MPLS VIRTUAL PRIVATE NETWORKS

A review of the implementation options for MPLS VPNs including the ongoing standardization work in the IETF MPLS Working Group

November, 2000

Paul Brittain, European Product Manager, MetaSwitch (pjb@metaswitch.com)
Adrian Farrel, MPLS Architect, Data Connection (af@dataconnection.com)

Data Connection Limited
100 Church Street
Enfield, UK
Tel: +44 20 8366 1177
http://www.dataconnnection.com

Table of Contents

MPLS VIRTUAL PRIVATE NETWORKS

1. INTRODUCTION

Historically, private WANs were provisioned using dedicated leased line connections, each line providing a point-to-point connection between two customer sites. Such networks are expensive to put in place, especially if the connections between sites need to support some level of redundancy. There is also no scope in such a system to share under-utilized bandwidth across several customers or, conversely, to increase the bandwidth available between particular sites dynamically in order to meet short-term peaks in demand.

Virtual Private Networks (VPNs) are a method of interconnecting multiple sites belonging to a customer using a Service Provider (SP) backbone network in place of dedicated leased lines. Each customer site is directly connected to the SP backbone. The SP can offer a VPN service more economically than if dedicated private WANs are built by each individual customer because the SP can share the same backbone network resources (bandwidth, redundant links) between many customers. The customer also gains by outsourcing the complex task of planning, provisioning and managing a geographically distributed network to the SP.

Unfortunately, existing VPN solutions are not all interoperable and may be tied to one equipment vendor and/or a single SP. This has created strong interest in IP-based VPNs running over the public Internet using standards-based interoperable implementations that work across multiple SPs.

Many of these IP-based solutions require IP address-mapping or double encapsulation using two IP headers. This can require complex configuration management and requires additional processing at the entry to and exit from the SP's networks.

The new Internet technology, Multi-Protocol Label Switching (MPLS) forwards data using labels that are attached to each data packet. Intermediate MPLS nodes do not need to look at the content of the data in each packet. In particular the destination IP addresses in the packets are not examined, which enables MPLS to offer an efficient encapsulation mechanism for private data traffic traversing the SP backbone. MPLS can, therefore, provide an excellent base technology for standards-based VPNs.

This white paper reviews the requirements placed on a base technology for VPNs, how MPLS meets these requirements, and the state of the ongoing standardization efforts within the IETF. Alternative VPN technologies are touched on briefly, but a detailed review of such alternatives is outside the scope of this paper.

The structure of this white paper is shown in the table of contents. Readers who are already familiar with RFC 2764 and MPLS concepts, in particular tunnels and label stacks, may prefer to skip to the *MPLS For VPNs* section.

A glossary of terms and a table of references are provided at the end of the paper.

2. BACKGROUND

2.1 Introduction to MPLS

Multi-Protocol Label Switching (MPLS) is a new technology that will be used by many future core networks, including converged data and voice networks. MPLS does not replace IP routing, but will work alongside existing and future routing technologies to provide very high-speed data forwarding between Label-Switched Routers (LSRs) together with reservation of bandwidth for traffic flows with differing Quality of Service (QoS) requirements.

MPLS enhances the services that can be provided by IP networks, offering scope for Traffic Engineering, guaranteed QoS and Virtual Private Networks (VPNs).

The basic operation of an MPLS network is shown in the diagram below.

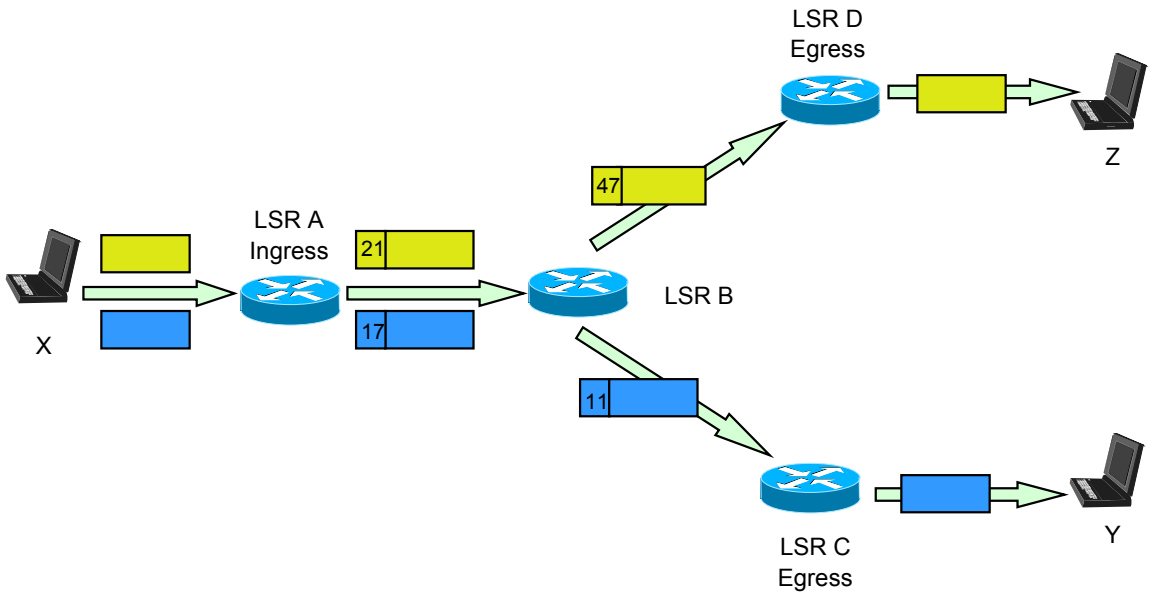

Fig.1: Two LSPs in an MPLS Network

MPLS uses a technique known as label switching to forward data through the network. A small, fixed-format label is inserted in front of each data packet on entry into the MPLS network. At each hop across the network, the packet is routed based on the value of the incoming interface and label, and dispatched to an outwards interface with a new label value.

The path that data follows through a network is defined by the transition in label values, as the label is swapped at each LSR. Since the mapping between labels is constant at each LSR, the path is determined by the initial label value. Such a path is called a Label Switched Path (LSP).

At the ingress to an MPLS network, each packet is examined to determine which LSP it should use and hence what label to assign to it. This decision is a local matter but is likely to be based on factors including the destination address, the quality of service requirements and the current state of the network. This flexibility is one of the key elements that make MPLS so useful.

The set of all packets that are forwarded in the same way is known as a Forwarding Equivalence Class (FEC). One or more FECs may be mapped to a single LSP.

The diagram shows two data flows from host X: one to Y, and one to Z. Two LSPs are shown.

- LSR A is the ingress point into the MPLS network for data from host X. When it receives packets from X, LSR A determines the FEC for each packet, deduces the LSP to use and adds a label to the packet. LSR A then forwards the packet on the appropriate interface for the LSP.

- LSR B is an intermediate LSR in the MPLS network. It simply takes each labeled packet it receives and uses the pairing {incoming interface, label value} to decide the pairing {outgoing interface, label value} with which to forward the packet. This procedure can use a simple lookup table and, together with the swapping of label value and forwarding of the packet, can be performed in hardware. This allows MPLS networks to be built on existing label switching hardware such as ATM and Frame Relay. This way of forwarding data packets is potentially much faster than examining the full packet header to decide the next hop.

  In the example, each packet with label value 21 will be dispatched out of the interface towards LSR D, bearing label value 47. Packets with label value 17 will be re-labeled with value 11 and sent towards LSR C.

- LSR C and LSR D act as egress LSRs from the MPLS network. These LSRs perform the same lookup as the intermediate LSRs, but the {outgoing interface, label value} pair marks the packet as exiting the LSP. The egress LSRs strip the labels from the packets and forward them using layer 3 routing.

So, if LSR A identifies all packets for host Z with the upper LSP and labels them with value 21, they will be successfully forwarded through the network.

Note that the exact format of a label and how it is added to the packet depends on the layer 2 link technology used in the MPLS network. For example, a label could correspond to an ATM VPI/VCI, a Frame Relay DLCI, or a DWDM wavelength for optical networking. For other layer 2 types (such as Ethernet and PPP) the label is added to the data packet in an MPLS “shim” header, which is placed between the layer 2 and layer 3 headers.

2.1.1 Label Distribution

In order that LSPs can be used, the forwarding tables at each LSR must be populated with the mappings from {incoming interface, label value} to {outgoing interface, label value}. This process is called LSP setup, or Label Distribution.

The MPLS architecture document [13] does not mandate a single protocol for the distribution of labels between LSRs. In fact it specifically allows multiple different label distribution protocols for use in different scenarios, including the following.

- LDP [11]
- CR-LDP [10]

- RSVP [9]
- BGP4
- OSPF

A detailed review of how these protocols are used for label distribution is outside the scope of this white paper. For a comparative analysis of RSVP and CR-LDP, refer to the white paper *MPLS Traffic Engineering: A choice of Signaling Protocols* [1] from Data Connection.

Several different approaches to label distribution can be used depending on the requirements of the hardware that forms the MPLS network, and the administrative policies used on the network. The underlying principles are that an LSP is set up either in response to a request from the ingress LSR (downstream-on-demand), or pre-emptively by LSRs in the network, including the egress LSR (downstream unsolicited). It is possible for both to take place at once and for the LSP setup to meet in the middle.

