client/server VPNs in terms of VPN service types and VPN transport types, where the term VPN service type refers to the VPN client layer and the term VPN transport type refers to the VPN server layer. The different VPN service (client) and transport (server) types are classified in ITU-T Rec. Y.1311 as described below in Table 5-1.

Table 5-1/Y.1314 – Y.1311 service types

Service Type	Description
Layer 1	Provides a physical layer service between customer sites belonging to the same VPN. Connections can be based on physical ports, optical wavelengths, SDH/SONET VCs, frequency channels, or timeslots.
Layer 2	Provides a data link layer service between customer nodes belonging to the VPN. Forwarding of user data packets is based on information in the packets' data link layer headers (e.g., DLCI, ATM VCI/VPI, or MAC addresses).
Layer 3	Provides a network layer service between customer nodes belonging to the VPN. Forwarding of user data packets based on information in the Layer 3 header (e.g., IPv4 or IPv6 destination address).

One drawback to the method of classification used in ITU-T Rec. Y.1311 is that MPLS does not fit into any of these categories and must therefore be treated as a unique layer network technology. Another drawback is that, from a functional viewpoint, network technologies within the same layer can have very different characteristics and requirements. For example, Ethernet and ATM are both Layer 2 technologies; however, Ethernet is a broadcast-based connectionless technology whereas ATM is a connection-orientated non-broadcast technology.

An alternative method to classify network technologies is to classify them by the network mode they belong to. All network technologies can be mapped to one of three modes: connectionless packet-switched (CL-PS), connection-orientated packet-switched (CO-PS), and connection-orientated circuit-switched (CO-CS). The functional requirements for each mode are different as each mode has different characteristics. Table 5-2 shows examples of VPN network layer technologies and which mode they belong to.

Table 5-2/Y.1314 – Network modes of operation and examples

Mode of operation	Examples
Connectionless packet-switched	IP, Ethernet, MPLS MP2P (Note 1)
Connection-orientated packet-switched	Frame Relay, MPLS P2P/P2MP (Note 2), ATM
Connection-orientated circuit-switched	SDH/SONET, TDM

NOTE 1 – MPLS multipoint-to-point (MP2P) LSPs established using LDP in downstream unsolicited or ordered control mode traversing directly adjacent LDP peers.

NOTE 2 – MPLS point-to-point (P2P) or point-to-multipoint (P2MP) LSPs established using RSVP-TE traversing RSVP-TE peers, or P2P LSPs established using targeted/directed LDP between non-adjacent LDP peers.

5.1 Client/server combinations

There are nine possible client/server combinations based on the three network modes, although some combinations are more compatible than others. Table 5-3 describes the client/server combinations possible and provides some information on their compatibility.

A VPN server layer network must support multiplexing/de-multiplexing to provide data plane separation between multiple VPN client layers. VPN server layers must also support client traffic adaptation, which are client/server specific and are dependent on the VPN client and server layer network modes and on the specific technologies employed. One important adaptation requirement for circuit-switched VPN clients being carried by a packet-switched VPN server layer is that the adaptation function must provide rate decoupling (i.e., idle fill) and delineation of the VPN client layer packets. A key requirement for situations where the client/server are both packet-switched (CO or CL) is that the adaptation function must support fragmentation and sequencing if the VPN server layer traffic unit (i.e., the packets MTU) is smaller than the VPN client layer traffic unit. Other adaptation functions that may be required depending on the specific VPN client/server technologies employed include: coding, rate changing, and aligning.

Table 5-3/Y.1314 – Network mode client/server combinations

	CL-PS VPN client layer	CO-PS VPN client layer	CO-CS VPN client layer
CL-PS VPN server layer	– Ideal, although providing per flow delivery guarantees introduces scaling challenges – A common approach, which does not provide per flow delivery guarantees, is to use over-provisioning and class-based priority queuing (to manage any-to-any bursty traffic and congestion) *Example: An Ethernet server layer supporting an IP client layer*	– Providing per flow delivery guarantees introduces scaling challenges – A common approach, which does not provide per flow delivery guarantees, is to use over-provisioning and class-based priority queuing – VPN client layer must be able to recover from out-of-sequence traffic units (due to the possibility of server layer packet re-ordering) *Example: An IP server layer supporting an ATM client layer*	– Providing per flow delivery guarantees introduces scaling challenges – A common approach, which does not provide per flow delivery guarantees, is to use over-provisioning and class-based priority queuing – Recovering clock timing is technically challenging – VPN client layer must be able to recover from out-of-sequence traffic units *Example: An IP server layer supporting a TDM client layer*
CO-PS VPN server layer	– Cost associated with maintaining connection state for on-demand VPNs with short hold times, i.e., SPVCs *Example: An ATM server layer supporting an IP client layer*	– Ideal *Example: A P2P MPLS server layer supporting an ATM client layer*	– Recovering clock timing is technically challenging *Example: An ATM server layer supporting a TDM client layer*
CO-CS VPN server layer	– No statistical multiplexing between aggregates – Bandwidth assigned permanently in course increments resulting in poor network utilization – Slow connection set up response times for on-demand VPNs with short hold times *Example: An SDH server layer supporting an Ethernet client layer*	– No statistical multiplexing between aggregates – Bandwidth assigned permanently in course increments resulting in poor network utilization – Slow connection set up response times for on-demand VPNs with short hold times *Example: An ATM server layer supporting a TDM client layer*	– Ideal *Example: An optical server layer (e.g., a DWDM channel) supporting a SDH/SONET client layer*

5.2 VPN client layer transparency

In a client/server VPN, the functional components (such as routing, signalling, OAM, management, etc.) belonging to the VPN client layer network should be completely independent of the functional components belonging to the VPN server layer network.

Although it is possible to design client/server VPN solutions where the VPN server layer network's functional components interact with the VPN client layer network's functional components, this approach leads to a number of undesirable consequences, for example:

1) The VPN service may break if the customer changes any of the VPN client layer functional components.

2) The VPN service provider needs to track developments in the customer's VPN client layer technology and implement upgrades in its network accordingly.

3) Under fault conditions it becomes difficult to establish if the fault is in the VPN client layer network or the VPN server layer network.

By requiring that the VPN client and server layer networks are able to be run independently of one another, it naturally follows that the VPN server layer should transparently transfer the VPN client layer. For example, if the VPN client layer network is ATM, the VPN client layer network may implement a proprietary feature (e.g., AAL, non-PNNI routing and signalling, OAM) which, if not carried transparently, would break the VPN service.

Client layer transparency is not only a technical requirement, but it also has commercial implications because a VPN service provider is likely to consider the details of its network to be commercially sensitive and will therefore wish to hide those details from any VPN client layer networks. For example, it would not be desirable for the VPN server layer network to peer with the routing and signalling of the VPN client layer in the above example.

6 Peer level VPNs

In clause 5 the VPN topologies described were based on a client/server relationship between a VPN client layer and a VPN server layer. In the client/server VPN model, the VPN server layer adaptation source function adapts the CI of the VPN client layer into AI in the VPN server layer, and the VPN server layer adaptation sink function adapts VPN server layer AI to VPN client layer CI. In basic terms, this adaptation refers to the encapsulation of the client layer frame/signal in a VPN server layer frame/signal.

However, not all VPN topologies are based on the client/server model. VPNs can also be provided using CL-PS network technologies based on a model in which VPN reachability isolation within a shared domain is achieved via some means other than client/server encapsulation. This Recommendation refers to this type of VPN as a peer level VPN. The term "peer level" refers to the fact that the provider transports the customers VPN packets across its shared infrastructure at the same network layer at which it receives the packets from the customer. It does not refer to customer/provider control plane peering, the customer and provider may peer with each other in the control plane regardless of the type of VPN. Only the CL-PS network mode supports this type of VPN because, in the CO-PS and CO-CS cases, the connection-orientated nature of the technology enforces reachability isolation, i.e., NEs can only communicate with NEs that belong to the same P2P or P2MP connection.

In order to support VPNs across a shared domain, the network technology used must have some means of providing VPN isolation, i.e., NEs must only be able to communicate with other NEs belonging to the same VPN or be able to decrypt packets from NEs belonging to the same VPN.

6.1 Packet/route filtering

One way of implementing VPN isolation across a shared domain is to use packet filters along with PEs that are shared between multiple customers. In this approach, all the nodes in the service provider's network know all the customers' routes. This includes the provider edge (PE) nodes that face the customers' sites, and the provider (P) nodes in the core. In this architecture, the PE nodes are shared by different customers. The service provider allocates a portion of its address space to a customer and manages the packet filters on the PE routers to ensure full reachability between sites of a single customer, and isolation between customers.

To overcome the need to maintain consistent routing tables and packet filters on a per customer per site basis, an alternative is to implement a solution based on route filtering rather than packet filtering along with dedicated PEs, i.e., one PE per VPN. In this architecture, the P nodes contain all customers' routes but the PE nodes only contain routes for a single customer. Isolation of customer routes is achieved by route filtering. PE nodes are configured with route filters that only allow customers to learn about routes that belong to them. Border Gateway Protocol (BGP) is an example of a protocol commonly used for this purpose inside the provider backbone due to its versatile route filtering tools. An alternative to route filtering would be to use a different routing protocol instance for each VPN. However, using this approach, the shared network would only be able to support a small number of VPNs due to P nodes only being able to support a finite number of routing protocol instances, and due to the operational complexity of managing multiple protocol instances.

To overcome the need to use a different PE node for each VPN, an alternative approach is to use virtual routers (VRs). In this approach, one physical node is effectively compartmentalized into a number of virtual routers. One (or several) virtual routers can be assigned to a particular customer. In this way, one node can provide compartmentalized routing instances for several customers. Individual virtual routers behave just like separate PE nodes dedicated to a particular VPN. As with the route filtering approach, P nodes contain all customers' routes and therefore route filtering is required at PEs.[1]

6.2 Encryption

An alternative to route/packet filtering is to provide full reachability between all customers connected to a shared infrastructure in conjunction with packet encryption. Packet encryption ensures that if customers receive packets from a VPN to which they do not belong, they cannot obtain the information contained inside the packet. The customer may encrypt VPN packets before the traffic goes out over the shared network and therefore the customer is responsible for managing the VPN. In this approach, traffic within the service provider network is routed the same as any other IP traffic, and the service provider has no visibility into the tunnel. Nor does the service provider network need to be configured in any special manner. Alternatively, VPN packets may be encrypted using equipment managed by the provider (i.e., PEs or provider managed CEs) at the edge of the provider's shared network. In this approach, the provider is responsible for managing the VPN.

[1] A natural evolution of this approach is to use MPLS or other tunnelling approaches so that VPN-specific routes do not have to be maintained on backbone routers. However, this creates a client/server VPN topology and therefore this approach is applicable to clause 5 rather than this clause.

An example of an architecture that supports encryption is RFC 2401 – Security Architecture for the Internet Protocol (IPsec). IPsec defines cryptographic algorithms, authentication and key[2] management routines for creating secure IP traffic tunnels between IPsec gateways/clients. IPsec guarantees the privacy, integrity and data origin authentication in a VPN as the information traverses a shared infrastructure. IPsec is particularly useful for providing VPNs across public networks such as the Internet for site-to-site and remote access VPNs. IPsec functionality may be provided by a PE, CE, or end user device (e.g., a laptop running an IPsec client).

Secure Socket Layer (SSL) VPNs are another type of VPN that use encryption to provide per VPN isolation. A typical use of SSL VPNs is to allow users to access applications and files securely over the Internet. The advantage of this approach is that it does not require any configuration changes to end users' systems, only standard applications need to be supported (e.g., Web browsers, email clients, etc.). Also, SSL VPNs are transparent to the VPN peer layer (as encryption is performed at the application layer) and therefore configuration of routing/switching nodes is not necessary in order to support SSL VPNs.

6.3 Ethernet VLANs

The IEEE 802.1Q standard defines the operation of Virtual LAN (VLAN) Bridges that permit the definition, operation and administration of Virtual LAN topologies within a Bridged LAN infrastructure. VLANs allow end-stations on multiple physical LAN segments to communicate as if they were connected to the same LAN segment. End users and hubs/switches can be moved to different VLANs by changing the VLAN configuration on the port/interface on the 802.1Q-compliant switching device that the end station or hub/switch connects to. Broadcast and multicast frames are constrained by VLAN boundaries so end-stations will only receive broadcast/multicast frames for the VLAN that they belong to. This, in conjunction with how MAC address learning works, ensures that only end-stations belonging to the same VLAN may communicate with each other, and can therefore be considered to be members of the same VPN.

Traffic separation for frames belonging to different VLANs across a shared infrastructure is achieved by inserting a tag with a VLAN identifier (VID) into each frame. A VID must be assigned for each VLAN (1 to 4 096) and must be globally unique within the same physical infrastructure. One of the drawbacks to this approach is that customers also use VLANs within their own network, which introduces VID allocation and limitation issues. To solve this problem a second IEEE 802.1Q-tag can be appended to customer IEEE 802.1Q tagged packets that enter the providers' network (Q-in-Q as defined in IEEE 802.1ad). This separates the providers VLAN space from the customers' VLAN space and allows customers to use whatever VIDs they want[3].

[2] A key is a piece of information that controls the operation of the encryption/decryption algorithm.

[3] Another option is to use a MAC-in-MAC approach (as defined in IEEE 802.1ah) in which a provider appends a second Ethernet header to the customers packet. However, this option creates a client/server VPN rather than a peer level VPN because the customers' frame is encapsulated inside a provider frame.

7 Functional architecture of VPNs

The VPN reference model from ITU-T Rec. Y.1311 is shown in Figure 7-1.

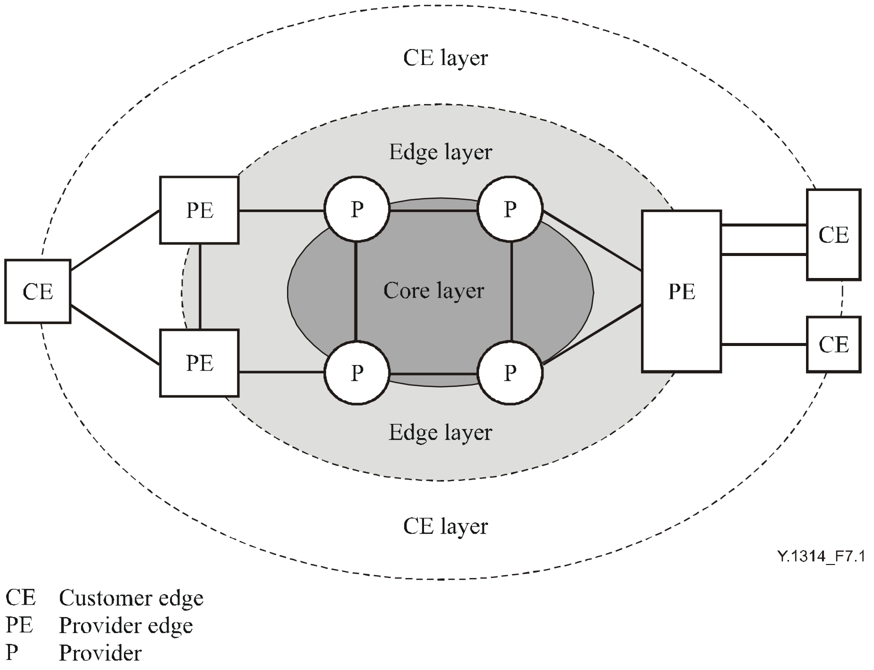


Figure 7-1/Y.1314 – ITU-T Rec. Y.1311 VPN reference model

Although this model shows the physical topology and the different network components, it does not show the different VPN server and client layer topologies or the location of inter-layer adaptation functions.

An alternative method of representing a client/server VPN network is to use functional modelling. The functional architecture of connection-orientated (CO-PS/CO-CS) and connectionless (CL-PS) layer networks can be described using ITU-T Rec. G.805, the "generic functional architecture of transport networks" and ITU-T Rec. G.809, the "functional architecture of connectionless layer networks" respectively.

ITU-T Recs G.805 and G.809 provide useful generic methods for modelling networks from a functional and structural architecture perspective. The terminology defined is technology-independent and can be used to describe the physical and logical components for any given network. This is particularly useful for network inventory and management as the complete network view can be modelled, from the optical fibres in the duct up to the VPN services running over them.

A VPN network can be decomposed into a number of independent layer networks with a client/server relationship between adjacent layer networks. As noted in ITU-T Rec. G.805, layer networks defined using functional modelling should not be confused with the layers of the Open System Interconnection (OSI) Model (ITU-T Rec. X.200). Each layer in the OSI model offers a specific service and the protocols defined at each layer perform a specific function corresponding to that layer, e.g., the transport layer (Layer 4) accepts data from the session layer, and passes it onto the network layer providing an end-to-end delivery service. On the contrary, each layer network in a functional model based on ITU-T Recs G.805 or G.809 offers the same service, i.e., the transport of bits/frames between inputs and outputs. Abstraction is commonly used to hide detail and to focus on the network layers/components of interest, but networks can be modelled right down to the network elements, e.g., Ethernet switches, copper pairs, SDH cross connects, etc.

7.1 Connection-orientated VPN layer networks

VPN client and server layer networks each have their own set of connectivity inputs and outputs known as access points (APs). These may be associated with each other to transfer information transparently across the layer network from input to output. Valid topology constructs of association between APs for CO layer networks are point-to-point (P2P) and point-to-multipoint (P2MP).

VPN server layer APs mark the functional boundary between the VPN server and client layer networks. From the VPN server layers perspective, a VPN server layer AP represents a routing destination that may support a trail. From the VPN client layer's perspective, a VPN server layer AP represents a point at which it is possible to procure link capacity. The functional components and reference points in a CO network layer are illustrated in Figure 7-2.

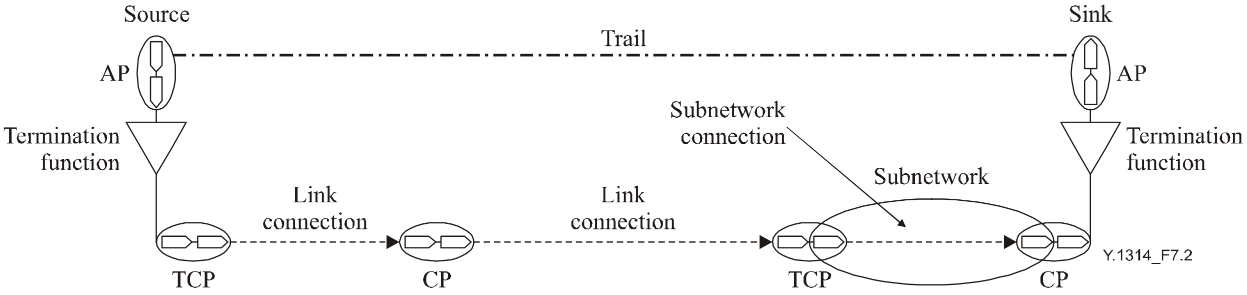


Figure 7-2/Y.1314 – CO functional components and reference points

Connections are transport entities in CO layer networks that consist of an associated pair of unidirectional connections capable of simultaneously transferring information in opposite directions between their respective inputs and outputs. A network connection is a transport entity in a CO layer network formed by a series of contiguous link connections and/or subnetwork connections between termination connection points (TCPs).

A subnetwork is a topological component in a CO layer network used to effect routing of specific characteristic information, and contains a set of points associated with a management function within a single CO network layer. A subnetwork connection transfers information across a subnetwork, and is formed by the association of ports (output of a trail termination source/input of a trail termination sink) on the boundary of the subnetwork.

Link connections interconnect topologically adjacent subnetworks that have a common subset of points. The point at which the input of a link connection is bound to the output of another link connection is a connection point (CP). The point at which a trail termination source output in a CO layer network is bound to the input of the network connection is a source TCP, and the point at which a trail termination sink input is bound to a network connection output is sink TCP. CPs and TCPs have a managed object associated with them and, it is therefore possible to group together TCPs and CPs belonging to the same VPN for management purposes.

7.2 Connectionless VPN layer networks

Unlike CO layer networks, CL layer networks support multipoint-to-multipoint (MP2MP) or any-to-any topologies. CL layer networks use flows rather than connections which are an aggregation of one or more traffic units with an element of common routing. Flows can be unidirectional or bidirectional, with bidirectional flows consisting of two contra-directional unidirectional flows. A network flow is a transport entity in a CL layer network formed by a series of contiguous flows between termination flow points (TFPs). The functional components and reference points in a CL network layer are illustrated in Figure 7-3.

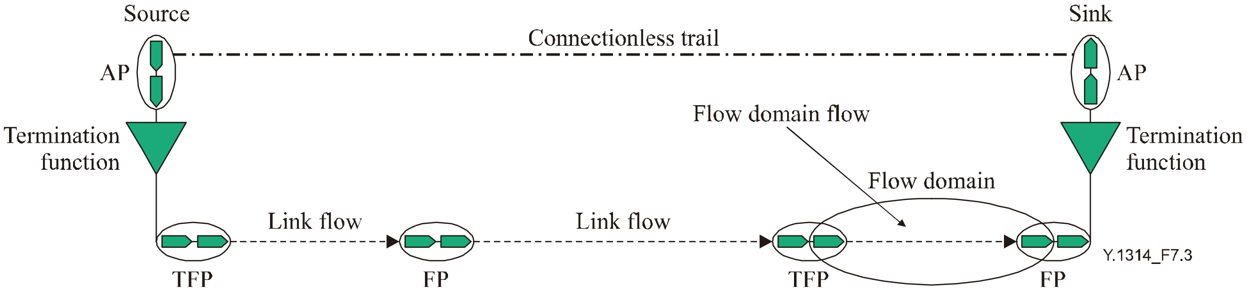


Figure 7-3/Y.1314 – CL functional components and reference points

A flow domain is a topological component in a CL layer network used to effect routing of specific characteristic information. A flow domain flow is a transport entity that transfers information across a flow domain, and is formed by the association of ports on the boundary of the flow domain. A flow domain contains a set of points associated with a management function within a single CL network layer.

