in order   develop he full   t of pecifications required for

Table of Contents

1.0 Introduction

This document describes a framework for virtual private networks

different types of VPNs, their respective requirements, and proposes specific mechanisms that could be used to implement each type of 7.1.N using existing or proposed specifications. The objective of this document is to serve as a framework for related protocol development in order to develop the full set of specifications required for widespread deployment of interoperable VPN solutions.

There is currently significant interest in the deployment of virtual private networks (7.1.N), across IP backbone facilities. The widespread deployment of 7.1.Ns has been hampered, however, by the lack of interoperable implementations, which, in turn, derives from the lack of gene.....agreement on the definition and scope of 7.1.Ns and confusion over the wide variety of solutions that are all described by the term 7.1.N. In the context of this document, a VPN is simply defined as the 'emulation of a private wide area network (WAN) facility using IP facilities' (including the public Internet, or private IP backbones). As such, there are as many types of 7PNs as there are types of WANs, hence the confusion over what exactly constitutes a 7.1.N.

In this document a 7PN is modelled as a connectivity object. Hosts may be attached to a 7PN, and 7.1.Ns may be interconnected together, in the same manne. as hosts today attach to physical networks, and physical networks are interconnected together (e.g. via bridges or routers). Many aspects of networking, such as addressing, forwarding mechanism, learning and advertising reachability, QoS, security, and firewalling, have common solutions across both physical and virtual networks, and many issues that arise in the discussion of 7PNs have

(7.1.N) running across IP backbones. It discusses the various

physical network environment, and then apply the same principle to an environment which contains virtual as well as physical networks, and to develop appropriate extensions and enhancements when necessary. Clearly having mechanisms that are common across both physical and virtual networks facilitates the introduction of nPNs into existing networks, and also reduces the effort needed for both standards and

- dedicated WANs that permanently connect together multiple sites,

B. Supppt for Data Security:

In general customers using nPNs require some form of data security. There are different trust models applicable to the use of nPNs. One such model is where the customer does not trust the service provider

CPE devices that implement firewall functionality and that are

the trust involved when a customer utilizes a public switched Frame

oith this model providing firewall functionality and secure packet transpnet services is the respnnsibility of the service provider.

two or more companies have networked access

operation of the VPN is outsourced to alicInternet service provider

lhe term 'extranet' is commonly used to refer to a scen

Together, the first two of these requrs 19ments imply that VPNs must be

unrelated to that used to route the tunneled packets across the IP

Furthermore, as discussed later, such tunneling mechanisms can also

2.2 VPNs and Extranets

lhere is also significant interest in 'network based VPNs', where the
apply only to network based VPNs.

customer. lhe document will indicate which techniques are likely to

might use an extranet for its suppliers to allow it to query databases for the pricing and availability of components, and then to order and track the status of outstanding orders. Another example is joint software development, for instance, company A allows one development group within company B to access its operating system source code, and company B allows one development group in company A to access its security software. Note that the access policies can get arbitrarily complex. For example company B may internally restrict access to its security software to groups in certain geographic locations to comply with export control laws, for example.

A key feature of an extranet is thus the control of who can access what data, and this is essentially a policy decision. Policy decisions are typically enforced today at the interconnection points between different domains, for example between a private network and the Internet, or between a software test lab and the rest of the company network. The enforcement may done via a firewall, router with access list functionality, application gateway, or any similar device capable of applying policy to transit traffic. Policy controls may be implemented within a corporate network, in addition to between corporate networks. Also the interconnections between networks could be a set of bilateral links, or could be a separate network, perhaps maintained by an industry consortium. This separate network could itself be a VPN or a physical network.

Introducing VPNs into a network does not require any change to this model. Policy can be enforced between two VPNs, or between a VPN and the Internet, in exactly the same manner as is done today without VPNs. For example two VPNs could be interconnected, which each administration locally imposing its own policy controls, via a firewall, on all traffic that enters its VPN from the outside, whether from another VPN or from the Internet.

This model of a VPN provides for a separation of policy from the underlying mode of packet transport used. For example, a router may direct voice traffic to ATM VCCs for guaranteed QoS, non-local internal company traffic to secure tunnels, and other traffic to a link to the Internet. In the past the secure tunnels may have been frame relay circuits, now they may also be secure IP tunnels or MPLS LSPs.

Other models of a VPN are also possible. For example there is a model whereby a set of application flows is mapped into a VPN. As the policy rules imposed by a network administrator can get quite complex, the number of distinct sets of application flows that are used in the policy rulebase, and hence the number of VPNs, can thus grow quite large, and there can be multiple overlapping VPNs. However there is little to be gained by introducing such new

complexity into a network. Instead a VPN should be viewed as a direct analogue to a physical network, as this allows the leveraging of existing protocols and procedures, and the current expertise and skillsets of network administrators and customers.

3.0 VPN Tunneling

As noted above in section 2.0, VPNs must be implemented using some form of tunneling mechanism. This section looks at the generic requirements for such VPN tunneling mechanisms. A number of characteristics and aspects common to any link layer protocol are taken and compared with the features offered by existing tunneling protocols. This provides a basis for comparing different protocols and is also useful to highlight areas where existing tunneling protocols could benefit from extensions to better support their operation in a VPN environment.

An IP tunnel connecting two VPN endpoints is a basic building block from which a variety of different VPN services can be constructed. An IP tunnel operates as an overlay across the IP backbone, and the traffic sent through the tunnel is opaque to the underlying IP backbone. In effect the IP backbone is being used as a link layer technology, and the tunnel forms a point-to-point link.

A VPN device may terminate multiple IP tunnels and forward packets between these tunnels and other network interfaces in different ways. In the discussion of different types of VPNs, in later sections of this document, the primary distinguishing characteristic of these different types is the manner in which packets are forwarded between interfaces (e.g. bridged or routed). There is a direct analogy with how existing networking devices are characterized today. A two-port repeater just forwards packets between its ports, and does not examine the contents of the packet. A bridge forwards packets using MAC layer information contained in the packet, while a router forwards packets using layer 3 addressing information contained in the packet. Each of these three scenarios has a direct VPN analogue, as discussed later. Note that an IP tunnel is viewed as just another sort of link, which can be concatenated with another link, bound to a bridge forwarding table, or bound to an IP forwarding table, depending on the type of VPN.

The following sections look at the requirements for a generic IP tunneling protocol that can be used as a basic building block to construct different types of VPNs.

3.1 Tunneling Protocol Requirements for VPNs

There are numerous IP tunneling mechanisms, including IP/IP [Perkins], GRE tunnels [Hanks], L2TP [Townsley], IPSec [IPSec], and MPLS [Rosen2]. Note that while some of these protocols are not often thought of as tunneling protocols, they do each allow for opaque transport of frames as packet payload across an IP network, with forwarding disjoint from the address fields of the encapsulated packets.

Note, however, that there is one significant distinction between each of the IP tunneling protocols mentioned above, and MPLS. MPLS can be viewed as a specific link layer for IP, insofar as MPLS specific mechanisms apply only within the scope of an MPLS network, whereas IP based mechanisms extend to the extent of IP reachability. As such, VPN mechanisms built directly upon MPLS tunneling mechanisms cannot, by definition, extend outside the scope of MPLS networks, any more so than, for instance, ATM based mechanisms such as LANE can extend outside of ATM networks. Note however, that an MPLS network can span many different link layer technologies, and so, like an IP network, its scope is not limited by the specific link layers used. A number of proposals for defining a set of mechanisms to allow for interoperable VPNs specifically over MPLS networks have also been produced ([Heinanen2] [Jamieson] [Casey1] [Li2], [Muthukrishnan] and [Rosen1]).

There are a number of desirable requirements for a VPN tunneling mechanism, however, that are not all met by the existing tunneling mechanisms. These requirements include:

3.1.1 Multiplexing

There are cases where multiple VPN tunnels may be needed between the same two IP endpoints. This may be needed, for instance, in cases where the VPNs are network based, and each end point supports multiple customers. Traffic for different customers travels over separate tunnels between the same two physical devices. A multiplexing field is needed to distinguish which packets belong to which tunnel. Sharing a tunnel in this manner may also reduce the latency and processing burden of tunnel set up. Of the existing IP tunneling mechanisms, L2TP (via the tunnel-id and session-id fields), MPLS (via the label) and IPSec (via the SPI field) have a multiplexing mechanism. Strictly speaking GRE does not have a multiplexing field. However the key field, which was intended to be used for authenticating the source of a packet, has sometimes been used as a multiplexing field. IP/IP does not have a multiplexing field.

The IETF [Fox] and the ATM Forum [Petri] have standardized on a single format for a globally unique identifier used to identify a VPN (a VPN-ID). A VPN-ID can be used in the control plane, to bind a tunnel to a VPN at tunnel establishment time, or in the data plane, to identify the VPN associated with a packet, on a per-packet basis. In the data plane a VPN encapsulation header can be used by MPLS, MPOA and other tunneling mechanisms to aggregate packets for different VPNs over a single tunnel. In this case an explicit indication of VPN-ID is included with every packet, and no use is made of any tunnel specific multiplexing field. In the control plane a VPN-ID field can be included in any tunnel establishment signalling protocol to allow for the association of a tunnel (e.g. as identified by the SPI field) with a VPN. In this case there is no need for a VPN-ID to be included with every data packet. This is discussed further in section 5.2.1.

3.1.2 Signalling Protocol

There is some configuration information that must be known by an end point in advance of tunnel establishment, such as the IP address of the remote end point, and any relevant tunnel attributes required, such as the level of security needed. Once this information is available, the actual tunnel establishment can be completed in one of two ways - via a management operation, or via a signalling protocol that allows tunnels to be established dynamically.

An example of a management operation would be to use an SNMP MIB to configure various tunneling parameters, e.g. MPLS labels, source addresses to use for IP/IP or GRE tunnels, L2TP tunnel-ids and session-ids, or security association parameters for IPSec.

Using a signalling protocol can significantly reduce the management burden however, and as such, is essential in many deployment scenarios. It reduces the amount of configuration needed, and also reduces the management co-ordination needed if a VPN spans multiple administrative domains. For example, the value of the multiplexing field, described above, is local to the node assigning the value, and can be kept local if distributed via a signalling protocol, rather than being first configured into a management station and then distributed to the relevant nodes. A signalling protocol also allows nodes that are mobile or are only intermittently connected to establish tunnels on demand. Signalling is particularly useful for the VPRN scenario described later (section 5.0).

When used in a VPN environment a signalling protocol should allow for the transport of a VPN identifier to allow the resulting tunnel to be associated with a particular VPN. It should also allow tunnel attributes to be exchanged or negotiated, for example the use of

frame sequencing or the use of multiprotocol transport. Note that the role of the signalling protocol need only be to negotiate tunnel attributes, not to carry information about how the tunnel is used, for example whether the frames carried in the tunnel are to be forwarded at layer 2 or layer 3. (This is similar to Q.2931 ATM signalling - the same signalling protocol is used to set up Classical IP LISs as well as LANE ELANs).

Of the various tunneling protocols, the following ones support a signalling protocol that could be adapted for this purpose: MPLS (the various mechanisms for label distribution, including the label distribution protocol (LDP) [Thomas]), L2TP (the L2TP control protocol) and IPSec (the Internet Key Exchange (IKE) protocol [Harkins]), and GRE (as used with mobile-ip tunneling [Calhoun3]).

3.1.3 Data Security

A VPN tunneling protocol must support mechanisms to allow for whatever level of security may be desired by customers, including authentication and/or encryption of various strengths. None of the tunneling mechanisms discussed, other than IPSec, have intrinsic security mechanisms, but rely upon the security characteristics of the underlying IP backbone. In particular, MPLS relies upon the explicit labeling of label switched paths (LSP) to ensure that packets cannot be misdirected, while the other tunneling mechanisms can all be secured through the use of IPSec. For VPNs implemented over non-IP backbones (e.g. MPOA, Frame Relay or ATM virtual circuits), data security is implicitly provided by the layer two switch infrastructure.

Overall VPN security is not just a capability of the tunnels alone, but has to be viewed in the broader context of how packets are forwarded onto those tunnels. For example with VPRNs implemented with virtual routers, the use of separate routing and forwarding table instances ensures the isolation of traffic between VPNs. Packets on one VPN cannot be misrouted to a tunnel on a second VPN since those tunnels are not visible to the forwarding table of the first VPN.