New ideas introduced by the IETF in the MPLS Generalized Signaling draft [14] also allow labels to be pushed from upstream to set up bi-directional LSPs.

Alternatively, LSPs may be programmed as static or permanent LSPs by programming the label mappings at each LSR on the path using some form of management such as SNMP control of the MIBs.

2.1.2 Tunnels and Label Stacks

A key feature of MPLS, especially when considering VPNs, is that once the labels required for an LSP have been exchanged between the LSRs that support the LSP, intermediate LSRs transited by the LSP do not need to examine the content of the data packets flowing on the LSP. For this reason, LSPs are often considered to form tunnels across all or part of the backbone MPLS network. A tunnel carries opaque data between the tunnel ingress and tunnel egress LSRs.

This means that the entire payload, including IP headers, may safely be encrypted without damaging the ability of the network to forward data.

In Fig.1, both LSPs are acting as tunnels. LSR B forwards the packets based only on the label attached to each packet. It does not inspect the contents of the packet or the encapsulated IP header.

An egress LSR may distribute labels for multiple FECs and set up multiple LSPs. Where these LSPs are parallel they can be routed, together, down a higher-level LSP tunnel between LSRs in the network. Labeled packets entering the higher-level LSP tunnel are given an additional label to see them through the network, and retain their first-level labels to distinguish them when they emerge from the higher-level tunnel. This process of placing multiple labels on a packet is known as label stacking and is shown in Fig.2.

Label stacks allow a finer granularity of traffic classification between tunnel ingress and egress nodes than is visible to the LSRs in the core of the network, which need only route data on the basis of the topmost label in the stack. This helps to reduce both the size of the forwarding tables that need to be maintained on the core LSRs and the complexity of managing data forwarding across the backbone.

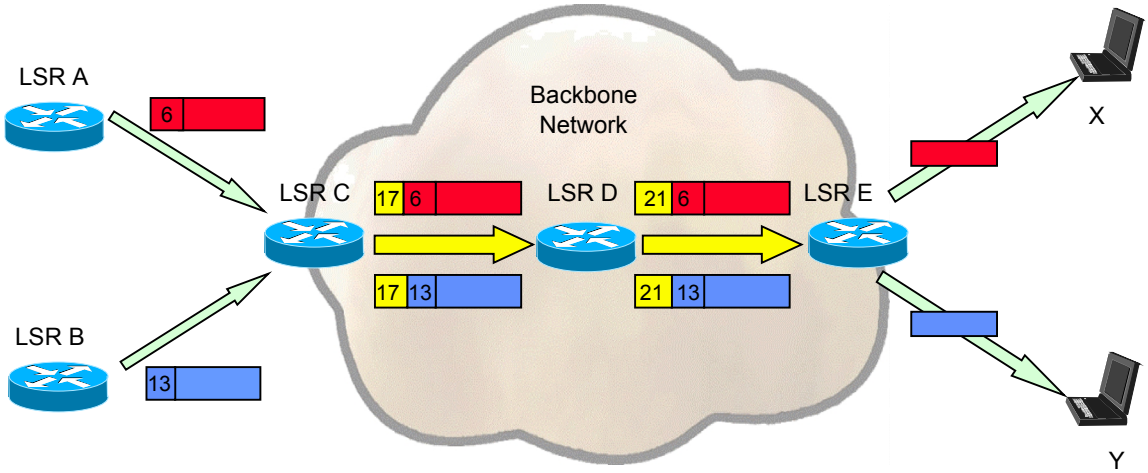

Fig.2: Label Stacks across the backbone

In Fig.2, two LSPs between LSR A and LSR E, and between LSR B and LSR E, shown as red and blue labels, are transparently tunneled across the backbone network in a single outer LSP between LSR C and LSR E.

At the ingress to the backbone network, LSR C routes both incoming LSPs down the LSP tunnel to LSR E, which is the egress from the backbone. To do this, it pushes an additional label onto the label stack of each packet (shown in yellow). LSRs within the backbone, such as LSR D, are aware only of the outer tunnel, shown by the yellow labels. Note that the inner labels are unchanged as LSRs C and D switch the traffic through the outer tunnel.

At the egress of the outer tunnel, the top label is popped off the stack and the traffic is switched according to the inner label. In the example shown, LSR E also acts as the egress for the inner LSPs, so it pops the inner label too and routes the traffic to the appropriate host. The egress of the inner LSPs could be disjoint from E in the same way that LSR A and LSR B are separate from LSR C. Equally, an LSR can act as the ingress for both levels of LSP.

A label stack is arranged with the label for the outer tunnel at the top and the label for the inner LSP at the bottom. On the wire (or fiber) the topmost label is transmitted first and is the only label used for routing the packet until it is popped from the stack and the next highest label becomes the top label.

The bottom label of a stack (the red and blue labels in Fig.2) is usually taken from a per platform label space (the Global Label Space) as this allows the outer tunnel to be re-routed when necessary. (Re-routing of an outer tunnel may result in that outer tunnel being received at its egress through a different physical interface from the one originally used when the inner tunnel was set up. This could lead to confusion about the interpretation of the lower label unless it is taken from a label space that is interpreted in the same way regardless of the incoming interface.)

For MPLS networks based on ATM equipment, it is attractive to consider using the VPI as the outer label and the VCI as the inner label. However, this places constraints on the number of outer and inner labels that may be too restrictive for an SP that needs to support many thousands of tunnels across the backbone. An alternative in such cases is to carry the inner label in a shim header below an outer VPI/VCI-based label. Although this method of label stacking in ATM means that the label stack cannot be fully implemented in standard ATM hardware, it does overcome other problems, not least of which is that some ATM hardware is incapable of performing VPI switching.

2.2 Overview of VPN Requirements

RFC 2764 [4] defines a generic framework for IP-based VPNs, including the following requirements for a VPN solution.

- Opaque transport of data between VPN sites, because the customer may be using non-IP protocols or locally administered IP addresses that are not unique across the SP network.
- Security of VPN data transport to avoid misdirection, modification, spoofing or snooping of the customer data.
- QoS guarantees to meet the business requirements of the customer in terms of bandwidth, availability and latency.

In addition, the management model for IP-based VPNs must be sufficiently flexible to allow either the customer or the SP to manage a VPN. In the case where an SP allows one or more customers to manage their own VPNs, the SP must ensure that the management tools provide security against the actions of one customer adversely affecting the level of service provided to other customers.

2.3 Summary of VPN Types

Four types of VPN are defined in RFC 2764.

- Virtual Leased Lines (VLL) provide connection-oriented point-to-point links between customer sites. The customer perceives each VLL as a dedicated private (physical) link, although it is, in fact, provided by an IP tunnel across the backbone network The IP tunneling protocol used over a VLL must be capable of carrying any protocol that the customer uses between the sites connected by that VLL.

- Virtual Private LAN Segments (VPLS) provide an emulated LAN between the VPLS sites. As with VLLs, a VPLS VPN requires use of IP tunnels that are transparent to the protocols carried on the emulated LAN. The LAN may be emulated using a mesh of tunnels between the customer sites or by mapping each VPLS to a separate multicast IP address.

- Virtual Private Routed Networks (VPRNs) emulate a dedicated IP-based routed network between the customer sites. Although a VPRN carries IP traffic, it must be treated as a separate routing domain from the underlying SP network, as the VPRN is likely to make use of non-unique customer-assigned IP addresses. Each customer network perceives itself as operating in isolation and disjoint from the Internet – it is, therefore, free to assign IP addresses in whatever manner it likes. These addresses must not be advertised outside the VPRN since they cannot be guaranteed to be unique more widely than the VPN itself.

- Virtual Private Dial Networks (VPDNs) allow customers to outsource to the SP the provisioning and management of dial-in access to their networks. Instead of each customer setting up their own access servers and using PPP sessions between a central location and remote users, the SP provides a shared, or very many shared access servers. PPP sessions for each VPDN are tunneled from the SP access server to an access point into each customer's network, known as the access concentrator.

The last of these VPN types is providing a specialized form of access to a customer network. The IETF has specified the Layer 2 Tunneling Protocol (L2TP), which is explicitly designed to provide the authentication and multiplexing capabilities required for extending PPP sessions from a customer's L2TP Access Concentrator (LAC) to the SP's L2TP Network Server (LNS).

3. MPLS FOR VPNS

MPLS is rapidly emerging as a core technology for next-generation networks, in particular optical networks. It also provides a flexible and elegant VPN solution based on the use of LSP tunnels to encapsulate VPN data. VPNs give considerable added value to the customer over and above a basic best effort IP service, so this represents a major revenue-generating opportunity for SPs.

The rest of this chapter gives an overview of the basic elements of an MPLS-based VPN solution and the applicability of MPLS to different VPN types. Subsequent chapters examine the trickier aspects of MPLS for VPNs in greater detail.

3.1 Elements of an MPLS VPN solution

Let us consider how MPLS can provide a VPN solution by examining how it would work at several different levels. We start with the data forwarding mechanics and work our way up to the network management considerations.

