MPLS Working Group Juha Heinanen
Internet Engineering Task Force Telia Finland, Inc.
INTERNET DRAFT Bryan Gleeson, Arthur Lin
Expires February 1998 Shasta Networks, Inc.

MPLS Mappings of Generic VPN Mechanisms
<draft-heinanen-generic-vpn-mpls-00.txt>

1. Status of this Memo

2. Abstract

This document describes a set of generic mechanisms which can be used to set up network based Virtual Private Networks (VPN) across IP networks. In particular, it describes how these mechanisms can be mapped into a network running the Multi-Protocol Label Switching (MPLS) specification. The mechanisms described, however, can apply to any type of IP network running various forms of IP tunneling mechanisms, and are not solely restricted to MPLS networks. This Draft serves to introduce these generic mechanisms, which are part of the broader VPN framework which will be described more fully in forthcoming Drafts.

3. Introduction

An earlier Draft [Heinanen] proposed a number of mechanisms for Virtual Private Network (VPN) support in networks running the Multi-Protocol Label Switching (MPLS) specification [Callon].

Subsequently, it was noted that most of the mechanisms proposed in that Draft are subsets of generic VPN mechanisms, and can also apply to current IP networks. Hence this Draft first discusses these generic mechanisms and shows how these may be applied in general IP networks. In particular, these can be applied in IP networks not running any major new protocol, such as MPLS, which may facilitate roll- out of VPN services on current IP networks, prior to the possible future deployment of MPLS. Subsequently, the Draft also discusses how these mechanisms can be applied to MPLS networks in particular.

As with the earlier Draft, this Draft is intended to serve as a framework, highlighting areas for more detailed specification. Neither has enough detail to allow for interoperable implementations. Hence more work is required to finalize a specification for VPN support, both on MPLS networks and on current networks. A future Internet Draft will propose a more general VPN framework and specific areas for future specification so as to allow more general interoperable VPN solutions.

4. VPN Definition and Scope

A VPN can be succinctly defined as the emulation of a private wide area network (WAN) facility using IP facilities (including the public Internet, or private IP backbones). There are a wide variety of VPN types, corresponding to the very wide variety of WAN facilities that are currently defined. Future Drafts will discuss the full range of possible VPN types, but the particular type of VPN specifically discussed in this Draft can be described as a 'virtual private routed network' (VPRN), in which a customer with multiple geographically dispersed sites wishes to connect each of these sites together into a private network. Such networks are routinely built today using, for instance, frame relay links and/or leased lines between the routers at each pair of sites, or, more likely, given the cost of such links, by star wiring each site to a single central site.

A VPRN emulates such a network using dedicated IP links. The nature of the connectivity of these sites is discussed further below, but for the moment it can be assumed that it is desired that each of these sites be logically meshed to each other site, since there is less cost assumed with full meshing in a virtual IP network, than in cases where physical resources (e.g. Frame Relay DLCI, or a leased line) must be allocated for each connected pair of sites. This yields optimal routing, since it precludes the need for traffic between two sites to traverse through a third.

VPNs of various sorts are today routinely implemented using a

combination of host applications and customer premises equipment (CPE) routers. The mechanisms discussed in both this Draft and its predecessor, however, apply specifically only to the class of 'network based VPNs', where the operation of the VPN is outsourced to an Internet service provider (ISP), and is implemented on network as opposed to CPE equipment. There is significant interest in such solutions both by customers seeking to reduce support costs and by ISPs seeking new revenue sources. The network based focus allows the use of particular mechanisms which may lead to highly efficient and cost effective VPRN solutions. However such mechanisms also leverage tools (e.g. piggybacking on routing protocols) which are accessible only to ISPs and which are unlikely to be made available to any customer, or even hosted on ISP owned and operated CPE, due to the problems of coordinating joint management of the CPE gear by both the ISP and the customer.

Hence, it is assumed that each customer site CPE router connects to an ISP edge router through one or more dedicated point-to-point stub links (e.g. leased lines, ATM or Frame Relay connections); the VPRN mechanisms discussed below will operate on each of these ISP edge routers, in order to route traffic received across a stub link to the appropriate destination customer site across its stub link. In particular, the edge router will hide all VPRN topology information from the CPE routers, hence significantly simplifying the operation of the CPE.

Note that a single ISP edge router could terminate multiple stub links belonging to the same VPRN. The means by which traffic is routed between such local interfaces is outside the scope of standardization, per se, though obviously these would leverage many of the same routing and forwarding mechanisms used for communication with remote VPN sites.