Link flows interconnect topologically adjacent flow domains that have a common subset of points. The point at which the input of a link flow is bound to the output of another link flow is a flow point (FP). The point at which a connectionless trail termination source output in a CL layer network is bound to the input of the network flow is a source TFP, and the point at which a connectionless trail termination sink input is bound to a network flow output is a sink TFP. As with CPs and TCPs in the CO case, in the CL case FPs and TFPs have a managed object associated with them and it is, therefore, possible to group together TFPs and FPs belonging to the same VPN for management purposes.

7.3 VPN client/server relationships

In functional terms, a VPN client layer network is a topological component in a client/server VPN that represents the set of access points of the same type associated for the purpose of transferring VPN client layer characteristic information, which are supported by a VPN server layer trail or a connectionless trail. The source/sink TCPs/TFPs for VPN client layer connections/flows may be located in CE nodes, or in nodes/end-systems elsewhere in the customer's network. For example, TCPs in an ATM VPN client layer are likely to be located in CE nodes, whilst TFPs in an Ethernet VPN client layer are likely to be located in end user computers or servers. The location of the VPN client flow/connection TFPs/TCPs is important from a customer perspective, as this is the point in the customer's network where adaptation between the VPN client layer and the layer above must take place. It is also important from an OAM perspective, as this is where the source and sink APs for the trail/connectionless trail associated with a VPN client layer flow/connection are located. Examples of client/server VPNs where the TFPs/TCPs are located in different places are provided in Appendix I.

A VPN server layer network is a topological component in a client/server VPN that represents the set of access points of the same type associated for the purpose of transferring adapted client layer information for one or multiple VPN client layer flows or connections. The VPN server layer contains source/sink adaptation functions that adapt the characteristic information in the VPN client layer into/from adapted information in the VPN server layer. The VPN client and server layers may belong to the same mode (i.e., when the client and server layers are both CO or the client and server layer are both CL), but combinations of the two are also possible, i.e., CO VPN server layers can support CL client layers, and likewise CL server layers can also support CO client layers. Figure 7-4 shows an example of a CL VPN server layer supporting a CL VPN client layer from a functional perspective based on the physical topology of the network model from ITU-T Rec. Y.1311 shown in Figure 7-1. The bottom layer shown in the model is the VPN server layer and the top layer is the VPN client layer. Only the VPN client/server layers are illustrated for simplicity, the customer client layer above the VPN client layer, and the server layer below the VPN server layer are not shown. In this example the VPN server layer is CO (e.g., ATM) whilst the VPN client layer is CL (e.g., Ethernet), although any combination of CO or CL pairs is possible.

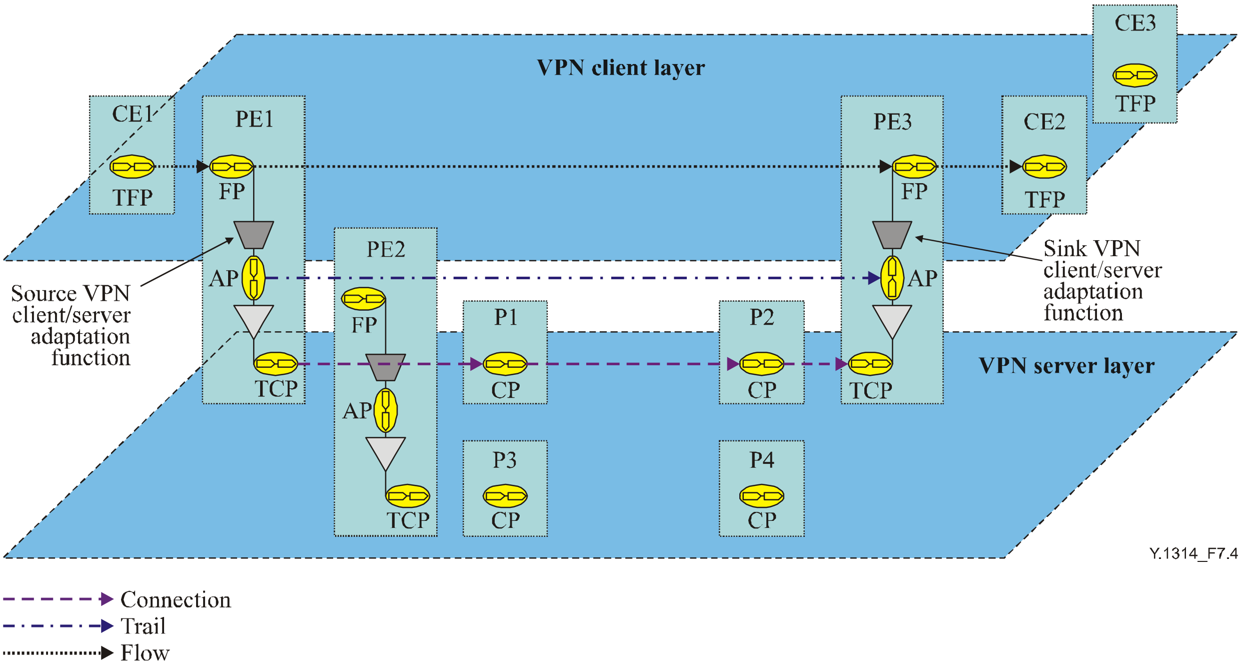


Figure 7-4/Y.1314 – Functional model of client/server VPN

Figure 7-4 shows how the functional model relates to the network diagram in Figure 7-1 by highlighting which functions and network reference points exist in which network element (i.e., CE, PE, or P node). The CE and P nodes belong to the VPN client and server layers respectively, whilst the PE nodes belong to both layers. The TFPs in the VPN client layer identify where (which CE node in this case) the P2P VPN client layer flow starts (its source) and ends (its sink), and the FPs identify which PE nodes the P2P flow passes through. Likewise, the TFPs in the VPN server layer identify the source and sink for the VPN server layer connection, and the FPs identify which P nodes the flow passes through. The APs in the VPN server layer identify the source/sink for the VPN server layer trail.

The following subclauses present each of the four possible client/server VPN combinations using functional models and describe the role of the client/server VPN adaptation functions.

7.3.1 CO VPN client layer supported by a CO VPN server layer

An example of a CO VPN client layer network that is supported by a CO VPN server layer network is illustrated in Figure 7-5.

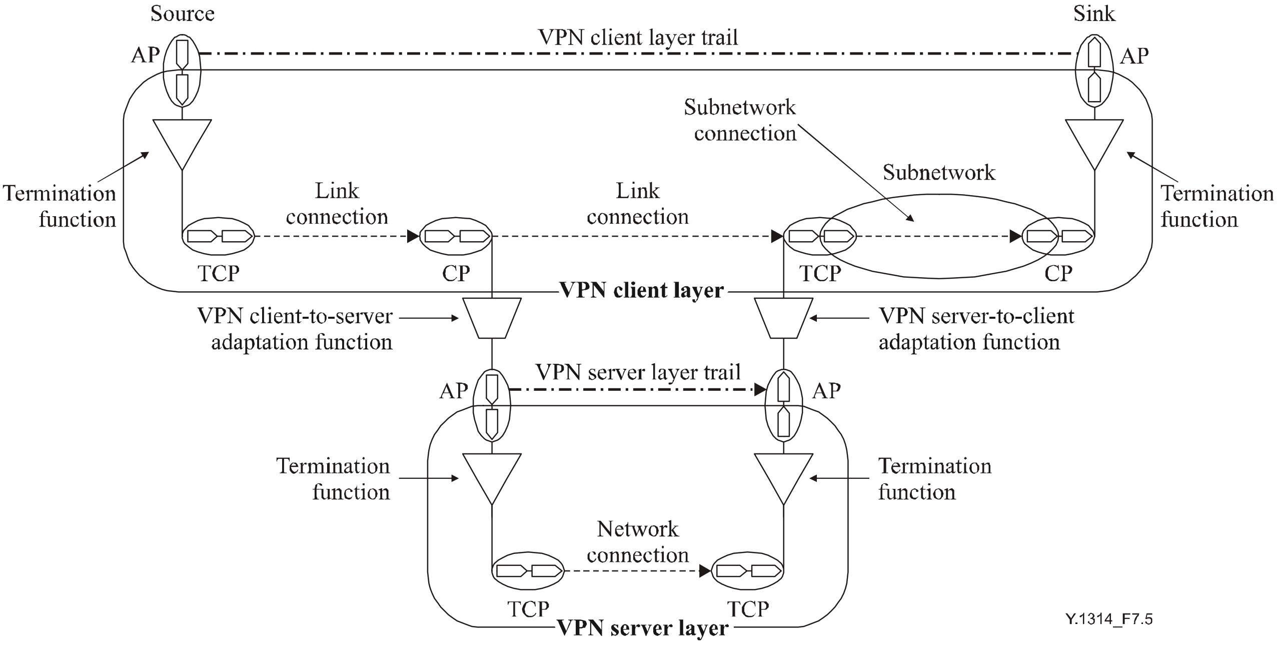


Figure 7-5/Y.1314 – CO VPN server layer with a CO VPN client

In this example, the CO VPN client layer connection is supported by a CO VPN server layer trail. The CO VPN server layer adaptation source function adapts the characteristic information (CI) of the CO VPN client layer into adapted information (AI) in the CO VPN server layer. The CO VPN server layer adaptation sink function adapts CO VPN server layer AI to CO VPN client layer CI.

7.3.2 CL VPN client layer supported by a CL VPN server layer

An example of a CL VPN client layer network that is supported by a CL VPN server layer network is illustrated in Figure 7-6.

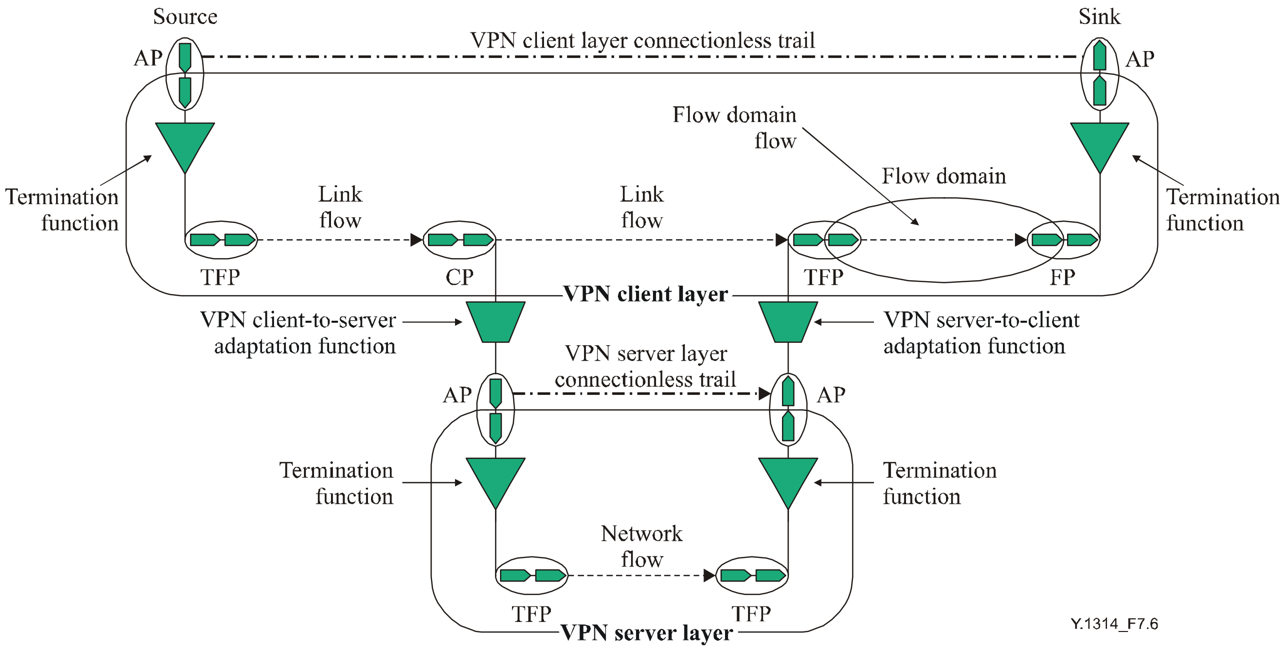


Figure 7-6/Y.1314 – CL VPN server layer with a CL VPN client

In this example, the CL VPN client layer flow is supported by a CL VPN server layer connectionless trail. The CL VPN server layer adaptation source function adapts the characteristic information (CI) of the CL VPN client layer into adapted information (AI) in the CL VPN server layer. The CL VPN server layer adaptation sink function adapts CL VPN server layer AI to CL VPN client layer CI.

7.3.3 CO VPN client layer supported by a CL VPN server layer

An example of a CO VPN client layer network that is supported by a CL VPN server layer network is illustrated in Figure 7-7.

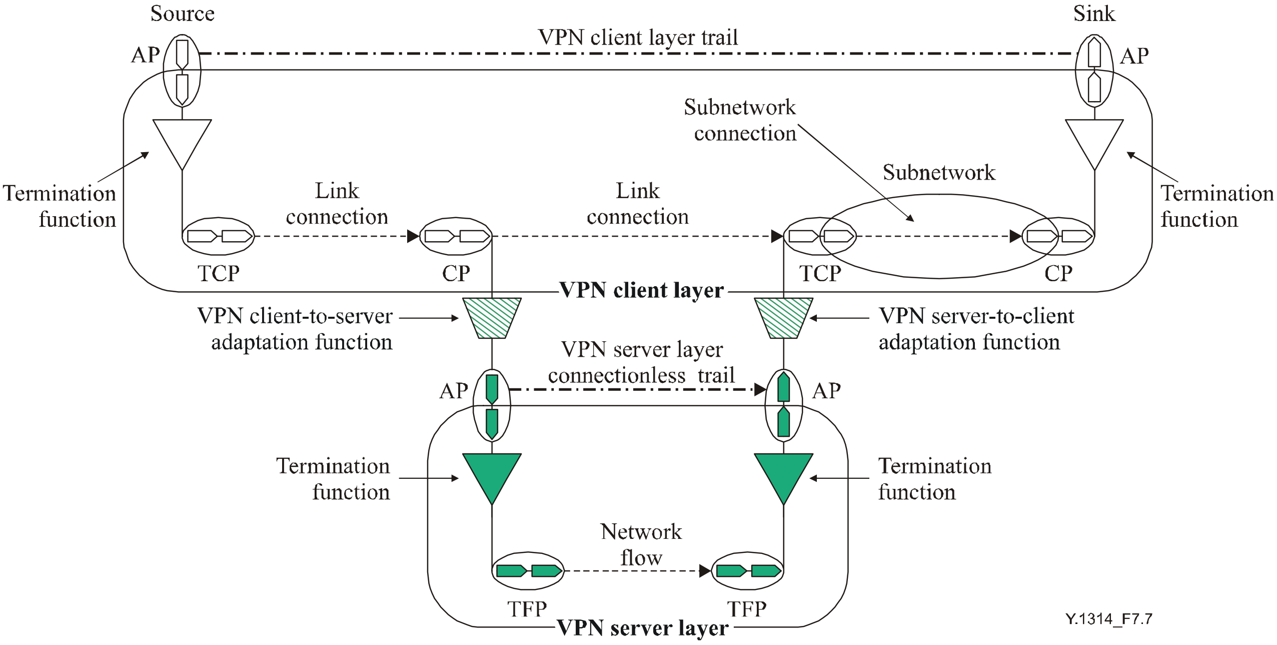


Figure 7-7/Y.1314 – CL VPN server layer with a CO client

In this example, the CO VPN client layer connection is supported by a CL VPN server layer connectionless trail. The CL VPN server layer adaptation source function adapts the characteristic information (CI) of the CO VPN client layer into adapted information (AI) in the CL VPN server layer. The CL VPN server layer adaptation sink function adapts CL VPN server layer AI to CO VPN client layer CI.

7.3.4 CL VPN client layer supported by a CO VPN server layer

An example of a CL VPN client layer network that is supported by a CO VPN server layer network is illustrated in Figure 7-8.

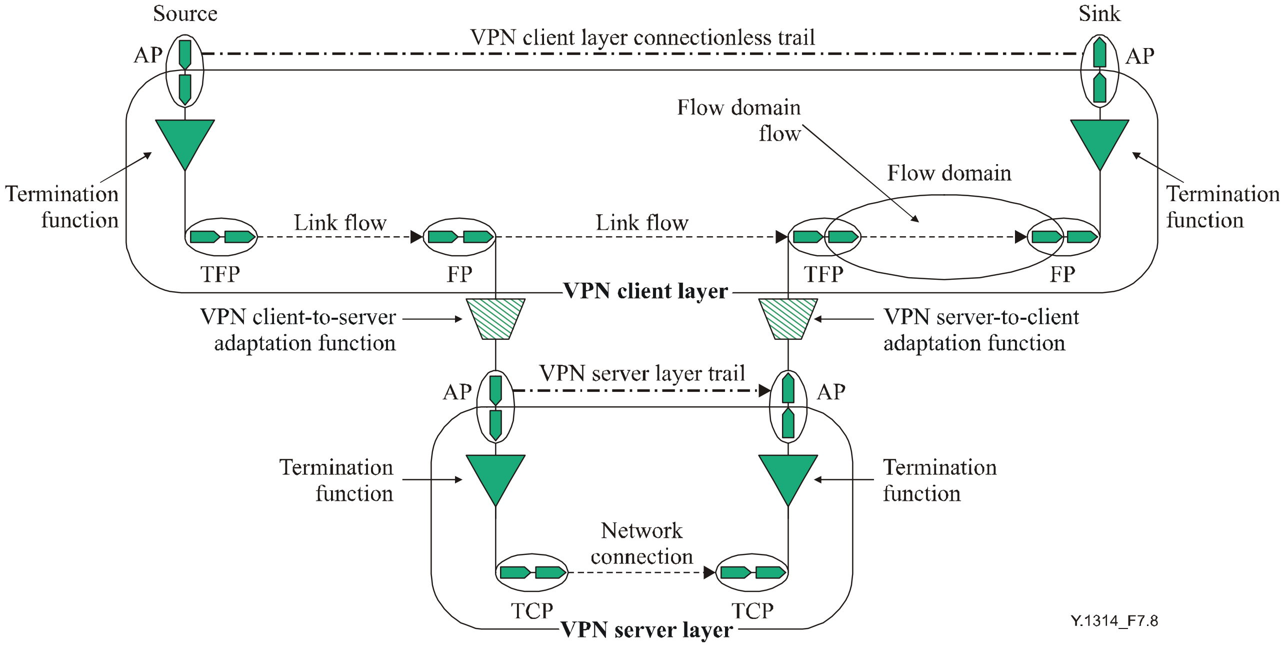


Figure 7-8/Y.1314 – CO VPN server layer with a CL client

In this example, the CL VPN client layer flow is supported by a CO VPN server layer trail. The CO VPN server layer adaptation source function adapts the characteristic information (CI) of the CL VPN client layer into adapted information (AI) in the CO VPN server layer. The CO VPN server layer adaptation sink function adapts CO VPN server layer AI to CL VPN client layer CI.

7.4 Multiple VPN client layers

In the examples so far in this clause, a single VPN client layer has been used end-to-end. However, this may not always be the case, a customer may wish to use one VPN client layer type at one side of a VPN, and a different VPN client type at the other side of a VPN. For example, at one side the VPN client layer might be IP and at the other side it might be MPLS, or one side might be Frame Relay (FR) and the other side might be ATM. In such cases, the two different VPN client layer networks must be networked on a peer level basis.

It should be noted that the term 'VPN client layer network' used here refers to a topological component in a client/server VPN that represents the set of access points of the same type associated for the purpose of transferring VPN client layer CI. It does not refer to network layering in the Layer 1, Layer 2, Layer 3 sense, i.e., the two network technologies being interworked at the VPN client layer may both be Layer 2 technologies (e.g., one may be ATM and one may be FR), but they are considered to be different layer networks as they contain different access points, which are also of a different type.

The interworking function may take place either before the VPN server layer source adaptation function or after the VPN server layer sink adaptation function. Figure 7-9 shows the physical topology of client/server VPN network which uses different VPN client layers at each side of the VPN.

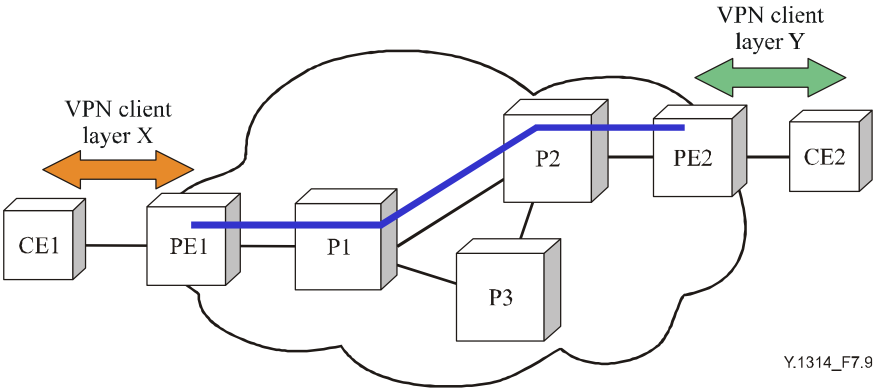


Figure 7-9/Y.1314 – VPN client peer level interworking physical topology

Figure 7-10 presents a generic functional model for peer level VPN client interworking based on the physical topology in Figure 7-9, where the interworking function takes place prior to VPN server layer source adaptation function.