Different implementation models for MPLS-based VPNs imply different interactions between these elements of a VPN solution. See the section *VPN Implementation Models* for further details.

3.1.1 LSP Tunnels

The basis of any MPLS solution for VPNs is the use of LSP tunnels for forwarding data between SP edge routers that border on a given VPN. By labeling the VPN data as it enters such a tunnel, the LSR neatly segregates the VPN flows from the rest of the data flowing in the SP backbone. This segregation is key to enabling MPLS to support the following characteristics of a VPN tunneling scheme, as identified in RFC 2764.

- Multiple protocols on the VPN can be encapsulated by the tunnel ingress LSR since the data traversing an LSP tunnel is opaque to intermediate routers within the SP backbone.
- Multiplexing of traffic for different VPNs onto shared backbone links can be achieved by using separate LSP tunnels (and hence separate labels) for each data source.
- Authentication of the LSP tunnel endpoint is provided by the label distribution protocols. See the section *VPN Security* for more details
- QoS for the VPN data can be assured by reserving network resources for the LSP tunnels. MPLS supports both Intserv and Diffserv. The implications of using each of these reservation styles are examined in the next section.
- Protection switching and automatic re-routing of LSP tunnels ensure that failure of a link or router that affects a VPN can be corrected without management intervention. These protection mechanisms operate at several different levels, including refresh/keep-alive messages on a hop-by-hop basis within the label distribution protocols, re-routing of LSP tunnels, pre-provisioning of alternative routes, and wavelength failure detection and management for optical networks.

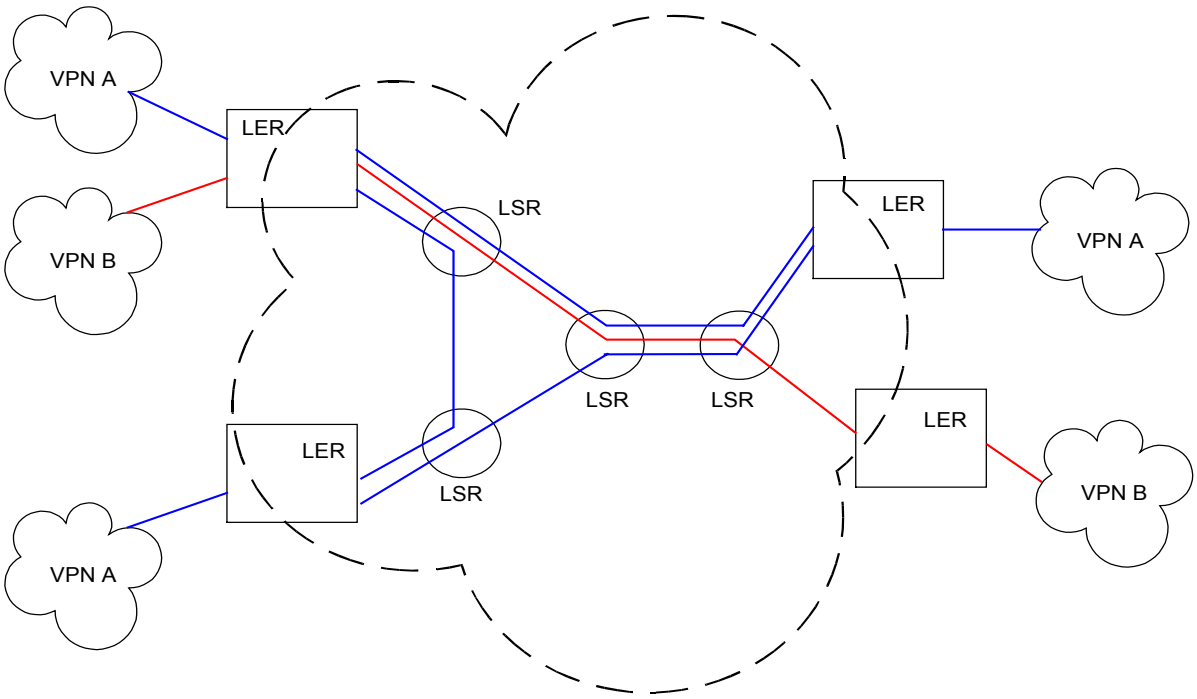

Fig.3: VPN Connectivity Using LSP Tunnels

Figure 3 shows simple interconnection between five VPN sites belonging to two different VPNs. A total of four LSPs are required in this topology, one to connect the two sites in VPN B, and three to connect the three sites in VPN A.

3.1.2 VPN Traffic Engineering

An LSP tunnel forms an excellent encapsulation scheme for VPN data flowing between two LSRs. But how do LSRs determine which LSPs to set up to provide connectivity for VPNs? In effect, how do LSRs decide which other LSRs provide access to the VPNs which they themselves serve? Even once this has been done, how should the different VPNs be mapped into LSP tunnels – a separate tunnel for each VPN, or a single tunnel for all VPNs?

These are complex questions that do not have a single "right" answer. There is a number of factors that determine what VPN Traffic Engineering (TE) scheme best suits the performance and scalability requirements of a particular customer and their SP.

- Identifying VPN peers

  This is the first problem facing an LSR that has been configured to support a VPN. The simplest scheme is to use explicit manual configuration of the VPN peers. This is the traditional solution providing obvious and deterministic control of resources and security, but it does not scale well as the size and complexity of the VPN increases.

  Alternative schemes automate the process of discovering VPN peers using a directory or by overlaying VPN membership information on one or more routing protocols used on the SP network. This greatly simplifies the configuration task for a VPN since it means that each SP edge router need only be configured with information about the VPNs serviced by each of its customer interfaces. There is clearly a potential security trade-off here as rogue routers can pretend to give access to a VPN.

  In comparison, an IPSEC-based solution [15] requires that each SP Edge router also be configured with security attributes for each peer in the VPN, which greatly increases the configuration complexity.

Several options for identifying VPN peers are examined in the *VPN Peer and Route Discovery* section below. See *MPLS VPN Security* for further discussion of IPSEC.

- Multiplexing VPNs on an LSP

  Although LSRs in the core of the SP network do not have to examine the data flowing on VPN LSP tunnels, they are still aware of the existence of these tunnels. This can represent a scalability problem if a separate mesh of LSP tunnels is used for each VPN, because the core LSRs must at least maintain a forwarding table entry and associated resource reservation for each tunnel.

  If the SP supports thousands of VPN customers, the core LSRs could be required to maintain millions of LSPs. This is the same problem faced by VPN solutions based on ATM or Frame relay technology. Depending on the network topology, this large number of labels may also be beyond the capacity of the LSR switching hardware.

An alternative approach is to multiplex the traffic from multiple VPNs that share the same ingress and egress SP edge routers within a single LSP tunnel between those LSRs. This is achieved using label stacks, with a single outer tunnel set up across the core and an inner LSP that identifies the VPN for which the data is bound. The lower label in the stack is known only to the ingress and egress LSRs.

This use of label stacks reduces the number of LSP tunnels exposed to the network core, but it ties VPNs together. The multiplexed VPNs cannot be routed separately or given different prioritization or drop priority by the core LSRs. The VPNs must also share a single network resource reservation within the network core, which may make it harder for the SP to guarantee the SLA for each individual customer.

In figure 4, two VPNs are connected across the MPLS network between a pair of LERs. The traffic for each VPN is carried on a distinct LSP shown as a red and a green line in the diagram. These two VPNs are nested within an outer LSP shown in blue.

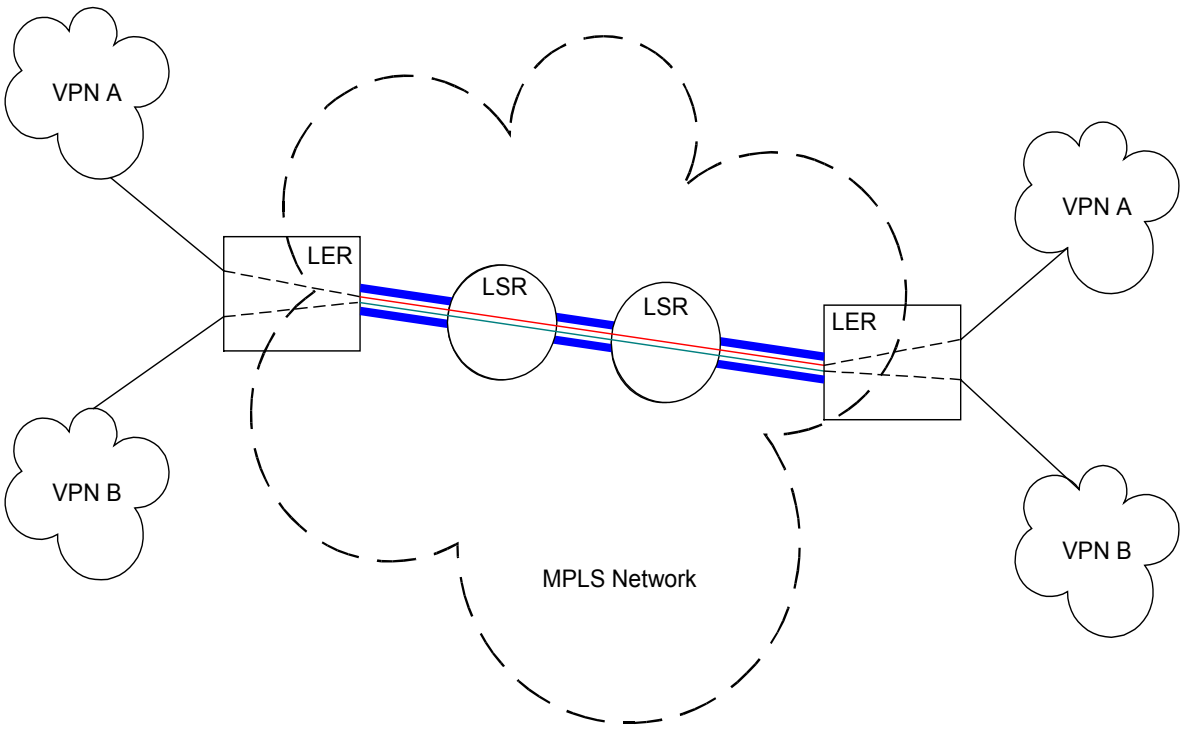

Fig.4: Nested LSPs Providing VPN Connectivity

- Separating QoS classes

  Multiplexing VPNs within a single tunnel helps to reduce the signaling load and forwarding table size in the core LSRs as the number and size of the VPNs increase.

