Network Working Group   F. Templin, Ed.
Internet-Draft   Boeing Research & Technology
Intended status: Standards Track   March 14, 2011
Expires: September 15, 2011

Virtual Enterprise Traversal (VET)
draft-templin-intarea-vet-24.txt

Abstract

Enterprise networks connect hosts and routers over various link types, and often also connect to provider networks and/or the global Internet. Enterprise network nodes require a means to automatically provision addresses/prefixes and support internetworking operation in a wide variety of use cases including Small Office, Home Office (SOHO) networks, Mobile Ad hoc Networks (MANETs), ISP networks, multi-organizational corporate networks and the interdomain core of the global Internet itself. This document specifies a Virtual Enterprise Traversal (VET) abstraction for autoconfiguration and operation of nodes in enterprise networks.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

Enterprise networks [RFC4852] connect hosts and routers over various link types (see [RFC4861], Section 2.2). The term "enterprise network" in this context extends to a wide variety of use cases and deployment scenarios. For example, an "enterprise" can be as small as a Small Office, Home Office (SOHO) network, as complex as a multi-organizational corporation, or as large as the global Internet itself. Internet Service Provider (ISP) networks are another example use case that fits well with the VET enterprise network model. Mobile Ad hoc Networks (MANETs) [RFC2501] can also be considered as a challenging example of an enterprise network, in that their topologies may change dynamically over time and that they may employ little/no active management by a centralized network administrative authority. These specialized characteristics for MANETs require careful consideration, but the same principles apply equally to other enterprise network scenarios.

This document specifies a Virtual Enterprise Traversal (VET) abstraction for autoconfiguration and internetworking operation, where addresses of different scopes may be assigned on various types of interfaces with diverse properties. Both IPv4/ICMPv4 [RFC0791][RFC0792] and IPv6/ICMPv6 [RFC2460][RFC4443] are discussed within this context (other network layer protocols are also considered). The use of standard DHCP [RFC2131] [RFC3315] is assumed unless otherwise specified.

```
                      Provider-Edge Interfaces
                           x   x        x
                           |   |        |
      +--------------------+---+--------+----------+    E
      |                    |   |        |          |    n
      |    I               |   |  ....  |          |    t
      |    n           +---+---+--------+---+      |    e
      |    t           |   +--------+      /|      |    r
      |    e  I   x----+   |  Host  |   I /*+------+--< p  I
      |    r  n        |   |Function|   n|**|      |    r  n
      |    n  t        |   +--------+   t|**|      |    i  t
      |    a  e   x----+              V e|**+------+--< s  e
      |    l  r      . |              E r|**|  .   |    e  r
      |       f      . |              T f|**|  .   |       f
      |    V  a      . |   +--------+   a|**|  .   |    I  a
      |    i  c      . |   | Router |   c|**|  .   |    n  c
      |    r  e   x----+   |Function|   e \*+------+--< t  e
      |    t  s        |   +--------+      \|      |    e  s
      |    u           +---+---+--------+---+      |    r
      |    a               |   |  ....  |          |    i
      |    l               |   |        |          |    o
      +--------------------+---+--------+----------+    r
                           |   |        |
                           x   x        x
                    Enterprise-Edge Interfaces
```

Figure 1: Enterprise Router (ER) Architecture

Figure 1 above depicts the architectural model for an Enterprise Router (ER). As shown in the figure, an ER may have a variety of interface types including enterprise-edge, enterprise-interior, provider-edge, internal-virtual, as well as VET interfaces used for encapsulating inner network layer protocol packets for transmission over outer IPv4 or IPv6 networks. The different types of interfaces are defined, and the autoconfiguration mechanisms used for each type are specified. This architecture applies equally for MANET routers, in which enterprise-interior interfaces typically correspond to the wireless multihop radio interfaces associated with MANETs. Out of scope for this document is the autoconfiguration of provider interfaces, which must be coordinated in a manner specific to the service provider’s network.

Enterprise networks require a means for supporting both Provider-(In)dependent (PI) and Provider-Aggregated (PA) addressing. This is especially true for enterprise network scenarios that involve mobility and multihoming. The VET specification provides adaptable mechanisms that address these and other issues in a wide variety of enterprise network use cases.

The VET framework builds on a Non-Broadcast Multiple Access (NBMA) [RFC2491] virtual interface model in a manner similar to other automatic tunneling technologies [RFC2529][RFC5214]. VET interfaces support the encapsulation of inner network layer protocol packets over IP networks (i.e., either IPv4 or IPv6). VET is also compatible with mid-layer encapsulation technologies including IPsec [RFC4301], and supports both stateful and stateless prefix delegation.

VET and its associated technologies (including the Subnetwork Encapsulation and Adaptation Layer (SEAL) [I-D.templin-intarea-seal]) are functional building blocks for a new Internetworking architecture based on the Internet Routing Overlay Network (IRON) [RFC6179] and Routing and Addressing in Networks with Global Enterprise Recursion (RANGER) [RFC5720][RFC6139]. Many of the VET principles can be traced to the deliberations of the ROAD group in January 1992, and also to still earlier initiatives including NIMROD [RFC1753] and the Catenet model for internetworking [CATENET] [IEN48] [RFC2775]. The high-level architectural aspects of the ROAD group deliberations are captured in a "New Scheme for Internet Routing and Addressing (ENCAPS) for IPNG" [RFC1955].

VET is related to the present-day activities of the IETF INTAREA, AUTOCONF, DHC, IPv6, MANET, and V6OPS working groups, as well as the IRTF RRG working group.

2. Terminology

The mechanisms within this document build upon the fundamental principles of IP encapsulation. The term "inner" refers to the innermost {address, protocol, header, packet, etc.} *before* encapsulation, and the term "outer" refers to the outermost {address, protocol, header, packet, etc.} *after* encapsulation. VET also accommodates "mid-layer" encapsulations including the Subnetwork Encapsulation and Adaptation Layer (SEAL) [I-D.templin-intarea-seal], IPsec [RFC4301], etc.

The terminology in the normative references apply; the following terms are defined within the scope of this document:

Virtual Enterprise Traversal (VET)
: an abstraction that uses encapsulation to create virtual overlays for transporting inner network layer packets over outer IPv4 and IPv6 enterprise networks.

enterprise network
the same as defined in [RFC4852]. An enterprise network is further understood to refer to a cooperative networked collective of devices within a structured IP routing and addressing plan and with a commonality of business, social, political, etc., interests. Minimally, the only commonality of interest in some enterprise network scenarios may be the cooperative provisioning of connectivity itself.

subnetwork
the same as defined in [RFC3819].

site
a logical and/or physical grouping of interfaces that connect a topological area less than or equal to an enterprise network in scope. From a network organizational standpoint, a site within an enterprise network can be considered as an enterprise network unto itself.

Mobile Ad hoc Network (MANET)
a connected topology of mobile or fixed routers that maintain a routing structure among themselves over links that often have dynamic connectivity properties. The characteristics of MANETs are described in [RFC2501], Section 3, and a wide variety of MANETs share common properties with enterprise networks.

enterprise/site/MANET
throughout the remainder of this document, the term "enterprise network" is used to collectively refer to any of {enterprise, site, MANET}, i.e., the VET mechanisms and operational principles can be applied to enterprises, sites, and MANETs of any size or shape.

VET link
a virtual link that uses automatic tunneling to create an overlay network that spans an enterprise network routing region. VET links can be segmented (e.g., by filtering gateways) into multiple distinct segments that can be joined together by bridges or IP routers the same as for any link. Bridging would view the multiple (bridged) segments as a single VET link, whereas IP routing would view the multiple segments as multiple distinct VET links. VET links can further be partitioned into multiple logical areas, where each area is identified by a distinct set of border nodes.

VET links configured over non-multicast enterprise networks support only Non-Broadcast, Multiple Access (NBMA) services; VET links configured over enterprise networks that support multicast can support both NBMA and native multicast services. All nodes connected to the same VET link appear as neighbors from the standpoint of the inner network layer.

Enterprise Router (ER)
As depicted in Figure 1, an Enterprise Router (ER) is a fixed or mobile router that comprises a router function, a host function, one or more enterprise-interior interfaces, and zero or more internal virtual, enterprise-edge, provider-edge, and VET interfaces. At a minimum, an ER forwards outer IP packets over one or more sets of enterprise-interior interfaces, where each set connects to a distinct enterprise network.

VET Border Router (VBR)
an ER that connects edge networks to VET links and/or connects multiple VET links together. A VBR is a tunnel endpoint router, and it configures a separate VET interface for each distinct VET link. All VBRs are also ERs.

VET Border Gateway (VBG)
a VBR that connects VET links to provider networks. A VBG may alternately act as "half-gateway", and forward the packets it receives from neighbors on the VET link to another VBG on the same VET link. All VBGs are also VBRs.

VET host
any node (host or router) that configures a VET interface for host-operation only. Note that a node may configure some of its VET interfaces as host interfaces and others as router interfaces.

VET node
any node (host or router) that configures and uses a VET interface.

enterprise-interior interface
an ER’s attachment to a link within an enterprise network. Packets sent over enterprise-interior interfaces may be forwarded over multiple additional enterprise-interior interfaces within the enterprise network before they reach either their final destination or a border router/gateway. Enterprise-interior interfaces connect laterally within the IP network hierarchy.

enterprise-edge interface
a VBR's attachment to a link (e.g., an Ethernet, a wireless personal area network, etc.) on an arbitrarily complex edge network that the VBR connects to a VET link and/or a provider network. Enterprise-edge interfaces connect to lower levels within the IP network hierarchy.

provider-edge interface
a VBR's attachment to the Internet or to a provider network via which the Internet can be reached. Provider-edge interfaces connect to higher levels within the IP network hierarchy.

internal-virtual interface
an interface that is internal to a VET node and does not in itself directly attach to a tangible link, e.g., a loopback interface.

VET interface
a VET node's attachment to a VET link. VET nodes configure each VET interface over a set of underlying enterprise-interior interfaces that connect to a routing region spanned by a single VET link. When there are multiple distinct VET links (each with their own distinct set of underlying interfaces), the VET node configures a separate VET interface for each link.

The VET interface encapsulates each inner packet in any mid-layer headers followed by an outer IP header, then forwards the packet on an underlying interface such that the Time to Live (TTL) - Hop Limit in the inner header is not decremented as the packet traverses the link. The VET interface therefore presents an automatic tunneling abstraction that represents the VET link as a single hop to the inner network layer.

Provider Aggregated (PA) prefix
a network layer protocol prefix that is delegated to a VET node by a provider network.

Provider-(In)dependent (PI) address/prefix
a network layer protocol prefix that is delegated to a VET node by an independent prefix registration authority.

Routing Locator (RLOC)
a public-scope or enterprise-local-scope IP address that can appear in enterprise-interior and/or interdomain routing tables. Public-scope RLOCs are delegated to specific enterprise networks and routable within both the enterprise-interior and interdomain routing regions. Enterprise-local-scope RLOCs (e.g., IPv6 Unique Local Addresses [RFC4193], IPv4 privacy addresses [RFC1918], etc.) are self-generated by individual enterprise networks and routable

only within the enterprise-interior routing region.

