Individual Submission
Internet-Draft
Expires: October 31, 2004

G. Huston
Telstra

May 2, 2004

Architectural Approaches to Multi-Homing for IPv6
draft-huston-multi6-architectures-00.txt

Status of this Memo

Copyright Notice

Abstract

This memo provides an analysis of the aspects of multi-homing support for the IPv6 protocol suite. The purpose of this analysis is to provide a taxonomy for classification of various proposed approaches to multi-homing. It is also an objective of this exercise to identify common aspects of this domain of study, and also to provide a framework that can allow exploration of some of the further implications of various architectural extensions that are intended to support multi-homing.

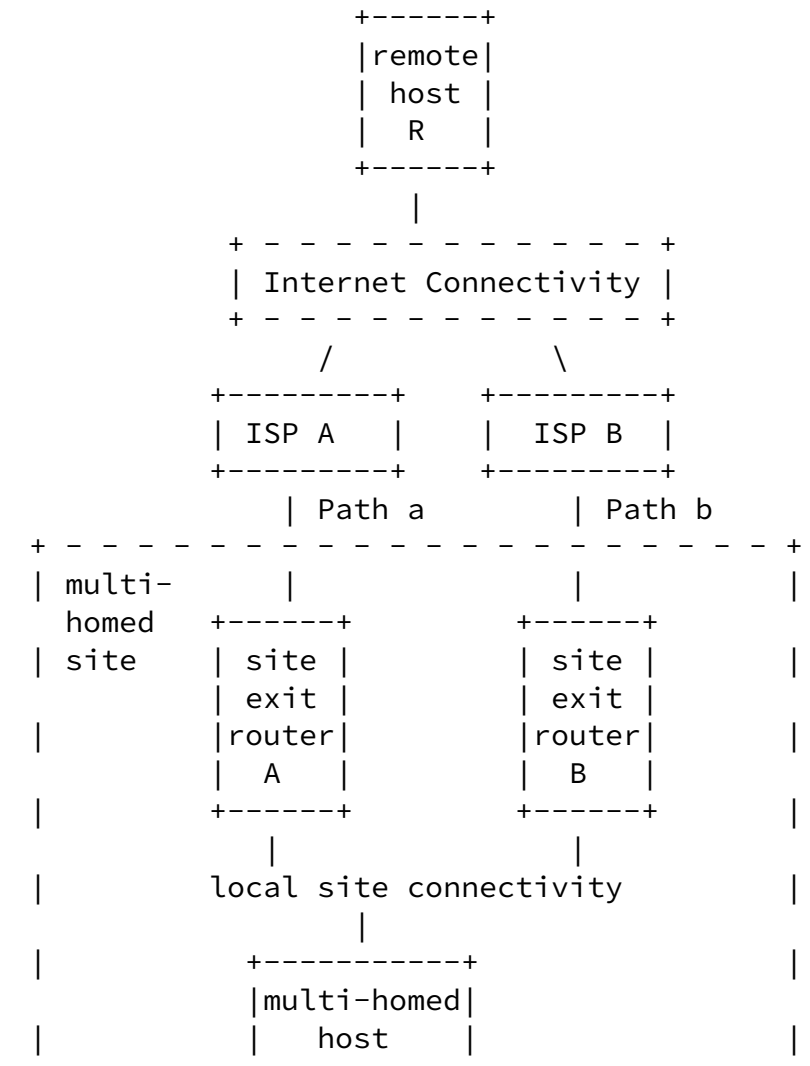
Table of Contents

1. Introduction

The objective of this exercise is to allow various technical proposals relating to the support of multi-homing environment in IPv6 to be placed within an architectural taxonomy. This is intended to allow these proposals to be classified and compared in a structured fashion. It is also an objective of this exercise to identify common aspects across all proposals within this domain of study, and also to provide a framework that can allow exploration of some of the further implications of various architectural extensions that are intended to support multi-homing. The scope of this study is limited to the IPv6 protocol suite architecture, although reference is made to IPv4 approaches as required.