However, once the data for multiple streams has been clustered together in a single LSP, it is hard to provide distinct management of the different flows. The encoding of an MPLS label allows three bits to encode the Differentiated Services Control Point (DSCP). Thus a total of eight classes of service (CoS) can be set for packets within any one LSP. These bits can define queuing rules and drop priorities for packets carried on the LSP.

In the case of an ATM-based network there is just one bit available to encode the DSCP and this is usually used simply to indicate the drop preference.

If a customer or SP needs to be able to differentiate more than eight DSCPs across the core, multiple outer LSP tunnels must be set up. Each outer tunnel carries a different CoS range and can be routed separately across the core.

The interaction between setting up multiple outer tunnels across the core to carry more CoSs, and the need to minimize the number of such tunnels using VPN multiplexing on a single tunnel is examined in more detail in the *VPN Multiplexing and Class of Service* section below.

The IETF draft draft-ietf-mpls-diff-ext [12] defines methods of signaling LSPs for CoS usage and ways of determining the interpretation of the DSCP bits.

- TE across the backbone

  MPLS TE can be used to distribute the load within a network and to guarantee bandwidth and QoS by controlling the routing of the outer VPN LSP tunnels across the SP backbone network. This is essentially the same problem as TE for non-VPN traffic, and hence is outside the scope of this paper. For details of the MPLS TE protocols, refer to the white paper *MPLS Traffic Engineering: A choice of Signaling Protocols* [1] from Data Connection.

3.1.3 Network Management

The management of a VPN falls into two categories.

- Defining the logical topology of the VPN.
- Mapping the VPN onto the SP's physical network, including routing the VPN data across the core network.

The second of these functions is always the preserve of the SP. The first feature may, however, be managed by the SP or by the customer on a self-serve basis.

3.2 Applicability of MPLS to VPN Types

MPLS LSP tunnels can be used to provide all or part of an implementation of any of the four types of VPN. The suitability of an MPLS solution to each VPN type is described below, including the scalability and management challenges such solutions present.

3.2.1 MPLS for VLL

Conceptually, this is the easiest application of MPLS to VPNs. Each point-to-point VLL is provisioned as an LSP tunnel between the appropriate customer sites. The customer is explicitly looking for the equivalent of leased lines, so it is very important that the SP meets any bandwidth guarantees in the SLA. This means that the LSP tunnels used in a VLL solution may have to be dedicated to that specific customer rather than multiplexing the VLL traffic with other VPNs. It is also possible to subdivide the resources of an outer tunnel to provide the QoS for inner LSPs.

The point-to-point connectivity of a VLL means that each VLL is most easily provisioned at the edge LSRs by manual configuration rather than an automatic scheme for detecting the VLL peers.

3.2.2 MPLS for VPLS

The most immediately obvious means of implementing a VPLS is to map the LAN segment to a unique IP multicast address perhaps using IP encapsulation of the VPN IP traffic. Such a solution could use existing IP multicast technologies, rather than MPLS. Indeed, such approaches are offered by many ISPs today.

However, technologies such as MOSPF and (non-labels) RSVP do not provide the full TE capabilities of MPLS, so the SP has less control over how the VPLS traffic is routed across the backbone network.

Very large SPs with many VPLS customers may also eventually find that there are too few administratively scoped IPv4 multicast addresses to represent each of the VPN LAN segments that they need to support, forcing them either to move to IPv6 or to multiplex several VPLSs on one multicast address. There are 2^{24} administratively scoped IP multicast addresses (239./8), but an SP may well wish to reserve only a portion of this address space for VPN services.

Current MPLS label distribution protocols are specified for unicast destination IP addresses only. This means that an MPLS-based implementation of a VPLS is, necessarily for now, based on one of the following network topologies.

- A full mesh of LSP tunnels connecting the customer sites, with each SP edge LSR responsible for the fan-out to all peers.

- Point-to-point or multipoint-to-point LSP tunnel connections to a “hub” LSR that handles fan-out to all sites using point-to-point LSP tunnels.

In both cases, but especially for the mesh of LSP tunnels, the MPLS-based topology may use more network bandwidth in total that the IP-multicast based solution. This is because multiple copies of each packet may be sent across any given link, each copy carried within one of several different LSP tunnels for the VPLS that transit that link. However, SPs may still choose to implement a VPLS using MPLS in order to exploit the TE capabilities of MPLS to give them better control of how the VPLS traffic is routed between SP edge LSRs.

Future standardization work on MPLS may extend the TE capabilities to cover point-to-multipoint or multipoint-to-multipoint LSP tunnels. Such an extension would allow MPLS-based implementations of a VPLS to avoid the bandwidth overhead compared to an IP-multicast based implementation.

3.2.3 MPLS for VPRN

LSP tunnels provide an excellent solution to VPRNs. A VPRN is routed, rather than requiring point-to-multipoint connectivity. This means that even if the SP edge routers set up a full mesh of LSP tunnels to all the other SP edge routers for a given VPRN, they can route each packet onto a single LSP tunnel according to the destination address for that packet rather than fanning out copies to all peers for that VPRN. This avoids the bandwidth wastage that can occur when using an MPLS-based VPLS, as described in the previous section.

Note that the routing protocols used on a VPRN are independent of the routing protocols used on the SP backbone. It is perfectly possible for an SP to use OSPF and BGP4 but for a VPN customer to use a much simpler protocol such as RIP.

3.2.4 MPLS for VPDN

MPLS could be used as the underlying transport mechanism between the LAC and LNS in an L2TP-based VPDN. This is no different from using MPLS to transport any other data that uses public IP addresses. The essential function of a VPDN is provided by L2TP. For this reason, no further consideration is given in this paper to the use of MPLS for VPDNs.

See the *References* section for sources of information about L2TP.

4. VPN PEER AND ROUTE DISCOVERY

Every edge LSR that participates in an MPLS-based VPN implementation needs some means of identifying the peer SP edge LSRs that have connectivity to the same VPN. Once it has this information, the LSR can set up the LSP tunnels that are needed to transport the VPN data across the SP backbone network. In the case of a VPRN, the LSR also needs to learn the VPN-specific routes accessible through each peer.

The main options for how each LSR obtains the list of VPN peer sites and routes are summarized in Table 1, including the VPN types each option is best suited to. The sections that follow described each option in more detail.

Discovery Method	Implementation	Scalability	Comments	Optimal VPN types
Manual configuration	Use TE MIB to configure LSP tunnels for VPN.	Geometric	Works well for small VPNs only.	VLL
Emulated LAN	Map each VPN to IP multicast address. Hello protocol & ARP used to discover peers.	Linear	Requires use of IP multicast. Scalability ultimately limited by available IP multicast addresses. Use a routing protocol between peers to distribute VPN routes.	VPLS VPRN
BGP or OSPF	Overlay VPN IDs and routes in IGP or EGP attributes.	Linear	Requires change to routing protocols and implementations EGP-based solutions allow inter-AS VPNs.	VPRN
Directory	Read VPN peers and routes from directory.	Linear	Transparent to existing routing or MPLS protocols and implementations. Other VPN attributes can also be held in the directory.	VLL VPLS VPRN

Table 1 VPN Peer and Route Discovery Methods

Some VPN topologies require that certain VPN routes are only accessible to a subset of the customer sites for that VPN. This can, obviously, be achieved by manual configuration of the exact topology at each VPN site, but that scales poorly. Alternative automatic methods for achieving this level of control are described in *VPRN Route Configuration.*

4.1 Manual Configuration

Each SP edge LSR that supports a given VPN could be manually configured, via the TE MIB [8], to set up an LSP to the peer LSRs in order to carry the VPN traffic. This works fine if a customer requires only a few VLLs or a very simple topology such as a star-shaped network, but it does not scale well for more complex VPN topologies. The automated means of detecting VPN peers described in the sections that follow provide better solutions for larger VPNs.

