Provider Provisioned VPN Working Group
Internet Draft
draft-knight-ppvpn-ipsec-dynroute-03.txt
Expires: April 2004

Paul Knight
Nortel Networks

Bryan Gleeson
Tahoe Networks

October 2003

A Method to Provide Dynamic Routing in IPsec VPNs

Status of this Memo

Abstract

Exchange of routing information between IPsec security gateways, using standard routing protocols across IPsec tunnels, can be a straightforward operation. Using the routing information to choose the proper path is also straightforward, when routing is functionally separated from the IPsec gateway operation. One of the most significant obstacles to widespread implementation of dynamic routing in IPsec VPNs has been agreement on a way to exchange and use the routing information. This document describes a simple and secure method of exchanging dynamic routing information between IPsec security gateways, using standard IPsec messages. This method is currently in use in a large number of installations, and has been demonstrated to be interoperable across several IPsec implementations from different vendors.

Table of Contents

1. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119].

2. Overview

Dynamic exchange of routing information between the various sites of a Virtual Private Network (VPN) can significantly simplify the management of the VPN. Without dynamic routing, it is difficult to configure a large number of routes correctly and in a timely manner.

It is even more difficult to set up load balancing and failover when the routing configuration is static. IPsec offers tremendous promise as a VPN technology, since it can securely connect sites over any IP network, within a service provider's IP network or over the Internet.

However, no specific method of accomplishing dynamic routing in an interoperable manner has been defined within existing IPsec standards. This document describes a method by which standard IPsec mechanisms can support the secure transport of dynamic IP routing information between sites in an IPsec-based VPN, and use that routing information to dynamically direct traffic. Most IP routing protocols can be transported natively over standards-compliant IPsec implementations using this method.

This method fits in the architecture for provider-provisioned customer edge (CE)-based IPsec VPNs discussed in [CE-BASED].

Three key elements are needed to support dynamic routing in an IPsec VPN environment:

1) Agreement between each pair of connected VPN sites to participate in a dynamic routing connection. This is usually accomplished by configuring a dynamic routing protocol on the gateways, and linking it to the IPsec Security Associations involved in the connection. The specific configuration details may vary by implementation.

2) Creation of a secure link between each pair of gateways at connected VPN sites. This is referred to as a "VPN tunnel" in this document, in accordance with the usage in [CE-BASED]. Any traffic passing through the VPN tunnel is protected by IPsec, including routing protocol exchanges.

3) Ability to use the VPN tunnels in a manner analogous to simple link connections, and route the traffic between sites over them. Route policy, packet filtering, and firewall capabilities can be applied to the traffic as it is being transmitted through the VPN tunnels, delivering overall functionality similar to the use of dedicated encryption/authentication hardware on dedicated links between routers.

The implementation described in this document uses a transport mode proposal in the Internet Key Exchange (IKE) Quick Mode exchange [RFC-2409] to properly express the restrictions on the traffic in an IPsec-compliant method. However, since the proposal also specifies the use of IP-in-IP [RFC-2003] encapsulation, the gateways actually send all inter-site traffic, including the routing information, in packets that are exactly the same as tunnel mode packets. Using the transport mode proposal to create an SA which protects the IP-in-IP

tunnel results in the creation of an IPsec-protected VPN tunnel which can carry all traffic, including routing updates.

Some of the perceived limitations of the direct transport of routing protocols within IPsec tunnels are the result of limitations of specific implementations, and not limitations of IPsec itself. In particular, IPsec does not preclude the transport of multicast or broadcast IP traffic. Thus OSPF [RFC-2328], which uses multicast, can be carried over IPsec tunnels and can be used by the IPsec gateways for dynamic routing.

3. Routing in IPsec

This section discusses the "routing problem" for IPsec, and describes how "tunnel links" can transport normal routing protocols as well as inter-site VPN traffic securely over IPsec.