2. The Multi-Homing Space

The simplest formulation of the multi-homing environment is indicated in Figure 1.


remote host R
Internet Connectivity
ISP A
ISP B
Path a
Path b
multi-homed site
site exit router A
site exit router B
local site connectivity
multi-homed host

The Multi-Homed Domain

Figure 1

The environment of multi-homing is one that is intended to provide sufficient support to local hosts so as to allow local hosts to exchange IP packets with remote hosts, such that this exchange of packets is to be seamlessly supported across dynamic changes in connectivity. This implies that if a local multi-homed-aware host establishes an application session with the remote host using "Path a", and this path fails, the application session should be mapped across to "Path b" without requiring any application-visible re-establishment of the session. In other words, the application session should not be required to be explicitly aware of underlying path changes at the level of packet forwarding paths chosen by the network.

This simple multi-homing scenario also includes "site-exit' routers, where the local site interfaces to the upstream Internet transit providers. The nature of the interactions between the external routing system and the site-exit routers, and interactions between the site-exit routers and the local multi-homed host, and the interactions between local connectivity forwarding and the local host and site exit routers are not defined a priori in this scenario, as they form part of the framework of interaction between the various multi-homing components.

The major characteristic of this scenario is that the address space used by, and advertised as reachable by, ISP A is distinct from the address space used by ISP B.

This simple scenario is intended to illustrate the basic multi-homing environment. Variations of this scenario include additional external providers of transit connectivity to the local site, complex site requirements and constraints, where the site may not interface uniformly to all external transit providers, sequential rather than simultaneous external transit reachability, communication with remote multi-homed hosts, multi-way communications, use of host addresses in

a referential context (third party referrals) and the imposition of policy constraints on path selection. However, the basic scenario is sufficient to illustrate the major architectural aspects of support for multi-homing, so this scenario will be used as the reference model for this analysis.

3. Requirements and Considerations

RFC 3582 [RFC3582] documents some requirements that a multi-homing approach should attempt to address. These requirements include:

- redundancy
- load sharing
- traffic engineering
- policy constraints
- simplicity of approach
- transport-layer survivability
- DNS compatibility
- packet filtering capability
- scaleability
- legacy compatibility

The reader is referred to [RFC3582] for a complete description of each of these requirements.

In addition, work in progress [draft-lear-multi6-things-to-think-about] documents further considerations for IPv6 multi-homing. Again, the reader is referred to this document for the detailed enumeration of these

considerations. The general topic areas considered in this study include:

- interaction with routing systems,
- aspects of a split between end-point-identifier and forwarding locator,
- changes to packets on the wire, and
- the interaction between names, endpoints and the DNS.

In considering various approaches, further consideration also include:

- the role of helpers and agents in the approach,
- modifications to host behaviors,
- the required trust model to support the interactions, and
- the nature of potential vulnerabilities in the approach.

4. Approaches to Multi-Homing

There appear to be four generic forms of architectural approaches to this problem, namely:

- Routing
  Use the IPv4 multi-homing approach
- New Protocol Element
  Insertion of a new element in the protocol stack that manages a persistent identity for the session
- Modify a Protocol Element
  Modify the Transport or IP protocol stack element in the host in order to support dynamic forwarding locator change
- Modified Site-Exit Router / Local Host interaction
  Modify the site-exit router and local forwarding system to allow various behaviors including source-based forwarding, site-exit hand-offs, and address rewriting by site-exit routers

These approaches will be described in detail in the following sections.

4.1 Multi-Homing: Routing

The approach used in IPv4 for multi-homing support is to preserve the semantics of the IPv4 address as both an endpoint identifier and a forwarding locator. For this to work in a multi-homing context it is necessary for the transit ISPs to announce the local site's address prefix as a distinct routing entry in the inter-domain routing system. This approach could be used in an IPv6 context, and, as with IPv4, no modifications to the IPv6 architecture are required to support this approach.

The local site's address prefix may be a more specific address prefix drawn from the address space advertised by one of the transit

providers, or from some third party provider not current directly connected to the local site. Alternatively the address space may be a distinct address block obtained by direct assignment from a Regional Internet Registry as Provider Independent space. Each host within the local site is uniquely addressed from the site's address prefix.

All transit providers for the site accept a prefix advertisement from

the multi-homed site, and advertise this prefix globally in the inter-domain routing table. When connectivity between the local site and an individual transit provider is lost, normal operation of the routing protocol will ensure that the routing advertisement corresponding to this particular path will be withdrawn from the routing system, and those remote domain domains who had selected this path as the best available will select another candidate path as the best path. Upon restoration of the path, the path is re-advertised in the inter-domain routing system. Remote domains will undertake a further selection of the best path based on this re-advertised reachability information. Neither the local or the remote host need to have multiple addresses, nor undertake any form of address selection. The path chosen for forward and reverse direction path flows is a decision made by the routing system.