If some form of signalling mechanism is used by one VPN end point to dynamically establish a tunnel with another endpoint, then there is a requirement to be able to authenticate the party attempting the tunnel establishment. IPSec has an array of schemes for this purpose, allowing, for example, authentication to be based on pre-shared keys, or to use digital signatures and certificates. Other tunneling schemes have weaker forms of authentication. In some cases no authentication may be needed, for example if the tunnels are provisioned, rather than dynamically established, or if the trust

model in use does not require it.

Currently the IPSec ESP protocol [Kent2] can be used to establish SAs that support either encryption or authentication or both. However the protocol specification precludes the use of an SA where neither encryption or authentication is used. In a VPN environment this "null/null" option is useful, since other aspects of the protocol (e.g. that it supports tunneling and multiplexing) may be all that is required. In effect the "null/null" option can be viewed as just another level of data security. Given that this option is of benefit in a VPN environment, it is recommended that the restrictive wording in the ESP protocol specification be removed.

3.1.4 Multiprotocol Transport

In many applications of VPNs, the VPN may carry opaque, multiprotocol traffic. As such, the tunneling protocol used must also support multiprotocol transport. L2TP is designed to transport PPP packets, and thus can be used to carry multiprotocol traffic since PPP itself is multiprotocol. GRE also provides for the identification of the protocol being tunneled. IP/IP and IPSec tunnels have no such protocol identification field, since the traffic being tunneled is assumed to be IP.

It is possible to extend the IPSec protocol suite to allow for the transport of multiprotocol packets. This can be achieved, for example, by extending the signalling component of IPSec (IKE) to indicate the protocol type of the traffic being tunneled, or to carry a packet multiplexing header (e.g. an LLC/SNAP header or GRE header) with each tunneled packet. This approach is similar to that used for the same purpose in ATM networks, where signalling is used to indicate the encapsulation used on the VCC, and where packets sent on the VCC can use either an LLC/SNAP header or be placed directly into the AAL5 payload, the latter being known as VC-multiplexing (see [Perez]).

3.1.5 Frame Sequencing

One quality of service attribute required by customers of a VPN may be frame sequencing, matching the equivalent characteristic of physical leased lines or dedicated connections. Sequencing may be required for the efficient operation of particular end to end protocols or applications. In order to implement frame sequencing, the tunneling mechanism must support a sequencing field. Both L2TP and GRE have such a field. IPSec has a sequence number field, but it is used by a receiver to perform an anti-replay check, not to guarantee in-order delivery of packets.

It is possible to extend IPSec to allow the use of the existing sequence field to guarantee in-order delivery of packets. This can be achieved, for example, by using IKE to negotiate whether or not sequencing is to be used, and to define an end point behaviour which preserves packet sequencing.

3.1.6 Tunnel Maintenance

The VPN end points must monitor the operation of the VPN tunnels to ensure that connectivity has not been lost, and to take appropriate action (such as route recalculation) if there has been a failure.

There are two approaches possible. One is for the tunneling protocol itself to periodically check in-band for loss of connectivity, and to provide an explicit indication of failure. For example L2TP has an optional keep-alive mechanism to detect non-operational tunnels.

The other approach does not require the tunneling protocol itself to perform this function, but relies on the operation of some out-of-band mechanism to determine loss of connectivity. For example if a routing protocol such as RIP or OSPF is run over a tunnel mesh, a failure to hear from a neighbour within a certain period of time will result in the routing protocol declaring the tunnel to be down. Another out-of-band approach is to perform regular ICMP pings with a peer. This is generally sufficient assurance that the tunnel is operational, due to the fact the tunnel also runs across the same IP backbone.

When tunnels are established dynamically a distinction needs to be drawn between the static and dynamic tunnel information needed. Before a tunnel can be established some static information is needed by a node, such as the identify of the remote end point and the attributes of the tunnel to propose and accept. This is typically put in place as a result of a configuration operation. As a result of the signalling exchange to establish a tunnel, some dynamic state is established in each end point, such as the value of the multiplexing field or keys to be used. For example with IPSec, the establishment of a Security Association (SA) puts in place the keys to be used for the lifetime of that SA.

Different policies may be used as to when to trigger the establishment of a dynamic tunnel. One approach is to use a data-driven approach and to trigger tunnel establishment whenever there is data to be transferred, and to timeout the tunnel due to inactivity. This approach is particularly useful if resources for the tunnel are being allocated in the network for QoS purposes. Another approach is to trigger tunnel establishment whenever the static tunnel configuration information is installed, and to attempt to keep the

tunnel up all the time.

3.1.7 Large MTUs

Since the traffic sent through a VPN tunnel may often be opaque to the underlying IP backbone, it cannot also generally be assumed that the maximum transfer unit (MTU) of the tunnel itself is less than or equal to the smallest MTU encountered on the path of the tunnel across the IP backbone. As such, fragmentation at some layer is needed.

If the frame to be transferred is mapped into one IP datagram, normal IP fragmentation will be used. An alternative approach is for the tunneling protocol itself to incorporate a segmentation and reassembly capability that operates at the tunnel level, (perhaps using the tunnel sequence number and an end-of-message marker of some sort) in order to avoid IP level fragmentation. None of the existing tunneling protocols support such a mechanism.

3.1.8 Minimization of Tunnel Overhead

There is clearly benefit in minimizing the overhead of any tunneling mechanisms. This is particularly important for the transport of jitter and latency sensitive traffic such as packetized voice and video. On the other hand, the use of security mechanisms, such as IPSec, do impose their own overhead, hence the objective should be to minimize overhead over and above that needed for security, and to not burden those tunnels in which security is not mandatory with unnecessary overhead.

One area where the amount of overhead may be significant is when voluntary tunneling is used for dial-up remote clients connecting to a VPN, due to the typically low bandwidth of dial-up links. This is discussed further in section 8.2.

3.1.9 Flow and congestion control

During the development of the L2TP protocol procedures were developed for flow and congestion control. These were necessitated primarily because of the need to provide adequate performance over lossy networks when PPP compression is used, which, unlike IP Payload Compression Protocol (IPComp) [Shacham], is stateful across packets. Another motivation was to accommodate devices with very little buffering, used for example to terminate low speed dial-up lines. However the flow and congestion control mechanisms defined in the final version of the L2TP specification are used only for the control channels, and not for data traffic.

In general the interactions between multiple layers of flow and congestion control schemes can be very complex. Given the predominance of TCP traffic in today's networks and the fact that TCP has its own end-to-end flow and congestion control mechanisms, it is not clear that there is much benefit to implementing similar mechanisms within tunneling protocols. Good flow and congestion control schemes, that can adapt to a wide variety of network conditions and deployment scenarios are complex to develop and test, both in themselves and in understanding the interaction with other schemes that may be running in parallel. There may be some benefit, however, in having the capability whereby a sender can shape traffic to the capacity of a receiver in some manner, and in providing the protocol mechanisms to allow a receiver to signal its capabilities to a sender. This is an area that may benefit from further study.

3.1.10 QoS / Traffic Management

As noted above, customers may require that VPNs yield similar behaviour to physical leased lines or dedicated connections with respect to such QoS parameters as loss rates, jitter, latency and bandwidth guarantees. How such guarantees could be delivered will, in general, be a function of the traffic management characteristics of the VPN nodes themselves, and the access and backbone networks across which they are connected.

A full discussion of QoS and VPNs is outside the scope of this document, however by modelling a VPN tunnel as just another type of link layer, many of the existing mechanisms developed for ensuring QoS over physical links can also be applied. For example at a VPN node, the mechanisms of policing, marking, queuing, shaping and scheduling can all be applied to VPN traffic with VPN-specific parameters, queues and interfaces, just as for non-VPN traffic. The techniques developed for Diffserv, Intserv and for traffic engineering in MPLS are also applicable. See also [Duffield] for a discussion of QoS and VPNs.

It should be noted, however, that this model of tunnel operation is not necessarily consistent with the way in which specific tunneling protocols are currently modelled. While a model is an aid to comprehension, and not part of a protocol specification, having differing models can complicate discussions, particularly if a model is misinterpreted as being part of a protocol specification or as constraining choice of implementation method. For example, IPSec tunnel processing can be modelled both as an interface and as an attribute of a particular packet flow.

3.2 Recommendations

IPSec is needed whenever there is a requirement for strong encryption or strong authentication. It also supports multiplexing and a signalling protocol (IKE). However extending the IPSec protocol suite to also cover the following areas would be beneficial, in order to better support the tunneling requirements of a VPN environment.

- the transport of a VPN-ID when establishing an SA (3.1.2)
- a null encryption and null authentication option (3.1.3)
- multiprotocol operation (3.1.4)
- frame sequencing (3.1.5)

L2TP provides no data security by itself, and any PPP security mechanisms used do not apply to the L2TP protocol itself, so that in order for strong security to be provided L2TP must run over IPSec. Defining specific modes of operation for IPSec when it is used to support L2TP traffic will aid interoperability. This is currently a work item for the proposed L2TP working group.

4.0 VPN Types: Virtual Leased Lines

The simplest form of a VPN is a 'virtual leased line' service. In this case a point-to-point link is provided to a customer, connecting two CPE devices, as illustrated below. The link layer type used to connect the CPE devices to the ISP nodes can be any link layer type, for example an ATM VCC or a Frame Relay circuit. The CPE devices can be either routers bridges or hosts.

The two ISP nodes are both connected to an IP network, and an IP tunnel is set up between them. Each ISP node is configured to bind the two links together at layer 2 (e.g. the ATM VCC and the IP tunnel). Frames are relayed between the two links. For example the AAL5 payload is taken and encapsulated in an IPSec tunnel, and vice versa. The contents of the AAL5 payload are opaque to the ISP node, and are not examined there.

To a customer it looks the same as if a single ATM VCC or Frame Relay circuit were used to interconnect the CPE devices, and the customer could be unaware that part of the circuit was in fact implemented over an IP backbone. This may be useful, for example, if a provider wishes to provide a LAN interconnect service using ATM as the network interface, but does not have an ATM network that directly interconnects all possible customer sites.

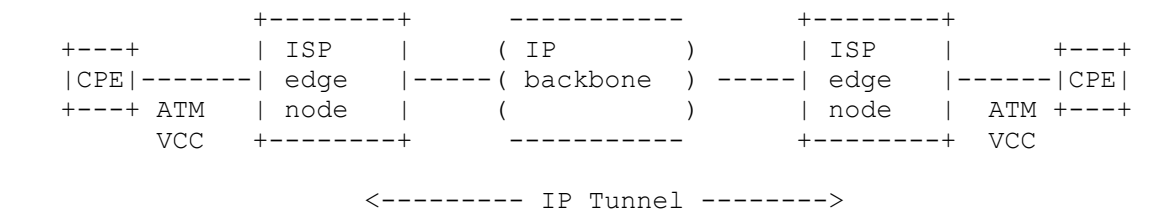

Figure 4.1: VLL Example

It is not necessary that the two links used to connect the CPE devices to the ISP nodes be of the same media type, but in this case the ISP nodes cannot treat the traffic in an opaque manner, as described above. Instead the ISP nodes must perform the functions of an interworking device between the two media types (e.g. ATM and Frame Relay), and perform functions such as LLC/SNAP to NLPID conversion, mapping between ARP protocol variants and performing any media specific processing that may be expected by the CPE devices (e.g. ATM OAM cell handling or Frame Relay XID exchanges).

The IP tunneling protocol used must support multiprotocol operation and may need to support sequencing, if that characteristic is important to the customer traffic. If the tunnels are established using a signalling protocol, they may be set up in a data driven manner, when a frame is received from a customer link and no tunnel exists, or the tunnels may be established at provisioning time and kept up permanently.

Note that the use of the term 'VLL' in this document is different to that used in the definition of the Diffserv Expedited Forwarding PHB [Jacobson]. In that document a VLL is used to mean a low latency, low jitter, assured bandwidth path, which can be provided using the described PHB. Although the use of the term VLL in this document shares some similarities with that in [Jacobson], in that a VLL can be viewed as 'a wire', its use here is different, in that it refers to a generic link layer pipe, one segment of which is an IP tunnel, and does not imply any specific QoS mechanism, Diffserv or otherwise.