3.1 IPsec background

The IP Security Architecture, defined in [RFC-2401], requires IPsec-compliant hosts or gateways to formulate a security policy regulating how they will communicate with other hosts and networks. The standard defines a Security Policy Database (SPD), which much be consulted for every inbound and outbound packet to decide whether that packet can bypass IPsec, or whether the packet requires IPsec processing, or perhaps that the packet must be dropped.

When IPsec processing is required, an appropriate IPsec security association (SA) for the packet needs to be found. If a SA does not currently exist, the Internet Key Exchange (IKE) is used to dynamically establish a pair of new SAs (one in each direction). As part of the IKE Quick Mode exchange defined in [RFC-2409] which establishes new IPsec SAs, the IKE peer initiating the Quick Mode exchange may send client identifiers which specify the traffic which will be allowed through the new security associations. The allowable traffic is specified in terms of source and destination addresses, or networks and ranges of addresses, with optional protocol and source and destination ports.

This background information is necessary to understand that an IPsec SA, whether operating in tunnel or transport mode, is not normally a "data pipe" through which any and all IP traffic may be sent. The client identifiers determine what is and is not allowed. Moreover, these identifiers are fixed at the time the SA is established. To allow any other traffic to pass, a new SA pair needs to be negotiated.

3.2 IPsec routing issues

Since the destination address of a packet (along with other selectors) can determine which SA applies to a packet and thus which tunnel the packet may pass through, it is difficult to apply dynamic

routing techniques to an IPsec VPN built with standard security associations. If the SAs specify particular networks, then these

specifications must override any routing protocol decisions, to comply with IPsec.

Support for dynamic routing between IPsec gateways must be added to the IPsec scenario just described, to make configuration of multiple IPsec VPN sites tractable, as opposed to the static routing - the full specification of the local and remote networks - which seems to be implied within IPsec. As described in [TOUCH], there are a number of difficulties with dynamic routing while using standard IPsec approaches, chief of which are:
- dynamic updating of SAs with each routing change is unwieldy, and may lead to security issues
- the SA database does not by definition contain interface information, and cannot adapt to changes.

Another discussion of issues constraining dynamic routing in IPsec VPNs is presented in [WANG]. Although generally useful, it appears to misidentify some implementation-specific issues as IPsec issues, such as IPsec tunnel endpoint attachment and a lack of support for multicast and broadcast traffic.

One method of supporting dynamic routing within IPsec would be to negotiate one security association just to pass the routing protocol in question, then once it is known what networks are available on the other side, negotiate separate security associations for each and every network on the fly, potentially dropping some as later routing updates show changes to the configuration.

Even if this were done, there is no guarantee that any other implementation would actually use the routing information to establish the necessary dynamic security policy database entries, since the IPsec specifications contain no requirement to support such dynamic changes to security policy. Therefore, this scheme is unlikely to lead to true interoperability, and the additional complexity of managing multiple security associations (and the dynamic policy entries necessary to support them) is a further drawback to this scheme.

What is proposed in this document instead is the "tunnel link" approach: one security association pair between the two gateways, which will carry both the routing protocols themselves and all subsequent traffic between the networks on both sides. This requires that the gateway must use the dynamic routing information in addition to the configured SA database entries for routing through the "tunnel links." Of course, any other required security restrictions must be applied by the routing functionality before traffic is allowed to enter the "tunnel links." This approach combines the proven encryption protection of IPsec with the manageability and traffic control capabilities of current routing

technology.

3.3 "Tunnel Link" Approach

The first step to dynamic routing is to set up a bi-directional pair of IPsec SAs between gateways that will allow ALL inter-site traffic to pass through, with IPsec protection. Since this IPsec "connection" can carry all IP traffic, including broadcast and multicast traffic, it is in some ways similar to a link layer connection, and is referred to here as a "tunnel link." A "tunnel link" is a specific "VPN tunnel" [CE-BASED] which uses IPsec encryption and/or authentication services, but which does not typically use the IPsec filtering mechanisms (selectors) for traffic; the filtering and routing will be performed by other processes before the traffic is sent through the "tunnel link."