This approach generally meets all the criteria for multi-homing approaches with one noteable exception: scaleability. Each site that multi-homes in this fashion adds a further entry in the global inter-domain routing table. Within the constraints of current routing and forwarding technologies it is not clearly evident that this approach can scale to encompass a population of multi-homed sites of the order of 10**7 such sites. The implication here is that this would add a similar number of unique prefixes into the inter-domain routing environment, which in turn would add to the storage and computational load imposed on inter-domain routing routing elements within the network. This scale of additional load is not supportable within the current capabilities of the IPv4 global Internet, nor is it clear at present that the routing capabilities of the entire network could be expanded to manage this load in a cost-effective fashion, within the bounds of the current inter-domain routing protocol architecture.

4.2 Multi-homing: Identity Considerations

The intent of multi-homing in the IPv6 domain is to achieve a comparable functional outcome for multi-homed sites without an associated additional load being imposed on the routing system. The overall intent of IPv6 is to provide a scaleable protocol framework to support the deployment of communications services for an extended period of time, and this implies that the scaling properties of the deployment environment remain tractable within projections of size of

deployment and underlying technology capabilities. Within the inter-domain routing space, the basic approach used in IPv4 and IPv6 is to attempt to align address deployment with network topology, so that address aggregation can be used to create a structured hierarchy of the routing space.

Within this constraint of topological-based address deployment and provider aggregatable addressing architectures, the local site that is connected to multiple providers is delegated addresses from each of these providers' address blocks. In the example network in Figure 1, the local multi-homed host will concievably be addressed in two ways: one using transit provider A's address prefix and the other using transit provider B's address prefix.

If remote host R is to initiate a communication with the local multi-homed host, it would normally query the DNS for an address for the local host. In this context the DNS would return 2 addresses (One using the A prefix and the other using the B prefix). The remote host would select one of these addresses and send a packet to this destination address. This would direct the pact to the local host along a path through A or B, depending on the selected address. If the path between the local site and the transit provider fails, then the address prefix announced by the transit provider to the inter-domain routing system will continue to be the provider's address prefix. The remote host will not see any change in routing, yet packets sent to the local host will now fail to be delivered. The question posed by the multi-homing problem is: "If the remote host is aware of multi-homing, how could it switch over to using the equivalent address for teh local multi-homed host hat transits the other provider?"

If the local multi-homed host wishes to initiate a session with remote host R, it needs to send a packet to R with a valid source and destination address. While the destination address is that of R, what source address should the local host use? There are two implications for this choice. Firstly the remote host will, by default use this source address as the destination address in its response, and hence this choice of source address will direct the reverse path from R to the local host. Secondly, the ISPs A and B may be using reverse unicast address filtering on source addresses of packets passed to the ISP, as a means of prevention of source address spoofing. This implies that if the multi-homed address selects a source address from address prefix A, and the local routing to R selects a best path via ISP B, then ISP B's ingress filters will discard the packet.

Within this addressing structure there is no form of routing-based repair of certain network failues. If the link between the local site and ISP A fails, there is no change in the route advertisements made

by ISP A to its external routing peers. Even though the multi-homed sitines to be reachable via ISP B, packets directed to the site using ISP A's prefix will be discarded by ISP A as the destination is unreachable. The implication here is that if the local host wishes to maintain a session across such events it needs to communicate to remote host R that it is possible to switch to using a destination address for the multi-homed host that is based on ISP B' address prefix.

In an aggregated routing environment multiple transit paths to a host imply multiple address prefixes for the host, where each possible transit path is identified by an address for the host. The implication of this constraint on multi-homing is that paths being passed to the local multi-homed site via transit provider ISP A must use a forwarding-level destination IP address drawn from ISP A's advertised address prefix set that maps to the multi-homed host. Equally, packets being passed via the transit of ISP B must use a destination address drawn from ISP B's address prefix set. The further implication here is that path selection (ISP A vs ISP B transit for incoming packets) is an outcome of the process of selecting an address for the destination host.

The architectural consideration here is that in the conventional IP protocol architecture the assumption is made that the transport-layer endpoint identity is the same identity used by the internet-layer forwarding layer, namely the IP address.