5.0 VPN Types: Virtual Private Routed Networks

A virtual private routed network (VPRN) is defined to be the emulation of a multi-site wide area routed network using IP facilities. This section looks at how a network-based VPRN service

can be provided. CPE-based VPRNs are also possible, but are not specifically discussed here. With network-based VPRNs many of the issues that need to be addressed are concerned with configuration and operational issues, which must take into account the split in administrative responsibility between the service provider (ISP) and the service user (customer).

A VPRN consists of a mesh of IP tunnels between ISP routers, together with the routing capabilities needed to forward traffic received at each VPRN node to the appropriate destination site. Attached to the ISP routers are CPE routers connected via one or more links, termed 'stub' links. There is a VPRN specific forwarding table at each ISP router that contains members of the VPRN. Traffic is forwarded between ISP routers, and between ISP routers and customer sites, using these forwarding tables, which contain network layer reachability information (in contrast to a Virtual Private LAN Segment type of VPN (VPLS) where the forwarding tables contain MAC layer reachability information - see section 7.0).

An example VPRN is illustrated in the following diagram, which shows 3 ISP edge routers connected via a full mesh of IP tunnels, used to interconnect 4 CPE routers. One of the CPE routers is multihomed to the ISP network. In the multihomed case, all stub links may be active, or, as shown, there may be one primary and one or more backup links to be used in case of failure of the primary. The term 'backdoor' link is used to refer to a link between two customer sites that does not traverse the ISP network.

The principal benefit of a VPRN is that the complexity and the configuration of the CPE routers is minimized. To a CPE router, the ISP edge router appears as a neighbour router in the customer's network, to which it sends all traffic, using a default route. The tunnel mesh that is set up to transfer traffic extends between the ISP edge routers, not the CPE routers. In effect the burden of tunnel establishment and maintenance and routing configuration is outsourced to the ISP. In addition other services needed for the operation of a VPN such as the provision of a firewall and QoS processing can be handled by a small number of ISP edge routers, rather than a large number of potentially heterogeneous CPE devices. The introduction and management of new services can also be more easily handled, as this can be achieved without the need to upgrade any CPE equipment. This latter benefit is particularly important when there may be large numbers of residential subscribers using VPN services to access private corporate networks. In this respect the model is somewhat akin to that used for telephony services, whereby new services (e.g. call waiting) can be introduced with no change in subscriber equipment.

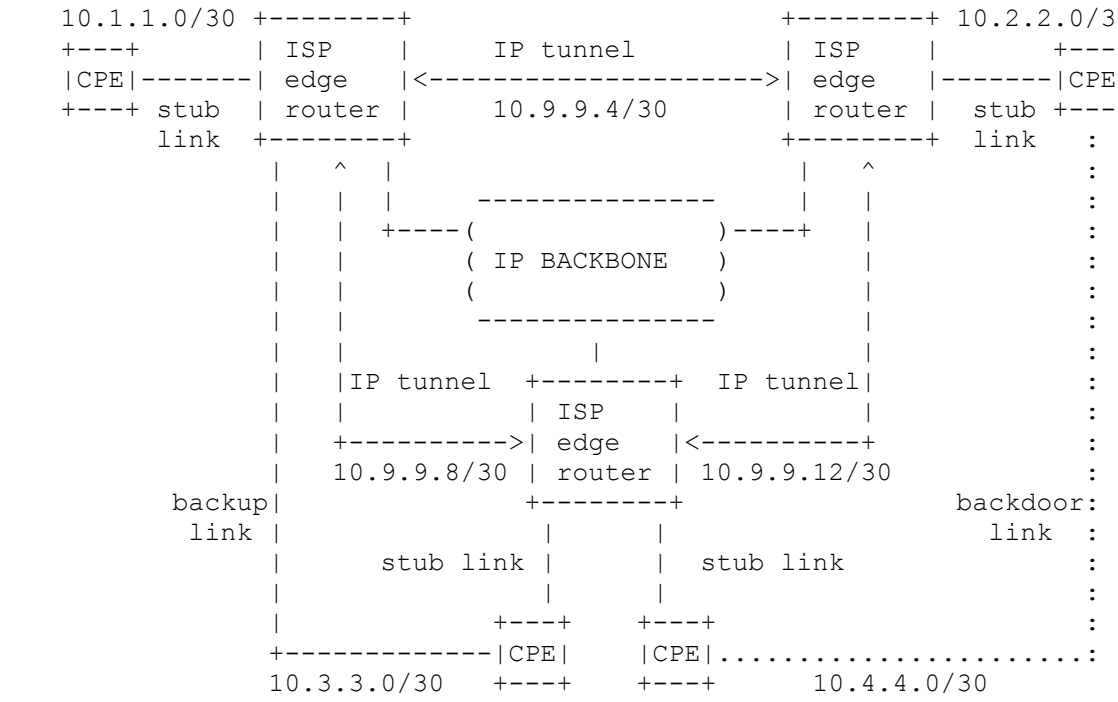

Figure 5.1: Example VPRN

The VPRN type of VPN is in contrast to one where the tunnel mesh extends to the CPE routers, and where the ISP network provides layer 2 connectivity alone. The latter case can be implemented either as a set of VLLs between CPE routers (see section 4.0), in which case the ISP network provides a set of layer 2 point-to-point links, or as a VPLS (see section 7.0), in which case the ISP network is used to emulate a multiaccess LAN segment. With these scenarios a customer may have more flexibility (e.g. any IGP or any protocol can be run across all customer sites) but this usually comes at the expense of a more complex configuration for the customer. Thus, depending on customer requirements, a VPRN or a VPLS may be the more appropriate solution.

Because a VPRN carries out forwarding at the network layer, a single VPRN only directly supports a single network layer protocol. For multiprotocol support, a separate VPRN for each network layer protocol could be used, or one protocol could be tunneled over another (e.g. non-IP protocols tunneled over an IP VPRN) or alternatively the ISP network could be used to provide layer 2

connectivity only, such as with a VPLS as mentioned above.

The issues to be addressed for VPRNs include initial configuration, determination by an ISP edge router of the set of links that are in each VPRN, and of the set of other routers that have members in the VPRN, determination by an ISP edge router of the set of IP address prefixes reachable via each stub link, determination by a CPE router of the set of IP address prefixes to be forwarded to an ISP edge router, the mechanism used to disseminate stub reachability information to the correct set of ISP routers, and the establishment and use of the tunnels used to carry the data traffic. Note also that, although discussed first for VPRNs, many of these issues also apply to the VPLS scenario described later, with the network layer addresses being replaced by link layer addresses.

Note that VPRN operation is decoupled from the mechanisms used by the customer sites to access the Internet. A typical scenario would be for the ISP edge router to be used to provide both VPRN and Internet connectivity to a customer site. In this case the CPE router just has a default route pointing to the ISP edge router, with the latter being responsible for steering private traffic to the VPRN, and other traffic to the Internet, and providing firewall functionality between the two domains. Alternatively a customer site could have Internet connectivity via an ISP router not involved in the VPRN, or even via a different ISP. In this case the CPE device is responsible for splitting the traffic into the two domains and providing firewall functionality.

A. Topology

The topology of a VPRN may consist of a full mesh of tunnels between each VPRN node, or may be an arbitrary topology, such as a set of remote offices connected to the nearest regional site, with these regional sites connected together via a full or partial mesh. With VPRNs using IP tunnels there is much less cost assumed with full meshing than in cases where physical resources (e.g. a leased line) must be allocated for each connected pair of sites, or where the tunneling method requires resources to be allocated in the devices used to interconnect the edge routers (e.g Frame Relay DLCIs). A full mesh topology yields optimal routing, since it precludes the need for traffic between two sites to traverse through a third. Another attraction of a full mesh is that there is no need to configure topology information for the VPRN. Instead, given the member routers of a VPRN, the topology is implicit. If the number of ISP edge routers in a VPRN is very large, however, a full mesh topology may not be appropriate, due to the scaling issues involved, for example, the growth in the number of tunnels needed between sites, (which for n sites is n(n-1)/2), or the number of routing

peers per router. Network policy may also lead to non full mesh topologies, for example an administrator may wish to set up the topology so that traffic between two remote sites passes through a central site, rather than go directly between the remote sites. It is also necessary to deal with the scenario where there is only partial connectivity across the IP backbone under certain error conditions (e.g. A can reach B, and B can reach C, but A cannot reach C directly), which can occur if policy routing is being used.

For a network-based VPRN, it is assumed that each customer site CPE router connects to an ISP edge router through one or more point-to-point stub links (e.g. leased lines, ATM or Frame Relay connections). The ISP routers are responsible for learning and disseminating reachability information amongst themselves. The CPE routers must learn the set of destinations reachable via each stub link, though this may be as simple as a default route.

The stub links may either be dedicated links, set up via provisioning, or may be dynamic links set up on demand, for example using PPP, voluntary tunneling (see section 6.2), or ATM signalling. With dynamic links it is necessary to authenticate the subscriber, and determine the authorized resources that the subscriber can access (e.g. which VPRNs the subscriber may join). Other than the way the subscriber is initially bound to the VPRN, (and this process may involve extra considerations such as dynamic IP address assignment), the subsequent VPRN mechanisms and services can be used for both types of subscribers in the same way.

B. Addressing

The addressing used within a VPRN may have no relation to the addressing used on the IP backbone over which the VPRN is instantiated. In particular non-unique private IP addressing may be used [Rekhter1]. Multiple VPRNs may be instantiated over the same set of physical devices, and they may use the same or overlapping address spaces.

C. Forwarding

For a VPRN the tunnel mesh forms an overlay network operating over an IP backbone. Within each of the ISP edge routers there must be VPN specific forwarding state to forward packets received from stub links ('ingress traffic') to the appropriate next hop router, and to forward packets received from the core ('egress traffic') to the appropriate stub link. For cases where an ISP edge router supports multiple stub links belonging to the same VPRN, the tunnels can, as a local matter, either terminate on the edge router, or on a stub link. In the former case a VPN specific forwarding table is needed for

egress traffic, in the latter case it is not. A VPN specific forwarding table is generally needed in the ingress direction, in order to direct traffic received on a stub link onto the correct IP tunnel towards the core.

Also since a VPRN operates at the internetwork layer, the IP packets sent over a tunnel will have their TTL field decremented in the normal manner, preventing packets circulating indefinitely in the event of a routing loop within the VPRN.

D. Multiple concurrent VPRN connectivity

Note also that a single customer site may belong concurrently to multiple VPRNs and may want to transmit traffic both onto one or more VPRNs and to the default Internet, over the same stub link. There are a number of possible approaches to this problem, but these are outside the scope of this document.

5.1 VPRN related work

VPRN requirements and mechanisms have been discussed previously in a number of different documents. One of the first was [Heinanen2], which showed how the same VPN functionality can be implemented over both MPLS and non-MPLS networks. Some others are briefly discussed below.

There are two main variants as regards the mechanisms used to provide VPRN membership and reachability functionality, - overlay and piggybacking. These are discussed in greater detail in sections 5.2.2, 5.2.3 and 5.2.4 below. An example of the overlay model is described in [Muthukrishnan], which discusses the provision of VPRN functionality by means of a separate per-VPN routing protocol instance and route and forwarding table instantiation, otherwise known as virtual routing. Each VPN routing instance is isolated from any other VPN routing instance, and from the routing used across the backbone. As a result any routing protocol (e.g. OSPF, RIP2, IS-IS) can be run with any VPRN, independently of the routing protocols used in other VPRNs, or in the backbone itself. The VPN model described in [Casey1] is also an overlay VPRN model using virtual routing. That document is specifically geared towards the provision of VPRN functionality over MPLS backbones, and it describes how VPRN membership dissemination can be automated over an MPLS backbone, by performing VPN neighbour discovery over the base MPLS tunnel mesh. [Casey2] extends the virtual routing model to include VPN areas, and VPN border routers which route between VPN areas. VPN areas may be defined for administrative or technical reasons, such as different underlying network infrastructures (e.g. ATM, MPLS, IP).

In contrast [Rosen1] describes the provision of VPN functionality using a piggybacking approach for membership and reachability dissemination, with this information being piggybacked in BGP. VPNs are constructed using BGP policies, which are used to control which sites can communicate with each other. [Li2] also uses BGP for piggybacking membership information, and piggybacks reachability information on the protocol used to establish MPLS LSPs (LDP or RSVP). Unlike the other proposals, however, this proposal requires the participation on the CPE router to implement the VPN functionality.