A "tunnel link" can be constructed using either IPsec tunnel mode or by using IP-in-IP encapsulation within IPsec transport mode. In either case, the packets meet the protection requirements of tunnel mode packets, as required by the IPsec Architecture [RFC-2401] for traffic between IPsec gateways.

Note that modifications to RFC 2401 (RFC2401bis) were proposed by the authors of RFC 2401 in the IPsec Working Group meeting at IETF 53. These modifications are expected to include recognition that IP-in-IP encapsulation within transport mode can provide the same protection to traffic between gateways as tunnel mode. However, there is currently no published document other than the WG session presentations as a reference.

3.3.1 Tunnel Mode "Tunnel Link"

Most standards-compliant IPsec gateway implementations can be configured with interoperable "tunnel links" using tunnel mode.

A "tunnel link" can be established by negotiating a tunnel mode SA as follows:

In the Quick Mode exchange, specify "wildcards" as client identifiers. This means the Identification Payload [RFC-2407] is set to ID_IPV4_ADDR_SUBNET (value 4) for ID Type, and the network mask of the Identification Data field is set to all zeros (wildcard). Although the IP address in the Identification Data field is irrelevant using the wildcard netmask, 0.0.0.0 should be used for clarity.

This might appear to be a good way to begin to solve the IPsec routing issue. However, in practice it has proven to provide limited interoperability among vendors for dynamic routing. This "wildcard" method is typically used by IPsec gateways with no dynamic routing capabilities, to establish a "default route" from a branch site to a

central hub site, or for simple site-to-site IPsec connections. An IPsec gateway that is capable of dynamic routing could establish an IPsec tunnel mode "tunnel link" with a non-routing-capable gateway,

but would be unable to exchange dynamic routes. This results in a link where the IPsec connection is functional, but traffic cannot flow in both directions due to the inability to exchange routes. One side may see the connection as a default route, while the other would receive no routing information and would send no traffic.

3.3.2 Transport Mode Proposal

In order to support dynamic routing unambiguously, another alternative is ideal. The signaling method proposed in this document is the use of a particular transport mode proposal. This is discussed in detail in section (3.4.3).

It should be noted that [RFC-2401] states in section 4.1 that security gateways must use tunnel mode SAs. In addition to the encapsulation protection of tunnel mode, this is also due to the need to avoid potential problems with fragmentation and reassembly of IPsec packets, and in circumstances where multiple paths exist to the same destination. Thus it is important that the packets in the "tunnel link" be encapsulated and decapsulated in the same manner as tunnel mode packets. A transport mode-based "tunnel link", using IP-in-IP encapsulation can meet these functional requirements of RFC 2401.

IPsec gateways that are incapable of dynamic routing will have no use for this particular transport mode proposal. Since there is little likelihood that gateways which do not support dynamic routing will be configured to accept this proposal, interoperability among implementations is more straightforward than with the tunnel mode approach. Interoperability between several leading vendors of IPsec gateways has already been demonstrated using this approach.

3.3.3 Tunnel vs. Transport as a "Tunnel Link"

As discussed in [TOUCH], the transport mode approach offers a cleaner and probably more secure method of separating routing from IPsec processing. With the tunnel mode approach, in order for dynamic routing to work in a hub site (with multiple "tunnel links" to various branch sites) the "wildcard" selectors (which allow ANY traffic to pass through ANY "tunnel link") must be explicitly ignored, in favor of a routing process. This is in contrast to the transport mode approach, where the selectors allow ONLY traffic that has been IP-encapsulated and explicitly passed to the IPsec process.

Thus for a hub site, it is clear that the transport mode approach provides a clean separation of routing functions from the IPsec processing: the routing functionality produces an IP-in-IP encapsulated packet with the destination set to the proper IPsec

next-hop, and the IPsec functionality encrypts it for transmission. In contrast, the tunnel mode approach does not allow for clean functional separation: the routing functionality must determine the

next hop and somehow communicate this to the IPsec process, so it can be used during encapsulation (as the destination address). While it is clear that gateway devices can be constructed to perform this combined operation, the lack of a well-defined method for doing it is likely to lead to non-interoperable implementations. Further, it breaks open the "black box" of IPsec processing so that it no longer complies with the specifications. Any given packet entering into the IPsec process could be encapsulated in various ways, depending on the input from the routing function. Unless the IPsec specifications are rewritten to include a mechanism by which routing protocols can direct encapsulation, then the use of tunnel mode with a wild card selector remains problematic for the hub site.