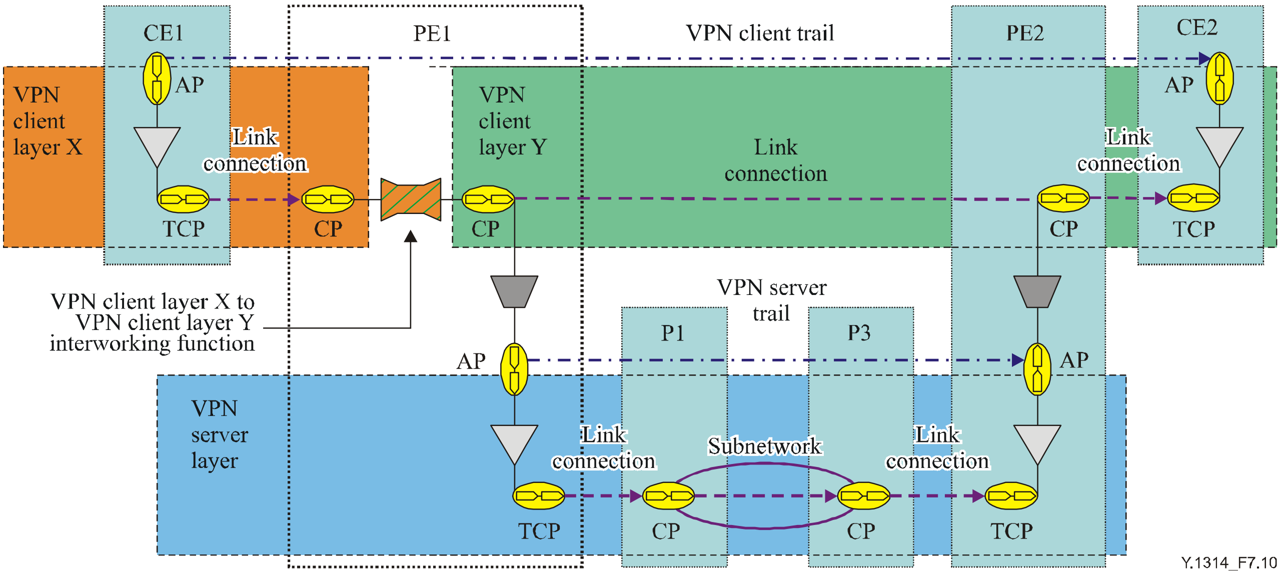


Figure 7-10/Y.1314 – VPN client peer level interworking (pre-VPN server source adaptation)

The two heterogeneous VPN client layers in this model are VPN client layer X and VPN client layer Y. In this example, the PE performs the interworking function but it could also be performed using a separate device. The interworking function converts the VPN client layer X CI into VPN client layer Y CI. The VPN server layer source adaptation function adapts the VPN client layer Y CI into AI in the VPN server layer and the VPN server layer AI is transmitted across the VPN server layer trail. At the VPN server layer sink the adaptation function adapts VPN server layer AI to VPN client layer Y CI. As an example, if VPN client layer X was FR and VPN client layer Y was ATM, then the source PE would convert the FR traffic into ATM traffic (e.g., using FRF.8) and the VPN client layer traffic would be carried as ATM over the VPN server layer.

Figure 7-11 presents a generic functional model for peer level VPN client interworking where the interworking function takes place after the VPN server layer sink adaptation function.

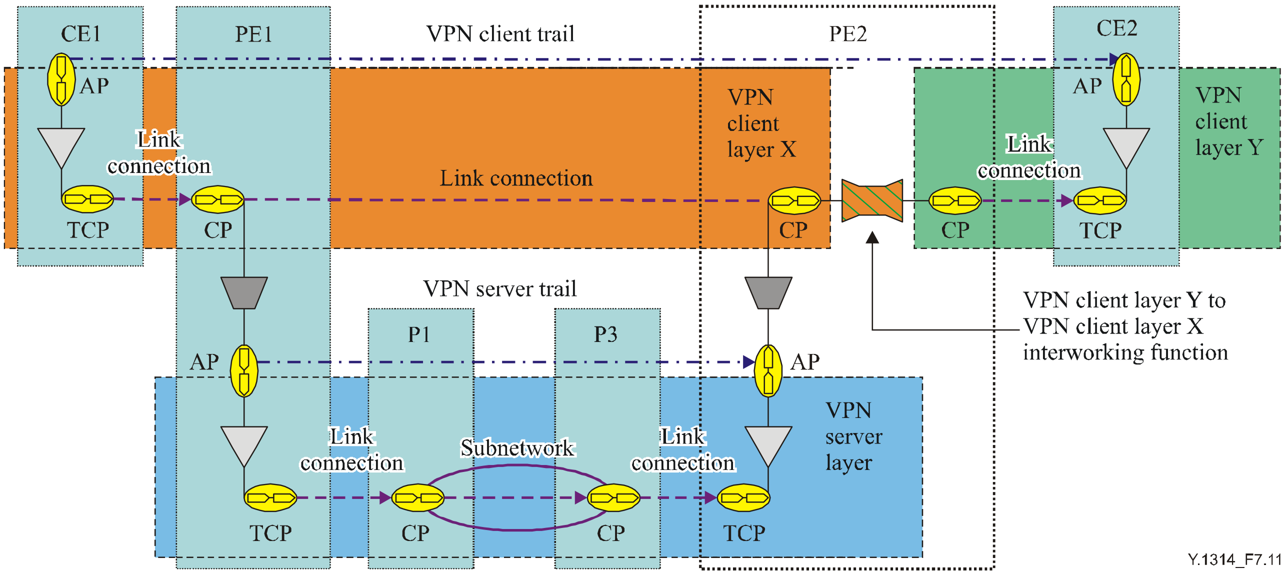


Figure 7-11/Y.1314 – VPN client peer level interworking (post VPN server sink adaptation)

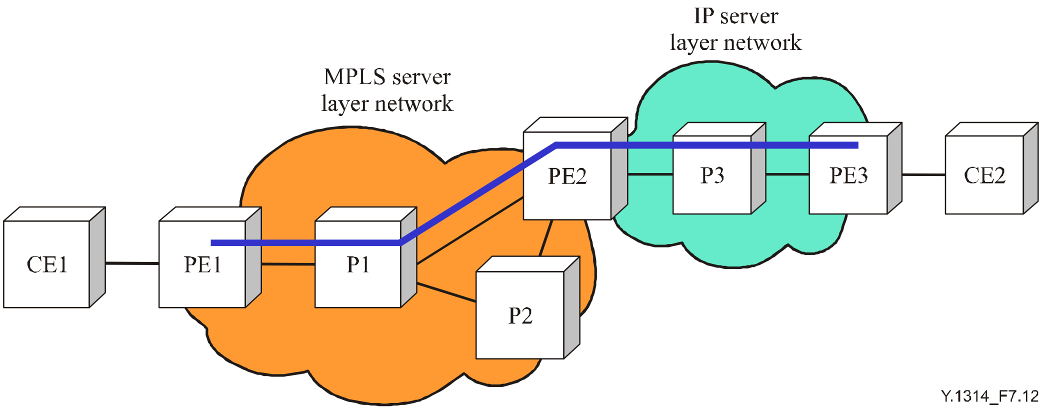
The VPN server layer source adaptation function adapts the VPN client layer X CI into AI in the VPN server layer and the VPN server layer AI is transmitted across the VPN server layer trail. At the VPN server layer sink, the adaptation function adapts VPN server layer AI to VPN client layer X CI. The interworking function converts the VPN client layer X CI into VPN client layer Y CI. If VPN client layer X was FR and VPN client layer Y was ATM, then the VPN client layer traffic would be carried as FR over the VPN server layer and converted into ATM by the sink PE.

7.5 Multiple VPN server layers

In the previous examples, a single VPN server layer was used end-to-end across the provider network to support the VPN client layer. However, this may not always be the case; for example, a provider may not be able to provide end-to-end connectivity using a single VPN server layer due to lack of network coverage, or a VPN client layer may need to cross multiple provider networks. Under these circumstances, multiple VPN server layers are required. Depending on the specific network technologies and the interworking capabilities of the provider's equipment, separate VPN server layers may be interworked on a peer level basis, or interworked with the VPN client on a client/server basis.

Although using multiple VPN server layers is possible, there are several factors that need to be taken into consideration when contemplating using multiple MPLS VPN server layers. The factors to be considered are dependent on the type of interworking required and the VPN server layer technologies to be employed. Examples of peer level and client/server interworking of multiple server layers along with some considerations for each are provided in Appendix II.

It should be noted that using multiple server layers below the VPN server layer should not be confused with using multiple VPN server layers. For example, as illustrated in Figure 7-12 a service provider may use a single MPLS VPN server layer end-to-end, but use a MPLS server layer (using MPLS label stacking) below the VPN server layer in one part of the network, and use an IP server layer (e.g., using GRE encapsulation) in another part of the network.



CE Customer edge node
PE Provider edge node
P Provider (core) node

— Physical link
▬ VPN

Figure 7-12/Y.1314 – Client/server VPN with MPLS and IP server layers

The PE and P routers must all support MPLS in the MPLS server layer network, however, only the PE routers in the IP server layer need to support MPLS, the P routers do not need to support MPLS. The functional model relating to the network illustrated in Figure 7-12 is described in Figure 7-13.

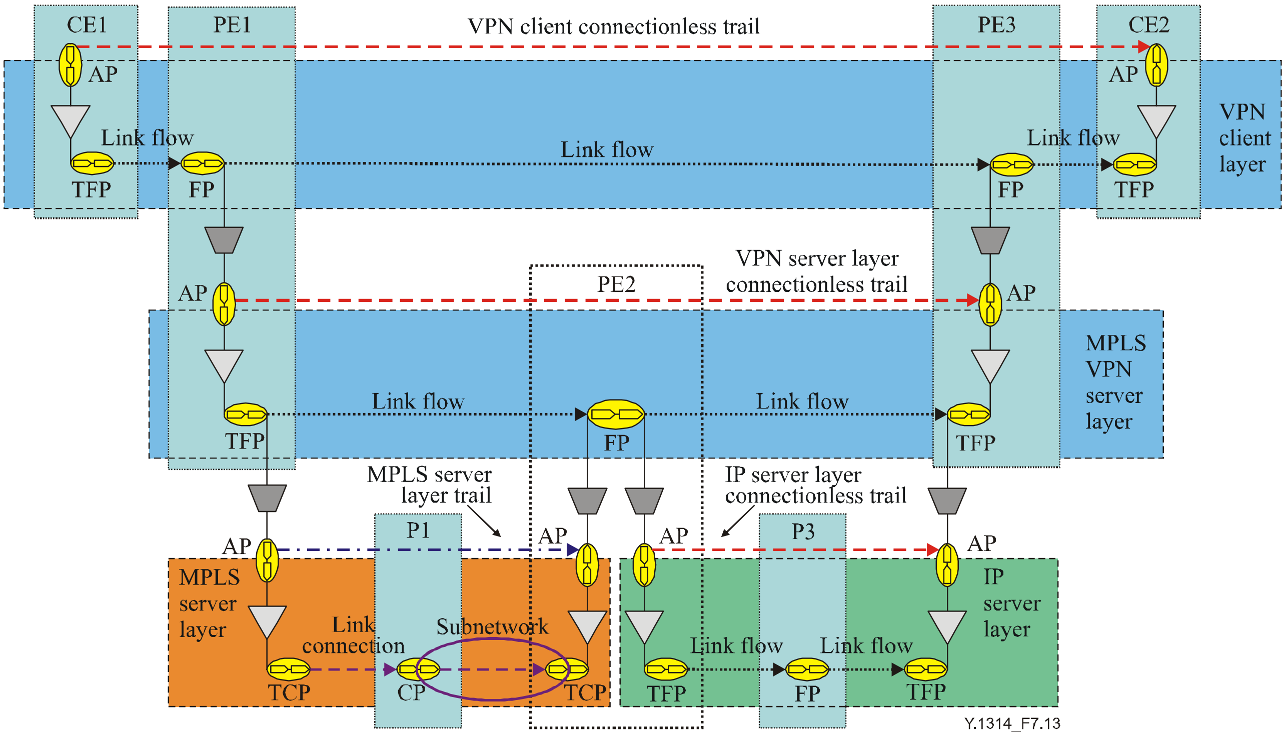


Figure 7-13/Y.1314 – VPN server layer supported by multiple server layers

In this example the MPLS server layer source adaptation function adapts the MPLS VPN server layer CI (which is a client of the MPLS server layer) into AI in the MPLS server layer, and the MPLS server layer sink adaptation function adapts the MPLS server layer AI to MPLS VPN server layer CI. The IP server layer source adaptation function adapts the MPLS VPN server layer CI into

AI in the IP server layer, and the IP server layer sink adaptation function adapts the IP server layer AI to MPLS VPN server layer CI.

7.6 VPN modelling using partitioning

The functional models provided in the previous clauses were developed using a layered approach. Decomposing networks into a number of independent layer networks allows the client/server relationship between adjacent layer networks to be modelled and the corresponding adaptation, termination and interworking functions to be described.

An alternative modelling approach is partitioning, which is used to define the network structure within a layer network and administrative/routing boundaries between network domains, e.g., networks owned by different operators. Partitioning allows a subnetwork at one level to be decomposed into its containing subnetworks and the links between them. This partitioning can continue until the limit of recursion is reached, that is, a single subnetwork within a network element. This is known as a matrix, as described in ITU-T Rec. G.805. Partitioning is illustrated in Figure 7-14.

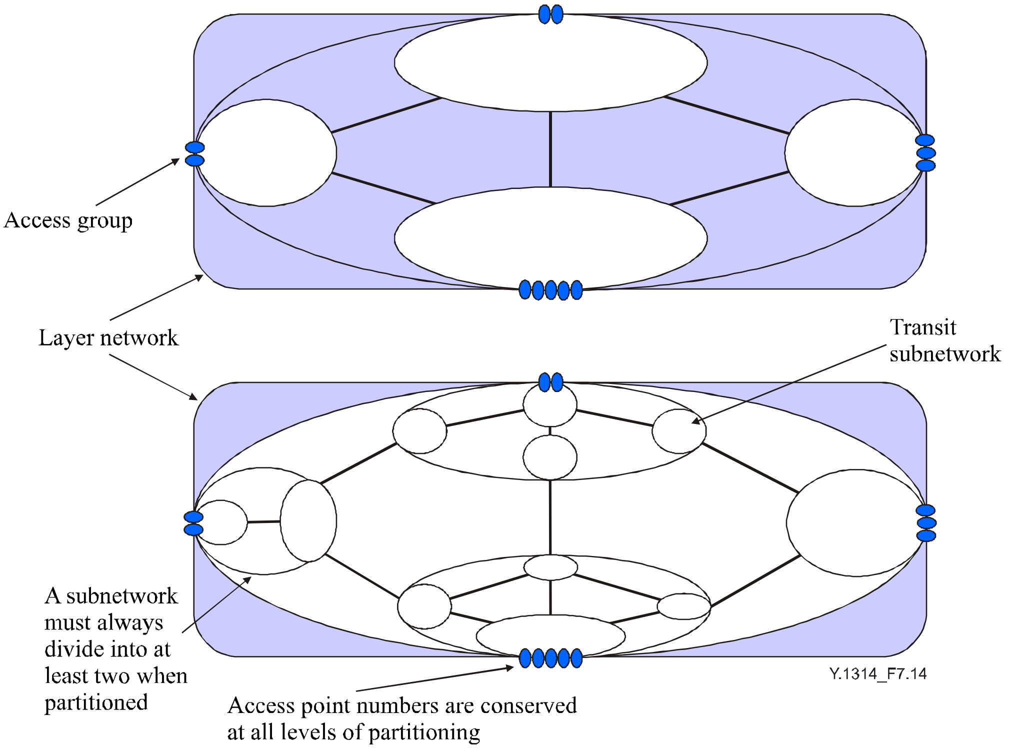

Access group
Layer network
Transit subnetwork
A subnetwork must always divide into at least two when partitioned
Access point numbers are conserved at all levels of partitioning
Y.1314_F7.14

Figure 7-14/Y.1314 – Partitioning of subnetworks within a layer network

As part of the process of partitioning, the number of flow/connection points in the largest subnetwork remains the same under partitioning, whilst the connection points internal to it at the next level of partitioning are revealed. From the perspective of connectivity, the subnetwork (flow domain) represents a point of flexibility between its inputs and outputs (e.g., source/sink access points or flow/connection points). Generally, this allows any input to be connected to any output.

This model is sufficient for public networks where the resources can all be assumed as being available for use. However it is not suitable for virtual private networks. The reason for this is that the connectivity between inputs and outputs on the subnetwork/flow domain is limited to those inputs and outputs belonging to the same VPN. To support the modelling of a VPN using the partitioning approach, Flow Domain Fragment (FDFr) constructs as described in ITU-T Rec. G.8010, and Subnetwork Connection (SNC) constructs are used. A FDFr/SNC is fragmented by means of dividing its inputs and outputs into different groups. Connectivity is limited to being between members of the same group. Such a group may be a VLAN on an Ethernet bridge (an Ethernet flow domain) or a VPN on a subnetwork or flow domain. Note that the fragment has no flow points; these are associated with the flow domain. A FDFr/SNC may be labelled by its associated layer network name and fragment number, or by the means of grouping flow points into a particular fragment, e.g., by VLAN identifier. An example of a network using VLANs to provide VPN isolation is shown in Figure 7-15.

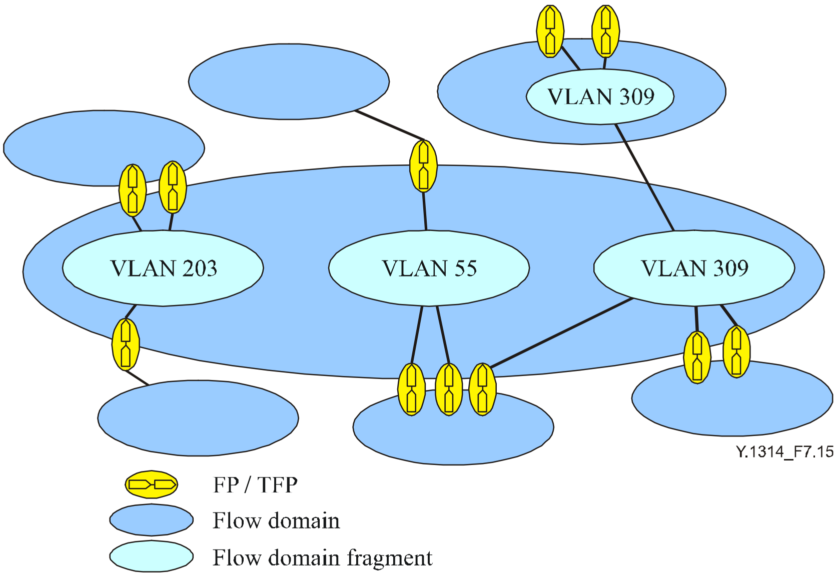


Figure 7-15/Y.1314 – Example of VPN partitioning functional model

An FDFr of one flow domain is associated with an FDFr in another flow domain by means of the interconnecting component link. Similarly, an SNC in one subnetwork is associated with an SNC in another subnetwork via the interconnecting link connection. This allows the construct to be partitioned or aggregated in line with the subnetwork model. As such, the model is very flexible and allows the VPN structure to be shown at any level of subnetwork partitioning.

7.7 VPN peer layer

Figure 7-16 shows the physical topology of a peer level VPN. In this example, the network cloud depicts the domain of the shared network and the grey line depicts a P2P VPN. VPN isolation could be achieved using any of the approaches defined in clause 6, e.g., an Ethernet VLAN, an IPsec tunnel, etc.

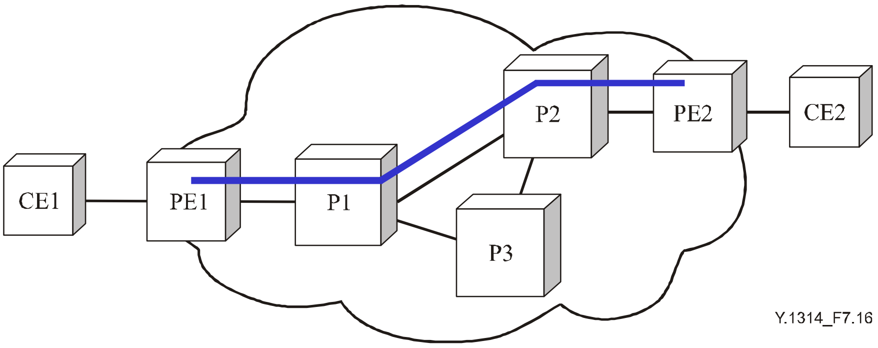


Figure 7-16/Y.1314 – Peer level VPN physical topology example

Figure 7-17 describes the VPN topology in Figure 7-1 from a functional perspective showing the VPN layer and a single underlying server layer between PEs. In this example the server layer is CO, but it could equally be CL.