In such a scenario, a VPN connecting each of these sites must generally meet a number of minimum requirements, which arise from the need to essentially emulate the facilities that customers expect from a leased line facility, and which, hence, they also generally expect from any emulation of such a facility. While there are a number of such requirements, three are of particular concern to this Draft:

A. Support for Disjoint Address Spaces:

The addressing used within the VPRN may have no relation to the addressing of the ISP, or ISPs, across which the VPRN may operate. In particular, the former may also be non-unique, private IP addressing [Rekhter1].

B. Support for Intra-VPN routing:

Since a VPRN will generally interconnect multiple sites, the VPRN mechanism must implement some mechanism by which intra-VPN traffic can be efficiently routed to the correct destination site within the VPRN.

C. Support for Data Security:

In general, customers using VPNs require some form of data security, given the general perceptions of the lack of security of IP networks, and particularly that of the Internet. Whether or not this perception is correct, it is one that must be addressed by any VPN implementation. Most recent VPN implementations are converging on the use of IPSec facilities [Kent] for this purpose.

Together, these requirements imply that VPRNs must be implemented through some form of IP tunneling mechanism, where the addressing used within the VPRN can be disjoint from that used to route the tunneled packets across the IP backbone. Such tunnels, depending upon their form, can provide some level of intrinsic data security, or this can also be enhanced using other mechanisms (e.g. IPSec).

Such tunnels together form an overlay network operating over and across the general Internet backbone, connecting each of the ISP edge routers supporting VPN stub links to each other. Within each of the ISP edge routers, there must be VPN specific IP forwarding mechanisms, to forward packets received across each of the stub links ('ingress' traffic) to the appropriate destination edge router, based upon the address space of the customer's network, and to forward packets received from the core ('egress' traffic) to the appropriate stub link, for cases where an edge router supports multiple stub links belonging to the same VPN (as will be noted below, VPN tunnels can, as a local matter, either terminate on the edge router, or on each stub link; in the former case, a VPN specific forwarding table is needed for egress traffic, in the latter case, it is not).

Note also that a single customer site may belong concurrently to multiple VPNs, of various sorts, and/or may wish to transmit traffic both onto one or more VPNs and to the default Internet. The mechanisms needed to do this are outside the scope of this Draft, and are not discussed further.

For the purposes of this Draft, it is also assumed that all of the traffic being sent across the VPRN is IP traffic, since the devices implementing the VPRN need to be able to interpret the packet header information to determine the appropriate end-point within the VPRN. However, that procedures discussed here could be readily extended for Multiprotocol transport, either by forming separate VPRNs for each protocol, or by running Multiprotocol routing and forwarding

procedures in each VPN router, and multiplexing multiple protocols across the VPN stub links. IP encapsulation methods could also be used to transport different protocols across IP links. These multiprotocol transport mechanisms are left for further study.

5. Generic VPN Mechanisms

ISPs wishing to offer tunnel based VPRN services need to be able to do so with minimal configuration, since this yields the most cost effective solution. The generic VPN mechanisms discussed here apply to the means by which this can be done. The following sections discuss these in detail.

5.1 VPN Membership Information Configuration and Dissemination

In order to establish a VPRN, or to insert new customer sites into an established VPRN, the stub links on each edge router from those sites in the particular VPRN must first be configured with the identity of the particular VPRN to which the stub links belong. Note that this first step, of stub link configuration, is unavoidable, since clearly the edge router cannot infer such bindings and hence must be configured with this information. The means by which this is done are outside the scope of this Draft, but a management information base (MIB) allowing for bindings between local stub links and VPN identities may be one obvious solution.

Thereafter, each edge router must learn either the identity of, or, at least, the router to, each other edge router supporting other stub links in that particular VPRN. Implicit in the latter is the notion that there exists some mechanism by which the configured edge routers can then use this edge router and/or stub link identity information to subsequently set up the appropriate tunnels between them; this is discussed further below.

In order to configure each stub link with the identity of the VPN to which it belongs, some form of VPN identifier is required; the scope of uniqueness of this identifier is a function of its usage, which is related to how VPRN membership is disseminated. This problem, of VPRN member dissemination between participating edge routers, can be solved in a variety of ways:

A. Directory Lookup:

The members of a particular VPRN, that is, at a minimum, the identity of the edge routers supporting stub links in the VPRN, and possibly also the identity of each of the stub links, could be configured into a directory, which edge routers could query, using some defined

mechanism (e.g. LDAP), upon configuration of their local stub interfaces and VPN identifier. The latter, in this case, need only be unique within the scope of the directory.

This mechanism allows for authorization checking prior to disseminating VPRN membership information, which may be desirable where VPRNs span multiple administrative domains. In such a case, directory to directory protocol mechanisms could also be used to propagate authorized VPRN membership information between the directory systems of the multiple administrative domains.

