Abstract

A variety of network virtualization over layer 3 methods are currently being developed and deployed. These methods treat IPv4 and IPv6 as equivalent underlay network technologies. This memo suggests how IPv6’s additional capabilities may be used to enhance Virtual Network encapsulation over an IPv6 Underlay Network.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on February 17, 2015.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.
1. Introduction

A variety of network virtualization over layer 3 methods are currently being developed and deployed [I-D.mahalingam-dutt-dcops-vxlan][I-D.sridharan-virtualization-nvgre][I-D.davie-stt]. Each of these methods treat both IPv4 and IPv6 as functionally equivalent underlay network technologies, with both providing general unicast and multicast capabilities.

IPv6 provides a number of capabilities not available in IPv4. This memo suggests how they may be used to enhance the encapsulation of Virtual Network traffic when forwarded over an IPv6 Underlay Network.

This memo does not consider how Virtual Network signalling protocols could be enhanced using IPv6. However, it may be possible to use techniques similar to those suggested in this memo to enhance these signalling protocols when being carried over IPv6.

2. Terminology

This memo adopts the terminology described in [I-D.ietf-nvo3-framework], summarised and supplemented below.
Tenant System - a device operated by the user of the Virtual Network service. It may be a host or a network element such as a router, and is not aware of the Virtual Network service.

IPv6 Underlay Network - the IPv6 network across which Tenant Packets are carried, encapsulated within IPv6, and possibly also within some other network virtualization header. It is assumed by this memo that this network supports both unicast and multicast IPv6 packet forwarding services.

Tenant Packet - a packet originated by a Tenant System, tunnelled within IPv6 across the IPv6 Underlay Network. The most common Tenant Packet is likely to be an Ethernet/IEEE 802.3 frame [IEEE8023], although other link-layer frame types, and other network layer packets such as IPv6 or IPv4 packets could be Tenant Packets.

Virtual Network - a single conceptual network interconnecting Tenant Systems, representing a single link or subnetwork. The IPv6 Underlay Network provides packet forwarding service for some or all of the Virtual Network.

Virtual Network Context Identifier (Virtual Network Context ID) - an identifier used to specify the Virtual Network a Tenant Packet belongs to while the packet is carried over the IPv6 Underlay Network.

Network Virtualization Edge (NVE) - a device or function within a device that performs IPv6 encapsulation of Tenant Packets on ingress to the IPv6 Underlay Network and IPv6 decapsulation of Tenant Packets on egress from the IPv6 Underlay Network. Other network virtualization related headers may be added or removed during the IPv6 encapsulation or decapsulation procedure.

3. Carrying the Virtual Network Context ID in the Flow Label Field

The IPv6 Flow Label field is a 20 bit field in a fixed location early in the IPv6 header [RFC2460][RFC6437]. It is intended to be used to identify flows between a pair of source and destination IPv6 addresses, as an alternative to identifying flows using transport layer header port numbers [RFC0793][RFC0768][RFC4960][RFC4340], which may be located deeper within the IPv6 packet, perhaps following a number of other IPv6 extension headers, or hidden by IPsec [RFC4301].

One of the expected and encouraged uses of the Flow Label field is as an input into link or path selection when using stateless load balancing of traffic across multiple links [RFC6436][RFC6438], using methods such as Equal Cost Multi-Path (ECMP) [RFC2991] or Link Aggregation Groups (LAGs) [IEEE8021AX].
The Flow Label field could be used to carry a whole or partial copy of the Virtual Network Context ID, providing it as an input into a stateless load balancing method, operating within the IPv6 Underlay Network.

Alternatively, the Flow Label field could carry the Virtual Network Context ID itself, providing support for up to 1 Million Virtual Networks. This would reduce the encapsulation overhead of tunnelling over IPv6.

A drawback of using the Flow Label to carry the Virtual Network Context ID is that it is a ‘best effort’ field, meaning that it may be changed as it transits the network without any protection by an end-to-end checksum, including when other fields in the IPv6 header are protected by the IPsec Authentication Header [RFC4302]. A change of the Flow Label field value when used to carry the Virtual Network Context ID would mean the Tenant Packet would either be delivered to the incorrect Virtual Network or would be dropped because the specified Virtual Network does not exist. Incorrect Virtual Network delivery would likely be unacceptable to the Virtual Network’s user for security reasons.