If multiple forwarding paths are to be supported for a single transport session, and path selection is to be decoupled from the functions of transport session initiation and maintenance, then the corollary of this requirement in architectural terms appears to be that some changes are required in the protocol architecture to decouple the concepts of identification of the endpoint and identification of the location and associated path selection for the endpoint. This change in the protocol architecture would permit a transport session to use an invariant endpoint identity value to initiate and maintain a session, while allowing the forwarding layer to dynamically change paths and associated endpoint locator identities without impacting on the operation of the session, nor would such a decoupled concept of identities and locators add any incremental load to the inter-domain routing system.

Some generic approaches to this form of separation of endpoint

identity and locator value are described in the following sections.

4.3 Multi-homing: Identity Protocol Element

One approach to this objective is to add a new element into the model

of the protocol stack.

The presentation to the upper level protocol stack element (ULP) would use endpoint identifiers to uniquely identify both the local stack and the remote stack. This will provide the ULP with stable identifiers for the duration of the ULP session.

The presentation to the lower level protocol stack element (LLP) would be of the form of a locator. This implies that the protocol stack element would need to maintain a mapping of endpoint identifier values to locator values. In a multi-homing context one of the essential characteristics of this mapping is that it needs to be dynamic, in that environmental triggers should be able to trigger a change in mappings, which in turn would correspond to a change in the paths (forward and/or reverse) used by the endpoints to traverse the network. In this way the ULP session is defined by a peering of endpoint identifiers that remain constant throughout the lifetime of the ULP session, while the locators may change to maintain end-to-end reachability for the session.

The operation of the new protocol stack element (termed here the "endpoint identity protocol stack element", or "EIP") is to establish a synchronized state with its remote counterpart. This would allow the stack elements to exchange a set of locators that may be used within the context of the session. A change in the local binding between the current endpoint identity value and a locator will cause a change in the source locator value used in the forwarding level packet header. The actions of the remote EIP upon receipt of this packet with the new locator is to firstly recognize this locator as part of an existing session, and, upon some trigger condition, to change its session view of the mapping of the remote endpoint identity to the corresponding locator, and use this locator as the destination locator in subsequent packets passed to the LLP.

From the perspective of the IP protocol architecture there are two possible locations to insert the EIP into the protocol stack.

One possible location is at the upper level of the transport protocol. Here the application program interface (API) of the application level protocols would interface to the EIP element, and use endpoint identifiers to refer to the remote entity. The EIP would pass locators to the API of the transport layer.

The second approach is to insert the EIP between the transport and internet protocol stack elements, so that the transport layer would function using endpoint identifiers, and maintain a transport session using these endpoint identifiers. The IP or internetwork layer would function using locators, and the mapping from endpoint identifier to

locator is undertaken within the EIP stack element.

4.4 Multi-homing: Modified Protocol Element

As an alternative to insertion of a new protocol stack element into the protocol architecture, an alternative approach is to modify an existing protocol stack element to include the functionality performed by the EIP element. This modification could be undertaken within the transport protocol stack element, or within the internetworking stack element. The functional outcome from these modifications would be to create a mechanism to support the use of multiple locators within the context of a single endpoint-to-endpoint session.

Within the transport layer, this functionality can be achieved, for example, by the binding of a set of locators to a single session, and then communicating this locator set to the remote transport entity. This would allow the local transport entity to switch the mapping to a different locator for either the local endpoint or the remote endpoint while maintaining the integrity of the ULP session.

Within the IP level this functionality could be supported by a form of dynamic rewriting of the packet header as it is processed by the protocol element. Incoming packets with the source and destination locators in the packet header are mapped to packets with the equivalent endpoint identifiers in both fields, and the reverse mapping is performed to outgoing packets passed from the transport layer. Mechanisms that support direct rewriting of the packet header are potential candidates in this approach, as are various forms of

packet header transformations of encapsulation, where the original endpoint identifier packet header is preserved in the packet and an outer level locator packet header is wrapped around the packet as it is passed through the internetworking protocol stack element.

In all these scenarios, there are common issues of what state is kept, by which part of the protocol stack, how state is maintained with additions, removals of locator bindings, and does only one piece of code have to be aware of the endpoint / locator split or do multiple protocol elements have to be modified? For example, if the functionality is added at the internetworking (IP) layer, there is no context of an active transport session, so that removal of identity / locator state information for terminated sessions needs to be triggered by some additional mechanism from the transport layer to the internetworking layer.