There would also need to be some form of database synchronization mechanism (e.g. triggered or regular polling of the directory by edge routers, or active pushing of update information to the edge routers by the directory) in order for all edge routers to learn the identity of newly configured sites inserted into an active VPRN.

B. Explicit Management Configuration:

A VPRN Management Information Base (MIB) could be defined which would allow a central management system to configure each edge router with the identities of each other participating edge router and possibly also the identity of each of the stub links. Similar mechanisms could also be used, as noted above, to configure the VPN bindings of the local stub links on the edge router. The scope of the VPN identifier in this case is related to the scope of the management system.

Note that this mechanism allows the management station to impose strict authorization control; on the other hand, it may be more difficult to configure edge routers outside the scope of the management system. The management configuration model can also be considered a subset of the directory method, in that the (management) directories could use MIBs to push VPRN membership information to the participating edge routers, either subsequent to, or as part of, the local stub link configuration process.

C. Piggybacking in Routing Protocols:

VPRN membership information could be piggybacked into the routing protocols run by each edge router, since this is an efficient means of automatically propagating information throughout the network to other participating edge routers. Specifically, each route advertisement by each edge router could include, at the minimum, the set of VPN identifiers associated with each edge router, and adequate information to allow other edge routers to determine the identity of, and/or, the route to, the particular edge router. Other edge routers would examine received route advertisements to determine if any

contained information relevant to a supported (i.e. configured) VPRN; this determination could be done by looking for a VPN identifier matching a locally configured VPN. The nature of the piggybacked information, and related issues, such as scoping, and the means by which the nodes advertising particular VPN memberships will be identified, will generally be a function both of the routing protocol and of the nature of the underlying transport, and is discussed further below.

The advantage of this last scheme is that it allows for very efficient information dissemination, particularly across multiple routing domains (e.g. across different autonomous systems/ISPs) but it does require that all nodes in the path, and not just the participating edge routers, be able to accept such modified route advertisements. Significant administrative complexity may also be required to configure scoping mechanisms so as to both permit and constrain the dissemination of the piggybacked advertisements.

Furthermore, unless some security mechanism is used for routing updates so as to permit only all relevant edge routers to read the piggybacked advertisements, this scheme generally implies a trust model where all routers in the path must perforce be authorized to know this information. Depending upon the nature of the routing protocol, piggybacking may also require intermediate routers, particularly autonomous system (AS) border routers, to cache such advertisements and potentially also re-distribute them between multiple routing protocols.

Each of the schemes described above have merit in particular situations. The earlier Draft [Heinanen] discussed the last scheme only, and that is further spelled out below, but the other two schemes may also offer important practical advantages. In particular, note that, in practice, there will almost always be some directory or management system which will maintain VPN membership information, since, as noted above, the binding of VPNs to stub links must be configured, hence, presumably, such information would be obtained from, and stored within, some database. Hence the additional steps to facilitate the configuration of such information into edge routers, and/or, facilitate edge router access to such information, may not be excessively onerous. These methods will be discussed in greater detail in forthcoming Drafts.

5.1.1 VPN Identifier

A principal benefit of the router piggybacking model is that it allows for simple dissemination of VPN membership information across multiple ASs. This implies the need for a VPN identifier than can be

unique across multiple ASs. To that end, [Heinanen] proposed a globally unique VPN identifier (note that such an identifier may be useful for VPN types other than only VPRNs) made up of the concatenation of an AS number, and a label assigned by the AS administrator which is locally unique within the particular AS. It is proposed that this be adopted as the VPN identifier, with the further stipulation that the VPN ID be coded as a four octet BGP Communities Attribute [Chandra], made up a two octet AS number and a 2 octet, unstructured integer VPN number, to allow for sufficient numbers of VPNs per AS. The specific details of this proposed format are for future clarification.

Note that where a VPN crosses multiple ASs, then there must be some administrative mechanisms to coordinate VPN ID assignment e.g. through the notion of a 'home AS' for a particular VPN, which is used in the VPN ID of that VPN. A VPN ID coded as proposed could also be easily piggybacked in BGP, and could also be easily specified within BGP policy filters in AS border routers for scoping and administrative purposes.

For the remainder of the discussion, it is assumed that the VPN identifier will be as so described.

5.2 Tunneling Mechanisms

Once VPRN membership information has been disseminated, the tunnels comprising the VPRN can be constructed. While this can be done through manual configuration, this is clearly not likely to be a scalable solution, given the o(n^2) problem of meshed links. As such, tunnel set up should use some form of signaling protocol which would allow two nodes to construct a tunnel to each other knowing only each other's identity. Note also that there are some tunneling mechanisms which allow for multiple disjoint calls or sessions within the same tunnel - in such a 'shared tunnel' case, the signaling protocol could also be used to assign a call within an existing tunnel between two edge routers for a new VPN between them.