This could be resolved by protecting the integrity of the Flow Label field value using a checksum carried in some other Virtual Network related header, and validating that checksum when the IPv6 tunnelling header is removed, before delivery to the corresponding Virtual Network.

[RFC6437] advises that Flow Label values should be uniformly distributed. If the Flow Label field carries the Virtual Network Context IDs, then ideally they should also be uniformly distributed. This would be easier to achieve if Virtual Network Context IDs are generated algorithmically, rather than chosen by a human operator. Note that some of the load distribution mechanisms described in [I-D.ietf-opsawg-large-flow-load-balancing] may reduce the importance of uniform distribution of Flow Label values when used in a closed network, such as the IPv6 Underlay Network.

[RFC6437] also advises that forwarding nodes must not depend upon uniform distribution of Flow Label values. When used as a hash key for load distribution, the Flow Label bits must be combined with other sources within the packet. If a 3-tuple of Flow Label, Source and Destination Addresses fields are used as hash keys, the method of carrying Tenant Packet addresses in the IPv6 Underlay Network packet source and destination address IIDs, or source address IIDs, described later in Section 5, should result in a constant hash value across a flow of IPv6 Underlay Network packets between a pair of
Tenant Systems, or one source Tenant System and a single broadcast or multicast destination, within the same Virtual Network.

A Flow Label value of zero has been deemed to mean that the Flow Label value has not been set [RFC6437], and can therefore be changed as the the IPv6 packet traverses the network. This would preclude the use of the Flow Label field to carry a Virtual Network Context ID value of zero, as if it was changed by an intermediary device it would fail the Flow Label integrity check using checksum information carried by some other Virtual Network header.

Carrying Virtual Network Context ID information in the Flow Label field is also likely to assist with IPv6 Underlay Network troubleshooting and facilitate traffic analysis using IPv6 tools that can analyse the Flow Label field.

4. Using /64s to Identify NVEs

Networks operating IPv6 have large numbers of /64 subnets; at a minimum, even the smallest end site is expected to be assigned a /56 or 256 /64s [RFC6177], where as a single ULA /48 prefix [RFC4193] provides more than 65 000 /64 subnets.

Instead of identifying an NVE in the IPv6 Underlay Network using a single unicast IPv6 address, an NVE could be identified using a /64 prefix. An NVE would then announce its /64 prefix into the IPv6 Underlay Network’s routing domain, using an IGP or EGP such as OSPFv3 [RFC5340] or BGP [RFC4271][I-D.lapukhov-bgp-routing-large-dc]. This would provide reachability and availability information to other NVEs, and support multihoming and load sharing when an NVE has multiple attachments to the IPv6 Underlay Network. Automated discovery of NVEs could be facilitated by attaching a widely known identifier to the NVE /64 route announcements, using mechanisms such as OSPFv3’s External Route Tag or a BGP community [RFC1997][RFC5701].

If multiple NVEs are attached to the same Virtual Network segments, they could be assigned and announce the same /64 prefix within the IPv6 Underlay Network. This would result in unicast Tenant Packets encapsulated in unicast IPv6 packets being more optimally forwarded to the closest NVE that provides access to the Tenant System network segment, and would also provide redundancy if one of the NVEs announcing the same /64 prefix fails. In this scenario, it is essential that all NVEs sharing the same /64 prefix provide access to the same set of Virtual Network segments. Note that this would likely mean there is a possible forwarding loop in the Virtual Network’s topology, which may need to be suppressed using a mechanism that is out of scope for this memo.
As the NVEs are now identified by /64 prefixes, for unicast Tenant Packets, the source and destination IPv6 addresses used for the IPv6 encapsulation can be the Subnet-Router anycast address, the result of the NVE /64 prefix and an IID portion value of all zeros [RFC4291]. For multicast traffic, the source used can be the Subnet-Router anycast address, while the destination address used is an IPv6 multicast address used to reach the appropriate NVEs.