ERs use RLOCs for operating the enterprise-interior routing protocol and for next-hop determination in forwarding packets addressed to other RLOCs. End systems can use RLOCs as addresses for end-to-end communications between peers within the same enterprise network. VET interfaces treat RLOCs as *outer* IP addresses during encapsulation.

Endpoint Interface iDentifier (EID)
a public-scope network layer address that is routable within enterprise-edge and/or VET overlay networks. In a pure mapping system, EID prefixes are not routable within the interdomain routing system. In a hybrid routing/mapping system, EID prefixes may be represented within the same interdomain routing instances that distribute RLOC prefixes. In either case, EID prefixes are separate and distinct from any RLOC prefix space, but they are mapped to RLOC addresses to support packet forwarding over VET interfaces.

VBRs participate in any EID-based routing instances and use EID addresses for next-hop determination. End systems can use EIDs as addresses for end-to-end communications between peers either within the same enterprise network or within different enterprise networks. VET interfaces treat EIDs as *inner* network layer addresses during encapsulation.

Note that an EID can also be used as an *outer* network layer address if there are nested encapsulations. In that case, the EID would appear as an RLOC to the innermost encapsulation.

The following additional acronyms are used throughout the document:

CGA - Cryptographically Generated Address
DHCP(v4, v6) - Dynamic Host Configuration Protocol
ECMP - Equal Cost Multi Path
FIB - Forwarding Information Base
ICMP - either ICMPv4 or ICMPv6
IP - either IPv4 or IPv6
ISATAP - Intra-Site Automatic Tunnel Addressing Protocol
NBMA - Non-Broadcast, Multiple Access
ND - Neighbor Discovery
PIO - Prefix Information Option
PRL - Potential Router List
PRLNAME - Identifying name for the PRL
RIB - Routing Information Base
RIO - Route Information Option
SCMP - SEAL Control Message Protocol

SEAL - Subnetwork Encapsulation and Adaptation Layer
SLAAC - IPv6 StateLess Address AutoConfiguration
SNS/SNA - SEAL Neighbor Solicitation/Advertisement
SRS/SRA - SEAL Router Solicitation/Advertisement

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 [RFC2119]. When used in lower case (e.g., must, must not, etc.), these words MUST NOT be interpreted as described in [RFC2119], but are rather interpreted as they would be in common English.

3. Enterprise Network Characteristics

Enterprise networks consist of links that are connected by Enterprise Routers (ERs) as depicted in Figure 1. ERs typically participate in a routing protocol over enterprise-interior interfaces to discover routes that may include multiple Layer 2 or Layer 3 forwarding hops. VET Border Routers (VBRs) are ERs that connect edge networks to VET links that span enterprise networks. VET Border Gateways (VBGs) are VBRs that connect VET links to provider networks.

Conceptually, an ER embodies both a host function and router function, and supports communications according to the weak end-system model [RFC1122]. The router function engages in the enterprise-interior routing protocol on its enterprise-interior interfaces, connects any of the ER's edge networks to its VET links, and may also connect the VET links to provider networks (see Figure 1). The host function typically supports network management applications, but may also support diverse applications typically associated with general-purpose computing platforms.

An enterprise network may be as simple as a small collection of ERs and their attached edge networks; an enterprise network may also contain other enterprise networks and/or be a subnetwork of a larger enterprise network. An enterprise network may further encompass a set of branch offices and/or nomadic hosts connected to a home office over one or several service providers, e.g., through Virtual Private Network (VPN) tunnels. Finally, an enterprise network may contain many internal partitions that are logical or physical groupings of nodes for the purpose of load balancing, organizational separation, etc. In that case, each internal partition resembles an individual segment of a bridged LAN.

Enterprise networks that comprise link types with sufficiently similar properties (e.g., Layer 2 (L2) address formats, maximum transmission units (MTUs), etc.) can configure a subnetwork routing service such that the inner network layer sees the underlying network as an ordinary shared link the same as for a (bridged) campus LAN (this is often the case with large cellular operator networks). In that case, a single inner network layer hop is sufficient to traverse the underlying network. Enterprise networks that comprise link types with diverse properties and/or configure multiple IP subnets must also provide an enterprise-interior routing service that operates as an IP layer mechanism. In that case, multiple inner network layer hops may be necessary to traverse the underlying network such that care must be taken to avoid multi-link subnet issues [RFC4903].

In addition to other interface types, VET nodes configure VET interfaces that view all other nodes on the VET link as neighbors on a virtual NBMA link. VET nodes configure a separate VET interface for each distinct VET link to which they connect, and discover neighbors on the link that can be used for forwarding packets to off-link destinations. VET interface neighbor relationships may be either unidirectional or bidirectional.

A unidirectional neighbor relationship is typically established and maintained as a result of network layer control protocol messaging in a manner that parallels IPv6 neighbor discovery [RFC4861]. A bidirectional neighbor relationship is typically established and maintained as result of a short transaction between the neighbors carried by a reliable transport protocol such as TCP. The protocol details of the transaction are out of scope for this document, and indeed need not be standardized as long as both neighbors observe the same specifications.

For each distinct VET link , a trust basis must be established and consistently applied. For example, for VET links configured over enterprise networks in which VBRs establish symmetric security associations, mechanisms such as IPsec [RFC4301] can be used to assure authentication and confidentiality. In other enterprise network scenarios, VET links may require asymmetric securing mechanisms such as SEcure Neighbor Discovery (SEND) [RFC3971]. VET links configured over still other enterprise networks may find it sufficient to employ additional encapsulations (e.g., SEAL [I-D.templin-intarea-seal]) that include a simple per-packet nonce to detect off-path attacks.

Finally, for VET links configured over enterprise networks with a centralized management structure (e.g., a corporate campus network, an ISP network, etc.), a hybrid routing/mapping service can be deployed using a synchronized set of VBGs. In that case, the VBGs can provide a "default mapper" [I-D.jen-apt] service used for short-term packet forwarding until route-optimized paths can be established. For VET links configured over enterprise networks with a distributed management structure (e.g., disconnected MANETs), peer-to-peer coordination between the VET nodes themselves without the assistance of VBGs may be required. Recognizing that various use cases will entail a continuum between a fully centralized and fully distributed approach, the following sections present the mechanisms of Virtual Enterprise Traversal as they apply to a wide variety of scenarios.

4. Autoconfiguration

ERs, VBRs, VBGs, and VET hosts configure themselves for operation as specified in the following subsections.

4.1. Enterprise Router (ER) Autoconfiguration

ERs configure enterprise-interior interfaces and engage in any routing protocols over those interfaces.

When an ER joins an enterprise network, it first configures an IPv6 link-local address on each enterprise-interior interface that requires an IPv6 link-local capability and configures an IPv4 link-local address on each enterprise-interior interface that requires an IPv4 link-local capability. IPv6 link-local address generation mechanisms include Cryptographically Generated Addresses (CGAs) [RFC3972], IPv6 Privacy Addresses [RFC4941], StateLess Address AutoConfiguration (SLAAC) using EUI-64 interface identifiers [RFC4291] [RFC4862], etc. The mechanisms specified in [RFC3927] provide an IPv4 link-local address generation capability.

Next, the ER configures one or more RLOCs and engages in any routing protocols on its enterprise-interior interfaces. The ER can configure RLOCs via administrative configuration, pseudo-random self-generation from a suitably large address pool, DHCP autoconfiguration, or through an alternate autoconfiguration mechanism.

Pseudo-random self-generation of IPv6 RLOCs can be from a large public or local-use IPv6 address range (e.g., IPv6 Unique Local Addresses [RFC4193]). Pseudo-random self-generation of IPv4 RLOCs can be from a large public or local-use IPv4 address range (e.g., [RFC1918]). When self-generation is used alone, the ER continuously monitors the RLOCs for uniqueness, e.g., by monitoring the enterprise-interior routing protocol. (Note however that anycast RLOCs may be assigned to multiple enterprise-interior interfaces; hence, monitoring for uniqueness applies only to RLOCs that are provisioned as unicast.)

DHCP autoconfiguration of RLOCs uses standard DHCP procedures, however ERs acting as DHCP clients SHOULD also use DHCP Authentication [RFC3118] [RFC3315] as discussed further below. In typical enterprise network scenarios (i.e., those with stable links), it may be sufficient to configure one or a few DHCP relays on each link that does not include a DHCP server. In more extreme scenarios (e.g., MANETs that include links with dynamic connectivity properties), DHCP operation may require any ERs that have already configured RLOCs to act as DHCP relays to ensure that client DHCP requests eventually reach a DHCP server. This may result in considerable DHCP message relaying until a server is located, but the DHCP Authentication Replay Detection vector provides relays with a means for avoiding message duplication.

In all enterprise network scenarios, the amount of DHCP relaying required can be significantly reduced if each relay has a way of contacting a DHCP server directly. In particular, if the relay can discover the unicast addresses for one or more servers (e.g., by discovering the unicast RLOC addresses of VBGs as described in Section 4.2.2) it can forward DHCP requests directly to the unicast address(es) of the server(s). If the relay does not know the unicast address of a server, it can forward DHCP requests to a site-scoped DHCP server multicast address if the enterprise network supports site-scoped multicast services. For DHCPv6, relays can forward requests to the site-scoped IPv6 multicast group address 'All_DHCP_Servers' [RFC3315]. For DHCPv4, relays can forward requests to the site-scoped IPv4 multicast group address 'All_DHCPv4_Servers', which SHOULD be set to 239.255.2.1 unless an alternate multicast group for the enterprise network is known. DHCPv4 servers that delegate RLOCs SHOULD therefore join the 'All_DHCPv4_Servers' multicast group and service any DHCPv4 messages received for that group.

A combined approach using both DHCP and self-generation is also possible when the ER configures both a DHCP client and relay that are connected, e.g., via a pair of back-to-back connected Ethernet interfaces, a tun/tap interface, a loopback interface, inter-process communication, etc. The ER first self-generates an RLOC taken from a temporary addressing range used only for the bootstrapping purpose of procuring an actual RLOC taken from a delegated addressing range. The ER then engages in the enterprise-interior routing protocol and performs a DHCP exchange as above using the temporary RLOC as the address of its relay function. When the DHCP server delegates an actual RLOC address/prefix, the ER abandons the temporary RLOC and re-engages in the enterprise-interior routing protocol using an RLOC taken from the delegation.

Alternatively (or in addition to the above), the ER can request RLOC prefix delegations via an automated prefix delegation exchange over an enterprise-interior interface and can assign the prefix(es) on enterprise-edge interfaces. Note that in some cases, the same enterprise-edge interfaces may assign both RLOC and EID addresses if there is a means for source address selection. In other cases (e.g., for separation of security domains), RLOCs and EIDs are assigned on separate sets of enterprise-edge interfaces.

In some enterprise network scenarios (e.g., MANETs that include links with dynamic connectivity properties), assignment of RLOCs on enterprise-interior interfaces as singleton addresses (i.e., as addresses with /32 prefix lengths for IPv4, or as addresses with /128 prefix lengths for IPv6) MAY be necessary to avoid multi-link subnet issues.

4.2. VET Border Router (VBR) Autoconfiguration

VBRs are ERs that configure and use one or more VET interfaces. In addition to the ER autoconfiguration procedures specified in Section 4.1, VBRs perform the following autoconfiguration operations.