This solution is still very attractive to ISPs who want to maintain strict control of all LSPs and traffic in the system. It also provides an extra level of security control.

4.2 Emulated LAN

An emulated LAN (ELAN) can be set up between peer edge LSRs for a given VPN by mapping the VPN ID to an IP multicast address. A Hello protocol can then be used to discover peer routers on the ELAN. Addresses used on the ELAN may be taken from a VPN-specific address range, in which case ARP can be used on the ELAN to discover the public IP addresses of peer SP edge router. Once the peer LSRs have been discovered in this manner, a mesh of LSP tunnels can be set up between the SP edge routers.

For a VPLS, a mesh of LSP tunnels provides sufficient routing information. However, if the VPN is a VPRN, a routing protocol, such as OSPF, should be used in place of a separate Hello protocol. Routing adjacencies can then be set up between the VPN peers for the exchange of VPN routes.

The use of an emulated LAN is described in more detail in an Internet draft on the subject [5]. Although that draft focuses on the Virtual Router (VR)-based implementation of a VPRN, this method for discovering peer VPN LSRs can be used independent of VRs.

Using an IP multicast address for an emulated LAN is superficially similar to a non-MPLS-based implementation of a VPLS. However, in this case the emulated LAN is used solely to provide an automated method of detecting peer SP edge routers for a VPN, rather than to carry the VPN data. This technique can be deployed for any type of VPN, not just VPLSs, and scales well as the size of any given VPN increases.

The number of available IP multicast addresses may limit the ability of an SP to apply this technique to very large numbers of VPNs. Multiple VPNs could be hashed to each IP multicast address, but that reduces efficiency and would require the addition of a VPN ID to the Hello protocol used on the ELAN.

4.3 Overlay IGP or EGP

VPN membership and routing information can be overlaid onto one of the IGP or EGP routing protocols already deployed in a SPs network. This approach allows the VPN data to piggyback the reliable distribution mechanisms built into such protocols to ensure that this membership information is available to all peer VPN LSRs. The routing protocol can also be used to distribute inner labels for each VPN, if required (see *Distributing Label Stacks*).

This style of VPN peer and route discovery is best suited to a VPRN because it focuses on distributing VPN routes rather than just membership information for a VPLS.

Using any modern routing protocol as a base (such as BGP4 or OSPF) gives good scalability provided that the number of VPN routes carried in the routing protocol remains small relative to the base routing data. If the overlaid VPN data becomes very large, for example because the number of VPN ports per SP edge router is very large, the size of the overlaid data may affect the scalability or the speed of convergence of the underlying routing protocol.

RFC 2547 [2] defines a BGP overlay implementation using a new class of VPN-specific route, which conveys accessibility information and labels for each route. This RFC uses a non-standard form of VPN ID (i.e. not RFC 2685 [3]) in combination with route target discriminators to allow control of the accessibility of routes to each VPN site. See *VPRN Route Configuration* for more details.

If VPN data is overlaid in an EGP such as BGP, it is possible to set up inter-AS VPNs for a customer who uses more than one SP. There are significant security implications for both the customer and the SP choosing to support inter-AS VPNs, which we do not have space in this paper to examine in detail. However, the fine degree of control provided by most BGP4 implementations on the import and export of routing data provides a better basis for resolving such security concerns than the use an emulated LAN based on IP multicast addresses.

One disadvantage of the overlay approach is that it requires that each routing protocol to be modified to carry the VPN data. Code changes are also required for existing implementations, which risk destabilizing the complex routing code.

4.4 Directory

Probably the most flexible method for discovering VPN peer sites is to read this information from an X.500 or LDAP directory. Each SP edge LSR is configured with just the VPN IDs of the VPNs to which it belongs. The LSR can then read the topology of all VPN LSP tunnels it is required to set up directly from the directory, together with the routes associated with each tunnel. A directory-based VPN solution can be used for any type of VPN.

The directory can also be used to hold other VPN-based data that needs to be synchronized across all SP LSRs that support the VPN, such as the mapping of DSCPs from the VPN into the DSCPs used across the network core, PKI-based encryption of sensitive data, or the setup and holding priority to be used for each VPN tunnel. Without the use of a directory, all this information would have to be configured at each SP edge LSR.

A VPN directory schema has yet to be defined and standardized. However, once this is in place, directory-based VPN peer identification will provide an excellent method for discovering VPN peer sites. This technique can be implemented without any modification to routing protocols.

In many implementations, the speed of update to a directory is slow. This means that a directory is well suited to storing information that changes infrequently, such as the set of VPN customers served by each SP edge router. In contrast, a directory is ill-suited to holding dynamically changing information, such as the precise routing of an LSP tunnel across the network. If inner labels are to be stored in a directory, they must be essentially static.

4.5 VPRN Route Configuration

Some VPRN topologies require the ability to control the VPN routes that are accessible from each customer site. For example, a customer may require that all of their satellite offices connect to a central location which routes between the sites, rather than allowing direct connectivity between the satellite offices. If all of this customer's satellite offices were allowed to learn the routes advertised by every other site, this topology would be impossible to set up.

Manual configuration or use of a central directory allows explicit control over the routes known to each VPN site. However, if an ELAN or a routing protocol overlay is used, one of the following techniques must be used.

- Each subset of the VPN in which all routes should be made accessible to all sites can be assigned a separate VPN ID. In the example of the single central office, however, this requires a separate VPN ID per satellite office, which is very wasteful of VPN IDs. If a separate VR is instantiated for each VPN ID, this is also very profligate in its use of router resources. It can still be useful to split a single customer's network into several logical VPNs (e.g. separate manufacturing and accounts networks), but this technique should not be used for fine-grained control of route availability.

- If the VPN is implemented using a routing protocol overlay, such as in RFC 2547, each VPN-specific route can be assigned a set of route target discriminator fields. Each VPN site is also configured with the route target discriminators that it can import, thus giving control over the VPN-specific routes visible to each customer site.

- If the VPN is based on VRs, the full set of configuration tools available for the routing protocol used by the VRs can be utilized to control the routes exported and imported by each VR. This technique has the advantage of using tools that may already be familiar to the SP or customer network administrators. See *MPLS VPN Implementation Models* for more details on VRs.

4.6 Management Information Bases

A new effort in the MPLS Working Group of the IETF is considering how to configure VPN edge points for MPLS/BGP VPNs. A new draft [16] defines the basic building blocks and will be tied into the MPLS model more closely at a later date.

5. VPN MULTIPLEXING AND CLASS OF SERVICE

A key decision for an SP providing a MPLS-based VPN service is how to balance the need to limit the number of LSP tunnels that cross the network core with the desire to offer SLAs specifically tailored to each customer's needs. It is easier to monitor and enforce the SLAs for each customer if separate LSP tunnels are used for each VPN, but this can become a problem in terms of both the resources needed on the core routers to track these tunnels and the effort needed to manage so many tunnels.

Label stacking allows multiple VPNs to be multiplexed into a single LSP tunnel, but this is purely a technical solution that needs to be backed up with a policy decision by the SP on how to perform the multiplexing. This policy decision breaks down into two parts: what classes of service (CoS) the SP wishes to offer, and how to multiplex VPNs and CoSs into LSP tunnels across the network core.

5.1 Class Of Service Options

Many VPN customers want to receive guaranteed minimum bandwidth on their VPN connections, but it would be inefficient and costly, for both the SP and their customers, to provision fixed bandwidth LSP tunnels that could support the maximum bandwidth needed between all VPN sites. A far better option is to provision the networks with spare capacity over and above the minimum bandwidth requirements and to share the spare capacity in the network between the VPN customers and public Internet traffic. The sharing of spare bandwidth could be on an unequal basis according to the CoS (Gold, Silver, Bronze etc.) that a customer has signed up for.

MPLS supports this style of provisioning. The sharing of spare bandwidth is similar to Diffserv, but this paper deliberately uses the CoS terminology to distinguish the service options available in the network core from the DSCPs used within any one VPN or the public Internet. In fact the Diffserv extensions to MPLS [12] can be used to signal the CoS that a tunnel carries as DSCPs within the SP network, but the interpretation of these DSCPs may well be different to that in the VPNs.

The ingress LSR is responsible for mapping the combination of VPN and DSCP (or equivalent for non-IP customer networks) to the LSP tunnels and CoS used to transport this data across the network core. The original DSCP is encapsulated and carried across the core network to the egress LSR, so the mapping to a different set of CoS in the core is transparent to the customer networks. This process is shown in Figure 5.

The CoS range used within the network core is an administrative decision for each SP to make according to the services they wish to be able to offer to their customers. Correctly, this is not standardized for all SPs. If a customer uses multiple SPs, the gateway nodes between the two SP networks must map the CoS ranges used by an inter-AS LSP tunnel into the ranges used by each SP.

Note that some customers may still demand fixed bandwidth connections between their VPN sites, equivalent to ATM or Frame Relay VCs – especially for VLL VPNs. MPLS can support such connections in parallel with variable bandwidth CoS-based services. See the next section for how this affects the multiplexing of VPNs.