[I-D.carpenter-6man-why64] reports that some IPv6 routers provide optimal forwarding performance for /64 or shorter prefixes. Assigning /64s to NVEs would gain the best performance from this class of IPv6 routers when carrying traffic across the IPv6 Underlay Network.

5. Carrying Tenant Packet Address and Other Information in IIDs

If /64s are used to identify NVEs, then the IPv6 Underlay Network’s packets’ 8 octet IID portions in their source and possibly destination addresses can be used to carry Tenant Packet address information and possibly other information. This is instead of setting the IID portions to the Subnet-Router anycast address IID suggested in Section 4.

Carrying Tenant Packet addresses and other fields in the address IID portions of the IPv6 Underlay Network header should improve load balancing, and would expose this information to IPv6 traffic analysis tools such as IPFIX [RFC7011][RFC7015], providing the IPv6 Underlay Network operator with information about individual Tenant Systems and the traffic volumes between them.

5.1. Carrying Tenant Packet Addresses

During encapsulation in IPv6, upon ingress to the IPv6 Underlay Network, Tenant Packet addresses could be copied into the IID portions of the IPv6 address fields. For unicast Tenant Packets, the source and destination addresses are copied into the corresponding IPv6 Underlay Packet address IID portions. For multicast Tenant Packets, the source address is copied into the IPv6 Underlay Packet source address IID portion, while the destination address is an appropriate IPv6 multicast address.

As the IPv6 source and destination address fields can be used as inputs for stateless load balancing across the IPv6 Underlay Network, the entropy in the IID portions of the address, as a result of being Tenant Packet address values, should improve the effectiveness of load balancing, while preserving in-order delivery of Tenant Packets between pairs of Tenant Systems. This will also assist with flow recognition when mechanisms described in
For most types of Tenant Packet addresses, the 8 octet IPv6 IID field will be large enough to hold a complete copy of the Tenant Packet addresses. To reduce tunnelling overhead, these address fields could be removed from the Tenant Packet while being tunnelled, and restored when the IPv6 packet arrives at the destination NVE, as part of the process of IPv6 decapsulation.

Note that the IPv6 header is not protected by an end-to-end checksum [RFC2460], so removing the Tenant Packet address fields during IPv6 encapsulation should only be performed when the removed fields are protected by a suitable network virtualization header checksum or a Tenant Packet checksum.

In the case of a network virtualization header checksum covering the Tenant Packet addresses when carried in the IPv6 address IID portions, the validation of this checksum would occur when the Tenant Packet is reconstructed by the destination NVE.

Alternatively, if the Tenant Packet checksum originally covered the Tenant Packet addresses, validation could be left to the Tenant Packet destination, increasing NVE performance at the cost of possibly forwarding corrupted Tenant Packets after IPv6 decapsulation. As Tenant Packet corruption is likely to be rare when forwarded across the IPv6 Underlay Network, it is recommended to leave this validation to the final Tenant Packet destination. It would be useful for a network operator to be able to switch on validation at an NVE temporarily for troubleshooting purposes.

If Tenant Packet addresses are larger than the IPv6 address IID portions, then the portion of the Tenant Packet addresses that would provide the best input into load balancing should be copied. For example, if Tenant Packets are raw IPv6 packets (i.e., without a link-layer header), then the Tenant Packet address 64 bit IID portions should be copied into the IPv6 Underlay Network packet address IID fields, and then perhaps removed from the Tenant Packets. Tenant Packets carrying IID portions generated using either [RFC7217] or [RFC4941] will provide the best IID values, as those IID values are the result of a pseudo-random or hash function.

When the IPv6 IID portions are used to carry Tenant Packet values, the receiving NVE would not consider any of the received IID values to have any significance. In other words, none of the IID values described in [RFC5453] are to be considered reserved. This is consistent with [RFC7136], which states that only a local context can give the IID bits semantic meaning.
5.2. Carrying Other Tenant Packet Information

If the Tenant Packet addresses are smaller than the IPv6 address IID portions, other Tenant Packet field values could be copied into the remaining parts of the IPv6 address IDs portion, and also possibly be removed from the Tenant Packet, which will further reduce tunnelling overhead, and may further increase stateless load-balancing effectiveness.