5.2 VPRN Generic Requirements

There are a number of common requirements which any network-based VPRN solution must address, and there are a number of different mechanisms that can be used to meet these requirements. These generic issues are

- The use of a globally unique VPN identifier in order to be able to refer to a particular VPN.

- VPRN membership determination. An edge router must learn of the local stub links that are in each VPRN, and must learn of the set of other routers that have members in that VPRN.

- Stub link reachability information. An edge router must learn the set of addresses and address prefixes reachable via each stub link.

- Intra-VPRN reachability information. Once an edge router has determined the set of address prefixes associated with each of its stub links, then this information must be disseminated to each other edge router in the VPRN.

- Tunneling mechanism. An edge router must construct the necessary tunnels to other routers that have members in the VPRN, and must perform the encapsulation and decapsulation necessary to send and receive packets over the tunnels.

5.2.1 VPN Identifier

The IETF [Fox] and the ATM Forum [Petri] have standardized on a single format for a globally unique identifier used to identify a VPN - a VPN-ID. Only the format of the VPN-ID has been defined, not its semantics or usage. The aim is to allow its use for a wide variety of purposes, and to allow the same identifier to used in with different technologies and mechanisms. For example a VPN-ID can be included in a MIB to to identify a VPN for management purposes. A

VPN-ID can be used in a control plane protocol, for example to bind a tunnel to a VPN at tunnel establishment time. All packets that traverse the tunnel are then implicitly associated with the identified VPN. A VPN-ID can be used in a data plane encapsulation, to allow for an explicit per-packet identification of the VPN associated with the packet. If a VPN is implemented using different technologies (e.g IP and ATM) in a network, the same identifier can be used to identify the VPN across the different technologies. Also if a VPN spans multiple administrative domains the same identifier can be used everywhere.

Most of the VPN schemes developed (e.g. [Muthukrishnan], [Jamieson], [Casey1], [Li2]) require the use of a VPN-ID that is carried in control and/or data packets, which is used to associate the packet with a particular VPN. Although the use of a VPN-ID in this manner is very common, it is not universal. [Rosen1] describes a scheme where there is no protocol field used to identify a VPN in this manner. In this scheme the VPNs as understood by a user, are administrative constructs, built using BGP policies. There are a number of attributes associated with VPN routes, such as a route distinguisher, and origin and target "VPN", that are used by the underlying protocol mechanisms for disambiguation and scoping, and these are also used by the BGP policy mechanism in the construction of VPNs, but there is nothing corresponding with the VPN-ID as used in the other documents.

Note also that [Grossman] defines a multiprotocol encapsulation for use over ATM AAL5 that uses the standard VPN-ID format.

5.2.2 VPN Membership Information Configuration and Dissemination

In order to establish a VPRN, or to insert new customer sites into an established VPRN, an ISP edge router must determine which stub links are associated with which VPRN. For static links (e.g. an ATM VCC) this information must be configured into the edge router, since the edge router cannot infer such bindings by itself. A management information base (MIB) allowing for bindings between local stub links and VPN identities is one solution.

For subscribers that attach to the network dynamically (e.g. using PPP or voluntary tunneling) it is possible to make the association between stub link and VPRN as part of the end user authentication processing that must occur with such dynamic links. For example the VPRN to which a user is to be bound may be derived from the domain name the used as part of PPP authentication. If the user is successfully authenticated (e.g. using a Radius server), then the newly created dynamic link can be bound to the correct VPRN. Note that static configuration information is still needed, for example to

maintain the list of authorized subscribers for each VPRN, but the location of this static information could be an external authentication server rather than on an ISP edge router. Whether the link was statically or dynamically created, a VPN-ID can be associated with that link to signify to which VPRN it is bound.

After learning which stub links are bound to which VPRN, each edge router must learn either the identity of, or, at least, the route 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. The problem of VPRN member dissemination between participating edge routers, can be solved in a variety of ways, discussed below.

A. Directory Lookup:

The members of a particular VPRN, that is, the identity of the edge routers supporting stub links in the VPRN, and the set of static stub links bound to the VPRN per edge router, could be configured into a directory, which edge routers could query, using some defined mechanism (e.g. LDAP), upon startup.

Using a directory allows either a full mesh topology or an arbitrary topology to be configured. For a full mesh, the full list of member routers in a VPRN is distributed everywhere. For an arbitrary topology, different routers may receive different member lists.

Using a directory 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 is 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, and also to learn of sites removed from a 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 the

identity of each of the static stub links bound to the VPRN. Like the use of a directory, this mechanism allows both full mesh and arbitrary topologies to be configured. Another mechanism using a centralized management system is to use a policy server and use the Common Open Policy Service (COPS) protocol [Boyle] to distribute VPRN membership and policy information, such as the tunnel attributes to use when establishing a tunnel, as described in [MacRae].

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 across the IP backbone, 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 was 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.

Using this method all the routers in the network will have the same view of the VPRN membership information, and so a full mesh topology is easily supported. Supporting an arbitrary topology is more difficult, however, since some form of pruning would seem to be needed.

The advantage of the piggybacking scheme is that it allows for 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. On the other hand, significant administrative complexity may be required to configure scoping mechanisms so as to both permit and constrain the dissemination of the piggybacked advertisements, and in itself this may be quite a configuration burden.

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. Note that, in practice, there will almost always be some centralized directory or management system which will maintain VPRN membership information, such as the set of edge routers that are allowed to support a certain VPRN, the bindings of static stub links to VPRNs, or authentication and authorization information for users that access the network via dynamics links. This information needs to be configured and stored in some form of database, so that the additional steps needed to facilitate the configuration of such information into edge routers, and/or, facilitate edge router access to such information, may not be excessively onerous.

5.2.3 Stub Link Reachability Information

There are two aspects to stub site reachability - the means by which VPRN edge routers determine the set of VPRN addresses and address prefixes reachable at each stub site, and the means by which the CPE routers learn the destinations reachable via each stub link. A number of common scenarios are outlined below. In each case the information needed by the ISP edge router is the same - the set of VPRN addresses reachable at the customer site, but the information needed by the CPE router differs.

1. The CPE router is connected via one link to an ISP edge router, which provides both VPRN and Internet connectivity.

This is the simplest case for the CPE router, as it just needs a default route pointing to the ISP edge router.

2. The CPE router is connected via one link to an ISP edge router, which provides VPRN, but not Internet, connectivity.

The CPE router must know the set of non-local VPRN destinations reachable via that link. This may be a single prefix, or may be a number of disjoint prefixes. The CPE router may be either statically configured with this information, or may learn it dynamically by running an instance of an IGP. For simplicity it is assumed that the IGP used for this purpose is RIP, though it could be any IGP. The ISP edge router will inject into this instance of RIP the VRPN routes which it learns by means of one of the intra-VPRN reachability mechanisms described in section 5.2.4. Note that the instance of RIP run to the CPE, and any instance of a routing protocol used to learn intra-VPRN reachability (even if also RIP) are separate, with the ISP edge router redistributing the routes from one instance to another.

3. The CPE router is multihomed to the ISP network, which provides VPRN connectivity.

In this case all the ISP edge routers could advertise the same VPRN routes to the CPE router, which then sees all VPRN prefixes equally reachable via all links. More specific route redistribution is also possible, whereby each ISP edge router advertises a different set of prefixes to the CPE router.

4. The CPE router is connected to the ISP network, which provides VPRN connectivity, but also has a backdoor link to another customer site

In this case the ISP edge router will advertise VPRN routes as in case 2 to the CPE device. However now the same destination is reachable via both the ISP edge router and via the backdoor link. If the CPE routers connected to the backdoor link are running the customer's IGP, then the backdoor link may always be the favoured link as it will appear an an 'internal' path, whereas the destination as injected via the ISP edge router will appear as an 'external' path (to the customer's IGP). To avoid this problem, assuming that the customer wants the traffic to traverse the ISP network, then a separate instance of RIP should be run between the CPE routers at both ends of the backdoor link, in the same manner as an instance of RIP is run on a stub or backup link between a CPE router and an ISP edge router. This will then also make the backdoor link appear as an external path, and by adjusting the link costs appropriately, the ISP path can always be favoured, unless it goes down, when the backdoor link is then used.

The description of the above scenarios covers what reachability information is needed by the ISP edge routers and the CPE routers, and discusses some of the mechanisms used to convey this information. The sections below look at these mechanisms in more detail.

A. Routing Protocol Instance:

A routing protocol can be run between the CPE edge router and the ISP edge router to exchange reachability information. This allows an ISP edge router to learn the VPRN prefixes reachable at a customer site, and also allows a CPE router to learn the destinations reachable via the provider network.

The extent of the routing domain for this protocol instance is generally just the ISP edge router and the CPE router although if the customer site is also running the same protocol as its IGP, then the domain may extend into customer site. If the customer site is running a different routing protocol then the CPE router redistributes the routes between the instance running to the ISP edge router, and the instance running into the customer site.

Given the typically restricted scope of this routing instance, a simple protocol will generally suffice. RIPv2 [Malkin] is likely to be the most common protocol used, though any routing protocol, such as OSPF [Moy], or BGP-4 [Rekhter2] run in internal mode (IBGP), could also be used.

Note that the instance of the stub link routing protocol is different from any instance of a routing protocol used for intra-VPRN reachability. For example, if the ISP edge router uses routing protocol piggybacking to disseminate VPRN membership and reachability information across the core, then it may redistribute suitably labeled routes from the CPE routing instance to the core routing instance. The routing protocols used for each instance are decoupled, and any suitable protocol can be used in each case. There is no requirement that the same protocol, or even the same stub link reachability information gathering mechanism, be run between each CPE router and associated ISP edge router in a particular VPRN, since this is a purely local matter.

This decoupling allows ISPs to deploy a common (across all VPRNs) intra-VPRN reachability mechanism, and a common stub link reachability mechanism, with these mechanisms isolated both from each other, and from the particular IGP used in a customer network. In the first case, due to the IGP-IGP boundary implemented on the ISP edge router, the ISP can insulate the intra-VPRN reachability mechanism from misbehaving stub link protocol instances. In the second case the ISP is not required to be aware of the particular IGP running in a customer site. Other scenarios are possible, where the ISP edge routers are running a routing protocol in the same instance as the customer's IGP, but are unlikely to be practical, since it defeats the purpose of a VPRN simplifying CPE router configuration. In cases where a customer wishes to run an IGP across multiple sites,

a VPLS solution is more suitable.

Note that if a particular customer site concurrently belongs to multiple VPRNs (or wishes to concurrently communicate with both a VPRN and the Internet), then the ISP edge router must have some means of unambiguously mapping stub link address prefixes to particular VPRNs. A simple way is to have multiple stub links, one per VPRN. It is also possible to run multiple VPRNs over one stub link. 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 VPRNs, or by tagging the routing advertisements from the CPE router with the appropriate VPN identifier. For example if MPLS was being used to convey stub link reachability information, different MPLS labels would be used to differentiate the disjoint prefixes assigned to particular VPRNs. 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 via the VPRN edge router, which may be appropriate for small customer sites, typically containing either a single host, or a single subnet. Address allocation can be carried out using protocols such as PPP or DHCP, with, for example, the edge router acting as a Radius client and retrieving the customer's IP address to use from a Radius server, or acting as a DHCP relay and examining the DHCP reply message as it is relayed to the customer site. In this manner the edge router can build up a table of stub link reachability information. Although these address assignment mechanisms are typically used to assign an address to a single host, some vendors have added extensions whereby an address prefix can be assigned, with, in some cases, the CPE device acting as a "mini-DHCP" server and assigning addresses for the hosts in the customer site.

Note that with these schemes it is the responsibility of the address allocation server to ensure that each site in the VPN received a disjoint address space. Note also that an ISP would typically only use this mechanism for small stub sites, which are unlikely to have backdoor links.

D. MPLS Label Distribution Protocol:

In cases where the CPE 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 a routing protocol per se, it may be useful to extend it for this particular constrained scenario.