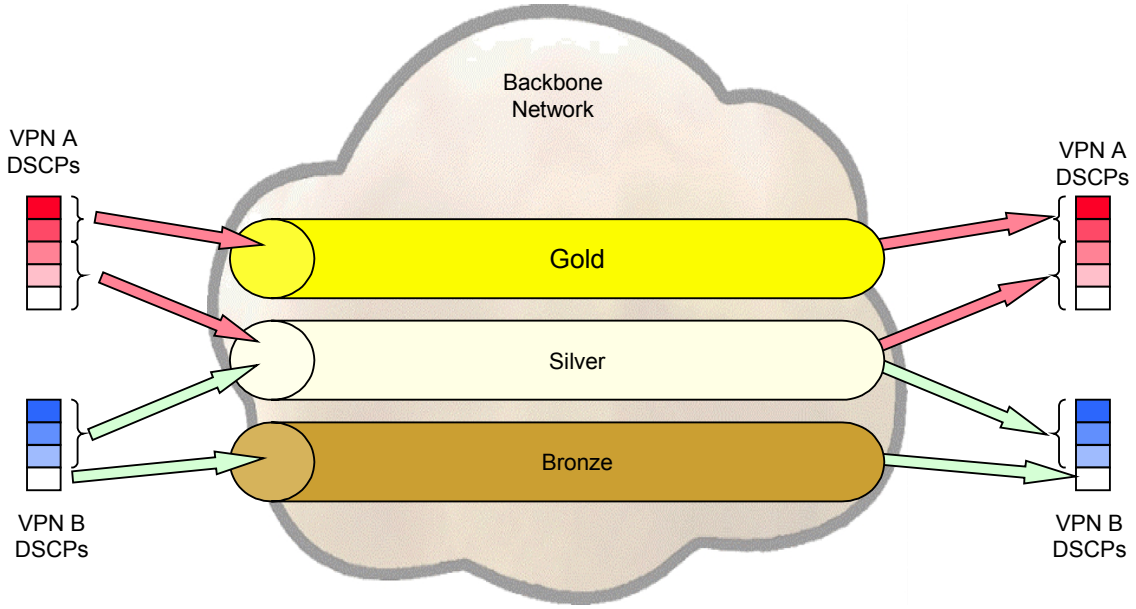

Fig.5: Separate CoS Mappings per VPN Customer

5.2 Multiplexing VPN and CoS

VPNs and CoS can be multiplexed into a single outer tunnel across the network core in a number of different ways. Table 2 summarizes the various possibilities. The following terms and conventions apply to this table.

- Items in *(brackets)* are optional.
- ***Peer*** represents a remote SP edge LSR that services one or more VPNs.
- ***VPN*** could be a group of customer IDs or a combination of customer and destination, as well as just a single VPN ID.
- ***CoS*** represents the service differentiation levels within the SP backbone. This is distinct from the DSCP (or equivalent) used on the customer network.

Outer Tunnel	Inner Tunnel	Comments	Max CoS in core[1]	VPN routing
Peer	VPN(+CoS)	Minimizes number of tunnels across the core. CoS can be combined with VPN, or a 3-level stack with VPN and CoS in either order.	8	Merged
Peer+VPN	(CoS)	Best solution for provisioning private VPN-specific tunnels.	8	Separate
Peer+CoS	VPN	Good solution if more CoS levels needed across core.	64	Merged
Peer+VPN +CoS	none	Maximum number of tunnels across the core.	64	Separate

Table 2 VPN and CoS Multiplexing

Notes

1. Assumes outer tunnel is provisioned using Diffserv extensions to MPLS and the shim header. If ATM is used instead of the shim header, replace all instances of "8" in this column with "1".

From Table 2, the following options for multiplexing VPNs and CoS emerge as being the most appropriate for different customer and SP situations.

- If a customer requires a VLL solution, or routing on specific private links, the outer tunnels must be provisioned per {peer+VPN}.
- If a SP requires more levels of CoS differentiation across the core than can be provided in a single LSP tunnel, the outer tunnels should be provisioned per {peer+CoS}, with per VPN inner tunnels.
- Otherwise the outer tunnels should be provisioned per peer, with inner tunnels per VPN and, optionally, per CoS.

In all these cases, further levels of inner tunnel could also be used to distinguish VPN-specific destinations on the remote LSR. For example, the remote LSR may provide access to more than one segment of the customer network. These segments could be treated as separate VPN IDs, thus incorporating the destination into the levels of the label stack covered in Table 2, or as a single VPN with a further level of label distinguishing the destination segment.

6. DISTRIBUTING LABELS FOR NESTED LSPS

MPLS-based VPN implementations require the use of label stacks for multiplexing of VPNs and/or CoS onto a single LSP tunnel in order to reduce the number of LSP tunnels that transit the network core. There are a number of different protocols and techniques that could be used to distribute or signal the labels for these stacks.

Labels for outer tunnels can be explicitly setup using one of the TE protocols such as RSVP-TE or CR-LDP. This allows bandwidth to be reserved for the outer tunnel. Alternatively, CoS-based services may be supplied to customers who do not require explicit bandwidth reservation using a simple egress-targeted label for the destination SP edge router. However, use of egress-targeted labels implies that the VPN traffic follows the same path through the SP network as any public IP traffic that is forwarded using these labels and shares the same CoS-based resources.

The suitability of a label distribution method for distributing the inner labels in VPN label stacks is determined by the following characteristics.

- Use of non-unique VPN IDs must be transparent to the network core otherwise it will be impossible to distinguish routes for different VPNs.
- Use of private IP addresses within VPNs must also be transparent to the network core since the address spaces may overlap.
- Unsolicited allocation of labels (downstream unsolicited) to all VPN peers (for use on inner LSPs) avoids the need for an explicit signaling exchange (downstream solicited) for each inner label used between every pair of VPN peer sites.

Table 3 compares several possible approaches to distributing the label stacks between peer LSRs, showing the characteristics and suitability of each protocol for signaling outer or inner tunnels. The following conclusions can be drawn from this table.

- Egress-targeted outer labels for remote SP edge routers can distributed using LDP, provided that bandwidth reservations are not required for these labels.
- Alternatively, outer LSP tunnels should be signaled using RSVP or CR-LDP TE protocols if bandwidth reservation is required.
- Piggybacking the distribution of inner labels on one of the routing protocols used by the SP is likely to be the best method for distributing VPN label stacks in most situations
- A directory could also be used to store inner labels, but would not be suitable for use with mobile VPN connectivity to multiple SP edge LSRs.

Protocol	Comments	Outer tunnels	Inner Tunnels	On-demand (O) / unsolicited (U)
Routing Protocol[1]	Dynamically allocated per VPN and/or CoS inner labels can be carried in routing data.	No[2,3]	Yes	U
LDP	Can be used to distribute egress-targeted labels for each SP edge router	Yes[3]	No[4]	O/U
RSVP CR-LDP	Extensions[5] currently being defined to allow VPN and CoS to be signaled for LSP tunnels. TE MIB is also being extended to allow an explicit route hop to specify that a new LSP tunnel is stacked within an existing tunnel.	Yes	Yes	O
Directory	Static per VPN or CoS inner labels can be read from a directory.	No[6]	Yes	U
Network Management SNMP	Static LSPs can be configured at each LSR across the network by writing to the LSR MIB.			

Table 3 Label Stack Distribution Methods

Notes

1. RFC 2547 defines a BGP4-based implementation that distributes per VPN inner tunnel labels.
2. Distance vector routing protocols could easily carry outer labels, but there would be a consequent increase in the size of the routing data. Link state protocols cannot easily be enhanced to carry labels. However, use of distance vector protocols as the SP IGP is rare.
3. Does not allow resource reservation, which may be required if the VPN customer wants a minimum bandwidth guarantee.
4. VPN-specific addressing would not be transparent to the network core if distributed using LDP. However, LDP could be used to distribute FEC-based labels within a VPN, which would be encapsulated within the label stacks covered by this table.
5. See [12].
6. No routing participation, so cannot set up labels across the network core.

7. MPLS VPN SECURITY

Customers expect VPN data to remain private, including the topology and addressing scheme for their network as well as the data carried on the VPN. Historically, VPN implementations based on ATM or Frame Relay VCs have provided this security by virtue of the connection-oriented nature of the physical network. However, the connectionless public IP network cannot provide the same protection, and IP VPNs have relied on cryptographic means to provide security and authentication.

MPLS brings to IP security benefits similar to layer 2 VCs. This means that the customer equipment connected to the VPN does not need to run IPSEC or other cryptographic software, representing a considerable saving for the customer in terms of equipment expense and management complexity.

MPLS VPN security is achieved as described below.

- At the ingress SP edge router, all data for a VPN is assigned a label stack that is unique to the VPN destination. This ensures that the data is delivered only to that destination, so data does not leak out of the VPN.

- Any other packet entering the SP network is either routed without the use of MPLS or is assigned a different label stack, so a malicious third-party cannot insert data into the VPN from outside the SP network.

- SP routers can use the Cryptographic Algorithm MD5, or similar techniques, to protect against insertion of fake labels or LSRs into the label distribution protocols.

There are two situations when a customer may still require the use of cryptographic security measures even when using an MPLS VPN solution.

- If the customer data is considered sufficiently sensitive that it must be protected against snooping even from within the SP network, IPSEC or similar cryptographic techniques must be applied to the VPN data before it enters the SP network. In this case the customer retains responsibility for distributing the cryptographic keys.