Given the need to separate the routing from the IPsec processing, it appears that the IPsec selectors will provide better security against implementation or configuration errors if the transport mode approach is used. Traffic which "leaks" past the routing process with the tunnel mode approach may go anywhere; traffic which "leaks" past the routing/encapsulation process in the transport mode approach will go nowhere.

Further, as noted in [TOUCH] Section 4.6, "IPsec selectors under IIPtran can express the same set of policies as conventional IPsec tunnel mode," so the full range of IPsec selectors can be applied to the inner IP packet. This is not possible with tunnel mode using a wildcard selector.

It is not currently feasible for one IPsec gateway to determine if another IPsec gateway can provide the capabilities for dynamic routing over the "tunnel link," when it is constructed using a tunnel mode proposal. Since a number of IPsec implementations can provide a "tunnel link" over tunnel mode connections, but not support dynamic routing over it, an approach should be used which will avoid confusion.

In particular, it is more important to have one agreed-upon way to support dynamic routing in IPsec than to support two methods. Negotiation of IPsec parameters between two different implementations will not work unless they both have implemented dynamic routing over a common approach; and if they share a common approach, there is no need for an alternative approach.

Given the interoperability limitations of dynamic routing in tunnel mode "tunnel links", and the discussion of the advantages of the transport mode approach above and in [TOUCH], we suggest that the transport mode approach should become the standard method for supporting dynamic routing over IPsec.

3.3.4 Routing over a "Tunnel Link"

The following discussion applies to any kind of "tunnel link."

It should be emphasized that routing information can be carried through a "tunnel link." IPsec gateways can encapsulate normal routing update messages and send them to each other through the "tunnel link." RIP can be propagated with no special considerations. OSPF can treat the "tunnel link" as an unnumbered point-to-point network. BGP can also be implemented.

There may be a limitation in some IPsec implementations that does not allow an IPsec tunnel to be recognized as an IP interface. For such an implementation, routing protocols such as RIP and OSPF that are dependent on IP interfaces cannot be defined directly on the IPsec tunnel, and may not be able to send and receive routing information in this way. This is not an IPsec issue, but is implementation-specific. In these cases, additional virtual IP interfaces may be required for terminating IP-in-IP or GRE tunnels, which will use the IPsec transport mode SA.

A "tunnel link" essentially provides a connection for any IP traffic between IPsec gateways. The gateways must have the ability to use them for dynamic routing. The gateway implementation must coordinate the application of routing updates with other security restrictions expressed in IPsec or configured by other methods. A description of these implementation details is beyond the scope of this document, since it depends intimately on the internal architecture of each IPsec gateway. However, interoperability of dynamic routing between most IPsec gateway implementations that provide routing capabilities should be feasible using the method described in this document.

3.4 Methods of Establishing a "Tunnel Link"

One crucial issue is how to express in IKE Quick Mode the desire to establish a "tunnel link" security association which allows for dynamic routing, while at the same time ensuring that various IPsec gateway implementations can still interoperate with "static" (SA-based) routing, and not cause confusion between the two (such as another implementation trying to negotiate something which the security gateway would interpret as being capable of dynamic routing).

Three approaches are discussed below. The third method is the method suggested for adoption.

3.4.1 Method 1 - Vendor ID compatibility

A specific IKE Vendor ID payload can be used on both sides in the Phase I negotiation to ensure that both peers support this method, and thus would be capable of allowing dynamic routing through a tunnel. During Quick Mode, assuming compatible gateways on both sides, send a Quick Mode proposal that is understood by both sides

to signal the creation of the tunnel link.