There are two specific cases of interest networks running MPLS, and current networks not running MPLS. These are discussed separately.

5.2.1 MPLS Networks

As noted in [Heinanen], MPLS can be considered to be a form of IP tunneling, since the labels of MPLS packets allow for routing decisions to be decoupled from the addressing information of the packets themselves. MPLS label distribution mechanisms can be used to

associate specific sets of MPLS labels with particular VPRN address prefixes supported on particular egress points (i.e. stub links of edge routers) and hence allow other edge routers to explicitly label and route traffic to particular VPRN stub links. The exact relationship of the various MPLS labels and the particular VPN to which they are bound is a function of whether or not the CPE edge routers participate or do not participate in MPLS with the ISP edge routers. These cases are discussed in greater detail below.

The principal attraction of MPLS as a tunneling mechanism is that it may require less processing within each edge router than alternative tunneling mechanisms. This is also a function of the fact that data security within a MPLS network is implicit in the explicit label binding, much as with a connection oriented network, such as Frame Relay. This may hence lessen customer concerns about data security and hence require less processor intensive security mechanisms (e.g. IPSec). As discussed below, however, such implicit mechanisms address only some of the potential security concerns of customers.

5.2.2 Non-MPLS Networks

For non-MPLS networks, VPNs in general require the use of an explicit IP tunneling mechanism. There are numerous IP tunneling mechanisms, including IP/IP [Simpson], GRE tunnels [Hanks], L2TP [Valencia] and IPSec [Kent]. Each of these allow for opaque transport of IP packets, with routing disjoint from the address fields of the encapsulated packets. Additional processing is required in edge routers for the use of any of these protocols, with some (e.g. IPSec) mandating significant processing capabilities. On the other hand, such tunneling protocols can provide significantly more comprehensive data security capabilities than the implicit security of MPLS.

It is the case, however, that none of the protocols listed above were originally designed to support VPNs of the type under consideration. As such, none provide all of the mechanisms likely to be needed for VPN applications. In particular, only L2TP and IPSec can be considered to have any form of signaling protocol (the L2TP control protocol, and the Internet Key Exchange protocol [Harkins], respectively) which could potentially be used to automate the process of tunnel set up. Furthermore, none of these tunneling protocols have support today for multicast (other than source replication), whereas MPLS does have such support, though the application of MPLS mechanisms to multicast transport within VPNs is not yet fully defined, and requires further study.

Given however, the current paucity of operational networks running MPLS, there is likely to be significant value in fully defining a VPN

tunneling mechanism which could be deployed in current networks. It may also be possible to readily extend other IP tunneling protocols to incorporate support for multicast. Subsequent Drafts will address these issues.

5.3 Stub Link Reachability Information

There must be mechanisms to allow ISP edge routers to determine the set of VPN addresses and address prefixes reachable at each stub link connecting to the edge router. There are a number of means by which this can be done:

A. Routing Protocol Instance:

A routing protocol can be run between the CPE edge router and the ISP edge router to exchange reachability information. Note that this routing exchange is asymmetrical since the CPE router views the ISP edge router as the default path into the VPN. Any suitable routing protocol could be used to exchange routing information between the CPE and ISP edge routers. It should be noted that the only function of this protocol is indeed to exchange reachability information, not to discover topology, since, by definition, there is only a single, point-to-point (logical) link between the CPE router and the ISP edge router, with the latter then discovering (and hiding) the VPN topology.

Likely protocols for this purpose include RIPv2, OSPF [Moy] and BGP-4 [Rekhter2]. Note that even if the same protocol is used between the CPE and ISP edge routers, and from the ISP edge routers into the core, these will be two quite distinct routing instantiations. If the ISP edge router uses routing protocol piggybacking to disseminate VPN membership and reachability information across the core, then it may redistribute suitably labeled routes from the CPE routing instantiation to the core routing instantiation (but never the other way round). There is no requirement that the same protocol, or even the same CPE reachability information gathering mechanism, be run between each CPE router and associated edge router in a particular VPRN, since this is purely local matter.

Note that if a particular customer site concurrently belongs to multiple VPNs (or wishes to concurrently communicate with both a VPN and the Internet), then the ISP edge router must have some means of unambiguously mapping stub link address prefixes to particular VPNs. This could be done either by ensuring (and appropriately configuring the ISP edge router to know) that particular disjoint address prefixes are mapped into separate VPNs, or by tagging the routing advertisements from the CPE edge router with the appropriate VPN

identifier. In the case of MPLS, as discussed below, different MPLS labels would be used to differentiate the disjoint prefixes assigned to particular VPNs. In any case, some administrative procedure would be required for this coordination.