4.2.1. VET Interface Initialization

VBRs configure a separate VET interface for each VET link, where each VET link spans a distinct sets of underlying links belonging to the same enterprise network. All nodes on the VET link appear as single-hop neighbors from the standpoint of the inner network layer protocol through the use of encapsulation.

The VBR binds each VET interface to one or more underlying interfaces, and uses the underlying interface addresses as RLOCs to serve as the outer source addresses for encapsulated packets. The VBR then assigns a link-local address to each VET interface if necessary. When IPv6 and IPv4 are used as the inner/outer protocols (respectively), the VBR can autoconfigure an IPv6 link-local address on the VET interface using a modified EUI-64 interface identifier based on an IPv4 RLOC address (see Section 2.2.1 of [RFC5342]). Link-local address configuration for other inner/outer protocol combinations is through administrative configuration, random self-generation (e.g., [RFC4941], etc.) or through an unspecified alternate method.

4.2.2. Potential Router List (PRL) Discovery

After initializing the VET interface, the VBR next discovers a Potential Router List (PRL) for the VET link that includes the RLOC addresses of VBGs. The PRL can be discovered through administrative configuration, information conveyed in the enterprise-interior routing protocol, an anycast VBG discovery message exchange, a DHCP option, etc. In multicast-capable enterprise networks, VBRs can also listen for advertisements on the 'rasadv' [RASADV] multicast group address.

When no other information is available, the VBR can resolve an identifying name for the PRL ('PRLNAME') formed as 'hostname.domainname', where 'hostname' is an enterprise-specific name string and 'domainname' is an enterprise-specific Domain Name System (DNS) suffix [RFC1035]. The VBR discovers 'PRLNAME' through administrative configuration, the DHCP Domain Name option [RFC2132], 'rasadv' protocol advertisements, link-layer information (e.g., an IEEE 802.11 Service Set Identifier (SSID)), or through some other means specific to the enterprise network. The VBR can also obtain 'PRLNAME' as part of an arrangement with a private-sector PI prefix vendor (see: Section 4.2.4).

In the absence of other information, the VBR sets the 'hostname' component of 'PRLNAME' to "isatapv2" and sets the 'domainname' component to an enterprise-specific DNS suffix (e.g., "example.com"). Isolated enterprise networks that do not connect to the outside world may have no enterprise-specific DNS suffix, in which case the 'PRLNAME' consists only of the 'hostname' component. (Note that the default hostname "isatapv2" is intentionally distinct from the convention specified in [RFC5214].)

After discovering 'PRLNAME', the VBR resolves the name into a list of RLOC addresses through a name service lookup. For centrally managed enterprise networks, the VBR resolves 'PRLNAME' using an enterprise-local name service (e.g., the DNS). For enterprises with no centralized management structure, the VBR resolves 'PRLNAME' using Link-Local Multicast Name Resolution (LLMNR) [RFC4795] over the VET interface. In that case, all VBGs in the PRL respond to the LLMNR query, and the VBR accepts the union of all responses.

4.2.3. Provider-Aggregated (PA) EID Prefix Autoconfiguration

VBRs that connect their enterprise networks to a provider network obtain Provider-Aggregated (PA) EID prefixes through stateful and/or stateless autoconfiguration mechanisms. The stateful and stateless approaches are discussed in the following subsections.

4.2.3.1. Stateful Prefix Delegation

For IPv4, VBRs acquire IPv4 PA EID prefixes through administrative configuration, an automated IPv4 prefix delegation exchange, etc.

For IPv6, VBRs acquire IPv6 PA EID prefixes through administrative configuration or through DHCPv6 Prefix Delegation exchanges with an VBG acting as a DHCP relay/server. In particular, the VBR (acting as a requesting router) can use DHCPv6 prefix delegation [RFC3633] over the VET interface to obtain prefixes from the VBG (acting as a delegating router). The VBR obtains prefixes using either a 2-message or 4-message DHCPv6 exchange [RFC3315]. When the VBR acts as a DHCPv6 client, it maps the IPv6 "All_DHCP_Relay_Agents_and_Servers" link- scoped multicast address to the VBG’s outer RLOC address.

To perform the 2-message exchange, the VBR’s DHCPv6 client function can send a Solicit message with an IA_PD option either directly or via the VBR’s own DHCPv6 relay function (see Section 4.1). The VBR’s VET interface then forwards the message using VET encapsulation (see: Section 5.4) to a VBG which either services the request or relays it further. The forwarded Solicit message will elicit a Reply message from the server containing prefix delegations. The VBR can also propose a specific prefix to the DHCPv6 server per Section 7 of [RFC3633]. The server will check the proposed prefix for consistency and uniqueness, then return it in the Reply message if it was able to perform the delegation.

After the VBR receives IPv4 and/or IPv6 prefix delegations, it can provision the prefixes on enterprise-edge interfaces as well as on other VET interfaces configured over child enterprise networks for which it acts as an VBG. The VBR can also provision the prefixes on enterprise-interior interfaces to service directly-attached hosts on the enterprise-interior link.

The prefix delegations remain active as long as the VBR continues to renew them via the delegating VBG before lease lifetimes expire. The lease lifetime also keeps the delegation state active even if communications between the VBR and delegating VBG are disrupted for a period of time (e.g., due to an enterprise network partition, power failure, etc.). Note however that if the VBR abandons or otherwise loses continuity with the prefixes, it may be obliged to perform network-wide renumbering if it subsequently receives a new and different set of prefixes.

Stateful prefix delegation for non-IP protocols is out of scope.

4.2.3.2. Stateless Prefix Delegation

When IPv6 and IPv4 are used as the inner and outer protocols, respectively, a stateless IPv6 PA prefix delegation capability is available using the mechanisms specified in [RFC5569][RFC5969]. VBRs can use these mechanisms to statelessly configure IPv6 PA prefixes that embed one of the VBR’s IPv4 RLOCs.

Using this stateless prefix delegation, if the IPv4 RLOC changes the IPv6 prefix also changes and the VBR is obliged to renumber any interfaces on which sub-prefixes from the delegated prefix are assigned. This method may therefore be most suitable for enterprise networks in which IPv4 RLOC assignments rarely change, or in enterprise networks in which only services that do not depend on a long-term stable IPv6 prefix (e.g., client-side web browsing) are used.

Stateless prefix delegation for other protocol combinations is out of scope.

4.2.4. Provider-(In)dependent (PI) EID Prefix Autoconfiguration

VBRs can acquire Provider (In)dependent (PI) prefixes to facilitate multihoming, mobility and traffic engineering without requiring site-wide renumbering events. These PI prefixes are made available to VBRs through a prefix delegation authority that may or may not be associated with a specific ISP.

VBRs that connect major enterprise networks (e.g., large corporations, academic campuses, ISP networks, etc.) to a parent enterprise network and/or the global Internet can acquire short PI prefixes (e.g., an IPv6 ::/20, an IPv4 /16, etc.) through a registration authority such as the Internet Assigned Numbers Authority (IANA) or a major regional Internet registry. VBRs that connect small enterprise networks (e.g., SOHO networks, MANETs, etc.) to a parent enterprise network can acquire longer PI prefixes through arrangements with a PI prefix delegation vendor.

After a VBR receives PI prefixes, it can sub-delegate portions of the prefixes on enterprise-edge interfaces, on child VET interfaces for which it is configured as a VBG and on enterprise-interior interfaces to service directly-attached hosts on the enterprise-interior link. The VBR can also sub-delegate portions of its PI prefixes to requesting routers connected to child enterprise networks. These requesting routers consider their sub-delegated portions of the PI prefix as PA, and consider the delegating routers as their points of connection to a provider network.

4.3. VET Border Gateway (VBG) Autoconfiguration

VBGs are VBRs that connect VET links configured over child enterprise networks to provider networks via provider-edge interfaces and/or via VET links configured over parent enterprise networks. A VBG may also act as a "half-gateway", in that it may need to forward the packets it receives from neighbors on the VET link via another VBG connected to the same VET link. This arrangement is seen in the IRON [RFC6179] client/server/relay architecture, in which a server "half-gateway" is a VBG that forwards packets with off-link destinations via a relay "half-gateway" VBG that connects the VET link to the provider network.

VBGs autoconfigure their provider-edge interfaces in a manner that is specific to the provider connections, and they autoconfigure their VET interfaces that were configured over parent VET links using the VBR autoconfiguration procedures specified in Section 4.2. For each of its VET interfaces connected to child VET links, the VBG initializes the interface the same as for an ordinary VBR (see Section 4.2.1). It then arranges to add one or more of its RLOCs associated with the child VET link to the PRL.

VBGs configure a DHCP relay/server on VET interfaces connected to child VET links that require DHCP services. VBGs may also engage in an unspecified anycast VBG discovery message exchange if they are configured to do so. Finally, VBGs respond to LLMNR queries for 'PRLNAME' on VET interfaces connected to VET links that span child enterprise networks with a distributed management structure.

4.4. VET Host Autoconfiguration

Nodes that cannot be attached via a VBR's enterprise-edge interface (e.g., nomadic laptops that connect to a home office via a Virtual Private Network (VPN)) can instead be configured for operation as a simple host on the VET link. Each VET host performs the same enterprise interior interfaces RLOC configuration procedures as specified for ERs in Section 4.1. The VET host next performs the same VET interface initialization and PRL discovery procedures as specified for VBRs in Section 4.2, except that it configures its VET interfaces as host interfaces (and not router interfaces). Note also that a node may be configured as a host on some VET interfaces and as an VBR/VBG on other VET interfaces.

A VET host may receive non-link-local addresses and/or prefixes to assign to the VET interface via DHCP exchanges and/or through information conveyed in Router Advertisements (RAs). If prefixes are provided, however, there must be assurance that either 1) the VET link will not partition, or 2) that each VET host interface connected to the VET link will configure a unique set of prefixes. VET hosts therefore depend on DHCP and/or RA exchanges to provide only addresses/prefixes that are appropriate for assignment to the VET interface according to these specific cases, and depend on the VBGs within the enterprise keeping track of which addresses/prefixes were assigned to which hosts.

When the VET host solicits a DHCP-assigned EID address/prefix over a (non-multicast) VET interface, it maps the DHCP relay/server multicast inner destination address to the outer RLOC address of a VBG that it has selected as a default router. The VET host then assigns any resulting DHCP-delegated addresses/prefixes to the VET interface for use as the source address of inner packets. The host will subsequently send all packets destined to EID correspondents via a default router on the VET link, and will discover more-specific routes based on any redirect messages it receives.

5. Internetworking Operation

Following the autoconfiguration procedures specified in Section 4, ERs, VBRs, VBGs, and VET hosts engage in normal internetworking operations as discussed in the following sections.

5.1. Routing Protocol Participation

ERs engage in any RLOC-based routing protocols over enterprise-interior interfaces to exchange routing information for forwarding IP packets with RLOC addresses. VBRs and VBGs can additionally engage in any EID-based routing protocols over VET, enterprise-edge and provider-edge interfaces to exchange routing information for forwarding inner network layer packets with EID addresses. Note that any EID-based routing instances are separate and distinct from any RLOC-based routing instances.

VBR/VBG routing protocol participation on non-multicast VET interfaces uses the NBMA interface model, e.g., in the same manner as for OSPF over NBMA interfaces [RFC5340]. (VBR/VBG routing protocol participation on multicast-capable VET interfaces can alternatively use the standard multicast interface model, but this may result in excessive multicast control message overhead.)