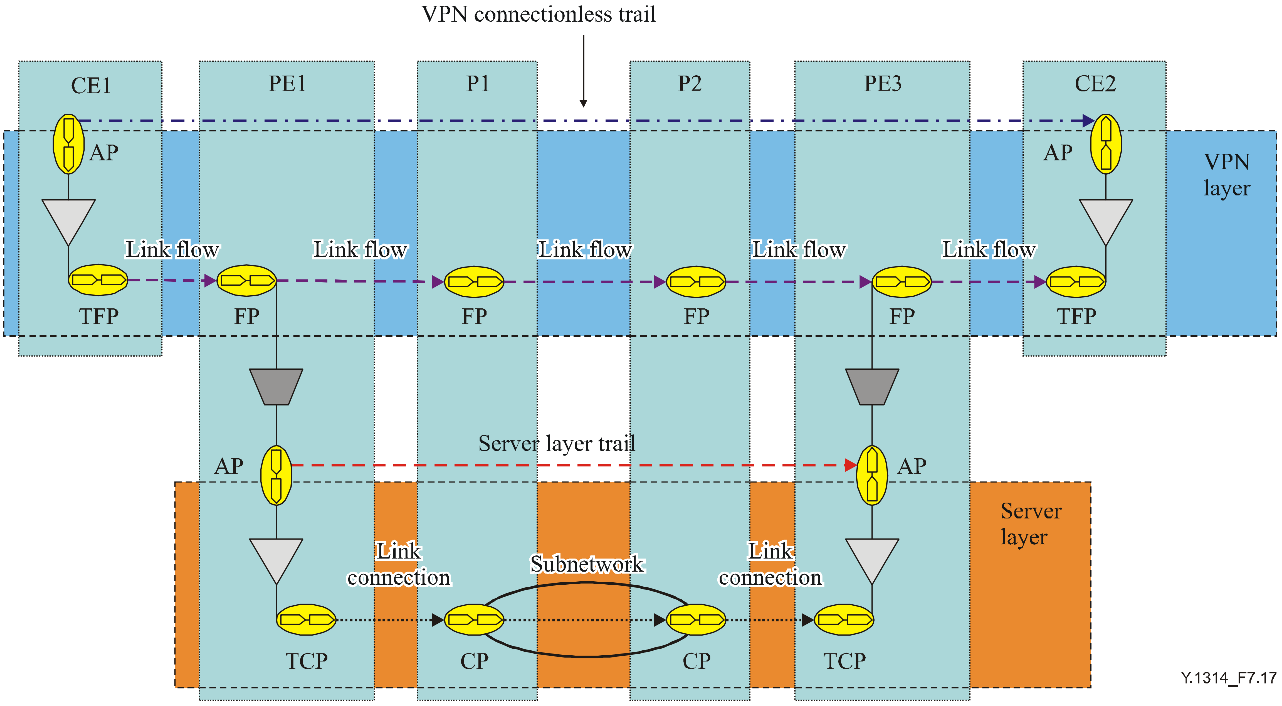


Figure 7-17/Y.1314 – Layer model of a single layer VPN

As Figure 7-17 shows, all the nodes in the network (including the P nodes) belong to the VPN layer and, therefore, they must be able to forward packets onto the correct destination using information in the VPN layer packet headers. Due to the single layer VPN architecture, the layering model does not provide as much information as it does when it is used in the client/server VPN case. In particular, the presentation format of Figure 7-17 does not provide any information about where the VPN starts and ends. One way to incorporate this information is to expand on the VPN/server layer adaptation functions. Figure 7-18 shows two different examples of VPN/server layer adaptation functions, one which uses IPsec and one which uses Ethernet VLAN tags.

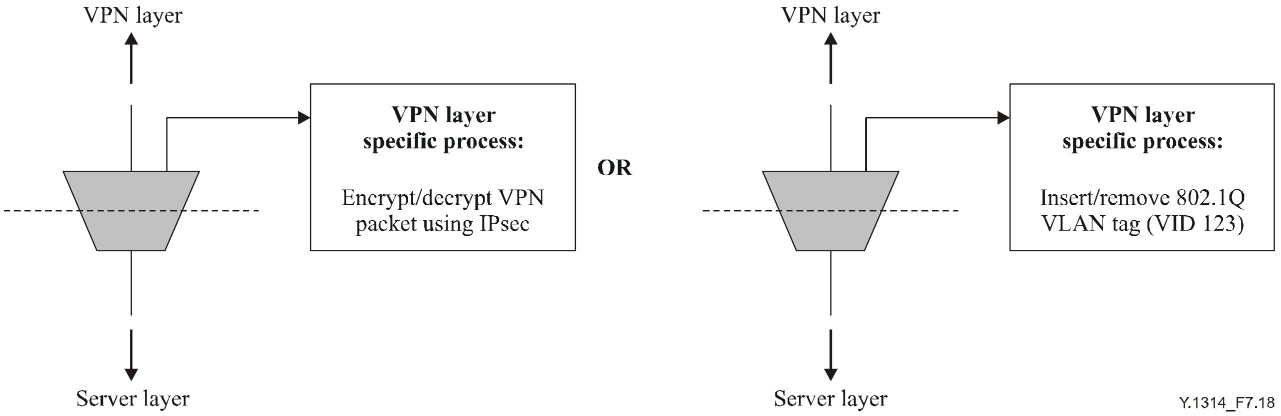

VPN layer
VPN layer specific process:
Encrypt/decrypt VPN packet using IPsec
Server layer
OR
VPN layer
VPN layer specific process:
Insert/remove 802.1Q VLAN tag (VID 123)
Server layer
Y.1314_F7.18

Figure 7-18/Y.1314 – Expansion of VPN/server layer adaptation functions

Another way of describing a peer level VPN is to use the partitioning concept introduced in 7.6. An example of how partitioning may be used in this way is provided in Figure 7-19.

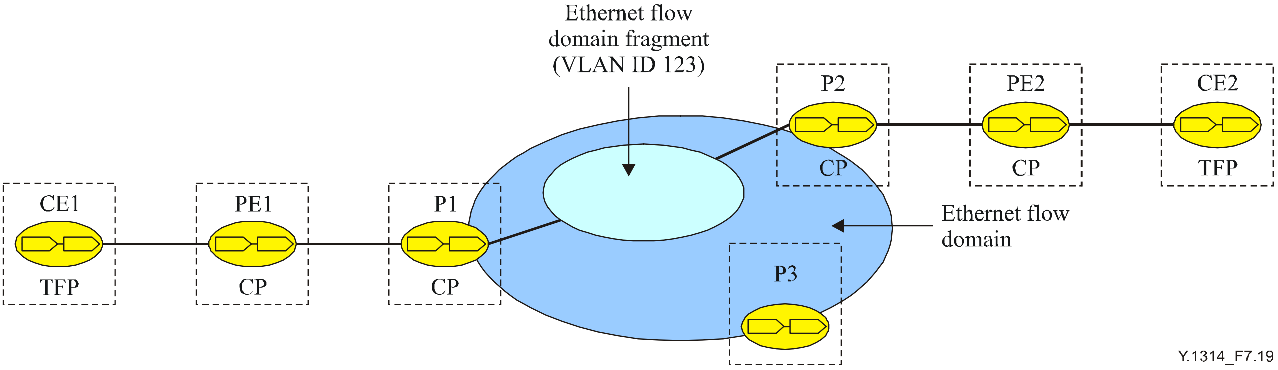


Figure 7-19/Y.1314 – Peer level VPN modelled using partitioning

Figure 7-19 describes the peer level VPN topology, and also shows the corresponding VLAN (123), but does not provide information about where the VPN starts and ends, i.e., where the IEEE 802.1Q VLAN tags are inserted/removed. It is easy to see from the model in Figure 7-19 that Nodes P1 and P2 are part of VLAN 123, but although PE1 and PE2 are the start/end points for the VPN, the model does not provide this information. However, incorporating the VPN/server layer adaptation functions into the partitioning model and expanding on the VPN layer-specific processing could provide this information (as shown in Figure 7-18).

8 VPN topology support

The term 'VPN topology' used within this Recommendation refers to the network topology from the perspective of the VPN customer, i.e., the topology between VPN sites which may be CE nodes or end systems. Connectivity between VPN sites can only be provided if VPN server layer or peer layer trails have been established between them. In general, the topology at layer n is dependent on the topology provided by the server layer trails at layer n–1. Once the VPN server or peer layer trails have been established, if the VPN client layer or peer layer technology is packet switched then it is possible to prune the VPN topology by restricting connectivity between certain sites within the VPN. One method of restricting connectivity between VPN members is to control route distribution at the VPN client layer (VPN sites cannot communicate if they do not have routes to reach each other). Another method that can be used to restrict connectivity is to use packet filtering (e.g., based on VPN client layer or peer layer source/destination addresses). The three basic VPN topologies are full mesh, partial mesh and hub and spoke, and are described in clauses 8.1, 8.2 and 8.3.

8.1 Full mesh VPN topologies

In a full mesh VPN topology, every VPN site has a route/connection to every other VPN site as depicted in Figure 8-1.

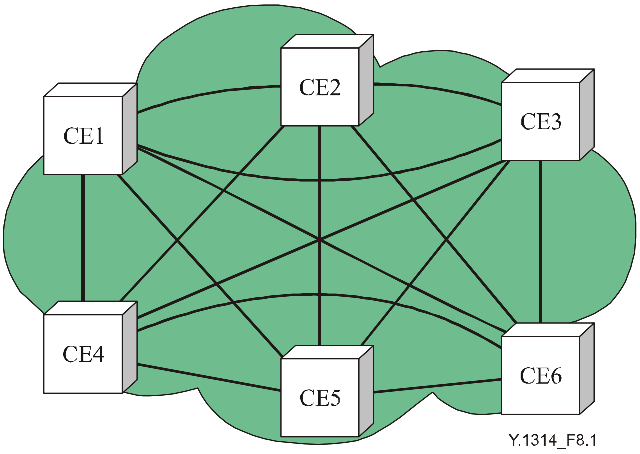


Figure 8-1/Y.1314 – An example of a full mesh VPN topology

A full mesh topology provides complete redundancy and can also provide efficient network utilization and performance as VPN sites can use the shortest/best paths/routes to reach each other. A disadvantage of the full mesh approach is that a full mesh can be expensive to implement, although this is dependent on the VPN network modes/technologies employed (e.g., a VPN network made up of a full mesh of ATM VCs is likely to be more expensive than an Ethernet VPN that supports any-to-any connectivity). Another disadvantage is that as the number of sites in the full mesh increases, the number of connections/routes and control plane adjacencies increases proportionately (the number of connections in a full mesh is n(n–1)/2, where n is the number of VPN sites). Supporting large numbers of connections/routes and control plane adjacencies introduces scaling issues due to an increase in the bandwidth and CPU resources required.

8.2 Partial mesh VPN topologies

In a partial mesh topology, VPN sites have routes/connections to some VPN sites but not all. An example of a partial mesh topology is provided in Figure 8-2.

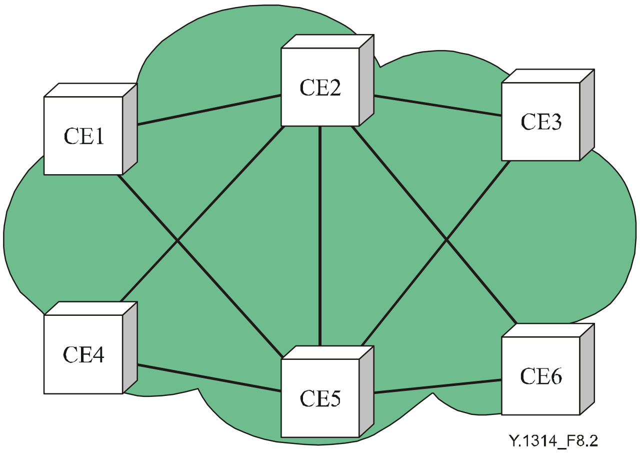


Figure 8-2/Y.1314 – An example of a partial mesh VPN topology

In some cases, VPN sites may be able to reach other VPN sites that do not have direct routes/connections to via transit VPN sites. However, in other cases, if VPN sites do not have direct routes/connections to reach each other then communication between them may not be possible. The ability for communication to take place between nodes that do not have direct routes or connections to reach each other is dependent on the existence of VPN server layer or peer layer trails and on any topology connectivity restrictions (e.g., routing policies or packet filters). Partial mesh topologies are more scalable than full mesh topologies because the bandwidth and CPU resources required are reduced, although this is at the expense of optimal routing and efficient network utilization (if some CEs are used as transit nodes). Network redundancy is also reduced, although partial mesh networks are usually designed so that redundant routes/connections are used where they are needed the most. For example, in Figure 8-2, CE nodes CE2 and CE5 could be core nodes and the other CE nodes could be edge nodes. In which case, in this topology, the edge nodes have redundant connections/routes to reach the core. Customers are often forced to use partial mesh topologies due to factors such as cost (i.e., full mesh networks are more expensive) and due to geographic restrictions.

8.3 Hub and spoke VPN topologies

In a hub and spoke (or star) topology, a VPN site can either be a spoke or a hub for a particular VPN (although if a VPN site belongs to multiple VPNs it may be a hub for some VPNs and a spoke for others). All the spokes in a hub and spoke topology have direct routes/connections to reach the hub, but do not have direct routes/connections to reach each other. Figure 8-3 shows an example of a hub and spoke topology in which CE2 is the hub and all the other CE nodes are spokes.

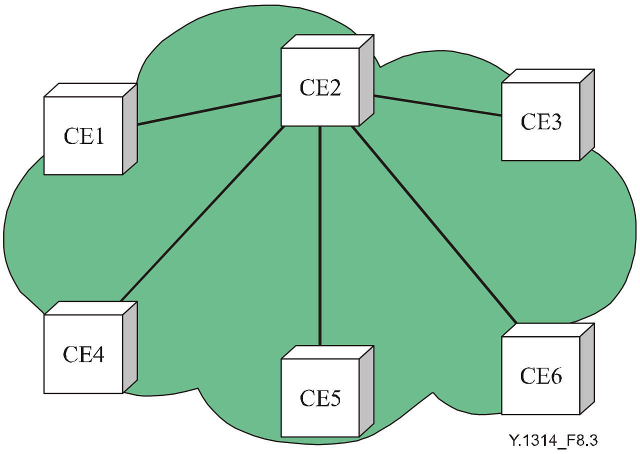


Figure 8-3/Y.1314 – An example of a hub and spoke VPN topology

In some cases, the hub may be configured as a transit node so that spokes can communicate with each other via the hub. However, in other cases, connectivity between spoke nodes may not be permitted. A common use of a hub and spoke topology is for connecting offices (spokes) to a corporate headquarters (hub). Using a hub and spoke topology allows the use of centralized network resources (e.g., Internet access, firewalls, and email servers), which can lead to a reduction in costs compared with a distributed network resources approach.

9 VPN QoS considerations

There are a large number of sources of information about Quality of service (QoS), which contain different definitions of what QoS actually means. ITU-T Rec. E.800 defines QoS as being the collective effect of service performances, which determine the degree of satisfaction of a user of the service. ITU-T Rec. G.1000 provides a framework and definitions for communications quality of service, and ITU-T Rec. G.1010 defines a model for multimedia Quality of Service (QoS) categories from an end-user viewpoint. The functions required to meet the quality of service requirements defined in these Recommendations and elsewhere are dependent on the network mode of operation. Therefore, quality of service requirements may have an impact on a VPN service provider's choice of VPN server layer technology and the VPN client layer's technologies that can be supported.

9.1 Circuit-switched layer networks

In connection-orientated circuit-switched layer networks, a path based on a physical link, optical wavelength, SDH/SONET VC, or TDM timeslot is established and dedicated to a single connection between APs in the network for the duration of the connection. When a new connection is requested, the network must decide whether or not to accept the connection; and if so, how to route it through the network and what resources to reserve for the connection. Connection Admission Control (CAC) mechanisms are used to accept a connection if bandwidth is available, or to reject it when the bandwidth of a requested connection exceeds the available bandwidth.

Data is transmitted at a constant bit rate in exactly the same order in which it was sent. Connections can be set up manually using static provisioning, or dynamically using signalling mechanisms or automated provisioning tools. The ability to set up new connections is dependent on having spare capacity in the network. If a connection is established then delivery of data across the connection is guaranteed.

In CO-CS networks such as a PSTN, delay is primarily a function of transmission distance. Switching delay in CO-CS network nodes is relatively small when compared to transmission (propagation) delay, especially when calls traverse long-distance trunks.

9.2 Packet-switched layer networks

In packet-switched networks, packets are forwarded based on information in the packet header. Packet switching provides connectivity while making efficient use of network resources by sharing them with many users (based on the assumption that not all users need to use the resources all of the time). Packet forwarding behaviour for flows or connections can be described by a set of parameters called traffic descriptors. Examples of traffic descriptors include mean packet/bit rate, maximum burst length/packet size, and probability of packet arrival within a fixed interval. User quality requirements are often expressed in terms of acceptable packet loss, delay and jitter.

Traffic shaping mechanisms may be used to regulate the amount of traffic admitted to the network, generally on a per queue/flow, per connection, or per interface basis. Congestion can occur in packet-based networks if traffic volume exceeds the forwarding capabilities of a network entity (NE) or available network capacity. When the network becomes congested, packets may be buffered which introduces a delay, or they may be dropped.

In packet-switched networks, delay is dependent on the transmission distance associated with the underlying physical server layer, plus a number of other factors at the packet-switched layer. The factors that introduce delay at the packet-switched layer include packet size, link speeds, per hop forwarding delay (which can be broken down into packetization, compression/decompression, switching/routing, and buffering delay), and number of hops. Priority control is required in packet-based networks in order to guarantee a diverse range of quality levels. Generally, priority control is implemented using separate queues per connection, per flow, or per QoS class at each interface, and controlling the priority of each queue. Packet scheduling mechanisms are used to allocate packets to a particular queue according to specific policies.

9.2.1 Connection-orientated packet-switched

In connection-orientated packet-switched layer networks, connections are established and maintained until connectivity is no longer required (regardless of whether data is being transmitted or not). Just as with connection-orientated circuit-switched layer networks, connections can be established via manual provisioning, a management system, or a signalling protocol. The current state of the network can be determined by monitoring the utilization of network resources and/or by characterizing the behaviour of connections already admitted. CAC mechanisms can be used to reserve the peak bandwidth of the connection required for constant bit rate (CBR) traffic sources. Alternatively, statistical multiplexing schemes may be used with CAC mechanisms to assign less than the peak bandwidth required in order to increase network efficiency. However, it can be difficult to characterize the bandwidth of a requested connection as the bandwidth required may vary significantly over time.

In a CO-PS network (e.g., an ATM network), if CBR services are supported (no over-subscription) per hop-forwarding delay remains constant and, therefore, delay/jitter can be calculated/guaranteed. If, however, services are overbooked to increase network utilization (which they usually are), then delay/loss will be introduced at congested nodes due to buffering or dropping of out of contract traffic. Although per hop forwarding delays become variable, the other factors such as link speeds, distance/number of hops (and packet size in ATMs case) remain constant.

9.2.2 Connectionless packet-switched

In connectionless packet-switched networks, once the data is sent, the connection is broken until further information is either sent or received (a packet can be thought of as a connection that exists for the time duration it takes the packet to be transmitted and received). There is no connection state stored and, therefore, successive packets do not necessarily follow the same path or arrive in the order in which they were sent. Traffic is sent at a variable bit rate and resources are normally assigned as needed on a first-come, first-served basis.

In CL-PS networks (e.g., IP networks), factors that determine delay such as packet size, link speeds, number of hops, and per hop forwarding delay are variable, especially when techniques are used to provide load balancing. Rate limiting/traffic shaping can be implemented at the edge to limit the amount of traffic entering a network, but due to the any-to-any nature of CL-PS traffic (and the increase in peer-to-peer networking), it is difficult to predict per link bandwidth utilization across a CL-PS network. Traffic monitoring along with modelling techniques can be used to develop a traffic matrix, and IGP metrics can be tweaked to provide increased link utilization but, due to the bursty and unpredictable nature of CL-PS traffic, the simplest/safest way to ensure service guarantees can be met is to over-provision the network.

However, even with over-provisioning, due to the non-deterministic nature of connectionless traffic, nodes/links in a CL-PS network can become congested, especially in the event of link/node failure or denial of service (DoS) attack. Also, the impact of link/node failure is not limited to traffic traversing the failed link/node, rerouting can cause congestion elsewhere in the network. A common approach to protecting premium traffic against network congestion is to use priority-based queuing (e.g., based on RFC 2475 Differentiated Services Architecture for IP) to control per class forwarding behaviour, i.e., higher priority traffic receives preferential treatment over lower priority traffic. This allows a provider to offer customers multiple levels of service (e.g., premium, real-time, best-effort) and to price services accordingly. The drawback to the Differentiated Services (Diffserv) approach is that bandwidth can only be reserved on a per aggregate basis and, therefore, delivery of individual flows within an aggregate cannot be guaranteed.

An alternative (or complementary) approach is to use an Integrated Services Architecture (based on RFC 1633) in which Resource Reservation Protocol (RSVP, RFC 2205) is used to reserve capacity along an end-to-end path by signalling a flow's requirements before sending any packets. Due to the fact that bandwidth can be reserved on a per flow basis, it is possible to provide guaranteed delivery for individual flows. This replicates the CAC model used in CO networks in which traffic is not sent until CAC has been performed to ensure there is sufficient capacity in the network. The major drawback to this approach is that it can place a significant (RSVP) processing load on core routers, which increases proportionately with the number of packet flows that require resource reservation.