4.5 Modified Site-Exit and Host Behaviors

The above approaches all assume that the hosts are explicitly aware

of the multi-homed environment and use modified protocol behavior to support multi-homing functionality. A further approach to this objective is to split this functionality across a number of network elements and potentially perform packet header rewriting from a persistent endpoint identity value to a locator value at a remote point.

One possible approach proposes the use of site-exit routers to perform some form of packet header manipulation as packets are passed out from the local multi-homed site to a particular transit provider. The local site routing system will select the best path to a destination host based on the remote hosts's locator value. The local host will write its endpoint identity as the source address of the packet. When the packet reaches a site-exit router, the site-exit router will rewrite the source field of the packet to a corresponding locator that selects a reverse path through the same transit ISP when the locator is used as a destination locator by the remote host. In order to preserve session integrity there is a need for a corresponding reverse transformation to be undertaken on incoming packets, where the destination locator has to be mapped back to the host's endpoint identifier. There are a number of considerations whether this is best performed at the site exit router on packet

ingress to the site, or by the local host.

Packet header rewriting by remote network elements has a large number of associated considerations, and documentation relating to the considerations of the use of Network Address Translators ([NAT Considerations] contains much of this material.

An alternative for packet header rewriting on site exit is for the host to undertake the endpoint-to-locator mapping, using one of the approaches outlined above. The consideration here is that there is some significant deployment of unicast reverse path filtering in Internet environments as a counter-measure to source address spoofing. Using the example in Figure 1, if a host selects a locator drawn from the ISP B address prefix, and local routing directs that packet to site-exit router A, then if the packet is passed to ISP A, the this would be discarded by such filters. Various approaches have been proposed to modify the behavior of the site forwarding environment all with the end effect that packets using a source locator drawn from the ISP B address prefix are passed to site-exit router B. These approaches include forms of source address routing and site-exit router hand-over mechanisms, as well as augmentation of the routing information between site-exit routers and local multi-homed hosts, so that the choice of locator by the local host for the remote host is consistent with the current local routing state for the local site to reach the remote host.

4.6 Approaches to Endpoint Identity

Both of the above mechanisms assume some form of exchange of information that allows both parties to the communication to be aware of the remote endpoint identity and the associated mapping to locators. There are a number of choices in terms of the way in which this information exchange can be implemented.

The first such possible approach is termed here a 'conventional' approach, where the mode of operation is in terms of encapsulating the protocol data unit (PDU) passed from the ULP with additional data elements that specifically refer to the function of the endpoint identity protocol stack element. The compound data element is passed to the LLP as its PDU. The corresponding actions on receipt of a PDU from a LLP is to extract the fields of the data unit that correspond

to the EIP function, and pass the reminder of the PSU to the ULP. The EIP operates in an "in-band" mode, communicating with its remote peer entity through additional information wrapped around the ULP PDU.

Another approach is to allow the EIP to communicate using a separate communications channel, where the EIP generates dedicated messages that are directed to its peer EIP, and passes these PDUs to the LLP independently of the PDUs that are passed top the EIP from the ULP. This allows the EIP to exchange information and synchronize state with the remote EIP semi-independently of the ULP protocol exchange. As a part of the EIP function is to transform the ULP PDU to include locator information there is an associated requirement to ensure that the EIP peering state remains synchronized to the exchange of ULP PDUs, so that the remote EIP can correctly recognize the locator to endpoint mapping for each active session.

Another potential approach here is to allow the endpoint to locator mappings to be held at a third party point. This model is already used for supporting the name to IP address mappings performed by the Domain Name system, where the mapping is obtained by reference to a third party, namely a DNS resolver. A similar form of third party mapping between endpoints and a locator set could be supported through the use of the DNS, or a similar third party referential mechanism. Rather than have each party exchange endpoint to locator mappings, this approach would see this mapping being obtained as a result of a lookup for a DNS Endpoint to Locator set map contained as DNS Resource Records, for example.

4.7 Endpoint Identity Structure

The previous section has used the term "endpoint identity" without examining what form this identity may take. There are a number of salient considerations regarding the structure and form of this

identity that should be enumerated within an architectural overview of this space.