VBRs can use the list of VBGs in the PRL (see: Section 4.2.1) as an initial list of neighbors for EID-based routing protocol participation. VBRs can alternatively use the list of VBGs as potential default routers instead of engaging in an EID-based routing protocol instance. In that case, when the VBR forwards a packet via a default router it may receive a redirect message indicating a different VBR as a better next hop.

5.1.1. PI Prefix Routing Considerations

VBRs that connect large enterprise networks to the global Internet advertise their EID PI prefixes directly into the Internet default-free RIB via the Border Gateway Protocol (BGP) [RFC4271] the same as for a major service provider network. VBRs that connect large enterprise networks to provider networks can instead advertise their EID PI prefixes into the providers’ routing system(s) if the provider networks are configured to accept them.

VBRs that connect small enterprise networks to provider networks obtain one or more PI prefixes and register the prefixes with a serving VBG in the PI prefix vendor’s network (e.g., through a vendor-specific short http(s) transaction). The PI prefix vendor network then acts as a virtual "home" enterprise network that connects its customer small enterprise networks to the Internet routing system. The customer small enterprise networks in turn appear as mobile components of the PI prefix vendor’s network, i.e., the customer networks are always "away from home".

Further details on routing for PI prefixes is discussed in "The Internet Routing Overlay Network (IRON)" [RFC6179] and "Fib Suppression with Virtual Aggregation" [I-D.ietf-grow-va].

5.2. Default Route Configuration and Selection

Configuration of default routes in the presence of VET interfaces must be carefully coordinated according to the inner and outer network protocols. If the inner and outer protocols are different (e.g., IPv6 within IPv4) then default routes of the inner protocol version can be configured with next-hops corresponding to default routers on a VET interface while default routes of the outer protocol version can be configured with next-hops corresponding to default routers on an underlying interface.

If the inner and outer protocols are the same (e.g., IPv4 within IPv4), care must be taken in setting the default route to avoid ambiguity. For example, if default routes are configured on the VET interface then more-specific routes could be configured on underlying interfaces to avoid looping. In a preferred method, however, multiple default routes can be configured with some having next-hops corresponding to (EID-based) default routers on VET interfaces and others having next-hops corresponding to (RLOC-based) default routers on underlying interfaces. In that case, special next-hop determination rules must be used (see: Section 5.4).

5.3. Address Selection

When permitted by policy and supported by enterprise-interior routing, VET nodes can avoid encapsulation through communications that directly invoke the outer IP protocol using RLOC addresses instead of EID addresses for end-to-end communications. For example, an enterprise network that provides native IPv4 intra-enterprise services can provide continued support for native IPv4 communications even when encapsulated IPv6 services are available for inter-enterprise communications. In other enterprise network scenarios, the use of EID-based communications (i.e., instead of RLOC-based communications) may be necessary and/or beneficial to support address scaling, transparent Network Address Translator (NAT) traversal, security domain separation, site multihoming, traffic engineering, etc. .

VET nodes can use source address selection rules (e.g., based on name service information) to determine whether to use EID-based or RLOC-based addressing. The remainder of this section discusses internetworking operation for EID-based communications using the VET interface abstraction.

5.4. Next Hop Determination

VET nodes perform normal next-hop determination via longest prefix match, and send packets according to the most-specific matching entry in the FIB. If the FIB entry has multiple next-hop addresses, the VBR selects the next-hop with the best metric value. If multiple next hops have the same metric value, the VET node can use Equal Cost Multi Path (ECMP) to forward different flows via different next-hop addresses, where flows are determined, e.g., by computing a hash of the inner packet's source address, destination address and flow label fields.

If the VET node has multiple default routes of the same inner and outer protocol versions, with some corresponding to EID-based default routers and others corresponding to RLOC-based default routers, it must perform source address based selection of a default route. In particular, if the packet's source address is taken from an EID prefix the VET node selects a default route configured over the VET interface; otherwise, it selects a default route configured over an underlying interface.

As a last resort when there is no matching entry in the FIB (i.e., not even default), VET nodes can discover neighbors within the enterprise network through on-demand name service queries for the EID prefix taken from a packet's destination address (or, by some other inner address to outer address mapping mechanism). For example, for the IPv6 destination address '2001:DB8:1:2::1' and 'PRLNAME' "isatapv2.example.com" the VET node can perform a name service lookup for the domain name:
'0.0.1.0.0.0.8.b.d.0.1.0.0.2.ip6.isatapv2.example.com'.

Name-service lookups in enterprise networks with a centralized management structure use an infrastructure-based service, e.g., an enterprise-local DNS. Name-service lookups in enterprise networks with a distributed management structure and/or that lack an infrastructure-based name service instead use LLMNR over the VET interface.

When LLMNR is used, the VBR that performs the lookup sends an LLMNR query (with the prefix taken from the IP destination address encoded in dotted-nibble format as shown above) and accepts the union of all replies it receives from neighbors on the VET interface. When a VET node receives an LLMNR query, it responds to the query IFF it aggregates an IP prefix that covers the prefix in the query. If the name-service lookup succeeds, it will return RLOC addresses (e.g., in DNS A records) that correspond to neighbors to which the VET node can forward packets.

5.5. VET Interface Encapsulation/Decapsulation

VET interfaces encapsulate inner network layer packets in any necessary mid-layer headers and trailers (e.g., IPsec [RFC4301], etc.) followed by a SEAL header (if necessary) followed by an outer UDP header (if necessary) followed by an outer IP header. Following all encapsulations, the VET interface submits the encapsulated packet to the outer IP forwarding engine for transmission on an underlying interface. The following sections provide further details on encapsulation:

5.5.1. Inner Network Layer Protocol

The inner network layer protocol sees the VET interface as an ordinary network interface, and views the outer network layer protocol as an ordinary L2 transport. The inner- and outer network layer protocol types are mutually independent and can be used in any combination. Inner network layer protocol types include IPv6 [RFC2460] and IPv4 [RFC0791], but they may also include non-IP protocols such as OSI/CLNP [RFC0994][RFC1070][RFC4548].

5.5.2. Mid-Layer Encapsulation

VET interfaces that use mid-layer encapsulations encapsulate each inner network layer packet in any mid-layer headers and trailers as the first step in a potentially multi-layer encapsulation.

5.5.3. SEAL Encapsulation

Following any mid-layer encapsulations, VET interfaces that use SEAL add a SEAL header as specified in [I-D.templin-intarea-seal]. Inclusion of a SEAL header must be applied uniformly between all neighbors on the VET link. Note that when a VET interface sends a SEAL-encapsulated packet to a neighbor that does not use SEAL encapsulation, it may receive an ICMP "port unreachable" or "protocol unreachable" depending on whether/not an outer UDP header is included.

SEAL encapsulation is used on VET links that require path MTU mitigations due to encapsulation overhead and/or mechanisms for VET interface neighbor coordination. When SEAL encapsulation is used, the VET interface sets the 'Next Header' value in the SEAL header to the IP protocol number associated with either the mid-layer encapsulation or the IP protocol number of the inner network layer (if no mid-layer encapsulation is used). The VET interface sets the other fields in the SEAL header as specified in [I-D.templin-intarea-seal].

5.5.4. Outer UDP Header Encapsulation

Following any mid-layer and/or SEAL encapsulations, VET interfaces that use UDP encapsulation add an outer UDP header. Inclusion of an outer UDP header must be applied uniformly between all neighbors on the VET link. Note that when a VET interface sends a UDP-encapsulated packet to a neighbor that does not recognize the UDP port number, it may receive an ICMP "port unreachable" message.

VET interfaces use UDP encapsulation on VET links that may traverse NATs and/or legacy networking gear (e.g., Equal Cost MultiPath (ECMP) routers, Link Aggregation Gateways (LAGs), etc.) that only recognize well-known network layer protocols. When UDP encapsulation is used, the VET interface encapsulates the mid-layer packet in an outer UDP header then sets the UDP port numbers as specified for the outermost mid-layer protocol (e.g., IPsec [RFC3947][RFC3948], etc.).

When SEAL [I-D.templin-intarea-seal] is used as the outermost mid-layer protocol, the VET interface maintains per-neighbor local and remote UDP port numbers. For bidirectional neighbors, the interface sets the local UDP port number to the value reserved for SEAL and sets the remote UDP port number to the observed UDP source port number in packets that it receives from the neighbor. In cases in which one of the bidirectional neighbors is behind a NAT, this implies that the one behind the NAT initiates the neighbor relationship. If both neighbors have a way of knowing that there are no NATs in the path, then they may select and set port numbers as described for unidirectional neighbors below.

For unidirectional neighbors, the VET interface sets both the local and remote UDP port numbers to the value reserved for SEAL, and additionally selects a small set of dynamic port number values for use as additional local UDP port numbers. The VET interface then selects one of this set of local port numbers for the UDP source port for each inner packet it sends, where the port number is determined e.g., by a hash calculated over the inner network layer addresses and inner transport layer port numbers. The VET interface uses a hash function of its own choosing when selecting a dynamic port number value, but it should choose a function that provides uniform distribution between the set of values, and it shoud be consistent in the manner in which the hash is applied.

Finally, for VET links configured over IPv4 enterprise networks, the VET interface sets the UDP checksum field to zero. For VET links configured over IPv6 enterprise networks, considerations for setting the UDP checksum are discussed in [I-D.ietf-6man-udpzero].

5.5.5. Outer IP Header Encapsulation

Following any mid-layer, SEAL and/or UDP encapsulations, the VET interface adds an outer IP header. Outer IP header construction is the same as specified for ordinary IP encapsulation (e.g., [RFC2003], [RFC2473], [RFC4213], etc.) except that the "TTL/Hop Limit", "Type of Service/Traffic Class" and "Congestion Experienced" values in the inner network layer header are copied into the corresponding fields in the outer IP header. The VET interface also sets the IP protocol number to the appropriate value for the first protocol layer within the encapsulation (e.g., UDP, SEAL, IPsec, etc.). When IPv6 is used as the outer IP protocol, the VET interface sets the flow label value in the outer IPv6 header the same as described in [I-D.carpenter-flow-ecmp].

5.5.6. Decapsulation

When a VET interface receives an encapsulated packet, it retains the outer headers and processes the SEAL header as specified in [I-D.templin-intarea-seal].

Next, if the packet will be forwarded from the receiving VET interface into a forwarding VET interface, the VET node copies the "TTL/Hop Limit", "Type of Service/Traffic Class" and "Congestion Experienced" values in the outer IP header received on the receiving VET interface into the corresponding fields in the outer IP header to be sent over the forwarding VET interface (i.e., the values are transferred between outer headers and *not* copied from the inner network layer header). This is true even if the packet is forwarded out the same VET interface that it arrived on, and necessary to support diagnostic functions (e.g., traceroute) and avoid looping.

During decapsulation, when the next-hop is via a non-VET interface, the "Congestion Experienced" value in the outer IP header is copied into the corresponding field in the inner network layer header.