Another approach that supports the reservation of resources on a per flow basis is the use of flow based routers. Flow based routers maintain per flow state and only accept new flows if there is sufficient resource available. As with RSVP, the challenge with this approach is that processing loads increase as the number of flows increases. However, there are routers available today that do support per flow routing for large numbers of flows.

10 Functions required for client/server VPN establishment

In establishing a client/server VPN there is a strict ordering of events that must take place. VPN client layer flows/connections cannot be established until the VPN server layer flows/connections have been established. Likewise, the VPN server layer flows/connections cannot be established until the server layer connections/flows (for which the VPN server layer is a client) have been established. This ordering of flow/connection establishment is due to the fact that a client layer topology is determined by the topology of the underlying server layer, which is recursive right down to the duct.

10.1 VPN server layer establishment

Assuming that the underlying server layer topology has been established and the VPN server layer TCP/TFPs and CPs/FPs have been configured with addresses, there are three main steps involved in establishing VPN server layer connectivity between VPN client layer members:

Step 1: Discover VPN members and store VPN membership information.

Step 2: Calculate routes between VPN members at the VPN server layer.

Step 3: Establish connections/tunnels/VLANs between VPN members at the VPN server layer.

Each of the functions required to support VPN server layer establishment and maintenance along with the individual functional entities are described in further detail in Table 10-1.

Table 10-1/Y.1314 – VPN server layer functions

Function	Functional entities	Network elements	VPN server layer mode
VPN membership discovery	Discovery of VPN members (VPN client layer CPs/FPs belonging to the same VPN)	PE	All
	Distribution/collection of VPN membership information (including joins, leaves, availability)	PE	All
	VPN membership information maintenance	PE	All
	Mapping of VPN client layer CPs/FPs to VPN server layer APs	PE	All
VPN server layer routing	Distribution/collection of VPN server layer reachability/topology/resource information	PE, P	All
	Maintenance of VPN server layer reachability/topology/resource information	PE, P	All
	Calculation of the best route(s) between VPN server layer APs	PE, P	All
VPN server layer tunnel/connection establishment	Connection admission control (CAC)	PE, P	All
	Notification of success/failure of connection/tunnel request	PE, P	All
	Assignment and configuration of VPN server layer de-multiplexing fields	PE, P	All
	Distribution of connection/tunnel information, e.g., QoS, de-multiplexing fields, bandwidth, etc.	PE, P	All

10.1.1 VPN membership discovery

In order to establish the VPN server layer topology between PEs, it is necessary to first of all determine which PEs are connected to CEs that are members of the particular client/server VPN. This function may be performed manually by a human operator based on the known network topology. Alternatively, this function may be performed dynamically via a centralized server/system or distributed protocol in order to automate/simplify the provisioning process. In order to support dynamic discovery, PEs must be configured with VPN identifiers to indicate that they are connected to one or more CEs that belong to a particular VPN. An example of a centralized server/system for discovery is the use of an authentication server (e.g., RADIUS) to distribute information about VPN members as part of the client authentication process. An example of a distributed protocol is the use of BGP for RFC 2547 VPNs, which uses route targets as VPN identifiers to ensure PEs only receive information about the VPNs of which they are members.

10.1.2 VPN server layer routing

If the underlying server layer (the layer below the VPN server layer) between the source/sink VPN server layer termination points is a single one-hop P2P connection/flow, then no routing needs to take place as there is only one route/path available. On the other hand, if there are alternative paths/routes across intermediate nodes to the same destination, or if the underlying server layer provides a P2MP[4] topology, then routing must be performed at the VPN server layer in order to discover the topology and/or calculate the best route(s) to the destination.

10.1.2.1 The need for routing

In the case of CO layer networks, signalling cannot take place until a route/path at the given layer has been calculated. In the CL layer network case, a packet cannot be forwarded until a route to the destination has been calculated/configured. This is not to say that every node in the network needs to have an explicit route to every other node in the network. Network address summarization is commonly used in conjunction with routing domain hierarchy in order to improve scalability. The ultimate form of address summarization is the use of default routes, which can be used as a 'catch all' mechanism to forward a packet regardless of its destination address.

One exception to the rule that a CL packet cannot be forwarded until a route has been calculated (or default route configured) is when the CL technology supports broadcasting. Broadcasting refers to the replication and forwarding of packets with unknown destination addresses across all server layer trails in the topology (except the trail that the packet was received on). An example of technology that supports this functionality is Ethernet. Another exception to the rule is the way in which token ring networks operate. In token ring layer networks, when a node receives a packet it retransmits the packet sending it onto the next node in the ring until it circulates back to the source node where it is removed. The destination node retains a copy of the frame and indicates that it has received the frame by setting the response bits in the frame. Although there are technologies that do not require routing, it should be noted that these technologies are not ideal as VPN server layer technologies. For layer networks to scale to large numbers of nodes over a large geographical area, routing, along with hierarchical address structures, are fundamental requirements. Mechanisms such as broadcasting and token passing are inherently insecure from a VPN perspective and are highly inefficient for the transmission of unicast (P2P) traffic.

[4] The reference to P2MP here refers to the outbound server layer topology from the perspective of a single source PE. The actual overall layer network topology could be any-to-any based on a full/partial mesh of bidirectional connections/flows between PEs.

10.1.2.2 Example network topologies that require routing

Figure 10-1 shows an example of a network where there are two alternative routes/paths (A and B) to the same destination.

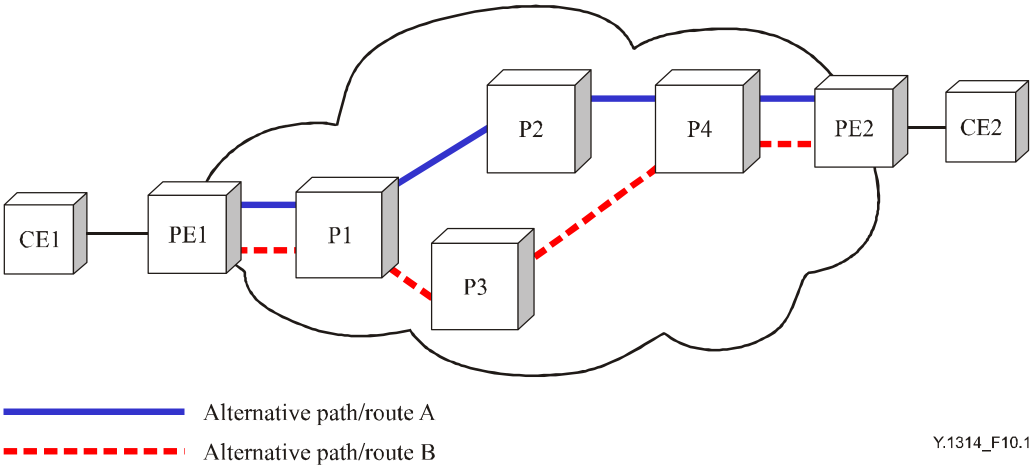


Figure 10-1/Y.1314 – Multiple routes/paths to the same destination

As Figure 10-1 shows, route A from CE1 to CE2 passes through PE1, P1, P2, P4 and PE2, whereas route B passes through PE1, P1, P3, P4 and PE2. This information is depicted using a functional model in Figure 10-2.

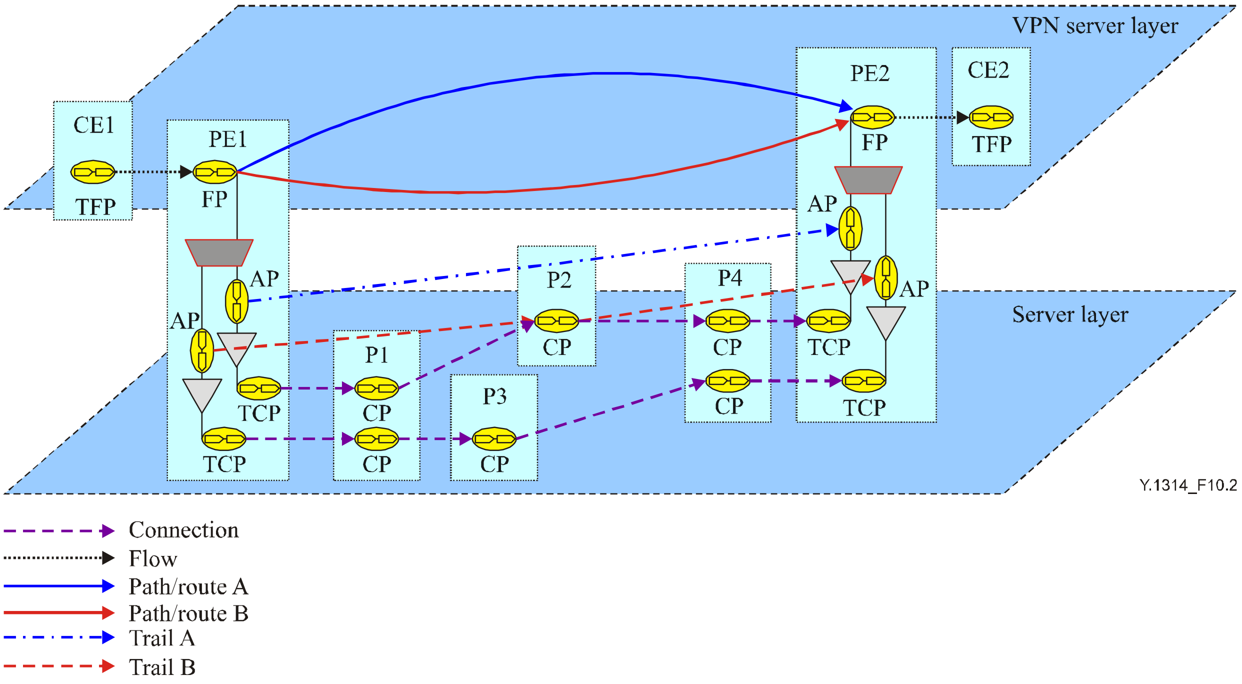


Figure 10-2/Y.1314 – Functional model for multiple paths/routes

Figure 10-2 shows two alternative server layer trails (A and B) that may be used by the VPN server layer. Based on the route calculated via the routing function at the VPN server layer, one of the server layer trails will be selected (or both if load balancing is required) in order to transmit the VPN server layer flow(s) between the VPN server layer source TFP located in CE1 and sink TFP located in CE2.

Figure 10-3 shows a case where a server layer provides P2MP connectivity from CE1 (the source) to CE2, CE3, and CE4 (the branch sinks in the P2MP topology).

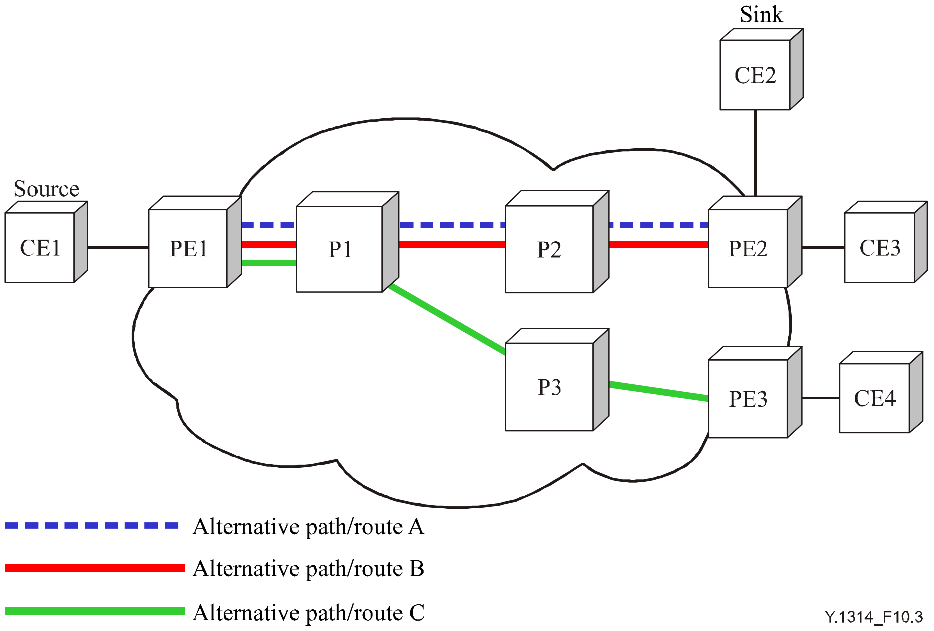


Figure 10-3/Y.1314 – P2MP server layer topology

The network in Figure 10-4 is shown as a functional model in Figure 10-4.

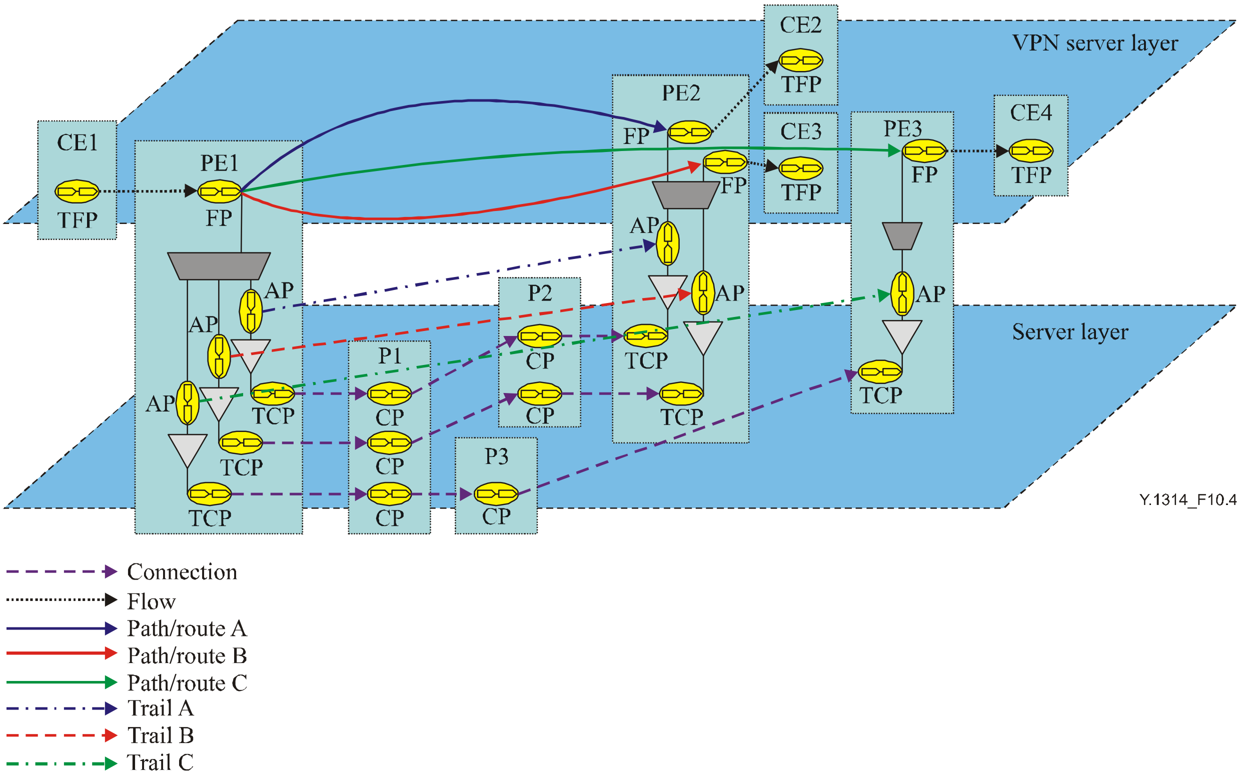


Figure 10-4/Y.1314 – Functional model of P2MP server layer topology

If CE1 is the source and CE2 is the sink for a particular VPN server layer flow, then CE1 needs to know what the route is to reach CE2. However, the routes/paths at the VPN server layer shown in Figure 10-4 are provided by P2P trails in the underlying server layer, i.e., only one route from the source TFP to each branch sink TFP in the P2MP topology exists. This means that the routing function only needs to discover the topology; it does not need to perform route calculation (as only one route to each sink exists). Following topology discovery, flows from CE1 destined for CE2 will use route/path A, provided by server layer trail A.

10.1.2.3 Alternative routing approaches

When routing is required, a human operator may perform the routing function in which case the human operator calculates routes across the network based on the known network topology and on resource utilization information. One example where routing may be performed manually is when configuring dual homed CE nodes when the VPN client layer is IP based. In this example, it may make sense to use static routes (i.e., one primary and one floating default routes) as only two alternative routes exist.

If a network management system (NMS) is used to perform the routing function, then the management system must discover the network topology by requesting or collecting reachability/topology/resource information, and then use this information to calculate routes and distribute the routing information to the network nodes. One example where routing is performed by a NMS is in establishing P2P connections across an SDH-based layer network. Before the connections can be established, the NMS must first calculate the best route(s) through the network.

If a dynamic routing protocol is used to perform the routing function, then reachability/topology/resource information is distributed across the network via the routing protocol to each node and used to calculate the best route/path to reach each destination. One example of a dynamic routing protocol is the routing component of PNNI used for ATM layer networks (although it can also be used with other network technologies) to discover the network topology and calculate routes for dynamic connections. Another example is RPR, which uses topology messages to discover the ring topology. When a node receives a topology message, it appends its MAC address and passes it to the next node in the ring, eventually the packet returns to its source with a topology map (list of addresses) of the ring.

An alternative to using a dynamic routing protocol in the control plane is to use address learning in the data plane, an example of a network technology that employs this mode of operation is Ethernet. Ethernet uses spanning tree (to avoid loops by pruning the network topology) and transparent bridging (based on source address learning) in the data plane to forward packets onto the correct destination without having to broadcast them to all nodes/end-stations. However, if address learning in the data plane is used then the network technology must also support broadcasting to forward packets with destination addresses that have not yet been learnt. Due to the fact that routes are not known until packets with the corresponding addresses have been received, address learning in the data plane cannot be used to perform the routing function for CO layer networks, it is only suitable for CL layer networks.

10.1.3 VPN server layer signalling

For the purposes of this Recommendation, signalling refers to the exchange of information required to establish CL tunnels (e.g., L2TP tunnels) and CO connections (e.g., ATM VPI/VCIs). The information required includes parameters such as multiplexing/de-multiplexing fields, QoS (e.g., delay, jitter), bandwidth, encryption keys, and resiliency (e.g., 1+1 protection).

One of the key differences between tunnels and connections is that connections always require signalling (or manual provisioning) to establish the connections before any user data can be sent. Although some tunnelling techniques (e.g., L2TP tunnels, explicitly configured GRE tunnels) also require signalling of tunnel parameters before user data can be sent, others such as soft/dynamic GRE and IP-in-IP tunnels do not require any signalling. These tunnelling techniques simply encapsulate a VPN client layer packet inside a VPN server layer packet header based on local policy/routing information. Intermediate (P) nodes encountered between the tunnel source/sink termination points only look at the VPN server layer packet header to determine whether and how to forward the packet towards the VPN server layer sink (destination PE). The VPN client layer headers are only used once the packet reaches the destination PE where the VPN server layer sink is located. Note also that intermediate nodes would often not be able to make any sense (e.g., be able to route) in the internal VPN client layer headers.

Connection admission control (CAC) is performed at connection set up time to determine whether sufficient bandwidth is available in the underlying server layer to maintain the QoS requirements of the client layer. Traffic descriptors (e.g., Peak Cell Rate (PCR) and Sustained Cell Rate (SCR) used in ATM) are used during client layer signalling to request the appropriate resource from the underlying server layer. The ability to determine how much bandwidth is available at the server layer, based on a request from a client layer, means that the CAC function must peer in the control plane with both the server and client layers.

In the case of CO-CS server layers, CAC is based on the amount of physical bandwidth available at the network layer at which it is requested (e.g., spare TDM timeslots or WDM wavelengths). In the case of CO-PS server layers, CAC is based on the amount of spare bandwidth not being used by existing connections. This information is made available by maintaining information about the state (e.g., up/down, amount of resource used) of each connection at each node for that particular layer network. Unlike CO-CS layer networks where the available bandwidth is constrained by the physical bandwidth available, in CO-PS networks per connection policing must be performed (especially if statistical multiplexing is assumed) at each node in the network to ensure that each connection only transmits/receives the amount of traffic agreed during connection establishment.

In the case of CL-PS server layer networks, CAC may be performed based on the bandwidth available at a physical/logical interface, or a queue/flow/class of service level. As with CO-PS layer networks, policing must be performed at each node in the network based on the bandwidth requested. However, unlike the CO-PS case where state is maintained for each connection, the equivalent (i.e., per flow) information is not normally maintained in CL layer networks[5]. This, combined with the non deterministic any-to-any nature of CL traffic means that CAC in CL layer networks relies on the use of extensive traffic monitoring and modelling to develop a traffic matrix, along with network over-provisioning to ensure that bandwidth is available, especially under failure conditions. If per VPN hard CAC and strong SLAs are required then a CO server layer network should be used rather than a CL server layer network.

10.2 VPN client layer authentication/configuration

The functions required to establish connectivity between CE and PE nodes at the VPN client layer may be performed using static provisioning or by using dynamic protocols. Static provisioning may be performed via manual configuration or via automated network management systems. The functional entities involved in establishing VPN client layer connectivity are shown in Table 10-2.