B. Configuration:

The reachability information across each stub link could be manually configured, which may be appropriate if the set of addresses or prefixes is small and static.

C. ISP Administered Addresses:

The set of addresses used by each stub site could be administered and allocated by the ISP, which may be appropriate for very small sites with little network administration resources. In such a case the ISP edge router could determine these addresses by proxying for the particular address administration mechanism (e.g. DHCP). Note that in this case it would be the responsibility of the ISP to ensure that each site in the VPN received a disjoint address space.

D. MPLS Label Distribution Protocol:

In cases where the CPE edge router runs MPLS, the MPLS LDP could be extended to convey the set of prefixes at each stub site, together with the appropriate labeling information. While LDP is not generally considered a routing protocol per se, it may be useful to extend it for this particular constrained scenario. This is for further study.

5.4 Intra-VPN Reachability Information

Once an edge router has determined the set of prefixes associated with each of its stub links, then this information must be disseminated to each other edge router in the VPRN. Note also that there is an implicit requirement that the set of reachable addresses within the VPRN be locally unique that is, each VPRN stub link (not performing load sharing) maintain an address space disjoint from any other, so as to permit unambiguous routing. In practical terms, it is also generally desirable, though not required, that this address space be well partitioned i.e. specific, disjoint address prefixes per stub link, so as to preclude the need to maintain and disseminate large numbers of host routes.

The intra-VPN reachability information dissemination can be solved in a number of ways, some of which include the following:

A. Directory Lookup:

Along with VPN membership information, a central directory could maintain a listing of the address prefixes associated with each end point. Such information could be obtained by the server through protocol interactions with each edge router. Note that the same directory synchronization issues discussed above would apply in this case.

B. Explicit Configuration:

The address spaces associated with each edge router could be explicitly configured into each other router. This is clearly a non-scalable solution, and also raises the question of how the management system learns such information in the first place.

C. Local Intra-VPRN Routing Instantiations:

In this approach, each edge router runs an instantiation of a routing protocol (a 'virtual router') per VPRN, running across the VPRN tunnels to each peer edge router, to disseminate intra-VPRN reachability information. The intra-VPN routing advertisements could be distinguished from normal tunnel data packets either by being addressed directly to the peer edge router, or by a tunnel specific mechanism.

Note that this intra-VPRN routing protocol need have no relationship with the routing protocols operated by the ISPs in the path. Specifically, the intra- VPRN routing protocol operates as an overlay over the IP backbone, and, given the very simple meshed topology of the VPRN, could be a very simple protocol, such as RIPv2 [Malkin], at least unless the VPRN spans a very large number of edge routers. Since the intra-VPN routing protocol runs as an overlay, it is also wholly transparent to any intermediate routers, and to any edge routers not within the VPRN. This also implies that such routing information can also remain opaque to such routers, which may be a necessary security requirements in some cases.

D. Link Reachability Protocol

Each edge router could run a link reachability protocol - for instance, some variation of the MPLS LDP - across the tunnel to each peer edge router in the VPRN, carrying the VPN ID and the reachability information of each VPRN running across the tunnel between the two edge routers. Such a protocol would need to be specified, and would require aspects of current routing protocols such as hello protocols, and re-transmit timers and/or positive acknowledgements. However, such an approach may reduce the

processing burden of running routing protocol instantiations per VPRN, and may be of particular benefit where a shared tunnel mechanism is used to connect a set of edge routers supporting multiple VPRNs.

E. Piggybacking in Routing Protocols:

As with VPN membership, the set of address prefixes associated with each stub interface could also be piggybacked into the routing advertisements from each edge router and propagated through the network. Other edge routers would extract this information from received route advertisements in the same way as they would obtain the VPRN membership information (which, in this case, is implicit in the identification of the source of each route advertisement). Note that this scheme may require, depending upon the nature of the routing protocols involved, that intermediate routers e.g. border routers cache intra-VPRN routing information in order to propagate it further. This also has implications for the trust model, and for the level of security possible for intra-VPRN routing information.

Note that in any of the cases discussed above, an edge router has the option of disseminating its stub link prefixes in a manner so as to permit tunneling from remote edge routers directly to the egress stub links. Alternatively, it could disseminate the information so as to associate all such prefixes with the edge router, rather than with specific stub links. In this case, the edge router would need to implement a VPN specific forwarding mechanism for egress traffic, to determine the correct egress stub link. The advantage of this is that it may significantly reduce the number of distinct tunnels or tunnel label information which need to be constructed and maintained. Note that this choice is purely a local manner and is not visible to remote edge routers.

The earlier Draft [Heinanen] discussed the last scheme only, and that is further spelled out below. A number of vendors have already announced, however, their intention to support variants of the virtual router scheme, which is also less disruptive to currently deployed routing protocols. As such, this scheme merits further investigation and will be addressed in future Drafts.