5.6. Mobility and Multihoming Considerations

VBRs that travel between distinct enterprise networks must either abandon their PA prefixes that are relative to the "old" network and obtain PA prefixes relative to the "new" network, or somehow coordinate with a "home" network to retain ownership of the prefixes. In the first instance, the VBR would be required to coordinate a network renumbering event on its attached networks using the new PA prefixes [RFC4192][RFC5887]. In the second instance, an adjunct mobility management mechanism is required.

VBRs can retain their PI prefixes as they travel between distinct network points of attachment as long as they continue to refresh their PI prefix to RLOC address mappings with their serving VBG as described in [RFC6179]. (When the VBR moves far from its serving VBG, it can also select a new VBG in order to maintain optimal routing.) In this way, VBRs can update their PI prefix to RLOC mappings in real time and without requiring an adjunct mobility management mechanism.

The VBGs of a multihomed enterprise network participate in a private inner network layer routing protocol instance (e.g., via an interior BGP instance) to accommodate network partitions/merges as well as intra-enterprise mobility events.

5.7. Neighbor Coordination on VET Interfaces using SEAL

VET interfaces that use SEAL use the SEAL Control Message Protocol (SCMP) as specified in Section 4.5 of [I-D.templin-intarea-seal] to coordinate reachability, routing information, and mappings between the inner and outer network layer protocols. SCMP directly parallels the IPv6 Neighbor Discovery (ND) [RFC4191][RFC4861] and ICMPv6 [RFC4443] protocols, but operates from within the tunnel and supports operation for any combinations of inner and outer network layer protocols.

VET and SEAL are specifically designed for encapsulation of inner network layer payloads over outer IPv4 and IPv6 networks as a link layer. VET interfaces that use SCMP therefore require a new Source/Target Link-Layer Address Option (S/TLLAO) format that encapsulates IPv4 addresses as shown in Figure 2 and IPv6 addresses as shown in Figure 3:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Type = 2   |   Length = 1  |           Reserved            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  IPv4 address (bytes 0 thru 3)                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: SCMP S/TLLAO Option for IPv4 RLOCs

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Type = 2   |   Length = 3  |           Reserved            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Reserved                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               IPv6 address (bytes 0 thru 3)                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               IPv6 address (bytes 4 thru 7)                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               IPv6 address (bytes 8 thru 11)                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               IPv6 address (bytes 12 thru 15)                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: SCMP S/TLLAO Option for IPv6 RLOCs

In addition, VET interfaces that use SCMP use a modified version of the Route Information Option (RIO) (see: [RFC4191]) formatted as shown in Figure 4:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Type = 24   |    Length     | Prefix Length |  AF |Prf|Resvd|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Route Lifetime                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Prefix (Variable Length)                    |
.                                                               .
.                                                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: SCMP Route Information Option Format

In this modified format, the VET interface sets the Route Lifetime and Prefix fields in the RIO option the same as specified in

[RFC4191]. It then sets the fields in the header as follows:

- o the 'Type', 'Prf', and 'Resvd' fields are set the same as specified in [RFC4191].

- o the 'Length' field is set to 1, 2, or 3 as specified in [RFC4191]. It is instead set to 4 if the 'Prefix Length' is greater than 128 and set to 5 if the 'Prefix Length' is greater than 192 (e.g., in order to accommodate longer prefixes of non-IP protocols).

- o the 'Prefix Length' field ranges from 0 to 255. The 'Prefix' field is 0, 8, 16, 24 or 32 octets depending on the Length, and the embedded prefix MAY be up to 255 bits in length.

- o bits 24 - 26 are used to contain an 'Address Family (AF)' value that indicates the embedded prefix protocol type. This document defines the following values for AF:

  * 000 - IPv4

  * 001 - IPv6

  * 010 - OSI/CLNP NSAP

The following subsections discuss VET interface neighbor coordination using SCMP:

5.7.1. Router Discovery

VET hosts and VBRs can send SCMP Router Solicitation (SRS) messages to one or more VBGs in the PRL to receive solicited SCMP Router Advertisements (SRAs).

When an VBG receives an SRS message on a VET interface, it prepares a solicited SRA message. The SRA includes Router Lifetimes, Default Router Preferences, PIOs and any other options/parameters that the VBG is configured to include. If necessary, the VBG also includes Route Information Options (RIOs) formatted as specified above.

The VBG finally includes one or more SLLAOs formatted as specified above that encode the IPv6 and/or IPv4 RLOC unicast addresses of its own enterprise-interior interfaces or the enterprise-interior interfaces of other nearby VBGs.

5.7.2. Neighbor Unreachability Detection

VET nodes perform Neighbor Unreachability Detection (NUD) on VET interface neighbors by monitoring hints of forward progress enabled by SEAL mechanisms as evidence that a neighbor is reachable. First, when data packets are flowing, the VET node can periodically set the A bit in the SEAL header of data packets to elicit SCMP responses from the neighbor. Secondly, when no data packets are flowing, the VET node can send periodic probes such as SCMP Neighbor Solicitation (SNS) messages for the same purpose.

Responsiveness to routing changes is directly related to the delay in detecting that a neighbor has gone unreachable. In order to provide responsiveness comparable to dynamic routing protocols, a reasonably short neighbor reachable time (e.g., 5sec) SHOULD be used.

Additionally, a VET node may receive outer IP ICMP "Destination Unreachable; net / host unreachable" messages from an ER on the path indicating that the path to a neighbor may be failing. The node SHOULD first check the packet-in-error to obtain reasonable assurance that the ICMP message is authentic. If the node receives excessive ICMP unreachable errors through multiple RLOCs associated with the same FIB entry, it SHOULD delete the FIB entry and allow subsequent packets to flow through a different route (e.g., a default route with a VBG as the next hop).

5.7.3. Redirect Function

[[UNDER CONSTRUCTION]]

This section will be updated to reflect the new technique known as "Predirection" as discussed for ISATAP updates in Section 5.14.

[[UNDER CONSTRUCTION]]

5.8. Neighbor Coordination on VET Interfaces using IPsec

VET interfaces that use IPsec encapsulation use the Internet Key Exchange protocol, version 2 (IKEv2) [RFC4306] to manage security association setup and maintenance. IKEv2 provides a logical equivalent of the SCMP in terms of VET interface neighbor coordinations; for example, IKEv2 also provides mechanisms for redirection [RFC5685] and mobility [RFC4555].

IPsec additionally provides an extended Identification field and integrity check vector; these features allow IPsec to utilize outer IP fragmentation and reassembly with less risk of exposure to data corruption due to reassembly misassociations. On the other hand, IPsec entails the use of symmetric security associations and hence may not be appropriate to all enterprise network use cases.

5.9. Multicast

5.9.1. Multicast over (Non)Multicast Enterprise Networks

Whether or not the underlying enterprise network supports a native multicasting service, the VET node can act as an inner network layer IGMP/MLD proxy [RFC4605] on behalf of its attached edge networks and convey its multicast group memberships over the VET interface to a VBG acting as a multicast router. Its inner network layer multicast transmissions will therefore be encapsulated in outer headers with the unicast address of the VBG as the destination.

5.9.2. Multicast Over Multicast-Capable Enterprise Networks

In multicast-capable enterprise networks, ERs provide an enterprise-wide multicasting service (e.g., Simplified Multicast Forwarding (SMF) [I-D.ietf-manet-smf], Protocol Independent Multicast (PIM) routing, Distance Vector Multicast Routing Protocol (DVMRP) routing, etc.) over their enterprise-interior interfaces such that outer IP multicast messages of site-scope or greater scope will be propagated across the enterprise network. For such deployments, VET nodes can optionally provide a native inner multicast/broadcast capability over their VET interfaces through mapping of the inner multicast address space to the outer multicast address space. In that case, operation of link-or greater-scoped inner multicasting services (e.g., a link-scoped neighbor discovery protocol) over the VET interface is available, but SHOULD be used sparingly to minimize enterprise-wide flooding.

VET nodes encapsulate inner multicast messages sent over the VET interface in any mid-layer headers (e.g., UDP, SEAL, IPsec, etc.) followed by an outer IP header with a site-scoped outer IP multicast address as the destination. For the case of IPv6 and IPv4 as the inner/outer protocols (respectively), [RFC2529] provides mappings from the IPv6 multicast address space to a site-scoped IPv4 multicast address space (for other encapsulations, mappings are established through administrative configuration or through an unspecified alternate static mapping).

Multicast mapping for inner multicast groups over outer IP multicast groups can be accommodated, e.g., through VET interface snooping of inner multicast group membership and routing protocol control messages. To support inner-to-outer multicast address mapping, the VET interface acts as a virtual outer IP multicast host connected to its underlying interfaces. When the VET interface detects that an inner multicast group joins or leaves, it forwards corresponding outer IP multicast group membership reports on an underlying interface over which the VET interface is configured. If the VET node is configured as an outer IP multicast router on the underlying interfaces, the VET interface forwards locally looped-back group membership reports to the outer IP multicast routing process. If the VET node is configured as a simple outer IP multicast host, the VET interface instead forwards actual group membership reports (e.g., IGMP messages) directly over an underlying interface.

Since inner multicast groups are mapped to site-scoped outer IP multicast groups, the VET node MUST ensure that the site-scoped outer IP multicast messages received on the underlying interfaces for one VET interface do not "leak out" to the underlying interfaces of another VET interface. This is accommodated through normal site-scoped outer IP multicast group filtering at enterprise network boundaries.

5.10. Service Discovery

VET nodes can perform enterprise-wide service discovery using a suitable name-to-address resolution service. Examples of flooding-based services include the use of LLMNR [RFC4795] over the VET interface or multicast DNS (mDNS) [I-D.cheshire-dnsext-multicastdns] over an underlying interface. More scalable and efficient service discovery mechanisms (e.g., anycast) are for further study.

5.11. VET Link Partitioning

A VET link can be partitioned into multiple distinct logical groupings. In that case, each partition configures its own distinct 'PRLNAME' (e.g., 'isatapv2.zone1.example.com', 'isatapv2.zone2.example.com', etc.).

VBGs can further create multiple IP subnets within a partition, e.g., by sending SRAs with PIOs containing different IP prefixes to different groups of VET hosts. VBGs can identify subnets, e.g., by examining RLOC prefixes, observing the enterprise-interior interfaces over which SRSs are received, etc.

In the limiting case, VBGs can advertise a unique set of IP prefixes to each VET host such that each host belongs to a different subnet (or set of subnets) on the VET interface.

5.12. VBG Prefix State Recovery

VBGs retain explicit state that tracks the inner network layer prefixes delegated to VBRs connected to the VET link, e.g., so that packets are delivered to the correct VBRs. When a VBG loses some or all of its state (e.g., due to a power failure), client VBRs must refresh the VBG's state so that packets can be forwarded over correct routes.

5.13. Legacy ISATAP Services

VBGs can support legacy ISATAP services according to the specifications in [RFC5214]. In particular, VBGs can configure legacy ISATAP interfaces and VET interfaces over the same sets of underlying interfaces as long as the PRLs and IPv6 prefixes associated with the ISATAP/VET interfaces are distinct.

Legacy ISATAP hosts acquire addresses and/or prefixes in the same manner and using the same mechanisms as described for VET hosts in Section 4.4 above.