[5] Exceptions include the use of RSVP RFC 2205 (end-to-end signalling based solution) and flow state routing (hop-by-hop solution), where the state of each flow is maintained and new flows are rejected if there is not enough bandwidth available.

Table 10-2/Y.1314 – VPN client layer authentication and configuration functions

Function	Functional entities	Network elements	VPN client layer mode
CE/user authentication, authorization, and accounting (AAA)	Authentication: Identification of the CE/user based on the authentication parameters, e.g., a valid username and password	CE, PE	All
	Authorization: Grant or deny access to the VPN client layer network resources/services	CE, PE	All
	Accounting: Measurement of resources/services used	CE, PE	All
VPN client layer network element configuration	Assignment and configuration of VPN client layer network addresses across VPN client layer CPs/FPs and TCPs/TFPs	CE, PE	All
	Assignment and configuration of VPN identifiers across VPN client layer CPs/FPs belonging to the same VPN	PE	All
	Configuration of per VPN profiles and policies	CE, PE	CO-PS, CL-PS

10.2.1 CE/user AAA

The CE/user AAA function controls access to the VPN client layer, enforces policies, supports usage audits, and provides the information necessary to bill for VPN services. The AAA functions may be performed by the PE device that the CE connects to, a separate device, or a mixture of the two.

In some cases a centralized authentication server is likely to be required for user/CE authentication, and in others only the CE and PE may be involved in the authentication process. An example of the former is when IEEE 802.1X is used for authentication of an Ethernet CE device. In this example the PE would be the authenticator, and a centralized authentication server would be used to carry out the authentication. An example of the latter is the authentication of control messages (e.g., BGP messages) from a CE to authenticate the message source and protect against spoofing.

10.2.2 VPN client layer network element configuration

During VPN client layer provisioning the network elements at the edge of the customer and provider network must be configured with the following parameters: VPN client layer network addresses, VPN client layer network de-multiplexing fields, VPN identifiers, and per VPN policies/profiles. The configuration could be performed during the authentication/authorization process or independently. An example of the former is following successful authentication, a CE could automatically be configured with a specific bandwidth allocation and packet marking profile based on information received from an authentication server. An example of the latter is the use of manual configuration or Dynamic Host Configuration Protocol (DHCP) to assign an IP address to a CE.

VPN client layer addresses to be configured at PE CPs/FPs and CE TCPs/TFPs or CPs/FPs are the addresses belonging to the VPN client layer network (e.g., IP addresses for an IP VPN client or ITU-T Rec. E.164/NSAP addresses for an ATM VPN client).

Client layer network de-multiplexing fields only need to be configured if multiple VPN clients are being carried across the same CE to PE link, or if the VPN client layer network technology employed always carries a de-multiplexing field. An example of the former is an Ethernet VPN client layer, which only needs to use VLAN tags if it needs to support multiple VPNs. An example of the latter is ATM, which always uses VPI/VCI values in traffic unit (cell) headers. In some cases, the de-multiplexing field configuration will be dependent on the physical configuration rather than configuring a value in a packet header (e.g., attaching a fibre to the correct ingress interface at a PE corresponding with the correct egress DWDM wavelength).

Although a VPN identifier is a name used to identify a particular VPN and only needs to be assigned/configured if support for dynamic VPN membership discovery and signalling is required, it may also be useful from an operational perspective (e.g., to aid troubleshooting, billing). An example of a VPN identifier used for dynamic discovery and signalling is the Route Target attribute used for RFC 2547 VPNs. A VPN identifier may be configured on a PE statically via manual/OSS provisioning, or dynamically (e.g., as part of the authentication process using RADIUS). If the VPN identifier is to be used for discovery/signalling, then it should be unique at least within a single routing/signalling domain (and ideally globally unique if support for inter-AS/provider VPNs is required).

Configuration of per VPN profiles and policies for packet-based VPN clients may be required in the CE device, in the PE device or in both. Examples of VPN profiles and policies that may need to be configured depending on the VPN service include: rate limiting/traffic shaping, packet marking/classification, and route/connection selection for multi-homed sites (i.e., one primary, one backup).

10.3 VPN client layer routing and signalling

As with the VPN server layer, VPN client layer routing is required where multiple routes/paths between source and sink TCPs/TFPs exist, or if the VPN server layer trails create a P2MP topology at the VPN client layer. If the VPN client layer is CO and dynamic provisioning at the VPN client layer is to be supported, then signalling is also required.

An important point to note here is that the VPN server layer trails must be established before VPN client layer routing/signalling may take place. The VPN client layer data plane topology is based on the topology of the underlying VPN server layer trails, and therefore it is not possible to perform route calculation or signal connections/tunnels until the VPN server layer trails have been established.

The VPN client layer routing and signalling functions along with the individual functional entities are described in Table 10-3.

Table 10-3/Y.1314 – VPN client layer routing and signalling functions

Function	Functional entities	Network elements	VPN client layer mode
VPN client layer routing	Distribution/collection of VPN client layer reachability/topology/resource information	CE, PE	All
	Maintenance of VPN client layer reachability/topology/resource information	CE, PE	All
	Calculation of the best route(s) between VPN client layer APs	CE, PE	All
VPN client layer tunnel/connection signalling	Connection admission control (CAC)	PE, P	CO-CS, CO-PS
	Notification of success/failure of connection/tunnel request	PE, P	All
	Assignment and configuration of VPN client layer de-multiplexing fields	PE, P	All
	Distribution of VPN client layer connection/tunnel information, e.g., QoS, de-multiplexing fields, bandwidth, etc.	PE, P	All

10.3.1 Any-to-any CL-PS VPN client layer connectivity

If VPN server layer trails provide an any-to-any full/partial mesh topology for a CL-PS VPN client layer network with multiple sites, then nodes containing VPN client layer TFPs/FPs (i.e., PE/CE nodes but not P nodes) must make forwarding decisions regarding where to forward a packet based on VPN client layer address information. This means that CE and PE nodes must exchange VPN client layer routing information using dynamic routing protocols via the control plane, or static routes must be configured using manual or OSS provisioning. An alternative to using dynamic routing protocols or static routing is to use address learning in the data plane, as is the case with Ethernet which uses source-based address learning to unicast traffic to the correct destination.

The routing information for each VPN must be isolated from the routing information from other VPNs. This is to provide VPN forwarding separation (i.e., to ensure packets are not routed to nodes belonging to a different VPN) and to allow overlapping VPN client layer address spaces to be used. This may be achieved using physically separate per VPN PEs, or common PEs with logically/virtually separate routing information databases. An alternative would be to use common PE devices and routing tables but allocate separate address spaces for each VPN client[6]. An example of a VPN solution that supports routing at the VPN client layer is RFC 2547. RFC 2547 uses dynamic or static CE to PE routing along with MP-BGP to distribute VPN client layer routing information between PEs and separate virtual routing tables to provide VPN client layer route isolation.

10.3.2 On-demand VPN client layer dynamic connection set up/teardown

In most cases, CO-CS and CO-PS VPN client layer connections will be statically configured via manual or OSS provisioning. However, if on-demand dynamic connection establishment is required, then peering in the control plane peering (routing and signalling) at the VPN client layer must take place between all CPs and TCPs (i.e., between PE and CE nodes). Also, CAC must be performed at connection set up time to determine whether sufficient bandwidth is available at the VPN server layer for the VPN client layer connection. This means that the CAC function must peer with both the VPN server layer network and VPN client layer network control planes. If the VPN server layer and client layer technologies are different, then peer level control plane interworking must take place between the VPN client and server layers.

10.3.3 Customer-controlled on-demand connections

Customer-controlled on-demand dynamic connections refers to the case where the customer has some (or total) control over the CE node which allows them to establish new VPN client layer connections. The advantage of this capability from a customer perspective is that it gives them the flexibility to dynamically set up VPNs as and when they are needed, and be charged for their use accordingly. For example, a customer may wish to establish an on-demand connection for a short period of time to download/upload a large file (e.g., an application or video file) or to set up a reliable connection for a videoconference. An example of the use of on-demand dynamic VPN client layer connection set up is the use of PNNI to set up/teardown SPVCs across VPN server layer trails provided using virtual paths.

One important factor to be taken into consideration when considering adding support for on-demand dynamic VPN client layer connections is the distribution of address/topology information. A service provider is unlikely to want to disclose their network topology or internal

[6] There are several major disadvantages to this approach: it requires careful management of the address space by the service provider, agreement by the customer to use addresses allocated by the service provider (the customer may want to use their own addresses), and packet filtering to guarantee isolation between VPNs, which is a tedious and error prone task.

network addressing to customers for security reasons. Therefore, it is desirable for the routing function at the PE to distribute reachability information to the CE only. Another important consideration is in deciding what action to take at connection set up time if bandwidth is not available. The ability to establish a new VPN client layer connection is dependent on the availability of server layer trails between the source and sink termination points. If trails do not exist or there is not enough spare bandwidth, then either the connection must be rejected, or a new VPN server layer connection/tunnel must be established (or the bandwidth increased for existing connections/tunnels). In order to establish new VPN server layer connections/tunnels or increase the bandwidth of existing connection/tunnels, CAC must be performed to ensure that bandwidth is available in the underlying server layer.

If the server layer is CL then it is not possible to perform hard CAC and, therefore, the network must be over-provisioned to allow new VPN server layer tunnels to be established. The disadvantage with this approach is that it requires careful network planning and control/policing to ensure that existing VPN server layer tunnels are not affected in any way. If the underlying server layer is CO then hard CAC can be performed to ensure that bandwidth is available to establish new VPN server layer connections/tunnels. However, every connection request at layer n has an impact on the bandwidth available at layer n–1, and this is recursive down to the duct. As we get closer to the duct, the bandwidth granularity and provisioning/hold times for the connections increase. In general, if there is insufficient capacity in an underlying server layer to support a new connection then the connection should be rejected. Server layer capacity should be provided as a result of capacity planning activities including network modelling and usage analysis/forecasting.

10.3.4 Service provider controlled on-demand connections

Service provider controlled on-demand dynamic connections refers to the scenario where the service provider manages the CE node and uses routing/signalling to dynamically establish new VPN client layer connections. The advantage of this capability from the service providers perspective is that it allows them to establish end-to-end VPN client layer connections dynamically rather than having to use static configuration (i.e., manual or OSS provisioning). An example scenario where VPN client layer dynamic connection set up may be useful is where two or more ATM access networks are interconnected via a MPLS core. In this example, PNNI could be used to set up/teardown SPVCs at the VPN client layer across MPLS VPN server layer trails. As the VPN client and server layer technologies are different, peer level interworking in the control plane must take place.

In the service provider controlled on-demand dynamic connections case, even if the provider manages the CE node on behalf of the customer, distributing internal addressing and topology information to the CE has risks associated with it, e.g., the CE is located within the customer's premises rather than the provider's premises. One way to avoid these security risks would be to use static/manual provisioning between the CE device and the adjacent intermediate node in the provider's network, and to use dynamic routing/signalling from that node back to the PE. For example, if the VPN client layer is ATM then the VC could be manually provisioned between the CE and the provider's ATM switch that it connects to, and then PNNI used end-to-end between the ATM switches. Regarding the control of connection/tunnel set up at different layers in the layer network hierarchy, as the provider controls the on-demand connections, the provider has more control over what happens in the network. However, careful network planning and NMS monitoring of connections at each layer must still take place, particularly if the department in the company responsible for managing the VPN client layer is different from the department responsible for managing the VPN server layer (and the server layers below).

11 Functions required for peer level VPN establishment

Assuming that the underlying server layer topology has been established and the VPN peer layer TFPs and FPs have been configured with addresses, there are three main steps involved in establishing VPN peer layer connectivity between VPN members:

Step 1: Discover and authenticate VPN members and store VPN membership information.

Step 2: Calculate routes between VPN members at the VPN peer layer.

Step 3: Configure VPN peer layer network elements to provide VPN isolation.

Each of the functions required to support VPN peer layer establishment and maintenance along with the individual functional entities are described in further detail in Table 11-1.

Table 11-1/Y.1314 – VPN server layer functions

Function	Functional entities	Network elements
VPN membership discovery	Discovery of VPN members	CE/PE
	Distribution/collection of VPN membership information (including joins, leaves, availability)	CE/PE
	VPN membership information maintenance	CE/PE
CE/user authentication, authorization, and accounting (AAA)	Authentication: Identification of the CE/user based on the authentication parameters, e.g., a valid username and password	CE, PE
	Authorization: Grant or deny access to the VPN client layer network resources/services	CE, PE
	Accounting: Measurement of resources/services used	CE, PE
VPN peer layer routing	Distribution/collection of VPN peer layer reachability/topology/resource information	CE, PE, P
	Maintenance of VPN peer layer reachability/topology/resource information	CE, PE, P
	Calculation of the best route(s) between VPN peer layer APs	CE, PE, P
VPN peer layer network element configuration	Configuration of per VPN packet filters	PE
	Configuration of per VPN route filters	PE
	Configuration and exchange of per VPN/CE encryption keys	ES, CE, PE
	Assignment and configuration of VLAN IDs	CE, PE, P

11.1 VPN membership discovery

In customer provisioned peer level VPN scenarios where the VPN is transparent to the provider (e.g., an IPsec VPN over the Internet), before establishing the VPN it is necessary to first of all determine which CEs belong to the VPN. In provider provisioned VPN scenarios (e.g., Ethernet VLAN based VPNs), the provider needs to discover which PEs are connected to CEs that are members of the VPN. Discovery may be performed manually by a human operator based on the known network topology, or may be performed dynamically via a centralized server/system or distributed protocol.

11.2 CE/user authentication, authorization, and accounting (AAA)

A CE/user AAA function is used in provider provisioned VPN scenarios to control access to VPN peer layer resources. AAA is also used to enforce policies, support usage audits, and to provide information necessary to bill the customer for VPN services. The AAA function may be performed by a PE, a separate device, or using a mixture of the two. For example, if IEEE 802.1X is used to authenticate a CE for an Ethernet VLAN based VPN, the PE would be the authenticator and a centralized authentication server could be used to carry out the authentication.

11.3 VPN peer layer routing

Where alternative paths/routes between VPN members exist, routing must be performed at the VPN peer layer in order to discover the topology and/or calculate the best route(s) between VPN members. As CE, PE, and P nodes all belong to the VPN peer layer, all three node types are involved in any route/path calculations. The routing function may be performed manually by a human operator, or may be performed dynamically via a centralized server/system or distributed routing protocol. For the purposes of this Recommendation, routing includes transparent bridging based on source address learning in the data plane.

11.4 VPN peer layer network element configuration

There are a number of alternative functions for providing VPN isolation. One option is to configure per VPN packet filters on shared PE nodes to ensure full reachability between sites of a single customer, but isolation between customers. Another option is to use dedicated PE nodes and configure route filters so that although P nodes contain all customers' routes, PE nodes only contain routes for a single customer. Packet/route filtering is only applicable to provider provisioned VPN scenarios and therefore must be carried out by PE nodes.

An alternative to using route/packet filtering where connectivity exists between customers (e.g., across the Internet) is to use packet encryption. Using packet encryption ensures that if customers receive packets from a VPN they do not belong to, they cannot obtain the data contained inside the packet. Packet encryption is performed by PE nodes for provider provisioned VPNs and by CE nodes or end systems for customer provisioned VPNs.

Common types of cryptography used to support encryption/decryption include secret key and public key cryptography. Secret key cryptography is most suited to closed user groups where secret keys can be held and securely distributed by a single authority, e.g., in an enterprise VPN environment. The advantage of public-key cryptography is that it allows users to communicate securely without having prior access to a shared secret key. This approach uses two keys, a private key that is kept secret and a public key that needs to be distributed to all VPN members. Public and private keys are mathematically related and anyone that does not possess a specific private key cannot decrypt the information in the encrypted packet. A common use for public key cryptography is for the exchange of secret keys to be used for secret key cryptography.

Where Ethernet is used as the VPN peer layer technology, VPN isolation can be achieved by assigning and configuring VLANs. VLANs are normally assigned and configured manually or via OSS, although dynamic protocols can also be used. To provide end-to-end connectivity between CEs, VLANs must be configured correctly on CE, PE and P nodes.

12 VPN OAM functions

OAM tools and functions are essential in maintaining operational efficiency in large-scale networks. Examples of important network connection/flow characteristics conveyed via OAM functions include quality, integrity and validity. If a layer network does not support OAM or has some missing OAM functionality, then that particular layer network is functionally deficient with regard to that OAM functionality. Higher/lower layer OAM functions/tools cannot be used as a replacement/substitute to provide the same functionality, particularly when it comes to fault localization. This is not to say that it is not possible to provide VPN services using network technologies that have OAM functions missing. However, missing OAM functionality is likely to significantly increase operating costs and operational complexity.

Table 12-1 presents some of the key OAM functions and identifies which network elements should support the associated functions.

Table 12-1/Y.1314 – Client/server OAM functions

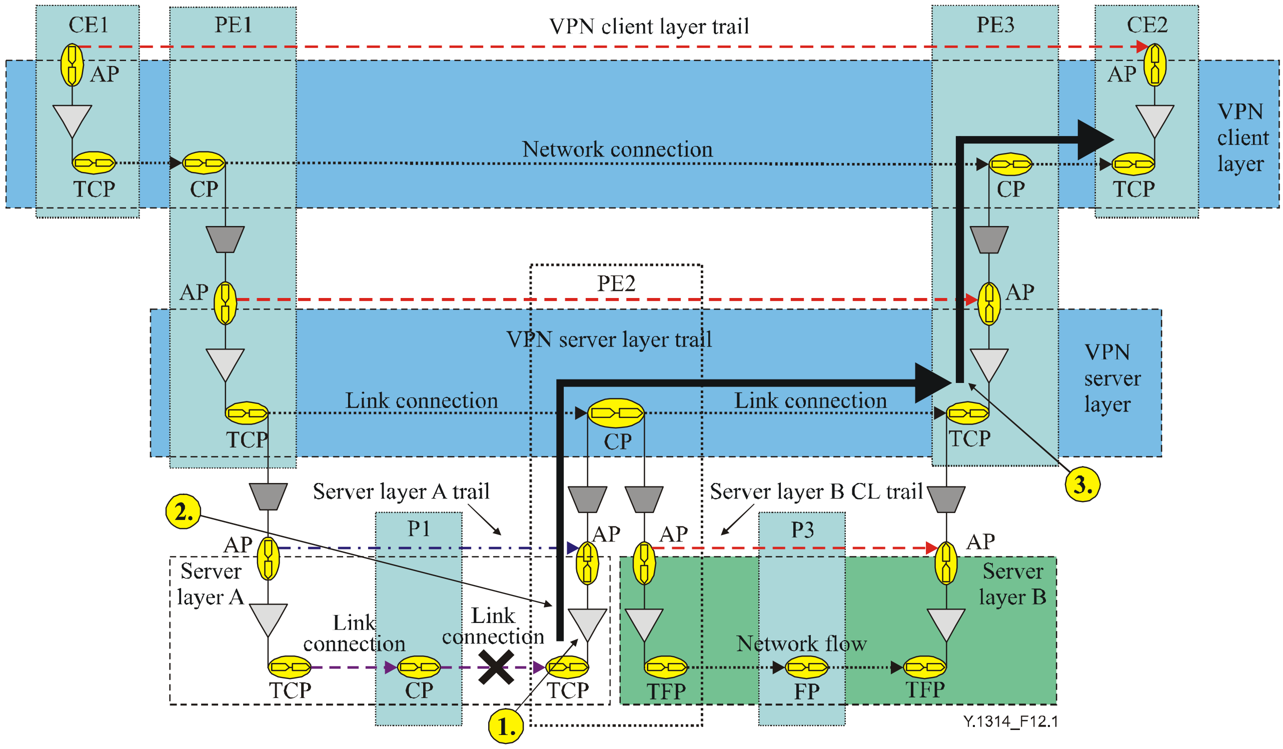
Function	Functional entities	Network elements
VPN client layer OAM	VPN client layer fault detection/management	CE and PE
	VPN client layer performance monitoring	CE and PE
	VPN client layer OAM activation and deactivation	CE and PE
VPN server layer OAM	VPN server layer fault detection/management	PE and P
	VPN server layer performance monitoring	PE and P
	VPN server layer OAM activation and deactivation	PE and P
VPN peer layer OAM	VPN peer layer fault detection/management	CE, PE, P (all)
	VPN peer layer performance monitoring	CE, PE, P (all)
	VPN peer layer OAM activation and deactivation	CE, PE, P (all)

12.1 Fault management

Fault management includes fault detection, localization, correction and on-demand diagnostic testing. Defects must be detected and processed at the connection/flow sink termination point in the layer network that they occur. Failure to do this will lead to ambiguous fault indications, which significantly increase operational complexity and the time taken to resolve a fault. Upon detecting a failure, in addition to generating and sending alarms to the NMS, to prevent alarm storms in client layer networks an FDI (Forward Defect Indication) or AIS (Alarm Indication Signal) signal should be passed to client layer network(s) using the appropriate syntax of the OAM used by the particular client layer technology affected (if one exists).

The most important fault detection mechanism is the use of Connectivity Verification (CV), which is a common requirement across all three networking modes. Put simply, it requires the source of a traffic flow to deterministically (in some manner) identify itself to the sink. How this is achieved is dependent on the networking mode and is explained in subsequent clauses. Fault localization is another key requirement across all three networking modes in order to determine the root cause of a failure. In addition to the initial failure information, on-demand diagnostic testing tools may be used to localize the fault.