For example, a tunnel mode proposal could be sent without client identifier payloads. Normally, negotiating tunnel mode security associations without client identifiers implies that only the tunnel endpoints (gateways) may communicate through the resulting security association pair. However, when using this vendor-specific ID, this use of Quick Mode client identifiers (or actually the lack of them) would be interpreted as defining a "tunnel link." This approach could be acceptable as a known proprietary solution.

The biggest problem with this method is the reliance on a Quick Mode proposal that can easily be interpreted differently by another implementation. Also, the use of the Vendor ID payload is problematic, as it then means that either all implementations would have to use the specific Vendor ID. It is far better to use a Quick Mode proposal that will be unambiguous to all implementations, even those that do not support dynamic routing. Method 3, described below, solves these issues.

3.4.2 Method 2 - Notify Message or new IPsec message

An alternative method might use an IPsec Notify Message, a newly-specified IPsec extension, or perhaps a new Encapsulation Mode value to communicate the intent to set up the "tunnel link." This approach is likely to be non-interoperable for some period, and has not been explored in detail.

3.4.3 Method 3 - Transport Mode with IP-in-IP

An interoperable approach must allow the Quick Mode proposal to express a request for a "tunnel link" connection with dynamic routing, while limiting the possibility of misinterpretation by an implementation that does not support dynamic routing. It uses a transport mode proposal with client identifiers and protocol identification to express the same capabilities as the tunnel mode "tunnel link" discussed above.

A more general approach to this method is discussed in detail in [TOUCH], and labeled "IIPtran."

We assume that the initial ISAKMP Security Association (SA) has already been established in Phase 1 between the gateways. This ISAKMP SA is used to protect the negotiations for Phase 2, in which a Quick Mode negotiation establishes the IPsec SA.

This Quick Mode proposal should specify a transport mode SA, with client identifiers consisting of the gateway addresses, and a protocol value of 4, signifying IP-in-IP.

The traffic sent will actually be constructed exactly the same as IPsec tunnel mode traffic, but the SA traffic restrictions will be

evaluated according to the proposal for transport mode.

Since transport mode identifiers are applied to the outermost IP header, the syntax is correct for expressing the "restrictions" on the traffic being passed. IPsec tunnel mode packets use the gateway addresses as the source and destination addresses. IPsec tunnel mode packets are constructed such that the "next payload" field of either AH or ESP is always IP-in-IP (4), so the IPsec packets passed through are consistent with the transport mode restrictions placed by the client identifiers.

Essentially, this method places no restrictions on the inner IP header, but it does specify, through its use of IP-in-IP as a transport mode protocol, that there will always be an inner IP header as there is in "normal" tunnel mode.

As noted in [TOUCH], although the packets themselves contain no differentiation as to whether they are tunnel mode or are transport mode carrying IP-in-IP, there are differences in the way that incoming transport and tunnel mode decapsulation is handled. For the method described in this document to work effectively, the receiver must implement a processing rule for type 4 (IP-in-IP) packets so that they will always be decapsulated upon receipt. That is, they must be processed as in tunnel mode. This can be accomplished either by explicitly configuring the IP-in-IP encapsulation as a tunnel interface, or by other implementation-specific mechanisms.

4. Other considerations

4.1 Interoperability Issues

Implementations that use this method should provide the following functionality:
a) be able to negotiate the Quick Mode proposal for the "tunnel link" described in this document in section 3.4.3
b) be able to send, receive, and appropriately process the routing messages over the "tunnel links"
c) be able to use the routing information to direct traffic to the appropriate "tunnel link", using the routes which have been learned
d) be able to apply configured route policies to the learned routes
e) be able to apply packet filtering, and (optionally) firewall capabilities to the traffic before it is sent over the "tunnel link."

This method of establishing the "tunnel link" should be restricted to this specific use. The only purpose for sending this proposal should be to advertise the gateway's ability to support dynamic routing. This Quick Mode proposal should be rejected by an implementation that does not support dynamic routing in this method.

There is no current alternate interpretation of the client

identifiers with IP-in-IP that would cause confusion to
implementations that do not support dynamic routing, so it should be

safe to use even without any requirement to have cooperating Vendor IDs or other signaling.