One possible form of an identity is the use of identity tokens lifted from the underlying protocol's "address space". In other words an endpoint identity is a special case instance of an IPv6 protocol address. There are a number of advantages in using this form of endpoint identity, observing that the suite of IP protocols and

associated applications already manipulate IP addresses. The essential difference in a domain that distinguishes between endpoint identity and locator is that the endpoint identity parts of the protocol would operate on those addresses that assume the role of endpoint identities, and the EIP mapping function would undertake a mapping from an endpoint "address" to a set of potential locator "addresses", and also undertake a reverse mapping from a locator "address" to the distinguished endpoint identifier "address". The address space is hierarchically structured, permitting a suitably efficient mapping to be performed in both directions, and the underlying semantics of addresses in the context of public networking includes the necessary considerations of global uniqueness of endpoint identity token values.

It is possible to take this approach further and allow the endpoint identifier to also be a valid locator. This would impliy the existence of a 'distinguished' or 'home' locator, and other locators could be dynamically mapped to this initial locator peering as required. The drawback of this approach is that the endpoint identifier is now based on one of the transit provider's address prefix, and a change of transit provider would necessarily require a change of endpoint identifier values within the multi-homed site. An alternative approach for address-formatted identifiers is to use address values which are not part of the global unicast locator space, allowing applications and protocol elements to distinguish between endpoint identity values and locators based on address prefix value. It is also possible to allow the endpoint identity and locator space to overlap, and distinguish between the two identity realms by the context of usage rather than by a prefix comparison.

It is also feasible to use the fully qualified domain name (FQDN) as an endpoint identity, undertaking a similar mapping as described above, using the FQDN as the lookup "key". The implication here is that there is no default 'address' that is to be associated with the endpoint identifier.

The syntactic properties of these two different identity realms have obvious considerations in terms of the manner in which these identities may be used within PDUs.

It is also an option to consider a new structured identity space

which is not generated through the reuse of IPv6 address values nor drawn from the FQDN. Given that the address space would need to be structured in such a fashion that permits it to be used as a lookup key to obtain the corresponding locator set, the obvious question in such an option is what additional or altered characteristics would be used in such an endpoint identity space that would distinguish it from either of the above approaches?

Instead of structured tokens that double as lookup keys to obtain mappings from endpoint identities to locator sets, the alternative is to use an unstructured token space, where individual token values are drawn opportunistically for use within a multi-homed session context. Here the semantics of the endpoint identity are subtly changed. The endpoint identity is not a persistent alias or reference to the identity of the endpoint, but a means to allow an EIP to confirm that two locators are part of the same mapped locator set for an endpoint. In this context the unstructured opportunistic endpoint identifier values are used in determining locator equivalence rather than in some form of lookup function.

5. Common Issues for Multi-Homing Approaches

The above overview encompasses a very wide range of potential approaches to multi-homing, and each particular approach necessarily has an associated set of considerations regarding its applicability.

There are, however, a set of considerations that appear to be common across all approaches, and they are examined in further detail in this section.

5.1 Triggering Locator Switches

Ultimately, regardless of the method of generation, a packet generated from a local multi-homed host to a remote host must have a source locator in the IP packet that is passed into the transit network. In a multi-homed situation the local multi-homed host has a number of self-referential locators that are equivalent aliases in almost every respect. The difference between locators is the inference that at the remote end the choice of locator may determine the path used to send a packet back to the local multi-homed host. The issue here is how does the local host make a selection of the "best" source locator to use? Obviously the parameters of this selection include the objective to select a locator that represents a currently viable path from the remote host to the local multi-homed host. Local routing information for the multi-homed host does not include this reverse path information. Equally, the local host does not necessarily know of any additional policy constraints that apply

to the remote host that may result in a remote host's preference to use one locator over another for the local host. Considerations of unicast reverse path forwarding filters also indicate that the selection of a source locator should result in the packet being passed to a site-exit router that is connected to the associated ISP transit provider, and that the site-exit router passes the packet to the associated ISP.

If the local multi-homed host is communicating with a remote multi-homed host, the local host may have some discretion in the choice of a destination locator. The considerations relating to the selection of a destination locator include considerations of local routing state (to ensure that the chosen destination locator reflects a viable path to the remote endpoint), policy constraints that may determine a "best" path to the remote endpoint. In such situations it may also be the case that the source address selection should also be considered in relation to the destination locator selection.

Another common issue is the consideration of the point when a locator is not considered to be viable, and the consequences to the transport session state.

- Transport Layer Triggers

  A change in state for a currently used path to another path could be triggered by indications of packet loss along the current path, or by transport session timeouts, assuming an internal signalling mechanism between the transport stack element and the locator pool management stack element.