6. Routing Protocol Piggybacking

As noted above, routing protocol piggybacking could be used to carry VPN membership information alone, or also VPN reachability information. The means by which this is done, and the nature of the piggyback information, is a function both of the particular routing protocol, and of the underlying network mechanism. The particular

cases of OSPF and BGP-4 are discussed below.

6.1 OSPF

OSPF is often used as an intra-AS routing protocol, and hence may be a required candidate for routing protocol piggybacking. One means by which VPN membership and reachability information could be piggybacked is through the use of the proposed OSPF opaque LSA [Coltun]. The exact details of how such a piggybacking advertisement might be coded are for further study. In particular, it may prove to be the case that opaque LSAs could be well suited for piggybacking VPN membership information, but not VPN reachability information, since opaque LSAs, at least as currently defined, are attributes of, not indexes into, reachability information. Using them in the latter manner, which would be required to piggyback VPN reachability information, may break some existing OSPF implementations. Opaque LSAs do, however, have a well defined scoping mechanism, that, at least within an AS, allows for control over the extent of dissemination of a VPN advertisement.

Note also that as a link state protocol OSPF advertisements always allow for the identification of the source of an advertisement. However, each router in the OSPF network, and not only edge routers, will also need to examine received advertisements, and explicitly ignore piggybacked VPN advertisements, unless configured to be a VPN terminator (i.e. edge router).

6.2 BGP-4

There are a number of potential mechanisms by which VPN information could be piggybacked into BGP-4, including the Multiprotocol Extensions attribute [Bates] or the BGP communities attribute. In the case where VPN reachability information is piggybacked, each VPN address prefix could be encoded as Network Layer Reachability Information (NLRI) and bound to the VPN identifier as a community attribute, if the VPN ID has the format proposed previously. Note that in cases where it was desired only to advertise VPN membership information, then advertising each VPN prefix may be onerous and undesirable, but there is no specific mechanism in BGP-4, as yet, to advertise anything other than NLRI.

In the case where this is done on an MPLS network, then the advertisement would carry each VPN prefix, together with the MPLS label(s) to be used to send packets to that stub link. As noted above, these labels, as a purely local matter, could identify either the route to each stub link, or only to the edge router itself, which would then use its own forwarding mechanisms for egress packets.

Since there is already defined a particular mechanism for carrying MPLS labels in BGP-4 using the Multiprotocol Extensions field [Rekhter3], this would suggest that this mechanism should be generalized for the purpose also of conveying VPN information; hence it is proposed that that Draft be amended for this purpose.

The use of BGP-4 for VPN piggybacking is more complex in cases where this is done on non-MPLS networks. In such a case, the piggybacked advertisements must allow for the explicit identification of the source of the advertisement. This is important for tunnel set up in non-MPLS networks, where each edge router needs to know the identity (i.e. IP address) of each of the other edge routers, in order to initiate whatever signaling mechanism may be used for tunnel set-up.

At present there is no means by which the original source of a BGP advertisement can be identified once that advertisement is redistributed (e.g. from an intra-AS protocol like OSPF into BGP at a border node, or from an edge router through a border router for distribution outside the original AS). Since VPN support in non-MPLS networks is an important requirement, it is proposed that whatever BGP-4 mechanism is chosen for the purpose of VPN advertisements also be amended to allow for explicit tagging with the IP address of the original source of the advertisement. One possible means by which this could be done may be to associate the VPN ID (coded as a Community Attribute) with the /32 prefix (i.e. IP address) of the edge router supporting the VPN. This issue is for further study.

Note that in the case where BGP advertisements are propagated across AS boundaries, then each border router must cache the full set of prefixes and associated label stacks of each advertised VPN. In such a case, further work is also needed to control scoping of BGP piggybacked advertisements. In particular, at AS boundaries, border routers would generally need to be manually configured with VPN route advertisement policies to determine whether such advertisements should be propagated, and, if so, to which peer ASs. In general ASs will also likely automatically reject VPN advertisements received from peer ASs unless specifically configured to pass them. Some administrative mechanism (e.g. manual procedures or some form of directory communication, for instance) would be needed for this purpose.

Note also that such scoping policy configurations would be needed not only in each border router of each AS with one or more VPN termination points, but also in each AS in the transit path between them. This last may pose problems if the trust system includes the terminating ASs, but excludes one or more of the transit ASs. These problems expose a particular artifact of router piggybacking - while VPN membership and reachability information is relevant only to the

particular edge routers concerned, router piggybacking necessarily requires also the active participation of all intermediate routers that need to process and propagate such advertisements. This may impose significant burdens on the operation and administration of such intermediate routers, as well as compromising the integrity of the VPNs concerned.