For example, for Ethernet/IEEE 802.3 Tenant Packets, both the 6 octet Ethernet/IEEE 802.3 source address and subsequent 2 octet type/length field values could be copied into the IPv6 source address 8 octet IID portion in a single operation, and then removed from the original Tenant Packet. This should be beneficial to stateless load-balancing when the type/length field is carrying a variety of payload type values.

If the Ethernet/IEEE 802.3 type/length field is carrying length values when copied into the IPv6 source address IID portion, out of original sending order delivery of Tenant Packets could be the result, caused by the stateless load-balancing method being used by the IPv6 Underlay network. This may negatively impact the performance or possibly in the worst case cause failure of the corresponding upper layer protocol.

If lower performance or possible upper layer protocol failure is unacceptable, only the Ethernet/IEEE 802.3 source address could be copied into the the IPv6 source address IID portion for these ‘length’ Ethernet/IEEE 802.3 frames. To distinguish between these Tenant Packets and those where both the Ethernet/IEEE 802.3 source address and type field values are copied into the IPv6 source address IID field, either a different Virtual Network Context ID could be used, or some other indicator field in an additional Virtual Network header could indicate the different ‘length’ value encapsulation. If a different Virtual Network Context ID is used for these ‘length’ Tenant Packets, at the decapsulating NVE, these frames would be merged back into the single Virtual Network.

Upon NVE encapsulation, rather than perform a less than or equal to 1500 comparison operation on the type/length field to identify ‘length’ Ethernet/IEEE 802.3 frames [IEEE8023], a simpler and likely faster implementation could perform an exact match comparison on the type/length field value against a set of common protocol types, such as IPv4 [RFC0894], ARP [RFC0826], IPv6 [RFC2464] and IEEE 802.1Q [IEEE8021Q]. For those frames that match, both the Ethernet/IEEE 802.3 source address and type/length field values are copied into the IPv6 Underlay Network packet source address IID portion, whereas for all other frames, just the Ethernet/IEEE 802.3 source address value.
5.3. Indicating IID Contents

This optimisation of carrying Tenant Packet field values in the IPv6 encapsulating header’s address field IIDs portions and removing them from the Tenant Packet could be indicated to the destination NVE implicitly by the Virtual Network Context ID, or via some other header carried in the IPv6 packet.

6. Permanent Virtual Network Multicast Group Identifier

To simplify and automate configuration, a permanent IPv6 multicast group identifier could be assigned by IANA, in accordance with the allocation guidelines specified in [RFC3307], to be used for encapsulation of multicast Tenant Packets in IPv6 multicast packets.

This group ID would be used to form Interface-Local, Link-Local, and Site-Local scope multicast addresses. Each NVE would then subscribe to these scoped multicast addresses for the permanent group ID. The range of different scopes will allow an origin NVE to constrain the forwarding domain of IPv6 multicast packets holding multicast Tenant Packets if necessary or useful.

Other multicast scopes that may be useful for NVE encapsulation operation might be the Realm-Local, Admin-Local, and Organization-Local scopes [I-D.droms-6man-multicast-scopes], also used with the IANA reserved group ID.

7. Per-Virtual Network Multicast Group Addresses

Using a single well known multicast group to flood IPv6 encapsulated multicast or broadcast Tenant Packets to all NVEs for all Virtual Networks may eventually impact network performance, due to the volume of multicast traffic being sent to NVEs at which the corresponding Virtual Network is not present.

Reducing network load may be achieved by using multiple multicast groups to distribute IPv6 encapsulated multicast or broadcast Tenant Packets to NVEs where the Virtual Network is present. Optionally, an NVE might only become and remain a member of the Virtual Network specific multicast group when it is aware that there is at least one Tenant System present in the local Virtual Network segment.