There may be a limitation in an IPsec implementation that does not allow an IPsec tunnel to be recognized as an IP interface. For such an implementation, routing protocols such as RIP and OSPF that are dependent on IP interfaces cannot be defined directly on the IPsec tunnel. An implementation of this type may not be able to use this method for RIP and OSPF, but it can work with BGP. It may have to use another layer of encapsulation, such as GRE [RFC-2784](which can support RIP and OSPF), on top of the IPSec tunnel, or over a transport mode connection. This introduces additional configuration as well as additional packet header overhead, not just for the routing packets, but also for all data that traverses the tunnel. This is not an IPsec issue, but may depend on a specific implementation. Methods of using GRE for this purpose are discussed in [KHETAN].

4.2 Scalability

The use of a single IPsec SA per "tunnel link," as described in this document, can significantly increase the number of IPsec VPN connections supported per gateway, compared to alternatives. Since IPsec normally requires a separate IPsec SA per subnet or IP address range, there can potentially be a large number of SAs needed to express the normal routing of multiple subnets between two VPN sites. For the hub sites of large VPNs, this can even drive requirements to use multiple gateways to support all the connections. Thus using a single SA per connection can simplify VPN design as well as improve system performance.

One alternative approach to support dynamic routing which has been proposed using GRE [KHETAN] requires maintaining both a GRE tunnel and an IPsec SA per VPN connection, leading to higher overhead and potentially lower performance than the method described in this document.

4.3 OSPF Routing over a "Tunnel Link"

Since the IPsec gateways will in most cases not have their interfaces in the same subnet, OSPF must be configured to treat the "tunnel link" as an unnumbered point-to-point network.

5. Security Considerations

This document describes a method of dynamic routing in which the gateway device can use standard routing functionality to perform dynamic routing decisions. Packets are assigned to a specific "tunnel link" SA based on the packet's destination address, using

routing information that has been dynamically learned. Each "tunnel link" provides standard IPsec protection for all traffic. All traffic carried in the "tunnel link" between the IPsec gateways is

encapsulated in a packet which is identical to a tunnel mode packet, as required by the Security Architecture for IP [RFC-2401].

The use of a "tunnel link" may be controversial at first glance, because traffic control is performed by a routing function rather than through detailed negotiation of multiple IPsec security associations. However, it should be noted that the "tunnel link" in this case actually expresses the desire of the VPN operator for a connection providing trusted encryption and security across a public network, while allowing manageable dynamic routing within the protection afforded by IPsec. It does not violate the letter or the spirit of IPsec.

It should be noted that alternative proposals for use of encapsulation over IPsec such as [KHETAN] are actually equivalent in terms of IPsec security, in that they use some routing functionality to determine the desired path, then hide the characteristics of the data within an encapsulated IP packet format, so that most IPsec selectors of the internal packet are not visible.

Dynamic routing opens possibilities for misdirection of traffic, due to transient conditions or intentional misconfiguration. Since all routing information will be coming from other trusted VPN sites, the security exposure from this source is equivalent to any private network or VPN that exchanges routing information. The internal routing functionality SHOULD provide support for routing policies to manage the learned routes, as well as traffic filtering.

All VPN traffic crossing the public network between the IPsec gateways will be protected by IPsec using the method described in this document. Packets that are identical to tunnel mode are used, although the creation of the "tunnel link" is signaled by a proposal for a transport mode SA. Appropriate levels of encryption should be chosen, commensurate with the level of confidentiality required.

6. Summary for Sub-IP Area

6.1 Summary

The PPVPN WG currently supports three types of VPNs: Provider Provisioned Network Based Layer 3 VPNs, Provider Provisioned Network Based Layer 2 VPNs and Provider Provisioned Customer-Edge Based VPNs. This draft discusses the use of standard IPsec capabilities to support routing for CE-based IPSec VPNs.