5.2.4 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 edge router, 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 VPRN 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 in section 5.2.2 also 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, particularly when arbitrary topologies are used, 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 instance 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. Both full-mesh and arbitrary VPRN

topologies can be easily supported, since the routing protocol itself can run over any topology. The intra-VPRN 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 either with the IGP of any customer site or with the routing protocols operated by the ISPs in the IP backbone. Depending on the size and scale of the VPRNs to be supported either a simple protocol like RIPv2 [Malkin] or a more sophisticated protocol like OSPF [Moy] could be used. Because the intra-VPRN routing protocol operates as an overlay over the IP backbone it is wholly transparent to any intermediate routers, and to any edge routers not within the VPRN. This also implies that such routing information can remain opaque to such routers, which may be a necessary security requirements in some cases. Also note that if the routing protocol runs directly over the same tunnels as the data traffic, then it will inherit the same level of security as that afforded the data traffic, for example strong encryption and authentication.

If the tunnels over which an intra-VPRN routing protocol runs are dedicated to a specific VPN (e.g. a different multiplexing field is used for each VPN) then no changes are needed to the routing protocol itself. On the other hand if shared tunnels are used, then it is necessary to extend the routing protocol to allow a VPN-ID field to be included in routing update packets, to allow sets of prefixes to be associated with a particular VPN.

D. Link Reachability Protocol

Given a full mesh topology, each edge router could run a link reachability protocol, for instance some variation of 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. If VPRN membership information has already been distributed to an edge router, then the neighbour discovery aspects of a traditional routing protocol are not needed, as the set of neighbours is already known. TCP connections can be used to interconnect the neighbours, to provide reliability. This approach may reduce the processing burden of running routing protocol instances 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.

Another approach to a link reachability protocol would be to base it on IBGP. The problem that needs to be solved by a link reachability protocol is very similar to that solved by IBGP - conveying address

prefixes reliably between edge routers.

Using a link reachability protocol it is straightforward to support a full mesh topology - each edge router conveys its own local reachability information to all other routers, but does not redistribute information received from any other router. However once an arbitrary topology needs to be supported, the link reachability protocol needs to develop into a full routing protocol, due to the need to implement mechanisms to avoid loops, and there would seem little benefit in reinventing another routing protocol to deal with this. Some reasons why partially connected meshes may be needed even in a tunneled environment are discussed in section 5.0A.

E. Piggybacking in IP Backbone Routing Protocols:

As with VPRN 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 extract this information from received route advertisements in the same way as they 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.

5.2.5 Tunneling Mechanisms

Once VPRN membership information has been disseminated, the tunnels comprising the VPRN core can be constructed.

One approach to setting up the tunnel mesh is to use point-to-point IP tunnels, and the requirements and issues for such tunnels have

been discussed in section 3.0. For example while tunnel establishment 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 signalling protocol to allow two nodes to construct a tunnel to each other knowing only each other's identity.

Another approach is to use the multipoint to point 'tunnels' provided by MPLS. As noted in [Heinanen1], 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.

One attraction of MPLS as a tunneling mechanism is that it may require less processing within each edge router than alternative tunneling mechanisms. This is 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). However there are other potential security concerns with MPLS. There is no direct support for security features such as authentication, confidentiality, and non-repudiation and the trust model for MPLS means that intermediate routers, (which may belong to different administrative domains), through which membership and prefix reachability information is conveyed, must be trusted, not just the edge routers themselves.

5.3 Multihomed Stub Routers

The discussion thus far has implicitly assumed that stub routers are connected to one and only one VPRN edge router. In general, this restriction should be capable of being relaxed without any change to VPRN operation, given general market interest in multihoming for reliability and other reasons. In particular, in cases where the stub router supports multiple redundant links, with only one operational at any given time, with the links connected either to the same VPRN edge router, or to two or more different VPRN edge routers, then the stub link reachability mechanisms will both discover the loss of an active link, and the activation of a backup link. In the former situation, the previously connected VPRN edge router will cease advertising reachability to the stub node, while the VPRN edge router with the now active link will begin advertising reachability, hence restoring connectivity.

An alternative scenario is where the stub node supports multiple active links, using some form of load sharing algorithm. In such a case, multiple VPRN edge routers may have active paths to the stub node, and may so advertise across the VPRN. This scenario should not cause any problem with reachability across the VPRN providing that the intra-VPRN reachability mechanism can accommodate multiple paths to the same prefix, and has the appropriate mechanisms to preclude looping - for instance, distance vector metrics associated with each advertised prefix.

5.4 Multicast Support

Multicast and broadcast traffic can be supported across VPRN either by edge replication or by native multicast support in the backbone. These two cases are discussed below.

5.4.1 Edge Replication

This is where each VPRN edge router replicates multicast traffic for transmission across each link in the VPRN. Note that this is the same operation that would be performed by CPE routers terminating actual physical links or dedicated connections. As with CPE routers, multicast routing protocols could also be run on each VPRN edge router to determine the distribution tree for multicast traffic and hence reduce unnecessary flood traffic. This could be done by running instances of standard multicast routing protocols, e.g. Protocol Independent Multicast (PIM) [Estrin] or Distance Vector Multicast Routing Protocol (DVMRP) [Waitzman], on and between each VPRN edge router, through the VPRN tunnels, in the same way that unicast routing protocols might be run at each VPRN edge router to determine intra-VPN unicast reachability, as discussed in section 5.2.4. Alternatively, if a link reachability protocol was run across the VPRN tunnels for intra-VPRN reachability, then this could also be augmented to allow VPRN edge routers to indicate both the particular multicast groups requested for reception at each edge node, and also the multicast sources at each edge site.

In either case, there would need to be some mechanism to allow for the VPRN edge routers to determine which particular multicast groups were requested at each site and which sources were present at each site. How this could be done would, in general, be a function of the capabilities of the CPE stub routers at each site. If these run multicast routing protocols, then they can interact directly with the equivalent protocols at each VPRN edge router. If the CPE device does not run a multicast routing protocol, then in the absence of IGMP-proxying [Fenner] the customer site would be limited to a single subnet connected to the VPRN edge router via a bridging device, as the scope of an IGMP message is limited to a single subnet. However

using IGMP-proxying the CPE router can engage in multicast forwarding without running a multicast routing protocol, in constrained topologies. On its interfaces into the customer site it performs the router functions of IGMP, and on its interface to the VPRN edge router it performs the host functions of IGMP.

5.4.2 Native Multicast Support

This is where VPRN edge routers map intra-VPN multicast traffic onto a native IP multicast distribution mechanism across the backbone. Note that the latter is not synonymous with the use of native multicast mechanisms per se - e.g. the use of IP multicast across the backbone - since intra-VPN multicast has the same requirements for isolation from general backbone traffic as intra-VPRN unicast traffic. Currently the only IP tunneling mechanism that has native support for multicast is MPLS. On the other hand, while MPLS supports native transport of IP multicast packets, additional mechanisms would be needed to leverage these mechanisms for the support of intra-VPN multicast.

For instance, each VPRN router could prefix multicast group addresses within each VPRN with the VPN-ID of that VPRN and then redistribute these, essentially treating this VPN-ID/intra-VPRN multicast address tuple as a normal multicast address, within the backbone multicast routing protocols, as with the case of unicast reachability, as discussed previously. The MPLS multicast label distribution mechanisms could then be used to set up the appropriate multicast LSPs to interconnect those sites within each VPRN supporting particular multicast group addresses. Note, however, that this would require each of the intermediate LSRs to not only be aware of each intra-VPRN multicast group, but also to have the capability of interpreting these modified advertisements. Alternatively, mechanisms could be defined to map intra-VPRN multicast groups into backbone multicast groups.

Other IP tunneling mechanisms do not have native multicast support. It may prove feasible to extend such tunneling mechanisms by allocating IP multicast group addresses to the VPRN as a whole and hence distributing intra-VPRN multicast traffic encapsulated within backbone multicast packets. Edge VPRN routers could filter out unwanted multicast groups. Alternatively, mechanisms could also be defined to allow for allocation of backbone multicast group addresses for particular intra-VPRN multicast groups, and to then utilize these, through backbone multicast protocols, as discussed above, to limit forwarding of intra-VPRN multicast traffic only to those nodes within the group.

A particular issue with the use of native multicast support is the

provision of security for such multicast traffic. Unlike the case of edge replication, which inherits the security characteristics of the underlying tunnel, native multicast mechanisms will need to use some form of secure multicast mechanism. The development of architectures and solutions for secure multicast is an active research area, for example see [Wallner] and [Hardjono]. The Secure Multicast Group (SMuG) of the IRTF has been set up to develop prototype solutions, which would then be passed to the IETF IPSec working group for standardization. However considerably more development is needed before scalable secure native multicast mechanisms can be generally deployed.

5.5 Recommendations

The various proposals that have been developed to support some form of VPRN functionality, can be broadly classified into two groups - those that utilize the router piggybacking approach for distributing VPN membership and/or reachability information ([Rosen1], [Li2]) and those that use the virtual routing approach ([Muthukrishnan], [Casey1]). In some cases the mechanisms described rely on the characteristics of a particular infrastructure (e.g. MPLS) rather than just IP.

Within the context of the virtual routing approach it may be useful to develop a membership distribution protocol based on a directory or MIB. When combined with the protocol extensions for IP tunneling protocols outlined in section 3.2, this would then provide the basis for a complete set of protocols and mechanisms that support interoperable VPRNs that span multiple administrations over an IP backbone. Note that the other major pieces of functionality needed - the learning and distribution of customer reachability information, can be performed by instances of standard routing protocols, without the need for any protocol extensions.

Also for the constrained case of a full mesh topology, the usefulness of developing a link reachability protocol could be examined, however the limitations and scalability issues associated with this topology may not make it worthwhile to develop something specific for this case, as standard routing will just work.

Extending routing protocols to allow a VPN-ID to carried in routing update packets could also be examined, but is not necessary if VPN specific tunnels are used.

6.0 VPN Types: Virtual Private Dial Networks

A virtual private dial network (VPDN) allows for a remote user to connect on demand through an ad hoc tunnel into another site. The

user is connected to a public IP network via a dial-up PSTN or ISDN dial-up link, and user packets are tunneled across the public network to the desired site, giving the impression to the user of being 'directly' connected into that site. A key characteristic of such ad hoc connections is the need for user authentication as a prime requirement, since anyone could potentially attempt to gain access to such a site using a switched dial network.

Today many corporate networks allow access to remote users through dial connections made through the PSTN, with users setting up PPP connections across an access network to a Network Access Server (NAS), at which point the PPP sessions are authenticated using AAA systems running such standard protocols as Radius [Rigney]. Given the pervasive deployment of such systems, any VPDN system must in practice allow for the near transparent re-use of such existing systems.

The IETF have developed the Layer 2 Tunneling Protocol (L2TP) [Townsley] which allows for the extension of of user PPP sessions from an L2TP access concentrator (LAC)to a remote L2TP network server (LNS). The L2TP protocol itself was based on two earlier protocols, the Layer 2 Forwarding protocol (L2F) [Valencia], and the Point-to-Point Tunneling Protocol (PPTP) [Hamzeh], and this is reflected in the two quite different scenarios for which L2TP can be used - compulsory tunneling and voluntary tunneling, discussed further below in sections 6.2 and 6.3.

This document focuses on the use of L2TP over an IP network (using UDP), but L2TP may also be run directly over other protocols such as ATM or Frame Relay. Issues specifically related to running L2TP over non-IP networks, such as how to secure such tunnels, are not addressed here.

6.1 L2TP protocol characteristics

This section looks at the characteristics of the L2TP tunneling protocol using the categories outlined in section 3.0.

6.1.1 Multiplexing

L2TP has inherent support for the multiplexing of multiple calls from different users over a single link. Between the same two IP endpoints, there can be multiple L2TP tunnels, as identified by a tunnel-id, and multiple sessions within a tunnel, as identified by a session-id.

6.1.2 Signalling

This is supported via the inbuilt control connection protocol, allowing both tunnels and sessions to be established dynamically.

6.1.3 Data Security

By allowing for the transparent extension of PPP from the user, through the LAC to the LNS, L2TP allows for the use of whatever security mechanisms, with respect to both connection set up, and data transfer, may be used with normal PPP connections. However this does not provide security for the L2TP control protocol itself. In this case L2TP could be further secured by running it in combination with IPSec through IP backbones [Patel1], [Srisuresh2], or related mechanisms on non-IP backbones [Calhoun2].

The interaction of L2TP with AAA systems for user authentication and authorization is a function of the specific means by which L2TP is used, and the nature of the devices supporting the LAC and the LNS. These issues are discussed in depth in [Aboba1].