- When a VPN is served by more than one SP, the SPs may choose to use IPSEC-based tunnels to carry the VPN traffic between their networks on the public IP network if a direct MPLS connection between the SPs is not available. In this case, the SPs are responsible for distributing cryptographic keys.

8. VPN IMPLEMENTATION MODELS

Two main implementation models for VPNs have been proposed.

- Two Internet drafts [5], [6] suggest implementing the VPN as a set Virtual Routers (VRs) which each correspond to a separate SP network IP address within a physical SP edge router. LSP tunnels between the VPN sites are seen by the VRs as virtual interfaces.

- RFC 2547 [2], [7] uses VPN-specific routing and forwarding (VRF) tables within a single router implementation.

In terms of data forwarding function and the VPN types and topologies they can support, both models are identical. They can both be used with any of the VPN peer discovery, VPN multiplexing, and label distribution solutions described in earlier sections of this paper. Both schemes can also be implemented in such a way that VPN membership information need only be configured once per customer interface to an SP edge router (though this may be configured to either a VR or the physical router instance). Both schemes can utilize a hardware data plane to give similar data forwarding performance.

The conceptual differences between these implementation models only show up when considering the management of the VPN and the SP network, in particular how this information may be presented to the SP and the customer, the router and network resources used by each model, and the implementation effort.

- VRs provide an easy means to separate out the management domain for a given customer from the rest of the SP network. This separation can be exploited to give the customer self-service management more easily than providing secure access for multiple customers to a single routing protocol stack running in the physical SP router.

- VRs allow the SP or customer to utilize the full set of existing management tools for the routing protocols running in the VRs to manage the VPN topology. These tools may already be familiar to the SP or customer network administrator.

- Similarly, VRs allow the SP or the customer to monitor more easily the network state or performance as seen by a given VPN. For example, VRs maintain traffic statistics for each virtual interface presented to the VR. In contrast, a VPN-specific table implementation may not maintain this data, though it could do so if the data plane keeps traffic counts by label stack.

- Set against these points, VRs consume additional SP router resources and network bandwidth. Although both models must distribute and maintain VPN-specific routing data, there is always some overhead incurred for each additional copy of the routing protocols running in a router or over a link. This overhead will be relatively larger for small VPNs, in the worst case requiring two VRs for a simple default route between two sites.

- A VRF implementation that is closely coupled with VPN route and label distribution, such as the way RFC 2547 uses BGP, concentrates all portions of the router control plane that are aware of VPNs into one protocol stack. This may allow a more compact implementation and shorter development timescale, but at the expense of tying the VPN solution to one underlying routing technology.

These differences mean that the choice of implementation model is determined by the VPN topologies that a SP needs to support and the resource constraints in the SP network.

- The RFC 2547 solution is very efficient in terms of bandwidth and router resources. It is a good choice for VPNs with relatively simple topologies, especially if the CoS requirements allow use of LDP to distribute egress-targeted outer labels.

- A Virtual Router (VR) implementation requires more bandwidth and router resources, but gives finer-grained control over the routing topology. It is suitable for SPs that need to support very complex VPN topologies, or for customers who desire self-service management.

Over time, VPN topologies are likely to become ever more complex. Since the management complexity of a VPN solution is the largest factor in determining the recurring costs incurred by an SP, the VR implementation model is likely to dominate in the long term.

9. STANDARDIZATION EFFORTS

9.1 Work to Date

Several RFCs relating to MPLS-based VPNs have been issued by the IETF.

- RFC 2764 is an informational RFC that defines a set of standard terminology for IP-based VPNs and a framework for describing VPN solutions. It sets requirements for VPN implementations, but does not define any specific MPLS implementation.

- RFC 2685 defines a standard 7-byte format for VPN ID agreed between the IETF and ATM Forum. This format is based on the 3-byte OUI of either the VPN customer or the SP that serves them plus a 4-byte index assigned by the OUI owner.

- RFC 2547 is an informational RFC that describes Cisco's MPLS VPN solution based on use of VPN-specific routing tables and an overlay on BGP for VPN route and label distribution. RFC 2547 does not use the RFC 2685 VPN ID format, but defines an alternative 8-byte format using an AS number or an IP address in place of the OUI.

In addition to these RFC, there have been many Internet drafts issued relating to MPLS-based VPNs. None of these has progressed to being standards track documents at the time of writing. See the *References* section for a listing of the relevant drafts, but remember that these are all work in progress and may never progress to being RFCs.

9.2 Outstanding Items

Some further effort is required by the MPLS working group, and other IETF working groups, to define the standards needed for interoperable MPLS-based VPN solutions. The main outstanding items that need to be resolved are listed below. At the time of writing, some of these efforts are already underway, but none has yet reached the standards track.

9.2.1 VPN ID

The VPN ID format defined in RFC 2685 guarantees uniqueness of VPN IDs, but it requires global knowledge of the VPNs served (and numbered) by an SP to assign these IDs correctly. This may be difficult if the SP network consists of a number of ASs that are administered separately, for example by separate organizations inherited from acquisitions. In this case the AS-assigned IDs from RFC 2547 may be more convenient.

These VPN ID formats should either be merged to give a single format, say by modifying the RFC 2547 format to use a single-byte type field and assigning a new OUI-based type equivalent to RFC 2685, or a decision taken to use just one of these formats in all future standardization work (for example when specifying MPLS TE extensions for VPNs).

9.2.2 Overall Approach to MPLS VPNs

Many different options for implementing MPLS-based VPNs have been discussed in this paper, but some make rather better sense than others. Interoperability between router vendors will be achieved more quickly is the spectrum of implementation possibilities is restricted to a few options, such as:

- Outer labels should be per peer and, optionally, CoS. Inner labels should be per VPN.
- VPN route and label distribution using BGP or OSPF overlays.
- A directory-based solution for VPN peer and route determination.
- TE MIB and RSVP/CR-LDP extensions to allow setup of label stacks.

9.2.3 Routing Protocol Overlays

VPN route and label distribution overlays for the routing protocols need to be defined to be consistent with the chosen format of VPN ID and the overall MPLS VPN approach. RFC 2547 does this for BGP, subject to the VPN ID and VPN/CoS multiplexing model chosen.

9.2.4 Directory Schema

A standard directory schema should be defined for the definition of VPN membership and routing information. This would allow interoperability between routers from multiple vendors with a single directory.

9.2.5 MPLS TE Extensions

Extensions to RSVP and CR-LDP are required to allow these protocols to be used for signaling outer and inner tunnels for MPLS VPNs:

- A new TLV is needed to carry the VPN ID for a tunnel between the ingress and egress points for the tunnel.

- The TE MIB explicit route hop objects should be extended to allow the specification of the outer tunnel through which a new inner tunnel should be routed.

9.2.6 Management Information Bases

A new effort in the MPLS Working Group of the IETF is considering how to configure VPN edge points for MPLS/BGP VPNs. A new draft [16] defines the basic building blocks and will be tied into the MPLS model more closely at a later date.

9.2.7 Other Items

Work may also need to be considered for LDP extensions for VPN-based FECs and CoS attributes for FECs depending on the overall approach taken for MPLS VPNs.

10. SUMMARY

MPLS provides a step-change improvement in the scalability and ease of provisioning of VPNs over IP networks. It also offers enhanced CoS support to allow SPs to offer differentiated service levels. By leveraging these MPLS facilities, SPs can offer highly cost-effective and competitive VPN solutions to their customers and maximize bandwidth usage across the core network.

LSP tunnels provide the encapsulation mechanism for VPN traffic. Automatic methods for determining VPN routes allow the configuration complexity of an MPLS VPN to scale linearly (order(n)) with the number of sites in the VPN, as opposed to geometric (order(n^2)) scaling for other IP-tunneling VPN solutions. Best scalability of peer discovery is achieved by overlaying the VPN peer and route discovery using a routing protocol or by use of a directory.

VPN traffic can be multiplexed onto common outer LSP tunnels in order that the number of tunnels scales according to the number of SP edge routers rather than the much larger number of VPN sites serviced by these routers. This avoids the scalability problems seen in some ATM or Frame Relay VPN solutions by reducing the problem to order(m) where m is the number of LSRs providing access to n VPN sites, and $m \leq n$.

Outer LSP tunnels can also be provisioned for different CoS ranges, allowing SPs to customize the way VPN traffic is treated in the network core to match the service levels they wish to make available to customers. This can be combined with bandwidth reservations for certain CoS ranges or particular dedicated LSP tunnels for a specific customer if required by their SLA.

In the short-term, RFC 2547 provides an efficient VPN implementation model. Longer-term, a Virtual Router (VR) based implementation is likely to provide easier management of very complex VPN topologies. In the interests of having a single implementation and management model, SPs may also come to use VRs for smaller VPNs despite its lack of efficiency in that case.

The benefits of using MPLS for VPNs will be magnified if SPs have a choice of interoperable multi-vendor equipment that supports the VPN solutions. Standardization efforts are under way in the IETF MPLS Working Group for the technologies required for such solutions. The main challenge over the coming months will be to whittle down the number of different possible approaches for VPN membership determination and VPN/CoS multiplexing to a few generally applicable solutions to maximize interoperability.