7. MPLS Mappings

The earlier Draft [Heinanen] proposed a number of mechanisms for facilitating VPN set up in MPLS networks. As noted above, most of these mechanisms are subsets of more generic VPN mechanisms, and some of the alternate mechanisms described above can also be applied to MPLS networks. There are specific issues with respect to mappings into MPLS networks due to the nature of the particular control and data planes of MPLS. Furthermore, the operation of these data and control planes is a function of whether or not the CPE router also runs MPLS. These cases are considered separately.

7.1 CPE Router Runs MPLS

In this case the CPE router and ISP edge router exchange, using one of the mechanisms discussed above, the set of address prefixes associated with that stub site and then, concurrently or subsequently, assign MPLS labels to each such prefix. Note that, as discussed above, the edge router could decide, as a local matter, to assign the same label to each such stub link, or distinct labels to each, depending upon whether or not it wished to explicitly forward egress packets.

If a CPE routers belongs concurrently to more than one VPN, then it must label the (disjoint) prefixes of each VPN differently, to allow for unambiguous routing at the edge router. Thereafter, the ISP edge router uses whatever routing and label assignment mechanisms may be used within its network to disseminate the prefixes, tagged with the appropriate VPN ID, and the locally assigned MPLS label, to each other peer Label Switching Router (LSR).

In the specific case where BGP-4 is used for piggybacking across the core network, this implies that each edge router and border router in the AS but not the intermediate LSRs - will receive the bindings of VPN IDs, VPN prefixes and associated labels, together with the label needed to forward traffic to the particular edge router from its BGP peers. If border routers were to propagate this information further across the core, they would then push into this label stack, information to identify the explicit tunnel route to the particular border router from its peer border routers. At a terminating edge

router, the edge router would maintain in its VPN specific Label Information Base (LIB), the mappings of particular VPN prefixes to the label stacks associated with each prefix. Note that in this case, edge routers will know, and need know, only the route (i.e. MPLS label stack) to each VPN prefix, and will not know the identity of the edge router through which that prefix is reached.

Each edge router may also advertise, using either LDP or a piggybacked routing protocol, a default label to be used by the CPE across its stub links to send data into the VPN, unless this binding was implicit (e.g. all traffic from a stub link only gets sent to one particular VPN).

Note that all of these label stacks are disjoint from the labels used for connectivity between the edge and border routers, through the intermediate LSR within the AS MPLS network. This level of intra-AS connectivity is a lower level than the BGP peering level; hence, disjoint MPLS label allocation mechanisms (e.g. LDP following prefix distribution using an intra-AS routing protocol) would be used to determine connectivity to, and the appropriate label stacks for, edge and border router connectivity across the AS.

Hence an edge or border router wishing to transmit to a particular VPN stub link would need to first determine the destination VPN prefix, and the VPN label stack associated with that prefix. Subsequently, it would then determine the label switched path (LSP) to the particular destination edge router, push the resultant label stack onto the VPN label and transmit the packet. At the destination edge router, the intra-AS routing label stack would be popped, and the packet sent to the appropriate stub link using either the VPN label, if explicitly tagged, or using a local forwarding mechanism, if not.

7.2 CPE Router Does Not Run MPLS

This case would work exactly as described above, except that the ISP edge router would proxy for the CPE router by assigning labels to each CPE prefix. Packets sent to the stub link would be routed as described above, except that at the destination edge router the label stack would be removed and an untagged packet sent to the CPE router. In this case, the edge router would also need some unambiguous means of determining the destination VPN, where a particular stub site supports multiple VPNs. Typically this will require disjoint address spaces in each VPN.

Note that in either case that all border routers will need to maintain label mappings for all prefixes associated with each VPN in

the AS. This is a consequence of the fact that BGP can, at present, only advertise routes to particular NLRI prefixes and hence, cannot, as discussed above, advertise only VPN to edge router bindings. It is also, of course, obvious, that such information cannot in any sense be aggregated.

7.3 Use of RSVP in MPLS Networks

There have been a number of proposals to use the Resource Reservation Protocol (RSVP) to allocate labels within MPLS networks, either for the purpose of setting up flow specific LSPs [Davie] or for administrating traffic engineered tunnels across multiple ASs, as in the PASTE proposal [Li]. In either case, VPN membership and, perhaps also VPN reachability information, could be carried using such RSVP based label allocation mechanisms, as with the use of the LDP, described above. In particular, in the case of the PASTE proposal, RSVP is used to set up and administer traffic engineered tunnels that span potentially multiple service provider domains, and provide non-default path forwarding. In such a case, router piggybacking may not be possible, and hence RSVP may be the only protocol available in which to piggyback VPN advertisements. This subject is for further study.