The means by which the host determines the correct LAC to connect to, and the means by which the LAC determines which users to further tunnel, and the LNS parameters associated with each user, are outside the scope of the operation of VPDN, but may be addressed, for instance, by evolving Internet roaming specifications [Aboba2].

6.1.4 Multiprotocol Transport

L2TP transports PPP packets (and only PPP packets) and thus can be used to carry multiprotocol traffic since PPP itself is multiprotocol.

6.1.5 Sequencing

L2TP supports sequenced delivery of packets. This is a capability that be negotiated at session establishment, and can be turned on and off by an LNS during a session. The sequence number field in L2TP can also be used to provide an indication of dropped packets, which is needed by various PPP compression algorithms to operate correctly. If no compression is in use, and the LNS determines that the protocols in use (as evidenced by the PPP NCP negotiations) can deal with out of sequence packets (e.g. IP), then it may disable the use of sequencing.

6.1.6 Tunnel Maintenance

A keepalive protocol is used by L2TP in order to allow it to distinguish between a tunnel outage and prolonged periods of tunnel inactivity.

6.1.7 Large MTUs

L2TP itself has no inbuilt support for a segmentation and reassembly capability, but when run over UDP/IP IP fragmentation will take place if necessary. Note that a LAC or LNS may adjust the MRU negotiated via PPP in order to preclude fragmentation, if it has knowledge of the MTU used on the path between LAC and LNS. To this end, there is a proposal to allow the use of MTU discovery for cases where the L2TP tunnel transports IP frames [Shea].

6.1.8 Tunnel Overhead

L2TP as used over IP networks runs over UDP and must be used to carry PPP traffic. This results in a significant amount of overhead, both in the data plane with UDP, L2TP and PPP headers, and also in the control plane, with the L2TP and PPP control protocols. This is discussed further in section 6.2

6.1.9 Flow and Congestion Control

L2TP supports flow and congestion control mechanisms for the control protocol, but not for data traffic. See section 3.1.9 for more details.

6.1.10 QoS / Traffic Management

An L2TP header contains a 1-bit priority field, which can be set for packets that may need preferential treatment (e.g. keepalives) during local queuing and transmission. Also by transparently extending PPP, L2TP has inherent support for such PPP mechanisms as multi-link PPP [Sklower] and its associated control protocols [Richard], which allow for bandwidth on demand to meet user requirements.

In addition L2TP calls can be mapped into whatever underlying traffic management mechanisms may exist in the network, and there are proposals to allow for requests through L2TP signalling for specific differentiated services behaviors [Calhoun1].

6.1.11 Miscellaneous

Since L2TP is designed to transparently extend PPP, it does not attempt to supplant the normal address assignment mechanisms

associated with PPP. Hence, in general terms the host initiating the PPP session will be assigned an address by the LNS using PPP procedures. This addressing may have no relation to the addressing used for communication between the LAC and LNS. The LNS will also need to support whatever forwarding mechanisms are needed to route traffic to and from the remote host.

6.2 Compulsory Tunneling

Compulsory tunneling refers to the scenario in which a network node - a dial or network access server, for instance - acting as a LAC, extends a PPP session across a backbone using L2TP to a remote LNS, as illustrated below. This operation is transparent to the user initiating the PPP session to the LAC. This allows for the decoupling of the location and/or ownership of the modem pools used to terminate dial calls, from the site to which users are provided access. Support for this scenario was the original intent of the L2F specification, upon which the L2TP specification was based. Note that the diagram below shows access to a corporate network, but other deployment scenarios are possible. For example an ISP might provide Internet access via an LNS.

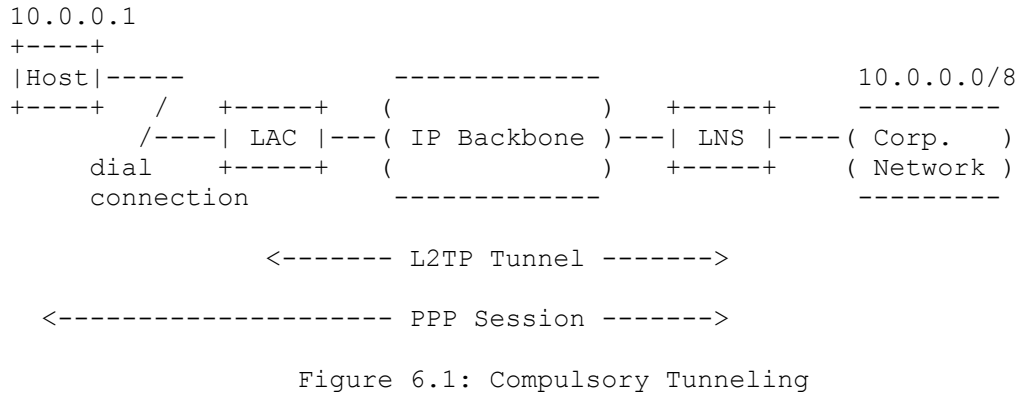

Figure 6.1: Compulsory Tunneling

Compulsory tunneling was originally intended for deployment on dial access servers supporting VPDN or wholesale dial services, allowing for remote dial access through common facilities to an enterprise site, while precluding the need for the enterprise to deploy its own dial servers. Another example of this is where an ISP outsources its own dial connectivity to an access network provider (such as a Local Exchange Carrier (LEC) in the US) removing the need for an ISP to maintain its own dial servers and allowing the LEC to serve multiple ISPs. More recently, compulsory tunneling mechanisms have also been proposed for evolving xDSL services [ADSL1], [ADSL2], which also seek

to leverage the existing AAA infrastructure.

Call routing for compulsory tunnels requires that some aspect of the initial PPP call set up can be used to allow the LAC to determine the identity of the LNS. As noted in [Aboba1], these aspects can include the user identity, as determined through some aspect of the access network, including calling party number, or some attribute of the called party, such as the fully qualified domain name (FQDN) of the CHAP/PAP user name.

It is also possible to chain two L2TP tunnels together, whereby a LAC initiates a tunnel to an intermediate relay device, which acts as an LNS to this first LAC, and acts as a LAC to the final LNS. This may be needed in some cases due to administrative, organizational or regulatory issues pertaining to the split between access network provider, IP backbone provider and enterprise customer.

6.3 Voluntary Tunnels

Voluntary tunneling refers to the case where an individual host connects to a remote site using a tunnel originating on the host, with no involvement from intermediate network nodes, as illustrated below. The PPTP specification, parts of which have been incorporated into L2TP, was based upon a voluntary tunneling model. Note that the diagram below shows access to a corporate network, but other deployment scenarios are possible. For example an ISP might provide Internet access via an LNS.

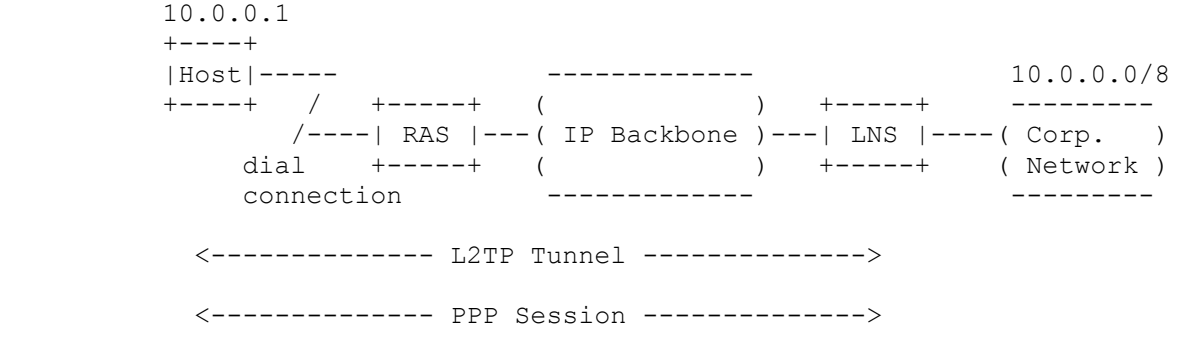

Figure 6.2: Voluntary Tunneling

The L2TP specification has support for voluntary tunneling, insofar as the LAC can be located on a host, not only on a network node. Note that such a host has two IP addresses - one for the LAC-LNS IP tunnel, and another, typically allocated via PPP, for the network to

which the host is connecting. The benefits of using L2TP for voluntary tunneling are that the existing authentication and address assignment mechanisms used by PPP can be reused without modification. For example an LNS could also include a Radius client, and communicate with a Radius server to authenticate a PPP PAP or CHAP exchange, and to retrieve configuration information for the host such as its IP address and a list of DNS servers to use. This information can then be passed to the host via the PPP IPCP protocol.

The above procedure is not without its costs, however. There is considerable overhead with such a protocol stack, particularly when IPSec is also needed for security purposes, and given that the host may be connected via a low-bandwidth dial up link. The overhead consists of both extra headers in the data plane and extra control protocols needed in the control plane. Using L2TP for voluntary tunneling, secured with IPSec, means a web application, for example, would run over the following stack

HTTP/TCP/IP/PPP/L2TP/UDP/ESP/IP/PPP/AHDLC

It is proposed in [Gupta] that IPSec alone be used for voluntary tunnels reducing overhead, using the following stack.

HTTP/TCP/IP/ESP/IP/PPP/AHDLC

In this case IPSec is used in tunnel mode, with the tunnels terminating on IPSec edge devices at the enterprise site. There are two possibilities for the IP addressing of the host. Two IP addresses could be used, in a similar manner to the L2TP case. Alternatively the host can use a single public IP address as the source IP address in both inner and outer IP headers, with the gateway performing NAT before forwarding the traffic to the enterprise network. To other hosts in the enterprise network the host appears to have an 'internal' IP address. Using NAT has some limitations and restrictions, also pointed out in [Gupta].

Another area of potential problems with PPP is due to the fact that the characteristics of a link layer implemented via an L2TP tunnel over an IP backbone are quite different to a link layer run over a serial line, as discussed in the L2TP specification itself. Poorly chosen PPP parameters may lead to frequent resets and timeouts, particularly if compression is in use. This is because an L2TP tunnel may misorder packets, and may silently drop packets, neither of which normally occurs on serial lines. The general packet loss rate could also be significantly higher due to network congestion. Using the sequence number field in an L2TP header addresses the misordering issue, and for cases where the LAC and LNS are coincident with the PPP endpoints, as in voluntary tunneling, the sequence

number field can also be used to detect a dropped packet, and to pass a suitable indication to any compression entity in use, which typically requires such knowledge in order to keep the compression histories in synchronization at both ends. (In fact this is more of an issue with compulsory tunneling since the LAC may have to deliberately issue a corrupted frame to the PPP host, to give an indication of packet loss, and some hardware may not allow this).

If IPSec is used for voluntary tunneling the functions of user authentication and host configuration still need to be carried out, however. A distinction needs to be drawn here between machine authentication and user authentication. 'Two factor' authentication is carried out on the basis of both something the user has, such as a machine or smartcard with a digital certificate, and something the user knows, such as a password. (Another example is getting money from an bank ATM machine - you need a card and a PIN number). Many of the existing legacy schemes currently in use to perform user authentication are asymmetric in nature, and are not supported by IKE. For remote access the most common existing user authentication mechanism is to use PPP between the user and access server, and Radius between the access server and authentication server. The authentication exchanges that occur in this case, e.g. a PAP or CHAP exchange, are asymmetric. Also CHAP supports the ability for the network to reauthenticate the user at any time after the initial session has been established, to ensure that the current user is the same person that initiated the session.

While IKE provides strong support for machine authentication, it has only limited support for any form of user authentication and has no support for asymmetric user authentication. While a user password can be used to derive a key used as a preshared key, this cannot be used with IKE Main Mode in a remote access environment, as the user will not have a fixed IP address, and while Agressive Mode can be used instead, this affords no identity protection. To this end there have been a number of proposals to allow for support of legacy asymmetric user level authentication schemes with IPSec. [Pereira1] defines a new ISAKMP message exchange - the transaction exchange - which allows for both Request/Reply and Set/Acknowledge message sequences. This draft also defines attributes that can be used for client IP stack configuration. [Pereira2] and [Litvin] describe mechanisms that use the transaction message exchange, or a series of such exchanges, carried out between the IKE Phase 1 and Phase 2 exchanges, to perform user authentication. A different approach, that does not extend the IKE protocol itself, is described in [Kelly]. With this approach a user establishes a Phase 1 SA with a security gateway and then sets up a Phase 2 SA to the gateway, over which an existing authentication protocol is run. The gateway acts as a proxy and relays the protocol messages to an authentication server.