An example of a failure scenario in a VPN server layer is described from a functional perspective in Figure 12-1.



1. Server layer A detects LOC based on non-receipt of CV packets

2. Server layer A forwards a FDI to the VPN server layer

3. The VPN server layer receives a FDI and propagates it up to the VPN client layer

Figure 12-1/Y.1314 – Client/server FDI propagation

In this example, detection of a link failure by the sink termination function in server layer A causes a FDI/AIS to be generated and to be passed onto the VPN server layer. The FDI is propagated to the VPN server layer sink termination function, which in turn sends an FDI to the client layer. This behaviour is recursive up to a layer network that does not support FDI. Therefore, although it is not shown here, on receipt of an FDI the VPN client layer may send an FDI to the layer above, depending on what technology the layer above is (e.g., ATM, Ethernet, IP, etc.).

The only place an alarm should be raised is at the trail termination point in the layer network where the original fault is detected. In particular, there should be no alarms raised in any client layers affected (this is the key purpose in sending FDI to them). Further, if single-ended monitoring of both directions is required, then a BDI (Backward Defect Indicator) can be sent in the other direction. Further details on how defect indicators/alarms (including details of defect and unavailability entry and exit criteria and consequent actions) can be found in Recommendations that deal with OAM for specific layer network technologies, e.g., ITU-T Rec. Y.1711 for MPLS OAM.

Fault Correction is responsible for the repair of a fault and for the control of procedures that use redundant resources to replace equipment or facilities that have failed. For example, in case of a fibre cut or node failure, protection switching or connection re-routing may be used to restore/maintain service.

On-demand diagnostic testing tools are generally used for fault localization, but can also be used for verification of correct connectivity/configuration of a connection/tunnel prior to it being brought into service. Loopback is one example of a diagnostic test during which a loop on a network connection from the source back towards the source via a connection or terminating connection point thereby isolating that section of the connection.

12.2 Performance management

Performance Monitoring (PM) is the process of collection, analysis, and reporting of performance data. This data is used to assess and maintain the network as well as to document the quality of service to customers. If multiple levels of class of service are supported (e.g., based on the Diffserv architecture) then performance monitoring should be performed on a per class of service basis. Performance monitoring includes, amongst other things, the detection of signal degradation, latency/jitter monitoring, and the counting of lost packets. There are a number of different objectives for performance monitoring including SLA maintenance, support for traffic engineering, per customer accounting, and service restoration/protection switching (e.g., due to signal degradation).

It is important to establish the relationship between defects, availability and PM. There is a particular ordering which can be summarized as follows:

1) The network mode defines the defects that are of relevance (which are different for each mode) and the nature of the OAM required.

2) All defects should be defined in terms of standardized entry/exit criteria and consequent actions.

3) The unavailability state is entered when a defect or unacceptable performance degradation has persisted for a consecutive number of seconds. In SDH the unavailable state is entered following 10 consecutive Severely Errored Seconds[7] (SES) and is exited following 10 consecutive non SES. To ensure harmonization, the unavailability period should be the same across all layer networks, i.e., 10 seconds.

4) PM for SLA purposes is only valid when in the available state and, therefore, PM for SLA purposes must be suspended when the unavailable state is entered.

When in the available state, PM for SLA purposes is a unidirectional measurement. However, as most applications require both directions (upstream and downstream) to be working, if either direction fails then both directions are deemed to have failed from an application's perspective. This means that unavailability is an "OR" function of each direction and, therefore, if either direction enters the unavailable state, then PM for SLA purposes should be suspended in both directions.

12.3 OAM activation/deactivation

For the CO-CS and CO-PS modes, the basic defect detection/handling OAM mechanisms should be activated/deactivated in synchronization with trail set up and teardown, which could be via NMS/OSS provisioning or via signalling. For example, CV generation should be activated at the source before activating CV detection at the sink in order to avoid meaningless alarms. The provisioning or signalling method used to set up the trail should also be able to tell the trail sink point what source identifier (e.g., TTSI in ITU-T Rec. Y.1711) to expect in the data plane for a particular trail in order to determine which trail the OAM packets it receives belong to.

[7] A SES is a period of one second with a bit error ratio equal to or higher than 1E-3, or during which an LOS or an AIS is detected.

12.4 Defects relevant to each network mode

The potential transport defects that may occur in a VPN client or server layer network are dependent on the network mode that the layer network technology belongs to. A summary of mode dependent potential defects is given below:

- **CL-PS**: Breaks only;
- **CO-PS**: Breaks, swaps, and merges;
- **CO-CS**: Breaks, swaps (but only between alike entities).

In the following clauses, each of the network modes is described in further detail to describe what the key OAM requirements and considerations are for that particular mode. It should be noted that this is not intended to be an in-depth list of OAM requirements for each mode. Only fundamental functional differences are highlighted in order to show how the network mode that the VPN client and server layers belong to impacts the OAM functions/tools required.

12.4.1 CL-PS layer networks

Under an assumption of consistent and valid routing information (which actually applies to all modes), misconnectivity defects (i.e., swaps or merging) cannot occur in CL-PS layer networks. Each packet contains both a source address (this is the CV function) and a destination address that contains all the information required for the packet to be routed correctly at each network node. Therefore, the only defect possible in a CL-PS layer network is the case where there is a break (e.g., due to routing, link, or node failures). In CL-PS layer networks, the CV function is an integral part of the packet header as each packet contains a network unique source/destination address. In CL-PS networks, the control and user data usually share the same data path and, therefore, if there is a failure in the control plane (e.g., a routing adjacency goes down), then by implication it can be assumed that connectivity has been lost and user data cannot be sent either. This is generally how faults are detected and corrected in CL-PS layer networks, e.g., non-receipt of routing hellos in the control plane indicates that there is a fault in the data plane and, therefore, corrective action (e.g., selection of an alternative route) must be taken. One case where this is not true, however, is when load balancing is used in IP layer networks. In this case, multiple routes to the same destination exist, therefore, if one route becomes unavailable this may not be detected by the control plane as control traffic may simply use one of the other routes available. To detect failures where load balancing is employed, an OAM mechanism must be used that tests connectivity across all available routes.

12.4.2 CO-PS layer networks

In the CO-PS case, only the layer network access points are aware of the network unique addresses used by the routing function to calculate the best route/path through the network for the connection. Once the route/path has been calculated, signalling (or manual provisioning) is used to allocate and configure locally significant ingress/egress multiplexing/de-multiplexing fields (or link connection identifiers), which are used in the data plane for switching the packet to the correct destination. As multiplexing/de-multiplexing fields are only locally significant, the same values can be reused by upstream/downstream nodes for the same connection, or for different connections. The reuse of multiplexing/de-multiplexing fields combined with the lack of network unique addressing in the data plane means that in the CO-PS layer networks, in addition to breaks, we can experience swapping and merging defects. As CO-PS packets are transmitted asynchronously and do not contain unique source/destination network addresses, the CV function needs to be added in some deterministic fashion, normally by transmitting CV packets at a specific rate. The rate at which CV packets are sent needs careful consideration to ensure that unnecessary actions are not taken in the event of transient error bursts.

12.4.3 CO-CS layer networks

CO-CS layer networks do not suffer from merging as the multiplexing/de-multiplexing fields are based on physical time/space/frequency link-connection identifiers with a constant bit rate. Defects that may occur in a CO-CS layer network include breaks and swapped connections, however, swapped connection defects can only occur between exactly alike trails, e.g., swaps cannot occur between a VC12 and VC4 in SDH. In the case of the CO-CS layer networks, as with CO-PS layer networks, the CV function must be added in some deterministic fashion. As a CO-CS frame is transmitted at a constant bit rate (whether there is data to be sent or not), the CV information can be carried in each frame using the frame transmit rate as the CV transmit rate, for example, the J0 trace message in an SDH VC4 frame has a base 125 µs insertion rate. In the CO-CS case, control traffic is always carried OOB and therefore OAM functions must be provided on a per connection basis for the user and control data planes.

12.4.4 Control and user data plane separation

The control data and user data in CO-PS networks can be transmitted using different data planes (often referred to as out of band (OOB) control); and, as noted in the previous paragraph, in the CO-CS mode this is forced behaviour in all cases. This separation of control and user data planes is advantageous for many reasons, and especially from a security and network stability perspective as it protects the control plane from user plane attacks and overloading/congestion problems caused by user plane traffic. When the user and control data planes are separated, it clearly cannot be assumed that a failure in the control plane indicates a failure in the user data plane (or indeed vice versa). Therefore, an OAM mechanism in CO-PS layer networks where OOB control is used must be used on a per data plane (i.e., per connection) basis. This is also the case where the control traffic may use the same data plane as some user traffic, but not all (e.g., in MPLS traffic engineering (TE) may be used to provide explicit routing for certain traffic types and, therefore, user data traffic need not follow the same path as the control packets used to set up the TE tunnels).

Failure to use data plane based OAM mechanisms may lead to a scenario in which a connection carrying data packets may experience a fault, but as control traffic is being transmitted using a separate connection, control information continues to flow and, therefore, the fault is not detected by the control plane. Without a data plane OAM detection mechanism the connection source will continue to send user data creating a traffic black hole, or worse, compromising security of customer data by sending traffic to the wrong location.

In order to unambiguously determine in which direction a fault has occurred, and to support correct fault handling for both P2P and P2MP connections, OAM should operate unidirectionally. Also, single-ended monitoring of faults in both directions should be supported if possible. This is particularly important where a customer or provider has control over one end of a connection/tunnel but not the other end, e.g., in an inter-provider VPN scenario where each end of a P2P VPN client layer connection is located in a different service provider's network.

13 Functional convergence and service scenarios

Mapping VPN service requirements to the functions described in this Recommendation allows network operators to select the most appropriate network technologies and mechanisms required to provide the VPN services they wish to offer. Selecting the best of breed mechanisms/protocols for each function allows individual functional components to evolve independently. This approach also supports the reuse of common mechanisms/protocols across different VPN network technologies (where appropriate) to reduce costs and complexity.

13.1 Client/server VPN services scenarios

The functions (and therefore mechanisms/protocols) required to support client/server VPNs are dependent on the client/server network modes as well as the actual VPN service being offered. For example, some customers may want to be able to set up on-demand SVCs between multiple sites as and when they are needed, whilst other customers may just want permanent connections based on a known static topology. As another example, some customers may want to use per user/CE authentication to increase security whilst other customers may believe restricting physical access to network infrastructure is adequate. Tables III.1 and III.2 provide some examples of different service scenarios and identify some example mechanisms/protocols that can be used to provide the functions required.

13.2 Peer level VPN scenarios

The functions required to support peer level VPNs are dependent on the peer layer network technology and the type of VPN service being offered. For example, authentication in the encryption based VPN case is mandatory in order to use the correct keys whereas in the Ethernet VLAN based VPN case authentication (e.g., using IEEE 802.1X) provides extra security but is not essential. Table III.3 provides some examples of different service scenarios and identifies some example mechanisms/protocols that can be used to provide the functions required.

14 VPN security considerations

This Recommendation does not introduce any new security issues. However, security is a fundamental factor to consider when designing/developing VPN networks in order to select the network technologies and functional components that meet a customer's security requirements. There are inherent security risks associated with all VPN technologies due to the fact that a shared infrastructure is being used to transport traffic for multiple customers.

Network security is a huge area in itself and is therefore not investigated in detail within this Recommendation. Looking at security from a high level, the physical VPN network infrastructure must be protected from unauthorized access or malicious attack (e.g., by restricting access to buildings containing network equipment). In addition, unauthorized remote access from outside the VPN network infrastructure must also be prevented (e.g., by using firewalls to protect against attacks sources from the Internet).

As described in 5.1, in the client/server VPN case, a VPN server layer network must support multiplexing/de-multiplexing to provide data plane separation between multiple VPN client layers. This traffic separation must be combined with effective VPN access control at the edge of the network based on per customer VPN policies.

In the peer level VPN case, as described in clause 6, in order to support VPNs across a shared domain, the network technology used must have some means of providing VPN isolation. CEs must only be able to communicate with other CEs belonging to the same VPN, or only be able to decrypt packets from CEs belonging to the same VPN.

Security can be increased for both client/server and peer level VPNs by using encryption to encrypt user/control traffic units and authentication can be used to authenticate users and network nodes. Authentication for client/server and peer level VPNs is described in more detail in 10.2.1 and 11.2 respectively. Encryption is described in further detail in 6.2 and 11.4.

Appendix I

Location of VPN client layer TCPs/TFPs

Figure I.1 gives an example of a client/server VPN network, it shows the physical topology of the network in which the black line represents the VPN server layer and the grey lines represent physical links between the nodes.

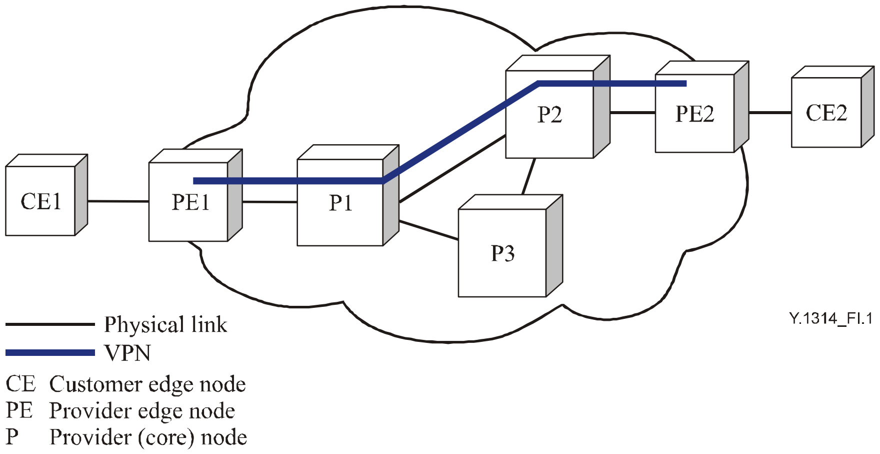


Figure I.1/Y.1314 – Client/server VPN physical topology – Example 1

Although Figure I.1 shows the physical topology and VPN server layer, it does not show the separate VPN client and server layer topologies or where the TCPs/TFPs are located. Figure I.2 shows a functional model based on the physical topology in Figure I.1 in which the TFPs are located in the CE nodes. In this example, the VPN server layer is CO (e.g., ATM) whilst the VPN client layer is CL (e.g., Ethernet), although any combination of CO or CL pairs are possible.

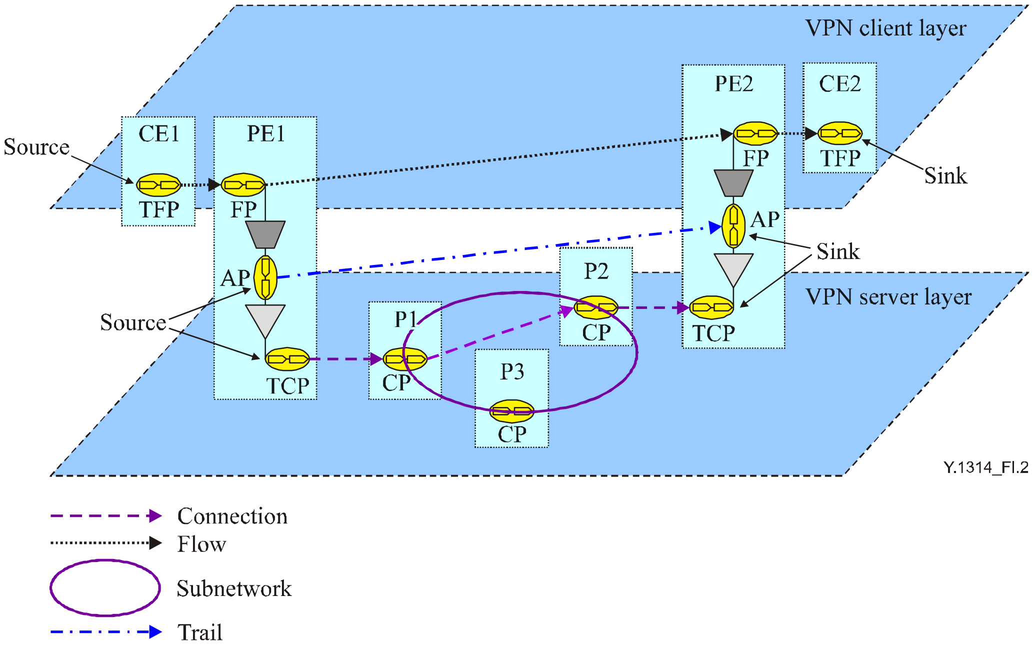


Figure I.2/Y.1314 – VPN client layer TFPs located in CE nodes

The CE and P nodes belong to the VPN client and server layers respectively, whilst the PE nodes belong to both layers. The TFPs in the VPN client layer identify where (which CE node in this case) the P2P VPN client layer flow starts (its source) and ends (its sink), and the FPs identify which PE nodes the P2P flow passes through. Likewise the TFPs in the VPN server layer identify the source and sink for the VPN server layer connection, and the FPs identify which P nodes the flow passes through. The APs in the VPN server layer identify the source/sink for the VPN server layer trail.

In the previous example, the VPN client layer TFPs were situated in the CE nodes (CE1 and CE2), however, this is not the case for all VPN client/server relationships. For example, the VPN client layer may be an Ethernet or IP layer network where the TFPs are located in hosts/end-systems.

Figure I.3 shows the physical topology of a client/server VPN network. If the VPN client layer was Ethernet then the C nodes would be Ethernet switches, and the end-systems/hosts would be computers/servers with Ethernet interfaces.

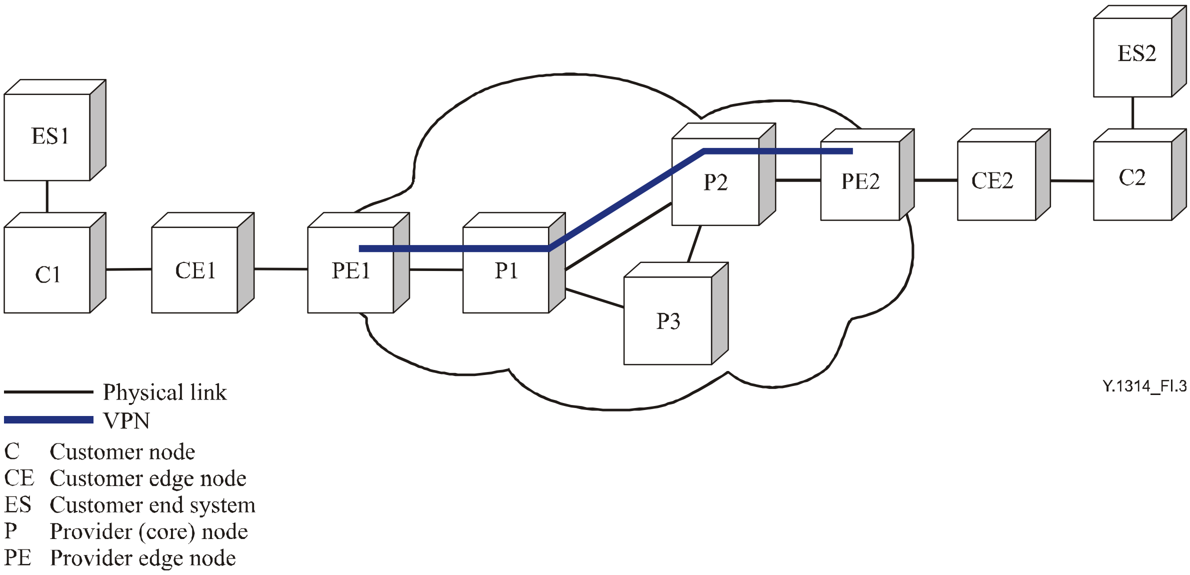


Figure I.3/Y.1314 – Client/server VPN physical topology – Example 2

A functional model based on the physical network depicted in Figure I.3 is presented in Figure I.4, where the VPN client layer's TFPs/TCPs are located in end-system/hosts rather than the CEs.