6.2 Where does it fit in the picture of the Sub-IP work?

This work fits in the PPVPN box. Although this work is based on IPsec, it is an application of IPsec, and does not involve changes to IPsec. Therefore it is not considered within the IPsec Working

Group. Exchange of dynamic routing information between CE-based VPN

devices provisioned by service providers is a key enabling technology for allowing scalable IPsec VPN deployments.

6.3 Why is it targeted at this Working Group?

This document describes how standard IPsec mechanisms can support the tunneling of IP multicast and broadcast packets, so that IPsec VPN tunnels can support dynamic routing protocols over the tunnel.

Under the current PPVPN WG charter, Provider Provisioned CE-based VPNs fits the scope of the WG, as stated from the following charter extract: "This working group is responsible for defining and specifying a limited number of sets of solutions for supporting provider-provisioned virtual private networks (PPVPNs). The work effort will include the development of a framework document, a service requirements document and several individual technical approach documents that group technologies together to specify specific VPN service offerings. The framework will define the common components and pieces that are needed to build and deploy a PPVPN. Deployment scenarios will include provider-managed VPN components located on customer premises."

6.4 Justification

This draft is justified since it targets the routing issue of CE-based VPNs, which are identified as a specific type of PPVPNs both in the WG charter and the general framework I-D. CE-based VPN has specific characteristics and operational requirements, including routing support.

7. Document Change History

Version -01:
- Added change history section
- Modified title to remove "signal," adjusted text to change emphasis from the concept of signaling. We felt that there is not a specific need to signal the ability to support routing protocols since routing protocols are typically configured on the nodes, and IPsec VPN tunnels should be considered similar to other links, not requiring additional signaling for this purpose.
- Clarified "tunnel link" terminology with respect to "VPN tunnel", to align with [CE-BASED].
- Added mention of Steve Kent's plans for RFC2401bis.
- Updated references.
- Added section discussing tunnel mode vs. transport mode for "tunnel links".

Version -02:
- Changed "tunnel link" terminology to "VPN tunnel" to correspond

with the usage of [CE-BASED].

8. References

[RFC 2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[CE-BASED] De Clercq, J., Paridaens, O., Iyer, M., Krywaniuk, A., and Wang, C., "An Architecture for Provider Provisioned CE-based Virtual Private Networks using IPsec", draft-ietf-l3vpn-ce-based-00.txt, work in progress, March 2003.

[RFC-2409] Harkins, D., and Carrel, D., "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

[RFC-2003] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996.

[RFC-2328] Moy, J., "OSPF Version 2", RFC 2328, STD 54, April 1998.

[RFC-2401] Kent, S. and Atkinson, R., "Security Architecture for the Internet Protocol", RFC 2401, November 1998.

[TOUCH] Touch, J., and Eggert, L., "Use of IPsec Transport Mode for Dynamic Routing," draft-touch-ipsec-vpn-06.txt, work in progress, September 2003.

[WANG] Wang, C., Beadles, M., and Khetan, A., "Routing Support in CE-based IPsec VPNs," draft-wang-cevpn-routing-00.txt, work in progress, October 2001.

[RFC-2407] Piper, D., "The Internet IP Security Domain of Interpretation for ISAKMP", RFC 2407, November 1998.

[RFC-2784] Farinacci, D., Li, T., Hanks, S., Meyer, D., and Traina, P., "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.

[KHETAN] Khetan, A., Wang, C., Beadles, M., French, L., and Vyncke, E., "Use of GRE for routing support in IPsec VPNs", draft-khetan-sp-greipsec-00.txt, work in progress, January 2002.

9. Acknowledgements

We would like to acknowledge the helpful comments and contributions of the following individuals: Larry DiBurro, Haixiang He, Bob Lee, Shawn Mamros (who implemented this approach), Simon McCormack, and Mark Duffy.

10. Authors' Addresses

Paul Knight
Nortel Networks
600 Technology Park Drive
Billerica, MA. 01821 USA
paknight@nortelnetworks.com
+1 (978) 288 6414

Bryan Gleeson
Tahoe Networks
3052 Orchard Drive
San Jose, CA 95134 USA
bryan@tahoenetworks.com

others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any