11. GLOSSARY

AS: Autonomous System. A part of the network under a single administration and usually running a single routing protocol for internal routing.

BGP: Border Gateway Protocol. The Exterior Gateway Protocol used for distributing routes over the Internet backbone.

CoS: Class of Service. Used in this paper to distinguish the service levels used in the network core from the DSCPs used in a particular VPN

CR-LDP: Constraint-based Routed Label Distribution Protocol. Extensions to LDP to set up Traffic Engineered LSPs, as defined in the Internet Draft "Constraint-based LSP Setup using LDP" [10].

DLCI: Data Link Circuit Identifier. The labels used in Frame Relay that are equivalent to MPLS labels.

DiffServ: Differentiated Services. A system of differentiating data packets for IP networks that is based on setting relative priorities and drop precedence for each DSCP. It is defined by the DiffServ Working Group.

DSCP: Differentiated Service Code Point. A DSCP identifies a particular flavor of differentiated service. DiffServ is defined for IP, but similar concepts exist for other network types. A few DSCPs have a globally assigned meaning, but most are administratively assigned.

EGP: Exterior Gateway Protocol. Any routing protocol used for distributing routes between Autonomous Systems such as BGP.

ER: Explicit Route. A route specified by the initiator during LSP setup and not determined by the routing protocol at each hop across the network.

FEC: Forwarding Equivalence Class. A logical aggregation of traffic that is forwarded in the same way by an LSR. A FEC can represent any aggregation that is convenient for the SP. FECs may be based on such things as destination address, VPN Id and DSCP.

IGP: Interior Gateway Protocol. Any routing protocol used for distributing routes within a single Autonomous System such as OSPF.

IPSEC: IP Security. A cryptographic security system for IP traffic defined in RFC 2401 and related RFCs issued by the IPSec Working Group.

L2TP: Layer 2 Tunneling Protocol. Allows the separation of remote access sessions into the access concentration function provided by the LAC, and the gateway to the a customers network, the LNS. A SP can provide the LAC for multiple LNSs from a shared modem bank.

Labels RSVP: See RSVP-TE.

LAC: L2TP Access Concentrator. The SP side of a L2TP VPDN.

LDP: Label Distribution Protocol. A protocol defined [11] by the IETF MPLS working group for distributing labels to set up MPLS LSPs.

LER: Label Edge Router. An LSR at the edge of the MPLS network. LERs typically form the ingress and egress points of LSP tunnels.

LNS: L2TP Network Server. The customer side of a L2TP VPDN.

LSP: Label Switched Path. A data forwarding path determined by labels attached to each data packet where the data is forwarded at each hop according to the value of the labels.

LSP Tunnel: A Traffic Engineered LSP capable of carrying multiple data flows.

LSR: Label Switching Router. A component of an MPLS network that forwards data based on the labels associated with each data packet.

MPLS: MultiProtocol Label Switching. A standardized technology that provides connection-oriented switching based on IP routing protocols and labeling of data packets.

OSPF: Open Shortest Path First. A common routing protocol that provides IGP function.

PPP: Point-to-Point Protocol. A common access protocol for VPNs particularly important in providing connection from roaming workstations.

RSVP: Resource ReSerVation Protocol (RFC 2205). A setup protocol designed to reserve resources in an Integrated Services Internet. RSVP has been extended to form Labels RSVP.

RSVP-TE: Extensions to RSVP to set up Traffic Engineered LSPs. Throughout this document, Labels RSVP or RSVP-TE is referred to simply as “RSVP”.

SLA: Service Level Agreement. The agreement between an end-user (or VPN customer) and a Service Provider that specifies the type of service that will be provided. This may define guaranteed bandwidth, network reliability and so forth.

SP: Service Provider.

VLL: Virtual Leased Line. The simplest form of VPN.

VPDN: Virtual Private Dial Network. Allows a customer to outsource the provision of dial-in access to their network to an SP.

VPI/VCI: Virtual Path Identifier / Virtual Channel Identifier. The labels used in ATM layer 2 networks that are equivalent to MPLS labels.

VPLS: Virtual Private LAN Segment. A VPN that emulates a LAN. A VPLS supports any-to-any connectivity, including multicast support, for any protocol used by the customer.

VPN: Virtual Private Network. A private network provided by securely sharing resources within a wider, common network.

VPRN: Virtual Private Routed Network. A VPN that uses IP routing between sites.

12. REFERENCES

The following documents are referenced within this white paper. All RFCs and Internet drafts are available from www.ietf.org URLs are provided for other references.

Note that all Internet drafts are "work in progress" and may be subject to change, or may be withdrawn, without notice.

1	White paper from Data Connection (www.dataconnection.com)	MPLS Traffic Engineering: A choice of Signaling Protocols
2	RFC 2547	BGP/MPLS VPNs
3	RFC 2685	Virtual Private Networks Identifier
4	RFC 2764	A Framework for IP Based Virtual Private Networks
5	draft-muthukrishnan-rfc2917bis	Core MPLS IP VPN Architecture
6	draft-ouldbrahim-vpn-vr	Network based IP VPN Architecture using virtual routers
7	draft-rosen-rfc2547bis	Updated version of RFC 2547
8	draft-ietf-mpls-te-mib	MPLS Traffic Engineering MIB
9	draft-ietf-mpls-rsvp-lsp-tunnel	Extensions to RSVP for LSP Tunnels
10	draft-ietf-mpls-cr-ldp	Constraint-based Routed LSP Setup Using LDP
11	draft-ietf-mpls-ldp	LDP specification
12	draft-ietf-mpls-diff-ext	MPLS Support of Differentiated Services
13	draft-ietf-mpls-arch	MultiProtocol Label Switching Architecture
14	draft-ietf-mpls-generalized-signaling	Generalized MPLS Signaling
15	RFC 2401	Security Architecture for IP (IPSEC)
16	draft-nadeau-mpls-vpn-mib	MPLS/BGP Virtual Private Network MIB

In addition, a number of other Internet drafts and RFCs for MPLS which may be of interest.

RFC 2205	Resource ReSerVation Protocol (RSVP)
RFC 2661	Layer Two Tunneling Protocol (L2TP)
draft-ietf-mpls-framework	A Framework for MPLS
draft-ietf-mpls-rsvp-tunnel-applicability	Applicability Statement for Extensions to RSVP for LSP-Tunnels
draft-ietf-mpls-crldp-applic	Applicability Statement for CR-LDP
RFC 1661	The Point-to-Point Protocol (PPP)

13. ABOUT DATA CONNECTION

Data Connection Limited (DCL) is the leading independent developer and supplier of MPLS, ATM, SS7, MGCP/Megaco, SSCTP, VoIP Conferencing, Messaging, Directory and SNA portable products. Customers include Alcatel, Cabletron, Cisco, Fujitsu, Hewlett-Packard, Hitachi, IBM Corp., Microsoft, Nortel, SGI and Sun.

Data Connection is headquartered in London UK, with US offices in Reston, VA and Alameda, CA. It was founded in 1981 and is privately held. During each of the past 19 years its profits have exceeded 20% of revenue. Last year sales exceeded $30 million, of which 90% were outside the UK, mostly in the US. In 1999 the company received its second Queen's Award for outstanding export performance.

The DC-MPLS product family provides OEMs with a flexible source code solution with the same high quality architecture and support for which Data Connection's other communications software products are renowned. It runs within Data Connection's existing high performance portable execution environment (the N-BASE). This provides extensive scalability and flexibility by enabling distribution of protocol components across a wide range of hardware configurations from DSPs to line cards to specialized signaling processors. It has fault tolerance designed in from the start, providing hot swap on failure or upgrade of hardware or software.

DC-MPLS is suitable for use in a wide range of IP switching and routing devices including Label Switch Routers (LSRs) and Label Edge Routers (LERs). Support is provided for a range of label distribution methods including Resource ReSerVation Protocol (RSVP), Constraint-based Routed Label Distribution Protocol (CR-LDP) and Label Distribution Protocol (LDP). The rich feature set gives DC-MPLS the performance, scalability and reliability required for the most demanding MPLS applications, including VPN solutions for massively-scalable access devices.

DC-MPLS integrates seamlessly with Data Connection's ATM, SS7, MGCP and other converged network software products, and uses the same proven N-BASE communications execution environment. The N-BASE has been ported to a large number of operating systems including VxWorks, Linux, OSE, pSOS, Chorus, Nucleus, Solaris, HP-UX and Windows NT, and has been used on all common processors including x86, i960, Motorola 860, Sparc, IDT and MIPS. Proprietary OSs and chipsets can be supported with minimal effort.

DC-Directory is a complete directory solution, which combines the best elements of X.500 and Internet standards (LDAP, HTTP) with comprehensive management and administration applications to provide an open, scalable directory service for enterprises and Service Providers. DC-Directory can be combined with DC-Metalink, a unique meta-directory product, to provide unrivalled flexibility in integrating legacy and proprietary directory systems in a coherent and manageable way.

Paul Brittain is a senior networking architect with Data Connection and European Product Manager for MetaSwitch.

Adrian Farrel is Architect and Development Manager for the DC-MPLS product family.