In addition there have also been proposals to allow the remote host to be configured with an IP address and other configuration information over IPSec. For example [Patel2] describes a method whereby a remote host first establishes a Phase 1 SA with a security gateway and then sets up a Phase 2 SA to the gateway, over which the DHCP protocol is run. The gateway acts as a proxy and relays the protocol messages to the DHCP server. Again, like [Kelly], this proposal does not involve extensions to the IKE protocol itself.

Another aspect of PPP functionality that may need to supported is multiprotocol operation, as there may be a need to carry network layer protocols other than IP, and even to carry link layer protocols (e.g. ethernet) as would be needed to support bridging over IPSec. This is discussed in section 3.1.4.

The methods of supporting legacy user authentication and host configuration capabilities in a remote access environment are currently being discussed in the IPSec working group.

6.3 Networked Host Support

The current PPP based dial model assumes a host directly connected to a connection oriented dial access network. Recent work on new access technologies such a xDSL have attempted to replicate this model [ADSL], so as to allow for the re-use of existing AAA systems. The proliferation of personal computers, printers and other network appliances in homes and small businesses, and the ever lowering costs of networks, however, are increasingly challenging the directly connected host model. Increasingly, most hosts will access the Internet through small, typically Ethernet, local area networks (LANs).

There is hence interest in means of accommodating the existing AAA infrastructure within service providers, whilst also supporting multiple networked hosts at each customer site. The principal complication with this scenario is the need to support the login dialogue, through which the appropriate AAA information is exchanged. A number of proposals have been made to address this scenario:

A. Extension of PPP to Hosts Through L2TP:

A number of proposals (e.g. [ADSL1]) have been made to extend L2TP over Ethernet so that PPP sessions can run from networked hosts out to the network, in much the same manner as a directly attached host.

B. Extension of PPP Directly to Hosts:

There is also a specification for mapping PPP directly onto Ethernet (PPPOE) [Mamakos] which uses a broadcast mechanism to allow hosts to find appropriate access servers with which to connect. Such servers could then further tunnel, if needed, the PPP sessions using L2TP or a similar mechanism.

C. Use of IPSec:

The IPSec based voluntary tunneling mechanisms discussed above can be used either with networked or directly connected hosts.

Note that all of these methods require additional host software to be used, which implements either LAC, PPPOE client or IPSec client functionality.

6.4 Recommendations

The L2TP specification has been finalized and will be widely used for compulsory tunneling. As discussed in section 3.2, defining specific modes of operation for IPSec when used to secure L2TP would be beneficial.

Also, for voluntary tunneling, completing the work needed to provide support for the following areas would be useful

- asymmetric / legacy user authentication (6.3)

- host address assignment and configuration (6.3)

along with any other issues specifically related to the support of remote hosts. Currently as there are many different non-interoperable proprietary solutions in this area.

7.0 VPN Types: Virtual Private LAN Segment

A virtual private LAN segment (VPLS) is the emulation of a LAN segment using Internet facilities. A VPLS can be used to provide what is sometimes known also as a transparent LAN service (TLS), which can be used to interconnect multiple stub CPE nodes, either bridges or routers, in a protocol transparent manner. A VPLS emulates a LAN segment over IP, in the same way as protocols such as LANE [ATMF1] emulate a LAN segment over ATM. The primary benefits of a VPLS are complete protocol transparency, which may be important both for multiprotocol transport and for regulatory reasons in particular service provider contexts.

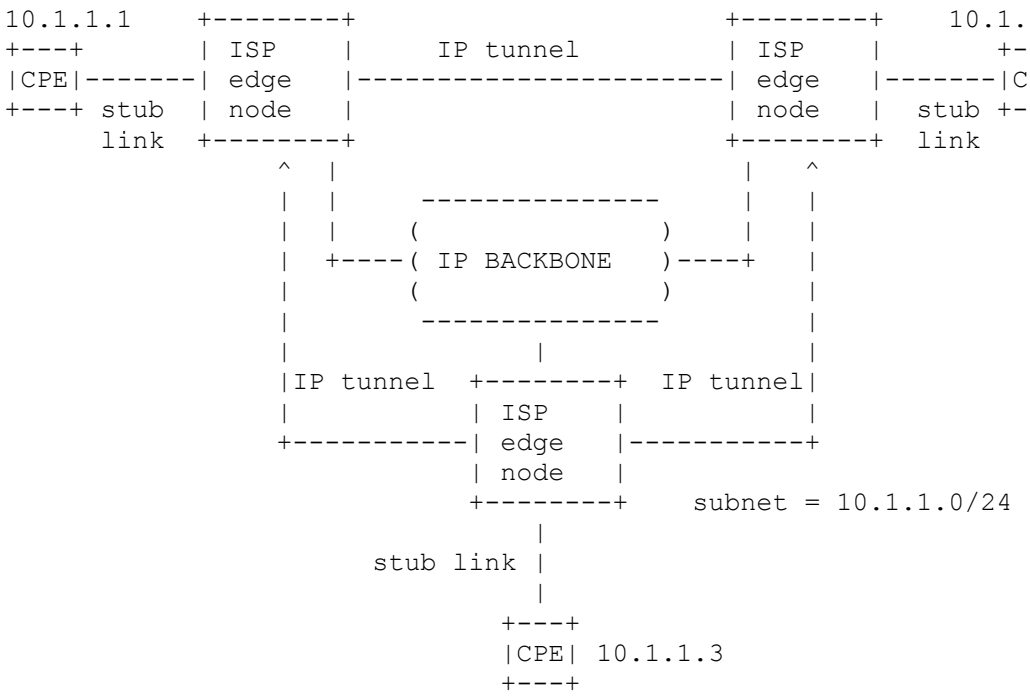

Figure 7.1: Example VPLS

7.1 VPLS Requirements

Topologically and operationally a VPLS can be most easily modelled as being essentially equivalent to a VPRN, except that each VPLS edge node implements link layer bridging rather than network layer forwarding. As such, most of the VPRN tunneling and configuration mechanisms discussed previously can also be used for a VPLS, with the appropriate changes to accommodate link layer, rather than network layer, packets and addressing information. The following sections discuss the primary changes needed in VPRN operation to support VPLSs.

7.1.1 Tunneling Protocols

The tunneling protocols employed within a VPLS can be exactly the same as those used within a VPRN, if the tunneling protocol permits the transport of multiprotocol traffic, and this is assumed below.

7.1.2 Multicast and Broadcast Support

A VPLS needs to have a broadcast capability. This is needed both for broadcast frames, and for link layer packet flooding, where a unicast frame is flooded because the path to the destination link layer address is unknown. The address resolution protocols that run over a bridged network typically use broadcast frames (e.g. ARP). The same set of possible multicast tunneling mechanisms discussed earlier for VPRNs apply also to a VPLS, though the generally more frequent use of broadcast in VPLSs may increase the pressure for native multicast support that reduces, for instance, the burden of replication on VPLS edge nodes.

7.1.3 VPLS Membership Configuration and Topology

The configuration of VPLS membership is analogous to that of VPRNs since this generally requires only knowledge of the local VPN link assignments at any given VPLS edge node, and the identity of, or route to, the other edge nodes in the VPLS; in particular, such configuration is independent of the nature of the forwarding at each VPN edge node. As such, any of the mechanisms for VPN member configuration and dissemination discussed for VPRN configuration can also be applied to VPLS configuration. Also as with VPRNs, the topology of the VPLS could be easily manipulated by controlling the configuration of peer nodes at each VPLS edge node, assuming that the membership dissemination mechanism was such as to permit this. It is likely that typical VPLSs will be fully meshed, however, in order to preclude the need for traffic between two VPLS nodes to transit through another VPLS node, which would then require the use of the spanning tree protocol [IEEE] for loop prevention.

7.1.4 CPE Stub Node Types

A VPLS can support either bridges or routers as a CPE device.

CPE routers would peer transparently across a VPLS with each other without requiring any router peering with any nodes within the VPLS. The same scalability issues that apply to a full mesh topology for VPRNs, apply also in this case, only that now the number of peering routers is potentially greater, since the ISP edge device is no longer acting as an aggregation point.

With CPE bridge devices the broadcast domain encompasses all the CPE sites as well as the VPLS itself. There are significant scalability constraints in this case, due to the need for packet flooding, and the fact that any topology change in the bridged domain is not localized, but is visible throughout the domain. As such this scenario is generally only suited for support of non-routable

protocols.

The nature of the CPE impacts the nature of the encapsulation, addressing, forwarding and reachability protocols within the VPLS, and are discussed separately below.

7.1.5 Stub Link Packet Encapsulation

A. Bridge CPE:

In this case, packets sent to and from the VPLS across stub links are link layer frames, with a suitable access link encapsulation. The most common case is likely to be ethernet frames, using an encapsulation appropriate to the particular access technology, such as ATM, connecting the CPE bridges to the VPLS edge nodes. Such frames are then forwarded at layer 2 onto a tunnel used in the VPLS. As noted previously, this does mandate the use of an IP tunneling protocol which can transport such link layer frames. Note that this does not necessarily mandate, however, the use of a protocol identification field in each tunnel packet, since the nature of the encapsulated traffic (e.g. ethernet frames) could be indicated at tunnel setup.

B. Router CPE:

In this case, typically, CPE routers send link layer packets to and from the VPLS across stub links, destined to the link layer addresses of their peer CPE routers. Other types of encapsulations may also prove feasible in such a case, however, since the relatively constrained addressing space needed for a VPLS to which only router CPE are connected, could allow for alternative encapsulations, as discussed further below.

7.1.6 CPE Addressing and Address Resolution

A. Bridge CPE:

Since a VPLS operates at the link layer, all hosts within all stub sites, in the case of bridge CPE, will typically be in the same network layer subnet. (Multinetting, whereby multiple subnets operate over the same LAN segment, is possible, but much less common). Frames are forwarded across and within the VPLS based upon the link layer addresses - e.g. IEEE MAC addresses - associated with the individual hosts. The VPLS needs to support broadcast traffic, such as that typically used for the address resolution mechanism used to map the host network addresses to their respective link addresses. The VPLS forwarding and reachability algorithms also need to be able to accommodate flooded traffic.

B. Router CPE:

A single network layer subnet is generally used to interconnect router CPE devices, across a VPLS. Behind each CPE router are hosts in different network layer subnets. CPE routers transfer packets across the VPLS by mapping next hop network layer addresses to the link layer addresses of a router peer. A link layer encapsulation is used, most commonly ethernet, as for the bridge case.

As noted above, however, in cases where all of the CPE nodes connected to the VPLS are routers, then it may be possible, due to the constrained addressing space of the VPLS, to use encapsulations that use a different address space than normal MAC addressing. See, for instance, [Jamieson], for a proposed mechanism for VPLSs over MPLS networks, leveraging earlier work on VPRN support over MPLS [Heinanen1], which proposes MPLS as the tunneling mechanism, and locally assigned MPLS labels as the link layer addressing scheme to identify the CPE LSR routers connected to the VPLS.

7.1.7 VPLS Edge Node Forwarding and Reachability Mechanisms

A. Bridge CPE:

The only practical VPLS edge node forwarding mechanism in this case is likely to be standard link layer packet flooding and MAC address learning, as per [IEEE]. As such, no explicit intra-VPLS reachability protocol will be needed, though there will be a need for broadcast mechanisms to flood traffic, as discussed above. In general, it may not prove necessary to also implement the spanning tree protocol [IEEE] between VPLS edge nodes, if the VPLS topology is such that no VPLS edge node is used for transit traffic between any other VPLS edge nodes - in other words, where there is both full mesh connectivity and transit is explicitly precluded. On the other hand, the CPE bridges may well implement the spanning tree protocol in order to safeguard against 'backdoor' paths that bypass connectivity through the VPLS.