8. Security Considerations

As noted above, VPN operation on MPLS networks relies upon the implicit security of explicitly labeled LSPs. Unless a particular edge router has been configured for membership into a particular VPN, then no CPE router connected to that edge router should be able to insert traffic into that VPN. Note, however, that this is only true if each edge router in the network, and not just those participating in the VPN, ensures that no CPE router can transmit packets with label information that may cause it to be inserted, at some merge point, into a LSP leading to the labeled VPN. As such, the trust model for MPLS based VPNs must encompass all MPLS edge routers, and not only those participating in the particular VPN.

The particular form of data security offered by MPLS based VPNs also does not address other potential security concerns e.g. data snooping, non- repudiation, etc. Such concerns can only be met through more explicit security mechanisms e.g. IPSec. As noted earlier, such mechanisms are required for any tunneling mechanism operating on non-MPLS networks where paths are not explicitly labeled. Note, however, that such security mechanisms - e.g. bilateral IPSec peerings between two edge routers in a VPN - have the advantage that the trust model need only include the relevant edge

routers, and not any of the intermediate routers (or administrative domains).

Particular VPN configuration mechanisms have their own security issues. In particular, piggybacking of VPN membership and routing information in routing protocols would potentially expose such information to all intermediate routing nodes. This may be a particular issue where this mechanism is used to distribute VPN information across multiple ASs or ISPs. Mechanisms to address this problem may be worthy of study e.g. the use of encryption and authentication mechanisms to protect such piggybacked information, with the use of key distribution mechanisms to restrict access only to trusted edge routers.

9. Intellectual Property Considerations

Cisco Systems has claimed potential intellectual property rights to certain aspects of the mechanisms discussed in the earlier Draft, and referred to here. Refer to [Heinanen] for the specific disclosure notice. The nature, extent and impact of these claims are unknown to the present authors.

10. Acknowledgements

Thanks to Tony Li, of Juniper Networks, for his helpful review and feedback and to Anthony Alles, of Shasta Networks, for his assistance in the generation of this Draft.

11. References

[Bates] Bates, T. "Multiprotocol Extensions for BGP-4", RFC 2283.

[Callon] Callon, R., et al "Multiprotocol Label Switching Architecture", draft-ietf-mpls-arch-02.txt.

[Chandra] Chandra, R. and Traina, P. "BGP Communities Attribute", RFC 1998.

[Coltun] Coltun, R. "The OSPF Opaque LSA Option", RFC 2370.

[Davie] Davie, B., et al - "Use of Label Switching with RSVP", draft-ietf-mpls-rsvp-00.txt

[Hanks] Hanks, S., et al "Generic Routing Encapsulation over Ipv4 Networks", RFC 1702.

[Harkins] Harkins, D. and Carrel, D. "The Internet Key Exchange (IKE)", draft-ietf-ipsec-isakmp-oakley-08.txt.

[Heinanen] Heinanen, J. and Rosen, E. "VPN Support with MPLS" draft-heinanen-mpls-vpn-01.txt.

[Kent] Kent, S. and Atkinson, R. "Security Architecture for the Internet Protocol", draft-ietf-ipsec-arch-sec-06.txt.

[Li] Li, T. and Rekhter, Y. - "Provider Architecture for Differentiated Services and Traffic Engineering (PASTE)", draft-li-paste-00.txt.

[Malkin] Malkin, G. "RIP Version 2 Carrying Additional Information", RFC 1723.

[Moy] Moy, J. "OSPF Version 2", RFC 2328.

[Rekhter1] Rekhter, Y., et al "Address Allocation for Private Internets", RFC 1918.

[Rekhter2] Rekhter, Y. and Li, T. "A Border Gateway Protocol 4 (BGP-4)", RFC 1771.

[Rekhter3] Rekhter, Y. and Rosen, E. "Carrying Label Information in BGP-4", draft-ietf-mpls-bgp4-mpls-00.txt.

[Simpson] Simpson, W. "IP in IP Tunneling", RFC 1853.

[Valencia], Valencia, A., et al "Layer Two Tunneling Protocol "L2TP"", draft-ietf-pppext-l2tp-11.txt.

11. Author Information

Juha Heinanen
Telia Finland, Inc.
Myyrmaentie 2
01600 VANTAA
Finland
Tel: +358 303 944 808
Email: jh@telia.fi

Bryan Gleeson
Shasta Networks
249 Humboldt Court
Sunnyvale CA 94089-1300
USA
Tel: +1 (408) 548 3711

Email: bgleeson@shastanets.com

Arthur Lin
Shasta Networks
249 Humboldt Court
Sunnyvale CA 94089-1300
USA
Tel: +1 (408) 548 3788
Email: alin@shastanets.com