- Routing Triggers

  Alternatively, in the absense of local transport triggers, the site exit router could communicate failure of the outbound forwarding path in the case where the remote host is multi-homed with an associated locator set. Conventional routing would be incapable of detecting a failure in the inbound forwarding path, so there are some limitations in the approach of using routing triggers to change locator bindings.

- Heartbeat Triggers

  An alternative to these approaches is the use of a session heartbeat protocol, where failure of the heartbeat would cause the session to seek a new locator binding that would re-establish the heartbeat.

The sensitivity of the locator-switch trigger is a consideration here. A very fine-grained sensitivity of the locator switch trigger

may generate false triggers arising from short-term transient path congestion, while coarse-grained triggers may impose an undue

performance penalty on the session due to an extended time to detect a path failure.

5.2 Session Startup and Maintenance

The next issue if that of the difference between the initial session startup mode of operation and the maintenance of the session state.

In a split endpoint identifier / locator environment there needs to be at least one initial locator associated with an endpoint identifier in order to establish an initial connection between the two hosts. This locator could be loaded into the DNS in a conventional fashion, or, if the endpoint identifier is a distinguished address value, the initial communication could be established using the endpoint identifier in the role of a locator (i.e. using this as a conventional address).

The initial actions in establishing a session would be simular. If the session is based on specification of a FQDN, the FQDN is first mapped to an endpoint identity value, and this endpoint identity value could then be mapped to a locator set. The locators in this set are then candidate locators for use in establishing an initial synchronized state between the two hosts. Once the state is established it is then possible to update the initial locator set with the current set of useable locators. This update could be part of the initial synchronization actions, or deferred until required.

This leads to the concept of the use of a 'distinguished' locator that acts as the endpoint identifier, and a pool of alternative locators that are associated with this 'home' locator. This association may be statically defined, using referential pointers in a third party referral structure (such as the DNS), or dynmically added to the session through the actions of the endpoint identity protocol stack element, or both.

6. Security Considerations

There are a significant number of security considerations that result from the action of distinguishing within the protocol suite endpoint

identity and locator identity.

It is not proposed to enumerate these considerations in detail within this draft, but to provide a distinct document that describes the security considerations of this domain. Subsequent revisions of this draft will refer the reader to this yet-to-be-drafted document.

7. Acknowledgements

The author acknowledges the exensive contribution of Margaret Wasserman in preparing the original draft of the summary of current approaches to multi-homing.

Normative References

Author's Address

Geoff Huston
Telstra

Appendix A. Notes on Various approaches

These notes were orginally drafted by Margaret Wasserman. The notes on various approaches are non-exclusive, i.e. solutions not reviewed or mentioned here are not ruled out of discussion. Also the review comments are not comprehensive, and the selection reflects the time constraints of the contributors to this section than any qualititive judgement on any of the approaches. The author is desirous, in future revisions of this draft, in augmenting this selection of reviewed approaches.

A.1 Host Identity Protocol (HIP)

HIP is an ID/Locator separation mechanism intended to solve a much wider problem space than site multi-homing. HIP uses cryptographic identifiers termed Host Identity Tags (HITs) at the application layer, which are mapped to locators (IP Addresses) by a HIP protocol stack layer that interfaces between the transport layer and IP

internetwork layer.

The essential characteristic of HIP is it use of opportunistic identity generation, as it uses a cyptographic host identifier as the basis of the persistent identity. The transport session cab be agile across locators, or even across IP protocol versions, as the HIT is used to determine session integrity. allowing the hosts to determine what packets legitimately form part of the session.

HIP is proposed as a new protocol element, located at layer 3.5 (i.e. above the internetwork IP layer and below the transport layer). The presentation to the transport layer uses 128 bit hash values (the HIT) in place of IP addresses, while the presentation to the internet layer uses conventional IP addresses.

Being opportunistic and unstructured, the HIT space is not an

efficient search space, nor can a HIT be used as a unique search key. HITs are part of an an equivalence function, to allow each host to determine that an incoming packet is part of an established session. HITs cannot be used as an identity value in a conventional referral sense (HostA wants to tell HostB to talk to HostC). While an application could pass a HIT to a third-party (and legacy applications would unknowingly do so), the third party would have no way to map that HIT to a locator (an IP address) as HIP does not include any global HIT->Locator mapping mechanism.