B. Router CPE:

Standard bridging techniques can also be used in this case. In addition, the smaller link layer address space of such a VPLS may also permit other techniques, with explicit link layer routes between CPE routers. [Jamieson], for instance, proposes that MPLS LSPs be set up, at the insertion of any new CPE router into the VPLS, between all CPE LSRs. This then precludes the need for packet flooding. In the more general case, if stub link reachability mechanisms were used to configure VPLS edge nodes with the link layer addresses of the CPE routers connected to them, then modifications of any of the intra-VPN

reachability mechanisms discussed for VPRNs could be used to propagate this information to each other VPLS edge node. This would then allow for packet forwarding across the VPLS without flooding.

Mechanisms could also be developed to further propagate the link layer addresses of peer CPE routers and their corresponding network layer addresses across the stub links to the CPE routers, where such information could be inserted into the CPE router's address resolution tables. This would then also preclude the need for broadcast address resolution protocols across the VPLS.

Clearly there would be no need for the support of spanning tree protocols if explicit link layer routes were determined across the VPLS. If normal flooding mechanisms were used then spanning tree would only be required again only if full mesh connectivity was not available and hence VPLS nodes had to carry transit traffic.

7.2 Recommendations

There is significant commonality between VPRNs and VPLSs, and, where possible, this similarity should be exploited in order to reduce development and configuration complexity. In particular, VPLSs should utilize the same tunneling and membership configuration mechanisms, with changes only to reflect the specific characteristics of VPLSs.

8.0 Summary of Recommendations

In this document different types of VPNs have been discussed individually, but there are many common requirements and mechanisms that apply to all types of VPNs, and many networks will contain a mix of different types of VPNs. It is useful to have as much commonality as possible across these different VPN types. In particular, by standardizing a relatively small number of mechanisms, it is possible to allow a wide variety of VPNs to be implemented.

The benefits of adding support for the following mechanisms should be carefully examined.

For IKE/IPSec:

- the transport of a VPN-ID when establishing an SA (3.1.2)
- a null encryption and null authentication option (3.1.3)
- multiprotocol operation (3.1.4)
- frame sequencing (3.1.5)

- asymmetric / legacy user authentication (8.2)

- host address assignment and configuration (8.2)

For L2TP:

- defining modes of operation of IPSec when used to support L2TP (5.2)

For VPNs generally:

- defining a VPN membership information configuration and dissemination mechanism, that uses some form of directory or MIB (section 5.1.2 A,B)

9.0 Security considerations

Security considerations are an integral part of any VPN mechanisms, and these are discussed in the sections describing those mechanisms.

10.0 Acknowledgements

Thanks to Anthony Alles, of Nortel Networks, for his invaluable assistance with the generation of this document, and who developed much of the material on which early versions of this document were based. Thanks also to Joel Halpern for his helpful review comments.

11.0 References

[Aboba1] Aboba, B. and Zorn, G. - "Implementation of PPTP/L2TP Compulsory Tunneling via RADIUS", draft-ietf-radius-tunnel-imp-05.txt.

[Aboba2] Aboba, B. and Zorn, G. - "Criteria for Evaluating Roaming Protocols", RFC 2477.

[ADSL1] ADSL Forum - "An Interoperable End-to-end Broadband Service Architecture over ADSL Systems (Version 3.0)", ADSL Forum 97-215.

[ADSL2] ADSL Forum - "Core Network Architectures for ADSL Access Systems (Version 1.01)", ADSL Forum 998-017.

[ATMF1] ATM Forum - "LAN Emulation over ATM 1.0", af-lane-0021.000, January 1995.

[ATMF2] ATM Forum - "Multi-Protocol Over ATM Specification v1.0",

af-mpoa-0087.000, June 1997.

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

[Bernet] Bernet, Y., et al - "A Framework for Differentiated Services", draft-ietf-diffserv-framework-02.txt.

[Boyle] Boyle, J. - " The COPS (Common Open Policy Service) Protocol", draft-ietf-rap-cops-07.txt.

[Brown] Brown, C. and Malis, A. - "Multiprotocol Interconnect over Frame Relay", RFC 2427.

[Calhoun1] Calhoun, P. and Peirce, K. - "Layer Two Tunneling Protocol "L2TP" IP Differential Services Extension", draft-ietf-pppext-l2tp-ds-03.txt.

[Calhoun2] Calhoun, P., et al - "Layer Two Tunneling Protocol "L2TP" Security Extensions for Non-IP networks", draft-ietf-pppext-l2tp-sec-04.txt.

[Calhoun3] Calhoun, P. et al - "Tunnel Establishment Protocol", draft-ietf-mobileip-calhoun-tep-02.txt.

[Casey1] Casey, L. et al - "IP VPN Realization using MPLS Tunnels", draft-casey-vpn-mpls-00.txt.

[Casey2] Casey, L. "An extended IP VPN Architecture", draft-casey-vpn-extns-00.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-02.txt

[Duffield] Duffield N, et al - "A Performance Oriented Service Interface for Virtual Private Networks", draft-duffield-vpn-qos-framework-00.txt.

[Estrin] Estrin, D., et al - "Protocol Independent Multicast-Sparse Mode (PIM-SM) Protocol Specification", RFC 2362.

[Fenner] Fenner, W. - "IGMP-based Multicast Forwarding (IGMP Proxying)", draft-fenner-igmp-proxy-01.txt

[Ferguson] Ferguson, P. and Huston, G. - "What is a VPN?", Revision, April 1 1998; http://www.employees.org:80/~ferguson/vpn.pdf.

[Fox] Fox, B. and Gleeson, B. "Virtual Private Networks Identifier", RFC 2685.

[Grossman] Grossman, D. and Heinanen, J. - "Multiprotocol Encapsulation over ATM Adaptation Layer 5", RFC 2684.

[Gupta] Gupta, V. - "Secure, Remote Access over the Internet using IPSec", draft-gupta-ipsec-remote-access-02.txt.

[Hamzeh] Hamzeh, K., et al - "Point-to-Point Tunneling Protocol (PPTP)", RFC 2637

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

[Hardjono] Hardjono, T. et al. - "Secure IP Multicast: Problem areas, Framework, and Building Blocks" draft-irtf-smug-framework-00.txt

[Harkins] Harkins, D. and Carrel, D. "The Internet Key Exchange (IKE)", RFC 2409.

[Heinanen1] Heinanen, J. and Rosen, E. "VPN Support with MPLS" draft-heinanen-mpls-vpn-01.txt.

[Heinanen2] Heinanen, J., et al - "MPLS Mappings of Generic VPN Mechanisms", draft-heinanen-generic-vpn-mpls-00.txt.

[Heinanen3] Heinanen, J. - "Multiprotocol Encapsulation over ATM Adaptation Layer 5", RFC1483.

[IEEE] ANSI/IEEE - 10038: 1993 (ISO/IEC) Information technology -- Telecommunications and information exchange between systems -- Local area networks -- Media access control (MAC) bridges, ANSI/IEEE Std 802.1D, 1993 Edition.

[Jacobson] Jacobson, V. et al - "An Expedited Forwarding PHB", RFC 2598.

[Jamieson] Jamieson, D., et al - "MPLS VPN Architecture", draft-jamieson-mpls-vpn-00.txt.

[Jamieson2] Jamieson, D and Wang, R. - "Solicitation Extensions for BGP-4" draft-jamieson-bgp-solicit-00.txt.

[Kelly] Kelly, S. et al. - "User-level Authentication Mechanisms for

IPsec", draft-kelly-ipsra-userauth-00.txt.

[Kent] Kent, S. and Atkinson, R. "Security Architecture for the Internet Protocol", RFC 2401.

[Kent2] Kent, S. and Atkinson, R. "IP Encapsulating Security Payload (ESP)", RFC 2406.

[Li] Li, T. and Rekhter, Y. - "Provider Architecture for Differentiated Services and Traffic Engineering (PASTE)", RFC 2430.

[Li2] Li, T. - "CPE based VPNs using MPLS", draft-li-mpls-vpn-00.txt.

[Litvin] Litvin, M. et al - "A Hybrid Authentication Mode for IKE", draft-ietf-ipsec-isakmp-hybrid-auth-02.txt.

[MacRae] MacRae, M. and Ayandeh, S. - "Using COPS for VPN Connectivity" draft-macrae-policy-cops-vpn-00.txt

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

[Mamakos] Mamakos, L. et al. - "A Method for Transmitting PPP Over Ethernet (PPPoE)" RFC 2516.

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

[Muthukrishnan] Muthukrishnan, K. and Malis A. - "Core IP VPN Architecture", draft-muthukrishnan-corevpn-arch-00.txt.

[Patel1] Patel, B. et al. - "Securing L2TP using IPSEC", draft-ietf-pppext-l2tp-security-04.txt.

[Patel2] Patel, B. - "Dynamic configuration of IPSEC VPN host using DHCP", draft-ietf-ipsec-dhcp-02.txt

[Pegrum] Pegrum, S. - "VPN Point to Multipoint Tunnel Protocol (VPMT)", draft-pegrum-vmmt-01.txt.

[Pereira1] Pereira, R. et al - "The ISAKMP Configuration Method", draft-ietf-ipsec-isakmp-mode-cfg-05.txt.

[Pereira2] Pereira, R. and Beaulieu, S. - "Extended Authentication Within ISAKMP/Oakley", draft-ietf-ipsec-isakmp-xauth-05.txt.

[Perez] Perez, M., Liaw, F. et al. "ATM Signaling Support for IP over ATM" RFC 1755.

[Perkins] Perkins, C. - "IP Encapsulation within IP" RFC 2003.

[Petri] Petri, B. (editor) - "MPOA v1.1 Addendum on VPN support", ATM Forum, af-mpoa-0129.000.

[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-03.txt.

[Rigney] Rigney, C., et al - "Remote Authentication Dial In User Service (RADIUS)", RFC 2138.

[Richard] Richard, C. and Smith, K. - "The PPP Bandwidth Allocation Protocol (BAP), The PPP Bandwidth Allocation Control Protocol (BACP)" RFC 2125.

[Rosen1] Rosen, E. and Rekhter, Y. - "BGP/MPLS VPNs", RFC 2457.

[Rosen2] Rosen, E., et al "Multiprotocol Label Switching Architecture", draft-ietf-mpls-arch-06.txt.

[Shacham] Shacham, A., et al - "IP Payload Compression Protocol (IPComp)", RFC 2393.

[Shea] Shea, R. - "L2TP-over-IP Path MTU Discovery (''L2TPMTU'')", draft- ietf-pppext-l2tpmtu-00.txt.

[Shieh1] Shieh, P et al. "The Architecture for Extending PPP Connections for Home Network Clients", ADSL Forum contribution 98-140.

[Shieh2] Shieh, P et al. "The Requirement and Comparisons of Extending PPP over Ethernet", ADSL Forum contribution 98-141.

[Sklower] Sklower, K., et al - "The PPP Multilink Protocol (MP)", RFC 1990.

[Srisuresh1] Srisuresh, P. and Holdrege, M. - "IP Network Address Translator (NAT) Terminology and Considerations", draft-ietf-nat-terminology-03.txt.

[Srisuresh2] Srisuresh, P. - "Secure Remote Access with L2TP", <draft-ietf-pppext-secure-ra-00.txt>

[Thomas] Thomas, B., et al - "LDP Specification", draft-ietf-mpls-ldp-06.txt.

[Townsley] Townsley, M., et al - "Layer Two Tunneling Protocol 'L2TP'", RFC 2661.

[Valencia] Valencia, A., et al "Cisco Layer Two Forwarding (Protocol) "L2F"", RFC 2341.

[Waitzman] Waitzman, D., et al - "Distance Vector Multicast Routing Protocol", RFC 1075.

[Wallner] Wallner, D., et al - "Key Management for Multicast: Issues and Architectures" RFC2627.

12.0 Author Information

Bryan Gleeson
Nortel Networks
4500 Great America Parkway
Santa Clara CA 95054
USA
Tel: +1 (408) 548 3711
Email: bgleeson@shastanets.com

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

Arthur Lin
Nortel Networks
4500 Great America Parkway
Santa Clara CA 95054
USA
Tel: +1 (408) 548 3788
Email: alin@shastanets.com

Grenville Armitage
Bell Labs Research Silicon Valley
3180 Porter Drive,
Palo Alto, CA 94304
USA
Email: gja@lucent.com

Andrew G. Malis
Lucent Technologies
1 Robbins Road
Westford, MA 01886
USA
Tel: +1 978 952 7414
Email: amalis@lucent.com

13.0 Full Copyright Statement