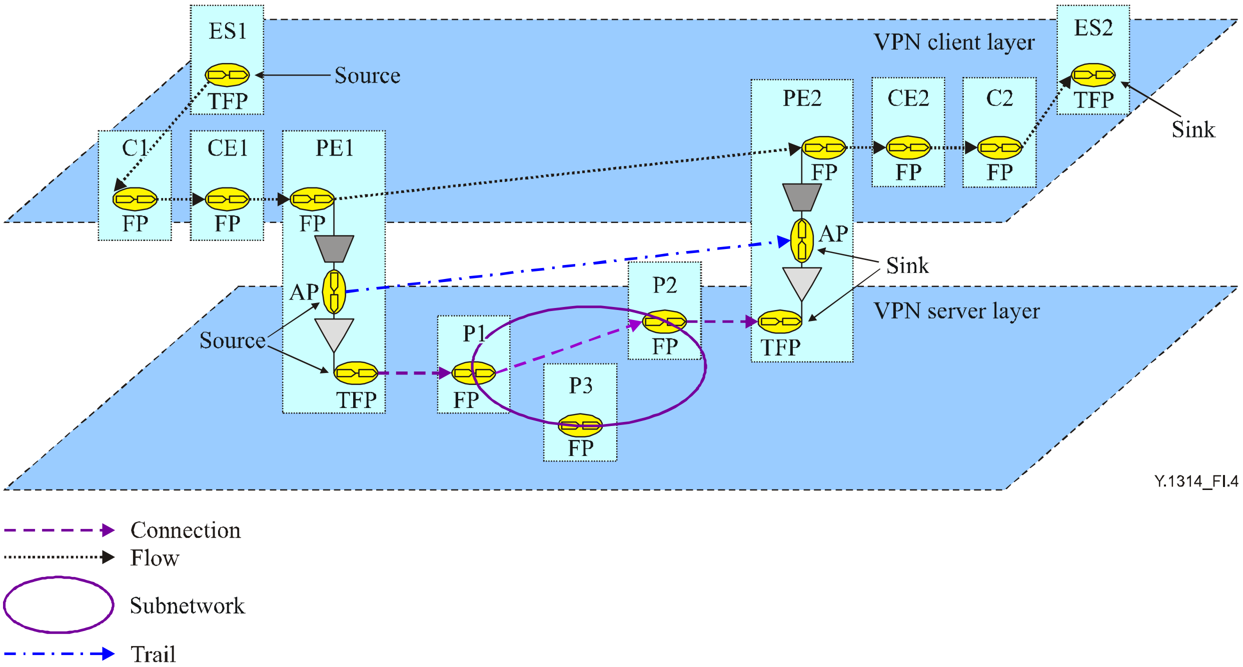


Figure I.4/Y.1314 – VPN client layer TFPs located in end-systems/hosts

The C and CE nodes along with the ESs belong to the VPN client layer. The PE nodes belong to the VPN sever layer and the client layer whilst the P nodes only belong to the VPN server layer. The TFPs in the VPN client layer identify the source and sink (i.e., ES1 and ES2 respectively) for the VPN client layer flow, and the FPs identify which C, CE, and PE nodes the flow passes through.

Although not illustrated in the previous examples, it is also possible that at one side of the VPN a source or sink TFP/TCP is located in the CE, whilst at the other end, the TFP is not located in the CE, i.e., a CP/FP is located in the CE and the TFP/TCP is located in a customer node or ES.

Appendix II

Client/server VPNs with multiple VPN server layers

Figure II.1 shows the physical topology of a client/server VPN network that uses two different VPN server layers, X and Y. Nodes PE1, P1 and P2 belong to VPN server layer X whilst nodes P3 and PE3 belong to VPN server layer Y. Node PE2 belongs to both server layers and acts as a gateway between the two.

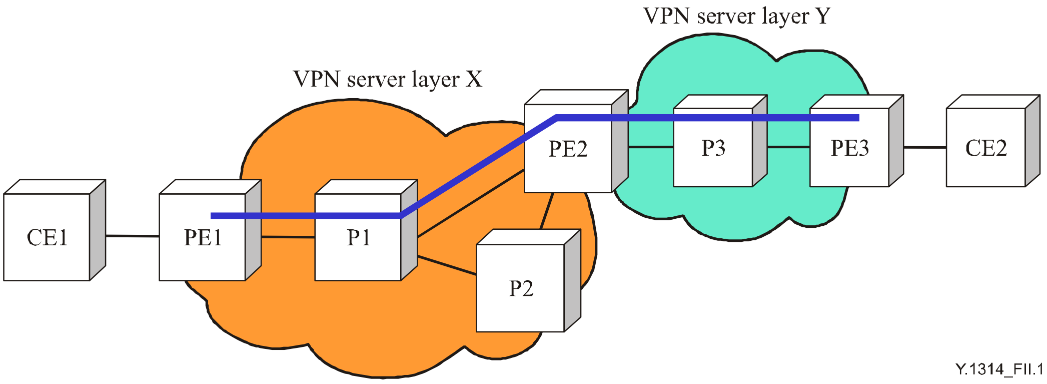


Figure II.1/Y.1314 – VPN server layer interworking physical topology

One method of interworking between VPN server layers X and Y would be to use client/server interworking as illustrated in Figure II.2. In this model, node PE2 belongs to the VPN server layer X and Y and all three PE nodes belong to the VPN client layer.

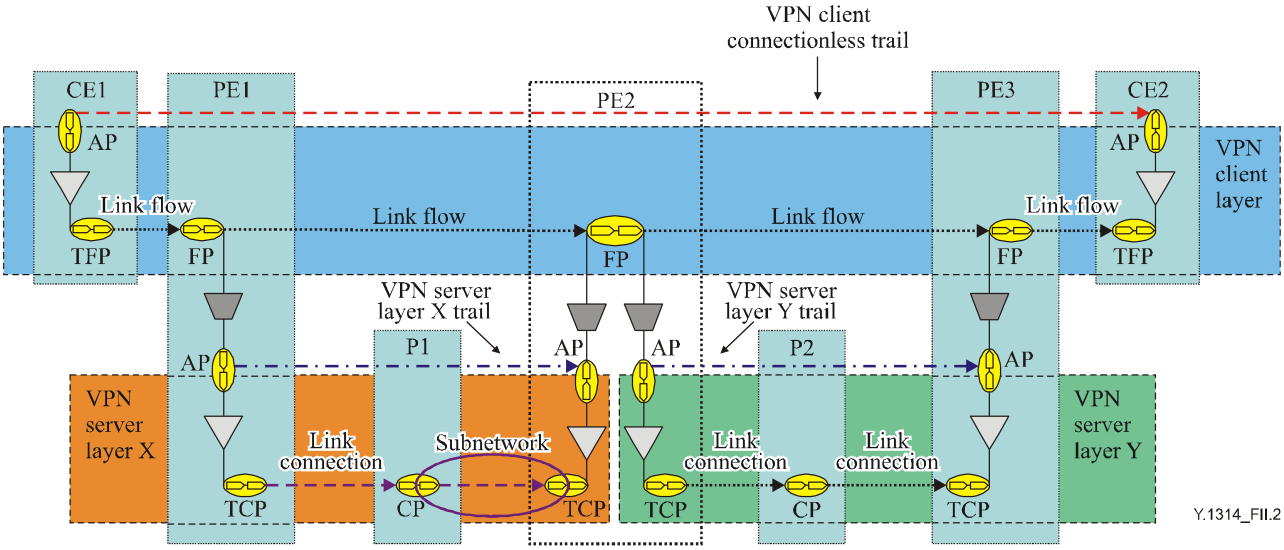


Figure II.2/Y.1314 – VPN server layer client/server interworking

The VPN server layer X source adaptation function adapts VPN client layer CI into AI in the VPN server layer X, and the sink adaptation function adapts VPN server layer X AI to VPN client layer CI. Similarly, the VPN server layer Y source adaptation function adapts the CI of the VPN client layer into AI in the VPN server layer Y, and the sink adaptation function adapts VPN server layer Y AI to VPN client layer CI.

The network elements in which the client/server adaptation takes place contain FPs or CPs belonging to the VPN client layer, which must be identified using VPN client layer addresses. So, for example, if the VPN client layer was IP, PE1, PE2, and PE3 would require IP addresses belonging to the VPN client layer.

Using multiple VPN server layers with client/server adaptation for CO VPN client layers means that a route/path must be dynamically/manually calculated across CPs and at least two link connections established end-to-end at the VPN client layer within the provider's network. Using multiple VPN server layers with client/server adaptation for CL VPN client layers means that a route/path must be dynamically/manually calculated across FPs, and that CL traffic units (i.e., packets) must be forwarded based on the address information at the VPN client layer. This is in contrast to the case where a single VPN server layer has been established end-to-end across the provider network between two CPs/FPs at the VPN client layer. In this case, only a single link connection/flow is required within the provider network between the VPN server layer trail source and sink and, therefore, a route/path does not need to be calculated across the provider network at the VPN client layer.

The alternative method of interworking between VPN server layers X and Y in Figure II.2 would be to use peer level interworking as illustrated in Figure II.3. In this model, node PE2 belongs to VPN server layer X and Y but does not belongs to the VPN client layer. PE1 and PE3 belong to VPN server layers X and Y respectively, and also belongs to the VPN client layer.

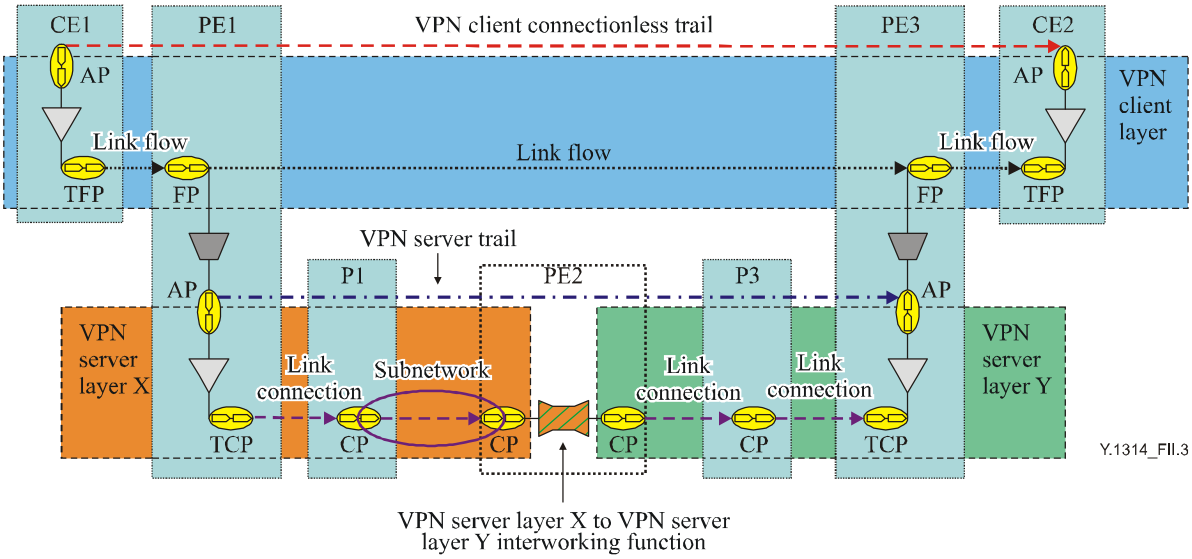


Figure II.3/Y.1314 – VPN server layer peer level interworking

The VPN server layer X source adaptation function adapts VPN client layer CI into AI in the VPN server layer X. The VPN server layer X to VPN server layer Y interworking function adapts the VPN server layer X AI to VPN server layer Y AI. The VPN server layer Y sink adaptation function adapts VPN server layer Y AI to VPN client layer CI.

The main factor to take into consideration when considering peer level interworking is that only certain network technologies can be interworked on a peer level basis, e.g., ATM and Frame Relay networks can be interworked on a peer level basis (using FRF.8), but IP and TDM networks can not. Peer level interworking requires interworking not only in the data plane but also in the control plane for functions such as routing, signalling, and OAM.

Appendix III

Examples of client/server and peer level VPN service scenarios

The tables below provide some examples of different VPN service scenarios and identify some example mechanisms/protocols that can be used to provide the functions required.

NOTE – Additional references related to the tables in this appendix are provided in the Bibliography.

Table III.1/Y.1314 – Client/server VPN service scenarios 1

	Layer 2 frame relay service over MPLS	**Layer 2 Ethernet VPWS service over IP/L2TPv3**	**Layer 3 RFC 2547 IP VPN service**
VPN client layer	Frame Relay	Ethernet	IP
VPN server layer	MPLS PW	IP/L2TPv3	MPLS
VPN membership discovery	RADIUS, BGP, Manual, NMS	RADIUS, BGP, LDP, RSVP-TE, Manual, NMS	BGP
VPN server layer routing	IGP, BGP, manual, NMS	IGP, BGP, manual, NMS	BGP
VPN server layer tunnel/connection establishment	LDP, BGP, Manual, NMS	L2TPv3 signalling	BGP
CE/user authentication, authorization, and accounting (AAA)	RADIUS, IEEE 802.1X, RMON, SNMP, NMS	RADIUS, IEEE 802.1X, RMON, SNMP, NMS	CE-PE routing protocol (e.g., EBGP with MD5), RMON, SNMP, NMS
VPN client layer network element configuration	NMS, manual	NMS, manual, E-LMI	DHCP, NMS, Manual
VPN client layer routing	NMS, manual	MAC address learning	EBGP, OSPF, manual/static
VPN client layer tunnel/connection signalling	NMS, manual	Non required as the client is CL-PS	Non required as the client is CL-PS
VPN client layer OAM	Frame Relay LMI	IEEE 802.1ag, E-LMI, IEEE 802.3ah, ITU-T Rec. Y.1731	IP Ping/traceroute
VPN server layer OAM	ITU-T Rec. Y.1711, ITU-T Rec. Y.1713, MPLS, VCCV, BFD/LSP ping	IP ping/traceroute	ITU-T Rec. Y.1711, ITU-T Rec. Y.1713, LSP Ping/traceroute

Table III.2/Y.1314 – Client/server VPN service scenarios 2

	Layer 1 SDH VPN service over OTN	**Layer 1 TDM VPN service over MPLS**	**Layer 2 ATM VPN service over SDH**
VPN client layer	SDH (e.g., STM-16)	TDM (e.g., E1)	ATM
VPN server layer	Lightpath / Optical Channel (OCh)	MPLS PW	SDH (e.g., VC4)
VPN membership discovery	ITU-T Rec. G.7714.1/Y.1705.1, Manual, NMS	RADIUS, BGP, LDP, Manual, NMS	Manual, NMS
VPN server layer routing	GMPLS/ASON routing protocols, manual, NMS	IGP, BGP, manual, NMS	GMPLS/ASON routing protocols, manual, NMS
VPN server layer tunnel/connection establishment	GMPLS/ASON signalling protocols, manual, NMS	LDP, BGP, Manual, NMS	GMPLS/ASON signalling protocols, manual, NMS
CE/user authentication, authorization, and accounting (AAA)	GMPLS/ASON protocols, SNMP, NMS	RMON, SNMP, NMS	ATM, PNNI/UNI security, RMON, SNMP, NMS
VPN client layer network element configuration	NMS, Manual	NMS, Manual	ATM UNI, Manual, NMS
VPN client layer routing	GMPLS/ASON routing protocols, manual, NMS	Manual, NMS	Manual/static, NMS, PNNI
VPN client layer tunnel/connection signalling	GMPLS/ASON signalling protocols, manual, NMS,	Manual, NMS	Manual, NMS, PNNI
VPN client layer OAM	SDH overhead (e.g., J0/J1/J2 trace bytes, G1 path status byte)	ITU-T Rec. G.775, AIS/LOS	F4 and F5 fault management, loopback, and continuity check (CC)
VPN server layer OAM	OCh overhead (e.g., Trail Trace Identifier (TTI) used in path/section monitoring (PM/SM))	ITU-T Rec. Y.1711, ITU-T Rec. Y.1713, MPLS, VCCV, BFD/LSP ping	SDH overhead (e.g., J0/J1/J2 trace bytes, G1 path status byte)

Table III.3/Y.1314 – Peer level VPN service scenarios

	IPsec VPN over the Internet	Ethernet VLAN VPN
VPN peer layer	IP	Ethernet
VPN membership discovery	Manual, NMS	Manual, NMS, RADIUS
CE/user authentication, authorization, and accounting (AAA)	IKE primary authentication (based on pre-shared keys or digital signatures), RMON, SNMP, NMS	IEEE 802.1x, RADIUS, RMON, SNMP, NMS
VPN peer layer routing	IGP routing protocols (e.g., ISIS, OSPF, RIP), BGP, manual, NMS	STP topology pruning and data plane address learning (transparent bridging)
VPN peer layer network element configuration	Configure the shared key, or request a certificate from the certification authority	Configure VLANs using manual configuration, NMS, or dynamic protocols
VPN peer layer OAM	IP ping, traceroute	IEEE 802.1ag, E-LMI, IEEE 802.3ah, ITU-T Rec. Y.1731

BIBLIOGRAPHY

The indicated references are subject to revision. Users of this Recommendation are encouraged to search for the most recent edition/draft of these references.

ATM UNI: ATM Forum UNI 4.1 (2002), "*ATM User Network Interface (UNI) Signalling Specification version 4.1*", af-sig-0061.001.

ATM Forum PNNI 1.1 (2002), *Private Network-Network Interface Specification v.1.1*, af-pnni-0055.001.

IEEE 802.1ad (2005, Draft 6.0), *Virtual Bridged Local Area Networks – Amendment 4: Provider Bridges*.

IEEE 802.1ag (2005, Draft 4.1), *Virtual Bridged Local Area Networks – Amendment 5: Connectivity Fault Management*, status: PAR approved, Task Group ballot in progress.

IEEE 802.1ah (Aug 2005, Draft 1.2), *Virtual Bridged Local Area Networks – Amendment 6: Provider Backbone Bridges*, status: PAR approved, Task Group ballot.

IEEE 802.1Q (2005), *Virtual Bridged Local Area Networks*, status: published.

IEEE 802.1X (2004), *Port-Based Network Access Control*, status: published.

IEEE 802.17 (2004), *Specific requirements – Part 17: Resilient packet ring (RPR) access method and physical layer specifications*, status: published.

IEEE 802.3ah (2004), *Specific requirements – Part 3: Carrier Sense Multiple Access With Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications* Amendment: *Media Access Control Parameters, Physical Layers, and Management Parameters for Subscriber Access Networks*, Ethernet in the First Mile amendment to IEEE Std 802.3.

IETF RFC 1633 (1994), *Integrated Services in the Internet Architecture: an Overview*.

IETF RFC 2205 (1997), *Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification*.

IETF RFC 2401 (1998), *Security Architecture for the Internet Protocol*.

IETF RFC 2409 (1998), *The Internet Key Exchange (IKE)*.

IETF RFC 2475 (1998), *An Architecture for Differentiated Services*.

IETF RFC 2547 (1999), *BGP/MPLS VPNs*.

IETF RFC 3036 (2001), *LDP Specification*.

IETF RFC 3209 (2001), *RSVP-TE: Extensions to RSVP for LSP Tunnels*.

IETF draft-ietf-bfd-base-03.txt (2005), *Bidirectional Forwarding Detection,* work in progress.

IETF draft-ietf-bfd-mpls-02.txt (2005), *BFD For MPLS LSPs*, work in progress.

IETF draft-ietf-l2tpext-l2vpn-05.txt (2005), *L2VPN Extensions for L2TP*, work in progress.

IETF draft-ietf-l2vpn-radius-pe-discovery-01.txt (2005), *Using RADIUS for PE-Based VPN Discovery,* work in progress.

IETF draft-ietf-l3vpn-bgpvpn-auto-06.txt (2005), *Using BGP as an Auto-Discovery Mechanism for Network-based VPNs*, work in progress.

IETF draft-ietf-l3vpn-rfc2547bis-03.txt (2004), *BGP/MPLS VPNs*, work in progress.

IETF draft-ietf-mpls-lsp-ping-09.txt (2005), *Detecting MPLS Data Plane Failures*, work in progress.

IETF draft-ietf-pwe3-control-protocol-17.txt (2005), *Pseudowire Setup and Maintenance using the Label Distribution Protocol*, work in progress.

IETF draft-ietf-pwe3-frame-relay-05.txt (2005), *Encapsulation Methods for Transport of Frame Relay Over MPLS Networks*, work in progress.

IETF draft-ietf-pwe3-vccv-06.txt (2005), *Pseudo Wire Virtual Circuit Connectivity Verification (VCCV)*, work in progress.

ITU-T Recommendation E.164 (2005), *The international public telecommunication numbering plan*.

ITU-T Recommendation E.800 (1994), *Terms and definitions related to quality of service and network performance including dependability*.

ITU-T Recommendation G.775 (1998), *Loss of Signal (LOS), Alarm Indication Signal (AIS) and Remote Defect Indication (RDI) defect detection and clearance criteria for PDH signals*.

ITU-T Recommendation G.826 (2002), *End-to-end error performance parameters and objectives for international, constant bit-rate digital paths and connections*.

ITU-T Recommendation G.827 (2003), *Availability performance parameters and objectives for end-to-end international constant bit-rate digital paths*.

ITU-T Recommendation G.1000 (2001), *Communications Quality of Service: A framework and definitions*.

ITU-T Recommendation G.1010 (2001), *End-user multimedia QoS categories*.

ITU-T Recommendation G.7714.1/Y.1705.1 (2003), *Protocol for automatic discovery in SDH and OTN networks*.

ITU-T Recommendation I.610 (1999), *B-ISDN operation and maintenance principles and functions*.

ITU-T Recommendation Q.933 (2003), *ISDN Digital Subscriber Signalling System No. 1 (DSS1) – Signalling specifications for frame mode switched and permanent virtual connection control and status monitoring*.

ITU-T Recommendation Q.2931 (1995), *Digital Subscriber Signalling System No. 2 – User-Network Interface (UNI) layer 3 specification for basic call/connection control*.

ITU-T Recommendation X.200 (1994), *Information technology – Open Systems Interconnection – Basic Reference Model: The basic model*.

ITU-T Recommendation Y.1413 (2004), *TDM-MPLS network interworking – User plane interworking*.

ITU-T Recommendation Y.1415 (2005), *Ethernet-MPLS network interworking – User plane interworking*.

ITU-T Recommendation Y.1711 (2004), *Operation & Maintenance mechanism for MPLS networks*.

ITU-T Recommendation Y.1713 (2004), *Misbranching detection for MPLS networks*.

ITU-T Recommendation Y.1731 (2006), *OAM functions and mechanisms for Ethernet based networks*.

MEF ETH OAM (2003), *Ethernet Services OAM*, Draft.

Frame Relay Forum FRF.8 (1995), *Frame Relay/ATM PVC Service Interworking Implementation Agreement*.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	**Global information infrastructure, Internet protocol aspects and next-generation networks**
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2006