[RFC3306] describes how to create multicast addresses using a unicast IPv6 prefix, between 0 and 64 bits in length. For each unicast IPv6 derived multicast prefix, 32 bits are available for the Group ID.
These group IDs are created using the guidelines specified in [RFC3307]. For dynamically created multicast addresses, [RFC3307] restricts the group ID range to (in IPv6 address form) ::8000:0000 to ::ffff:ffff, a range of 31 bits or approximately 2 billion unique groups. The leading high order bit in the Group ID corresponds to the 'T' bit value in the multicast address flag, which indicates a Temporary multicast address. This ensures that when the multicast group is mapped to a multicast link-layer address, by copying the lower 32 bits of the multicast address to the link-layer multicast address range (e.g., 33-33-XX-XX-XX-XX for Ethernet/IEEE 802.3 [RFC2464]), the link-layer multicast address does not collide with Permanent IPv6 multicast addresses at the link-layer.

These 31 bits of dynamic group IDs, available for a unicast prefix, could be used to form a unique multicast group address per Virtual Network, using the Virtual Network Context ID, by combining it with an IPv6 prefix used by all NVEs. The NVEs would be informed of the common IPv6 prefix using manual configuration or a signalling protocol.

The common IPv6 prefix used to form these addresses does not have to be related to any of the /64 prefixes being used by the NVEs. However it is recommended to relate them intuitively, by using a shorter aggregate prefix that covers the set of identifying /64 prefixes being used by the NVEs attached to the same IPv6 Underlay Network. This would simplify configuration, reduce errors and simplify troubleshooting.

With larger numbers of Virtual Networks, one multicast group per Virtual Network may exceed the IPv6 Underlay Network’s capacity to reliably track multicast group membership for all of the present multicast groups. NVEs would participate directly in the IPv6 Underlay Network’s multicast routing protocol [RFC5110], limiting the number of multicast groups to the IPv6 Underlay Network’s multicast routing protocol implementations’ maximum capacity.

The preferred option in this situation would be to create another IPv6 Underlay Network, and to move some, and ideally half of the Virtual Networks to the new IPv6 Underlay Network. This would preserve the efficiency of one multicast group per Virtual Network, as well as increasing encapsulation network unicast and multicast traffic capacity.

An alternative, although less efficient option would be to map multiple Virtual Networks onto each multicast group, on a many-to-one basis. A simple scheme would be to map the Virtual Networks equally onto the available multicast groups. This may be easier to implement if the Virtual Network Context IDs have been uniformly distributed as...
suggested previously in Section 3. More advanced mapping schemes might take into consideration other Virtual Network attributes such as the number of Tenant Systems attached to the individual Virtual Networks, the maximum allowed number of Tenant Systems in each Virtual Network or the number of NVEs where Tenant System network segments for each Virtual Network are present, with a goal of trying to equally balance multicast traffic across the available multicast groups.

8. Per-NVE Multicast Addresses

As each NVE is identified by a /64 prefix, the method of forming multicast addresses described in [RFC3306] could also be used by an NVE to generate multicast group addresses specific to its /64 prefix. This may be useful when multiple NVEs are using the same /64 prefix for performance and redundancy purposes, and the origin NVE can determine that it only needs to send encapsulated multicast Tenant Packets to the set of NVEs sharing a single /64 prefix.

NVEs creating multicast groups for all of their present Virtual Network Context IDs for their /64 prefix may not be practical, as it would increase the number of multicast group memberships the IPv6 Underlay Network needs to track. Mapping multiple Virtual Networks to a multicast group, on a many-to-one basis, may also consume excessive multicast membership tracking resources, as the amount of traffic towards one or more NVEs using a single /64 is likely to be small. In either case, this would be in addition to the per-Virtual Network multicast groups already being tracked by the IPv6 Underlay Network, suggested in Section 7.

Instead, the NVEs should use the IANA permanent multicast group ID to form the per-NVE /64 derived multicast addresses, used by all Virtual Networks. The NVE would then subscribe to the Interface-Local, Link-Local and Site-Local scope forms of this multicast address, and optionally other multicast scopes. It may also conditionally join these multicast groups based on Tenant System presence on the local Virtual Network segment.

9. Summary of Methods and Benefits

- Carrying a full or partial copy of the Virtual Network Context ID value in the Flow Label field would expose it to the IPv6 Underlay Network load balancing mechanisms and troubleshooting tools.

- Carrying the Virtual Network Context ID in the Flow Label field would reduce tunnelling overhead.
Using /64s to identify NVEs within the IPv6 Underlay Network would support the use of more conventional IPv6 routing methods and protocols, may provide more efficient forwarding, and allows for other uses of the IID portions of the IPv6 addresses.