In order to support dynamic on-demand routing on ISATAP interfaces, a new (and backwards-compatible) approach called "ISATAP Predirection" is specified in the following sections:

5.14. ISATAP Update

In order to support dynamic on-demand routing on ISATAP interfaces, a new (and backwards-compatible) approach called "ISATAP Predirection" is specified in the following sections. This section updates [RFC5214].

5.14.1. ISATAP Predirection

Figure 5 depicts a reference ISATAP network topology. The scenario shows an advertising ISATAP router ('A'), two non-advertising ISATAP routers ('B', 'D') and two ordinary IPv6 hosts ('C', 'E') in a typical deployment configuration:

```
                    .-(::::::::)
                 .-(::: IPv6 :::)-.
                (:::: Internet ::::)
                 '-(::::::::::::)-'
                    '-(::::::)-'
                         ,-.
           ,-----+-/-+--'   \+------.
          /     ,~~~~~~~~~~~~~~~~~,  :
         /      |companion gateway|  |.
      ,-'       '~~~~~~~~~~~~~~~~~'   '.
     ;           +--------------+       )
     :           |   Router A   |      /
      :          |   (isatap)   |     ;
      +-         +--------------+    -+
     ;         fe80::5efe:192.0.2.1   :
     |                                ;
     :       IPv4 Provider Network  -+-'
      '-.       (PRL: 192.0.2.1)     .)
         \                         _)
          '-----+--------)----+'----'
   fe80::5efe:192.0.2.2     fe80::5efe:192.0.2.3       .-.
     +--------------+         +--------------+     ,-(  _)-.
     |   (isatap)   |         |   (isatap)   |  .-(_ IPv6  )-.
     |   Router B   |         |   Router D   |--(__Edge Network )
     +--------------+         +--------------+   '-(______)-'
      2001:db8::/48            2001:db8:1::/48         |
            |                                    2001:db8:1::1
           .-.                                  +-------------+
         ,-(  _)-.       2001:db8::1            | IPv6 Host E |
      .-(_ IPv6  )-.   +-------------+          +-------------+
    (__Edge Network )--| IPv6 Host C |
       '-(______)-'    +-------------+
```

Figure 5: Reference ISATAP Network Topology

With reference to Figure 5, when router 'A' receives an IPv6 packet on an advertising ISATAP interface that it will forward back out the same interface, 'A' must arrange to redirect the originating ISATAP node 'B' to a better next hop ISATAP node 'D' that is closer to the final destination 'E'. First, however, 'A' must direct 'D' to establish a forwarding table entry so that it will have a means to determine that 'B' is authorized to produce packets using a given source address. This process is accommodated via a unidirectional reliable exchange in which 'A' first informs 'D', then 'D' informs 'B' via 'A' as a trusted intermediary. 'B' therefore knows that 'D' will accept the packets it sends as long as 'D' retains the forwarding table entry. We call this process "predirection", which stands in contrast to ordinary IPv6 redirection.

Consider the alternative in which 'A' informs both 'B' and 'D' separately via independent IPv6 Redirect messages (see: [RFC4861]). In that case, several conditions can occur that could result in communications failures. First, if 'B' receives the Redirect message but 'D' does not, subsequent packets sent by 'B' would disappear into a black hole since 'D' would not have a forwarding table entry to verify their source addresses. Second, if 'D' receives the Redirect message but 'B' does not, subsequent packets sent in the reverse direction by 'D' would be lost. Finally, timing issues surrounding the establishment and garbage collection of forwarding table entries at 'B' and 'D' could yield unpredictable behavior. For example, unless the timing were carefully coordinated through some form of synchronization loop, there would invariably be instances in which one node has the correct forwarding table state and the other node does not resulting in non-deterministic packet loss.

The following subsections discuss the predirection steps that support the reference operational scenario:

5.14.1.1. 'A' Sends Predirect Forward To 'D'

When 'A' forwards an original IPv6 packet sent by 'B' out the same ISATAP interface that it arrived on, it sends a "Predirect" message forward toward 'D' instead of sending a Redirect message back to 'B'. The Predirect message is simply an ISATAP-specific version of an ordinary IPv6 Redirect message as depicted in Section 4.5 of [RFC4861], and is identified by two new backward-compatible bits taken from the Reserved field as shown in Figure 6:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Type (=137)  |   Code (=0)   |          Checksum             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |I|P|                       Reserved                            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                                                               +
   |                                                               |
   +                       Target Address                          +
   |                                                               |
   +                                                               +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   +                                                               +
   |                                                               |
   +                     Destination Address                       +
   |                                                               |
   +                                                               +
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   Options ...
   +-+-+-+-+-+-+-+-+-+-+-+-
```

Figure 6: ISATAP-Specific IPv6 Redirect Message Format

Where the new bits are defined as:

I (1) the "ISATAP" bit. Set to 1 to indicate an ISATAP-specific Redirect message, and set to 0 to indicate an ordinary IPv6 Redirect message.

P (1) the "Predirect" bit. Set to 1 to indicate a Predirect message, and set to 0 to indicate a Redirect response to a Predirect message. (This bit is valid only when the I bit is set to 1.)

Using this new Predirect message format, ’A’ prepares the message in a similar fashion as for an ordinary ISATAP-encapsulated IPv6 Redirect message as follows:

o the outer IPv4 source address is set to ’A’s IPv4 address.

o the outer IPv4 destination address is set to ’D’s IPv4 address.

- o the inner IPv6 source address is set to 'A's ISATAP link-local address.

- o the inner IPv6 destination address is set to 'D's ISATAP link-local address.

- o the Predirect Target and Destination Addresses are both set to 'B's ISATAP link-local address.

- o the Predirect message includes Route Information Options (RIOs) [RFC4191] that encode an IPv6 prefix taken from 'B's address/prefix delegations that covers the IPv6 source address of the originating IPv6 packet.

- o the Predirect message includes a Redirected Header Option (RHO) that contains at least the header of the originating IPv6 packet.

- o the I and P bits in the Predirect message header are both set to 1.

'A' then sends the Predirect message forward to 'D'.

5.14.1.2. 'D' Processes the Predirect and Sends Redirect Back To 'A'

When 'D' receives the Predirect message, it decapsulates the message according to Section 7.3 of [RFC5214] since the outer IPv4 source address is a member of the PRL.

'D' then uses the message validation checks specified in Section 8.1 of [RFC4861], except that instead of verifying that the "IP source address of the Redirect is the same as the current first-hop router for the specified ICMP Destination Address" (i.e., the 6th verification check), it accepts the message if the "outer IP source address of the Predirect is the same as the current first-hop router for the destination address of the originating IPv6 packet encapsulated in the RHO". (Note that this represents an ISATAP-specific adaptation of the verification checks.) Finally, 'D' only accepts the message if the destination address of the originating IPv6 packet encapsulated in the RHO is covered by one of its CURRENT delegated addresses/prefixes (see Section 5.14.4).

'D' then either creates or updates an IPv6 forwarding table entry with the prefix encoded in the RIO option as the target prefix, and the IPv6 Target Address of the Predirect message (i.e., 'B's ISATAP link-local address) as the next hop. 'D' places the entry in the FILTERING state, then sets/resets a filtering expiration timer value of 40 seconds. If the filtering timer expires, the node clears the FILTERING state and deletes the forwarding table entry if it is not in the FORWARDING state. This suggests that 'D's ISATAP interface should maintain a private forwarding table separate from the common IPv6 forwarding table, since the entry must be managed by the ISATAP interface itself.

After processing the Predirect message and establishing the forwarding table entry, 'D' prepares an ISATAP Redirect message in response to the Predirect as follows:

- o the outer IPv4 source address is set to 'D's IPv4 address.
- o the outer IPv4 destination address is set to 'A's IPv4 address.
- o the inner IPv6 source address, is set to 'D's ISATAP link-local address.
- o the inner IPv6 destination address is set to 'A's ISATAP link-local address.
- o the Redirect Target and the Redirect Destination Addresses are both set to 'D's ISATAP link-local address.
- o the Redirect message includes RIOs that encode IPv6 prefixes taken from 'D's address/prefix delegations that covers the IPv6 destination address of the originating IPv6 packet encapsulated in the Redirected Header option of the Predirect.
- o the Redirect message includes an RHO copied from the corresponding Predirect message.
- o the (I, P) bits in the Redirect message header are set to (1, 0).

'D' then sends the Redirect message to 'A'.

5.14.1.3. 'A' Processes the Redirect then Proxies it Back To 'B'

When 'A' receives the Redirect message, it decapsulates the message according to Section 7.3 of [RFC5214] since the inner IPv6 source address embeds the outer IPv4 source address.

'A' next accepts the message only if it satisfies the same message validation checks specified for Predirects in Section 3.2.4.6.2.

'A' then locates a forwarding table entry that covers the IPv6 source address of the packet segment in the RHO (i.e., a forwarding table entry with next hop 'B'), then proxies the Redirect message back toward 'B'. Without decrementing the IPv6 hop limit in the Redirect message, 'A' next changes the IPv4 source address of the Redirect message to its own IPv4 address, changes the IPv4 destination address to 'B's IPv4 address, changes the IPv6 source address to its own IPv6 link-local address, and changes the IPv6 destination address to 'B's IPv6 link-local address. 'A' then sends the proxied Redirect message to 'B'.

5.14.1.4. 'B' Processes The Redirect Message

When 'B' receives the Redirect message, it decapsulates the message according to Section 7.3 of [RFC5214] since the outer IPv4 source address is a member of the PRL.

'B' next accepts the message only if it satisfies the same message validation checks specified for Predirects in Section 3.2.4.6.2.

'B' then either creates or updates an IPv6 forwarding table entry with the prefix encoded in the RIO option as the target prefix, and the IPv6 Target Address of the Redirect message (i.e., 'D's ISATAP link-local address) as the next hop. 'B' places the entry in the FORWARDING state, then sets/resets a forwarding expiration timer value of 30 seconds. If the forwarding timer expires, the node clears the FORWARDING state and deletes the forwarding table entry if it is not in the FILTERING state. Again, this suggests that 'B's ISATAP interface should maintain a private forwarding table separate from the common IPv6 forwarding table, since the entry must be managed by the ISATAP interface itself.

Now, 'B' has a forwarding table entry in the FORWARDING state, and 'D' has a forwarding table entry in the FILTERING state. Therefore, 'B' may send ordinary IPv6 data packets with destination addresses covered by 'D's prefix directly to 'D' without involving 'A'. 'D' will in turn accept the packets since it has a forwarding table entry authorizing 'B' to source packets from its claimed IPv6 address.

To enable packet forwarding from 'D' directly to 'B', a reverse-predirection operation is required which is the mirror-image of the forward-predirection operation described above. Following the reverse predirection, both 'B' and 'D' will have forwarding table entries in the "(FORWARDING | FILTERING)" state, and IPv6 packets can be exchanged bidirectionally without involving 'A'.

5.14.1.5. 'B' Sends Periodic Predirect Messages Forward to 'A'

In order to keep forwarding table entries alive while data packets are actively flowing, 'B' can periodically send additional Predirect messages via 'A' to solicit Redirect messages from 'D'. When 'B' forwards an IPv6 packet via 'D', and the corresponding forwarding table entry FORWARDING state timer is nearing expiration, 'B' sends

Predirect messages (subject to rate limiting) prepared as follows:

- o the outer IPv4 source address is set to 'B's IPv4 address.
- o the outer IPv4 destination address is set to 'A's IPv4 address.
- o the inner IPv6 source address is set to 'B's ISATAP link-local address.
- o the inner IPv6 destination address is set to 'A's ISATAP link-local address.
- o the Predirect Target and Destination Addresses are both set to 'B's ISATAP link-local address.
- o the Predirect message includes RIOs that encode IPv6 prefixes taken from 'B's address/prefix delegations that cover the IPv6 source address of the originating IPv6 packet.
- o the Predirect message includes an RHO that contains at least the header of the originating IPv6 packet.
- o the I and P bits in the Predirect message header are both set to 1.

When 'A' receives the Predirect message, it decapsulates the message according to Section 7.3 of [RFC5214] since the inner IPv6 source address embeds the outer IPv4 source address.

'A' next accepts the message only if it satisfies the same message validation checks specified for Predirects in Section 3.2.4.6.2.

'A' then locates a forwarding table entry that covers the IPv6 destination address of the packet segment in the RHO (in this case, a forwarding table entry with next hop 'D'). Without decrementing the IPv6 hop limit in the Redirect message, 'A' next changes the IPv4 source address of the Predirect message to its own IPv4 address, changes the IPv4 destination address to 'D's IPv4 address, changes the IPv6 source address to its own IPv6 link-local address, and changes the IPv6 destination address to 'D's IPv6 link-local address. 'A' then sends the proxied Predirect message to 'D'. When 'D' receives the proxied message, it processes the message the same as if it had originated from 'A' as described in Section 3.2.4.6.2.

5.14.2. Scaling Considerations

Figure 5 depicts an ISATAP network topology with only a single advertising ISATAP router within the provider network. In order to support larger numbers of non-advertising ISATAP routers and ISATAP hosts, the provider network can deploy more advertising ISATAP routers to support load balancing and generally shortest-path routing.

Such an arrangement requires that the advertising ISATAP routers participate in an IPv6 routing protocol instance so that IPv6 address/prefix delegations can be mapped to the correct router. The routing protocol instance can be configured as either a full mesh topology involving all advertising ISATAP routers, or as a partial mesh topology with each ISATAP router associating with one or more companion gateways and a full mesh between companion gateways.

5.14.3. Proxy Chaining

In large ISATAP deployments, there may be many advertising ISATAP routers, each serving many ISATAP clients (i.e., both non-advertising routers and simple hosts). The advertising ISATAP routers then either require full topology knowledge, or a default route to a companion gateway that does have full topology knowledge. For example, if Client 'A' connects to advertising ISATAP router 'B', and Client 'E' connects to advertising ISATAP router 'D', then 'B' and 'D' must either have full topology knowledge or have a default route to a companion gateway (e.g., 'C') that does.

In that case, when 'A' sends an initial packet to 'E', 'B' generates a Predirect message toward 'C', which proxies the message toward 'D' which finally proxies the message toward 'E'.

In the reverse direction, when 'E' sends a Redirect response message to 'A', it first sends the message to 'D', which proxies the message toward 'C', which proxies the message toward 'B', which finally proxies the message toward 'A'.

5.14.4. Mobility

An ISATAP router 'A' can configure both a non-advertising ISATAP interface on a provider network and an advertising ISATAP interface on an edge network. In that case, 'A' can service ISATAP clients (i.e. both non-advertising routers and simple hosts) within the edge network by acting as a DHCPv6 relay. When a client 'B' in the edge network that has obtained IPv6 addresses/prefixes moves to a different edge network, however, 'B' can release its address/prefix delegations via 'A' and re-establish them via a different ISATAP router 'C' in the new edge network.

When 'B' releases its address/prefix delegations via 'A', 'A' marks the IPv6 forwarding table entries that cover the addresses/prefixes as DEPARTED (i.e., it clears the CURRENT state). 'A' therefore ceases to respond to Predirect messages correlated with the DEPARTED entries, and also schedules a garbage-collection timer of 60 seconds, after which it deletes the DEPARTED entries.

When 'A' receives IPv6 packets destined to an address covered by the DEPARTED IPv6 forwarding table entries, it forwards them to the last-known edge network link-layer address of 'B' as a means for avoiding mobility-related packet loss during routing changes. Eventually, correspondents will receive new Redirect messages from the network to discover that 'B' is now associated with 'C'.

Note that this mobility management method works the same way when the edge networks comprise native IPv6 links (i.e., and not just for ISATAP links), however any IPv6 packets forwarded by 'A' via an IPv6 forwarding table entry in the DEPARTED state may be lost if the mobile node moves off-link with respect to its previous edge network point of attachment. This should not be a problem for large links (e.g., large cellular network deployments, large ISP networks, etc.) in which all/most mobility events are intra-link.

6. IANA Considerations

There are no IANA considerations for this document.

7. Security Considerations

Security considerations for MANETs are found in [RFC2501].

The security considerations found in [RFC2529][RFC5214][I-D.nakibly-v6ops-tunnel-loops] also apply to VET.

SEND [RFC3971] and/or IPsec [RFC4301] can be used in environments where attacks on the neighbor coordination protocol are possible. SEAL [I-D.templin-intarea-seal] provides a per-packet identification that can be used to detect source address spoofing.

Rogue neighbor coordination messages with spoofed RLOC source addresses can consume network resources and cause VET nodes to perform extra work. Nonetheless, VET nodes SHOULD NOT "blacklist" such RLOCs, as that may result in a denial of service to the RLOCs' legitimate owners.

VBRs and VBGs observe the recommendations for network ingress filtering [RFC2827].

8. Related Work

Brian Carpenter and Cyndi Jung introduced the concept of intra-site automatic tunneling in [RFC2529]; this concept was later called: "Virtual Ethernet" and investigated by Quang Nguyen under the guidance of Dr. Lixia Zhang. Subsequent works by these authors and their colleagues have motivated a number of foundational concepts on which this work is based.

Telcordia has proposed DHCP-related solutions for MANETs through the CECOM MOSAIC program.

The Naval Research Lab (NRL) Information Technology Division uses DHCP in their MANET research testbeds.

Security concerns pertaining to tunneling mechanisms are discussed in [I-D.ietf-v6ops-tunnel-security-concerns].

Default router and prefix information options for DHCPv6 are discussed in [I-D.droms-dhc-dhcpv6-default-router].

An automated IPv4 prefix delegation mechanism is proposed in [I-D.ietf-dhc-subnet-alloc].

RLOC prefix delegation for enterprise-edge interfaces is discussed in [I-D.clausen-manet-autoconf-recommendations].