Carrying full or partial copies of Tenant Packet addresses and other information in the IID portions of the IPv6 address fields would expose those values to the IPv6 Underlay Network load balancing mechanisms and other IPv6 troubleshooting and analysis tools, such as IPFIX.

Carrying Tenant Packet addresses and other field information in the IID portions of the IPv6 address fields, and then removing those fields from the original Tenant Packets while being carried across the IPv6 Underlay Network, would reduce tunnelling overhead.

Using an IANA assigned permanent multicast group ID, used by all NVEs, would both simplify and allow automated configuration of NVE multicast.

Per-Virtual Network multicast groups would reduce the volume of multicast or broadcast Tenant Packets being sent to NVEs at which the destination Virtual Network does not exist.

When the IPv6 Underlay Network’s multicast group capacity is exceeded, mapping multiple Virtual Networks to multiple multicast groups, on a many-to-one basis, would be more efficient than using a single multicast group for all Virtual Networks.

Use of the Interface-Local, Link-Local, and Site-Local scopes for the various Virtual Network multicast addresses would allow origin NVEs to better control the forwarding domain of multicast IPv6 Underlay Network traffic they send.

10. Security Considerations

Within a trusted IPv6 Underlay Network, copying or carrying Virtual Network or Tenant Packet attributes in IPv6 header fields will not significantly further expose them to untrusted parties, as they are likely to already exist in clear text within the IPv6 packet payload.

However, if the IPv6 Underlay Network is to span portions of the Internet, the IPv6 packets should be carried within IPsec [RFC4301] or some other secure tunnelling protocol that provides confidentiality, integrity and authenticity, to mitigate pervasive monitoring [RFC7258] and other security concerns.
In particular, when using IPsec, tunnel mode should be used with at least the IPsec Encapsulating Security Payload protocol [RFC4303], as the IPv6 Underlay Packets or their Tenant System packets would facilitate analysis of Tenant System traffic, by exposing detailed information about the numbers and identities of the Virtual Networks, possibly globally unique details of individual Tenant Systems, and volumes of traffic between distinct Tenant Systems.

To reduce the possibility of accidental forwarding of IPv6 Underlay Network traffic onto the Internet, it is recommended that the IPv6 Underlay Network is numbered using a single ULA /48, with egress packet filters dropping ULA source or destination packets at the network’s Internet boundary, as described in [RFC4193]. Additional egress packet filters at the edge of the IPv6 Underlay Network, for the ULA address space in use within the IPv6 Underlay Network, would provide further protection against accidental forwarding of IPv6 Underlay Network traffic onto the Internet.

11. Acknowledgements

Thanks to (in alphabetical order) Fred Baker and Brian Carpenter for their encouragement, review and comments.

This memo was prepared using the xml2rfc tool.

12. Change Log [RFC Editor please remove]

draft-smith-enhance-vne-with-ipv6-00, initial version, 2014-06-02

draft-smith-enhance-vne-with-ipv6-01, 2014-07-22
  - issues with ‘length’ Ethernet Tenant Packets
  - uniform distribution of flow label values/virtual network context IDs
  - more descriptive and clarifying text

draft-smith-enhance-vne-with-ipv6-02, 2014-07-28
  - methods and benefits summary section

draft-smith-enhance-vne-with-ipv6-03, 2014-07-29
  - fix some references

draft-smith-enhance-vne-with-ipv6-04, 2014-08-16
13. References

13.1. Normative References


13.2. Informative References


[I-D.ietf-nvo3-framework]

[I-D.ietf-opsawg-large-flow-load-balancing]

[I-D.lapukhov-bgp-routing-large-dc]

[I-D.mahalingam-dutt-dcops-vxlan]

[I-D.sridharan-virtualization-nvgre]

[IEEE8021AX]

[IEEE8021Q]

[IEEE8023]

[RFC1997]


Author’s Address

Mark Smith
In My Own Time
PO BOX 521
HEIDELBERG, VIC 3084
AU

Email: markzzzsmith@yahoo.com.au