MANET link types are discussed in [I-D.clausen-manet-linktype].

The LISP proposal [I-D.ietf-lisp] examines encapsulation/ decapsulation issues and other aspects of tunneling.

Various proposals within the IETF have suggested similar mechanisms.

9. Acknowledgements

The following individuals gave direct and/or indirect input that was essential to the work: Jari Arkko, Teco Boot, Emmanuel Bacelli, Fred Baker, James Bound, Scott Brim, Brian Carpenter, Thomas Clausen, Claudiu Danilov, Chris Dearlove, Remi Despres, Gert Doering, Ralph Droms, Washam Fan, Dino Farinacci, Vince Fuller, Thomas Goff, David Green, Joel Halpern, Bob Hinden, Sascha Hlusiak, Sapumal Jayatissa, Dan Jen, Darrel Lewis, Tony Li, Joe Macker, David Meyer, Gabi Nakibly, Thomas Narten, Pekka Nikander, Dave Oran, Alexandru Petrescu, Mark Smith, John Spence, Jinmei Tatuya, Dave Thaler, Mark Townsley, Ole Troan, Michaela Vanderveen, Robin Whittle, James Woodyatt, Lixia Zhang, and others in the IETF AUTOCONF and MANET

working groups. Many others have provided guidance over the course of many years.

10. Contributors

The following individuals have contributed to this document:

Eric Fleischman (eric.fleischman@boeing.com)
Thomas Henderson (thomas.r.henderson@boeing.com)
Steven Russert (steven.w.russert@boeing.com)
Seung Yi (seung.yi@boeing.com)

Ian Chakeres (ian.chakeres@gmail.com) contributed to earlier versions of the document.

Jim Bound's foundational work on enterprise networks provided significant guidance for this effort. We mourn his loss and honor his contributions.

11. References

11.1. Normative References

[I-D.templin-intarea-seal]
Templin, F., "The Subnetwork Encapsulation and Adaptation Layer (SEAL)", draft-templin-intarea-seal-28 (work in progress), February 2011.

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, September 1981.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, May 2000.

[RFC3118] Droms, R. and W. Arbaugh, "Authentication for DHCP Messages", RFC 3118, June 2001.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC3633] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, December 2003.

[RFC3971] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

[RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", RFC 3972, March 2005.

[RFC4191] Draves, R. and D. Thaler, "Default Router Preferences and More-Specific Routes", RFC 4191, November 2005.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[RFC4443] Conta, A., Deering, S., and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, March 2006.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[RFC5342] Eastlake, D., "IANA Considerations and IETF Protocol Usage for IEEE 802 Parameters", BCP 141, RFC 5342, September 2008.

11.2. Informative References

[CATENET] Pouzin, L., "A Proposal for Interconnecting Packet Switching Networks", May 1974.

[I-D.carpenter-flow-ecmp] Carpenter, B. and S. Amante, "Using the IPv6 flow label for equal cost multipath routing and link aggregation in tunnels", draft-carpenter-flow-ecmp-03 (work in progress), October 2010.

[I-D.cheshire-dnsext-multicastdns]
Cheshire, S. and M. Krochmal, "Multicast DNS", draft-cheshire-dnsext-multicastdns-14 (work in progress), February 2011.

[I-D.clausen-manet-autoconf-recommendations]
Clausen, T. and U. Herberg, "MANET Router Configuration Recommendations", draft-clausen-manet-autoconf-recommendations-00 (work in progress), February 2009.

[I-D.clausen-manet-linktype]
Clausen, T., "The MANET Link Type", draft-clausen-manet-linktype-00 (work in progress), October 2008.

[I-D.droms-dhc-dhcpv6-default-router]
Droms, R. and T. Narten, "Default Router and Prefix Advertisement Options for DHCPv6", draft-droms-dhc-dhcpv6-default-router-00 (work in progress), March 2009.

[I-D.ietf-6man-udpzero]
Fairhurst, G. and M. Westerlund, "IPv6 UDP Checksum Considerations", draft-ietf-6man-udpzero-02 (work in progress), October 2010.

[I-D.ietf-dhc-subnet-alloc]
Johnson, R., Kumarasamy, J., Kinnear, K., and M. Stapp, "Subnet Allocation Option", draft-ietf-dhc-subnet-alloc-11 (work in progress), May 2010.

[I-D.ietf-grow-va]
Francis, P., Xu, X., Ballani, H., Jen, D., Raszuk, R., and L. Zhang, "FIB Suppression with Virtual Aggregation", draft-ietf-grow-va-04 (work in progress), February 2011.

[I-D.ietf-lisp]
Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "Locator/ID Separation Protocol (LISP)", draft-ietf-lisp-10 (work in progress), March 2011.

[I-D.ietf-manet-smf]
Macker, J. and S. Team, "Simplified Multicast Forwarding", draft-ietf-manet-smf-11 (work in progress), March 2011.

[I-D.ietf-v6ops-tunnel-security-concerns]
Krishnan, S., Thaler, D., and J. Hoagland, "Security

Concerns With IP Tunneling", draft-ietf-v6ops-tunnel-security-concerns-04 (work in progress), October 2010.

[I-D.jen-apt]
Jen, D., Meisel, M., Massey, D., Wang, L., Zhang, B., and L. Zhang, "APT: A Practical Transit Mapping Service", draft-jen-apt-01 (work in progress), November 2007.

[I-D.nakibly-v6ops-tunnel-loops]
Nakibly, G. and F. Templin, "Routing Loop Attack using IPv6 Automatic Tunnels: Problem Statement and Proposed Mitigations", draft-nakibly-v6ops-tunnel-loops-03 (work in progress), August 2010.

[IEN48] Cerf, V., "The Catenet Model for Internetworking", July 1978.

[RASADV] Microsoft, "Remote Access Server Advertisement (RASADV) Protocol Specification", October 2008.

[RFC0994] International Organization for Standardization (ISO) and American National Standards Institute (ANSI), "Final text of DIS 8473, Protocol for Providing the Connectionless-mode Network Service", RFC 994, March 1986.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, November 1987.

[RFC1070] Hagens, R., Hall, N., and M. Rose, "Use of the Internet as a subnetwork for experimentation with the OSI network layer", RFC 1070, February 1989.

[RFC1122] Braden, R., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, October 1989.

[RFC1753] Chiappa, J., "IPng Technical Requirements Of the Nimrod Routing and Addressing Architecture", RFC 1753, December 1994.

[RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.

[RFC1955] Hinden, R., "New Scheme for Internet Routing and Addressing (ENCAPS) for IPNG", RFC 1955, June 1996.

[RFC2003] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996.

[RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, March 1997.

[RFC2473] Conta, A. and S. Deering, "Generic Packet Tunneling in IPv6 Specification", RFC 2473, December 1998.

[RFC2491] Armitage, G., Schulter, P., Jork, M., and G. Harter, "IPv6 over Non-Broadcast Multiple Access (NBMA) networks", RFC 2491, January 1999.

[RFC2501] Corson, M. and J. Macker, "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations", RFC 2501, January 1999.

[RFC2529] Carpenter, B. and C. Jung, "Transmission of IPv6 over IPv4 Domains without Explicit Tunnels", RFC 2529, March 1999.

[RFC2775] Carpenter, B., "Internet Transparency", RFC 2775, February 2000.

[RFC3819] Karn, P., Bormann, C., Fairhurst, G., Grossman, D., Ludwig, R., Mahdavi, J., Montenegro, G., Touch, J., and L. Wood, "Advice for Internet Subnetwork Designers", BCP 89, RFC 3819, July 2004.

[RFC3927] Cheshire, S., Aboba, B., and E. Guttman, "Dynamic Configuration of IPv4 Link-Local Addresses", RFC 3927, May 2005.

[RFC3947] Kivinen, T., Swander, B., Huttunen, A., and V. Volpe, "Negotiation of NAT-Traversal in the IKE", RFC 3947, January 2005.

[RFC3948] Huttunen, A., Swander, B., Volpe, V., DiBurro, L., and M. Stenberg, "UDP Encapsulation of IPsec ESP Packets", RFC 3948, January 2005.

[RFC4192] Baker, F., Lear, E., and R. Droms, "Procedures for Renumbering an IPv6 Network without a Flag Day", RFC 4192, September 2005.

[RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, October 2005.

[RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, October 2005.

[RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC4306] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol", RFC 4306, December 2005.

[RFC4548] Gray, E., Rutemiller, J., and G. Swallow, "Internet Code Point (ICP) Assignments for NSAP Addresses", RFC 4548, May 2006.

[RFC4555] Eronen, P., "IKEv2 Mobility and Multihoming Protocol (MOBIKE)", RFC 4555, June 2006.

[RFC4605] Fenner, B., He, H., Haberman, B., and H. Sandick, "Internet Group Management Protocol (IGMP) / Multicast Listener Discovery (MLD)-Based Multicast Forwarding ("IGMP/MLD Proxying")", RFC 4605, August 2006.

[RFC4795] Aboba, B., Thaler, D., and L. Esibov, "Link-local Multicast Name Resolution (LLMNR)", RFC 4795, January 2007.

[RFC4852] Bound, J., Pouffary, Y., Klynsma, S., Chown, T., and D. Green, "IPv6 Enterprise Network Analysis - IP Layer 3 Focus", RFC 4852, April 2007.

[RFC4903] Thaler, D., "Multi-Link Subnet Issues", RFC 4903, June 2007.

[RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, September 2007.

[RFC5214] Templin, F., Gleeson, T., and D. Thaler, "Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)", RFC 5214, March 2008.

[RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, July 2008.

[RFC5569] Despres, R., "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd)", RFC 5569, January 2010.

[RFC5685] Devarapalli, V. and K. Weniger, "Redirect Mechanism for the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 5685, November 2009.

[RFC5720] Templin, F., "Routing and Addressing in Networks with Global Enterprise Recursion (RANGER)", RFC 5720, February 2010.

[RFC5887] Carpenter, B., Atkinson, R., and H. Flinck, "Renumbering Still Needs Work", RFC 5887, May 2010.

[RFC5969] Townsley, W. and O. Troan, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification", RFC 5969, August 2010.

[RFC6139] Russert, S., Fleischman, E., and F. Templin, "Routing and Addressing in Networks with Global Enterprise Recursion (RANGER) Scenarios", RFC 6139, February 2011.

[RFC6179] Templin, F., "The Internet Routing Overlay Network (IRON)", RFC 6179, March 2011.

Appendix A. Duplicate Address Detection (DAD) Considerations

A priori uniqueness determination (also known as "pre-service DAD") for an RLOC assigned on an enterprise-interior interface would require either flooding the entire enterprise network or somehow discovering a link in the network on which a node that configures a duplicate address is attached and performing a localized DAD exchange on that link. But, the control message overhead for such an enterprise-wide DAD would be substantial and prone to false-negatives due to packet loss and intermittent connectivity. An alternative to pre-service DAD is to autoconfigure pseudo-random RLOCs on enterprise-interior interfaces and employ a passive in-service DAD (e.g., one that monitors routing protocol messages for duplicate assignments).

Pseudo-random IPv6 RLOCs can be generated with mechanisms such as CGAs, IPv6 privacy addresses, etc. with very small probability of collision. Pseudo-random IPv4 RLOCs can be generated through random assignment from a suitably large IPv4 prefix space.

Consistent operational practices can assure uniqueness for VBG-aggregated addresses/prefixes, while statistical properties for pseudo-random address self-generation can assure uniqueness for the RLOCs assigned on an ER’s enterprise-interior interfaces. Still, an RLOC delegation authority should be used when available, while a passive in-service DAD mechanism should be used to detect RLOC duplications when there is no RLOC delegation authority.

Appendix B. Anycast Services

Some of the IPv4 addresses that appear in the Potential Router List may be anycast addresses, i.e., they may be configured on the VET interfaces of multiple VBRs/VBGs. In that case, each VET router interface that configures the same anycast address must exhibit equivalent outward behavior.

Use of an anycast address as the IP destination address of tunneled packets can have subtle interactions with tunnel path MTU and neighbor discovery. For example, if the initial fragments of a fragmented tunneled packet with an anycast IP destination address are routed to different egress tunnel endpoints than the remaining fragments, the multiple endpoints will be left with incomplete reassembly buffers. This issue can be mitigated by ensuring that each egress tunnel endpoint implements a proactive reassembly buffer garbage collection strategy. Additionally, ingress tunnel endpoints that send packets with an anycast IP destination address must use the minimum path MTU for all egress tunnel endpoints that configure the same anycast address as the tunnel MTU. Finally, ingress tunnel endpoints should treat ICMP unreachable messages from a router within the tunnel as at most a weak indication of neighbor unreachability, since the failures may only be transient and a different path to an alternate anycast router quickly selected through reconvergence of the underlying routing protocol.

Use of an anycast address as the IP source address of tunneled packets can lead to more serious issues. For example, when the IP source address of a tunneled packet is anycast, ICMP messages produced by routers within the tunnel might be delivered to different ingress tunnel endpoints than the ones that produced the packets. In that case, functions such as path MTU discovery and neighbor unreachability detection may experience non-deterministic behavior that can lead to communications failures. Additionally, the fragments of multiple tunneled packets produced by multiple ingress tunnel endpoints may be delivered to the same reassembly buffer at a single egress tunnel endpoint. In that case, data corruption may result due to fragment misassociation during reassembly.

In view of these considerations, VBGs that configure an anycast address should also configure one or more unicast addresses from the Potential Router List; they should further accept tunneled packets destined to any of their anycast or unicast addresses, but should send tunneled packets using a unicast address as the source address.

Appendix C. Change Log

(Note to RFC editor - this section to be removed before publication as an RFC.)

Changes from -14 to -15:

- o new insights into default route configuration and next-hop determination

Changes from -13 to -14:

- o fixed Idnits

Changes from -12 to -13:

- o Changed "VGL" *back* to "PRL"
- o More changes for multi-protocol support
- o Changes to Redirect function

Changes from -11 to -12:

- o Major section rearrangement
- o Changed "PRL" to "VGL"
- o Brought back text that was lost in the -10 to -11 transition

Changes from -10 to -11:

- o Major changes with significant simplifications
- o Now support stateless PD using 6rd mechanisms
- o SEAL Control Message Protocol (SCMP) used instead of ICMPv6
- o Multi-protocol support including IPv6, IPv4, OSI/CLNP, etc.

Changes from -09 to -10:

- o Changed "enterprise" to "enterprise network" throughout
- o dropped "inner IP", since inner layer may be non-IP
- o TODO - convert "IPv6 ND" to SEAL SCMP messages so that control messages remain *within* the tunnel interface instead of being

exposed to the inner network layer protocol engine.

Changes from -08 to -09:

- o Expanded discussion of encapsulation/decapsulation procedures

- o cited IRON

Changes from -07 to -08:

- o Specified the approach to global mapping using virtual aggregation and BGP

Changes from -06 to -07:

- o reworked redirect function

- o created new section on VET interface encapsulation

- o clarifications on nexthop selection

- o fixed several bugs

Changed from -05 to -06:

- o reworked VET interface ND

- o anycast clarifications

Changes from -03 to -04:

- o security consideration clarifications

Changes from -02 to -03:

- o security consideration clarifications

- o new PRLNAME for VET is "isatav2.example.com"

- o VET now uses SEAL natively

- o EBGs can support both legacy ISATAP and VET over the same underlying interfaces.

Changes from -01 to -02:

o Defined CGA and privacy address configuration on VET interfaces

o Interface identifiers added to routing protocol control messages for link-layer multiplexing

Changes from -00 to -01:

o Section 4.1 clarifications on link-local assignment and RLOC autoconfiguration.

o Appendix B clarifications on Weak End System Model

Changes from RFC5558 to -00:

o New appendix on RLOC configuration on VET interfaces.

Author's Address

Fred L. Templin (editor)
Boeing Research & Technology
P.O. Box 3707 MC 7L-49
Seattle, WA 98124
USA

Email: fltemplin@acm.org