Summary:
- New Protocol Stack Element
- Layer 3.5 (Above IP, below Transport
- Unstructured, opportunistic identity values (non-referential)
- DNS rendezvous
- No Locator exchange protocol

Current IETF Documents:
- draft-moskowitz-hip-arch
- draft-moskowitz-hip
- draft-nikander-hip-mm
- draft-nikander-esp-beet-mode

A.2 Multihoming without IP Identifiers (NOID)

NOID proposes an approach for endpoint identifier and locator separation where the endpoint identifier space is drawn from the locator space. Instead of creating a new namespace for endpoint identifiers, the endpoint identifier may be chosen from the set of locators that can be used to reach a given endpoint. Until an event occurs that modifies the list of usable locators, the initial endpoint identifier value can serve as a locator.

NOID uses next-header values in the IPv6 header to indicate whether a given packet should be processed by the NOID layer. At a conceptual level, NOID adds a layer to the middle of IP above most IP processing, but below IPSec, fragmentation and reassembly functions.

NOID makes use of the global DNS as a mapping system between IDs and Locators. A node who wishes to communicate with another node can use the FQDN to get a list of possible locators (IP Addresses). That node will then choose one of the locators to use as an Application-level ID (AID).

NOID offers some support for application referrals. If Host A passes an AID to Host B that is supposed to point to Host C, Host B should be able to do a reverse DNS lookup to map the AID to an FQDN and then use the FQDN to get the complete set of locators. However, for this

to be effective, nodes would need to have both forward and reverse DNS entries. There might also be a need to dynamically update the DNS as a node becomes reachable or unreachable at certain locators.

Summary:
- New Protocol Stack Element
- Layer 3 (Inserted in the upper part of IP, below IPSEC and fragementation / reassembly
- Identity values based on locator set
- DNS rendezvous
- Identity peering protocol

Current IETF Documents:
- draft-nordmark-multi6-noid
- draft-templin-isnoid

A.3 Common Endpoint Locator Pools (CELP)

CELP explores the concept of sharing information about locator reachability between transport-layer "multi-addressing" mechanisms (such as SCTP and DCCP) and Internet-layer multiaddressing mechanisms, referred to in the draft as "wedge-layer approaches" (such as NOID). (This concept was originally discussed on the MULTI6 mailing list under the name 'SLAP'.)

The motivation behind CELP is that muliple multiaddressing mechanisms may be used (by different applications or for different connections) on a single endpoint, and that it would beneficial for those mechanisms to share information about the reachability of the IP addresses in a given locator pool. If a transport-layer mechansim, such as SCTP, could share its knowledge regarding the reachability of a certain locator, it might be possible to minimize or elimate Internet-layer control packets that are used to maintain that information at the Internet layer. In some ways, this is similar to IPv6 Neighbor Discovery's use of upper layer advice regarding neighbor reachability to avoid sending unncecessary ND packets.

This document offers a definition of the term "endpoint" that refers to a locator pool that may have a smaller scope than an entire IP node (i.e. a given locator pool may only contain a subset of the locators available on an IP node).

The CELP document is more of a consideration of approach than an actual proposal for a solution. It doesn't specify in detail how it would work with any particular transport-layer or Internet-layer multiaddressing mechanisms. However, it is an approach that could be applied to many combinations of solutions.

Summary:
- Considerations relating to sharing locator reachability information across session instances.

Current IETF Documents:
- draft-crocker-celp

A.4 Weak Identifier Multihoming Protocol (WIMP)

WIMP is an endpoint identifier / locator separation protocol that is heavily focused on mitigating the threats outlined in work in

progress on security threats in multi-homing scenarios [draft-nordmark-multi6-threats-00.txt]. The WIMP approach uses divided secrets and hash chaining to ensure that new locators are supplied by the same node that supplied the original locator.

WIMP uses a separate name space for 128-bit non-routable IDs that are never used in packets on the network. These IDs are locally generated for both local and remote nodes by hashing a nonce (for the initiator's endpoint identity) or the FQDN (for the responder's endpoint identity). (The approach assumes a requirement that all responders will have a FQDN.)

The WIMP protocol introduces a WIMP layer that maps between IDs and locators based on internal state. The WIMP layer is conceptually located within the network layer, above most IP processing and below IPsec, fragmentation/reassembly and destination options, similar to NOID.

Communication between two end-points requires establishment of a WIMP session. Once the session is established, it can be used for multiple simultaneous or sequential connections to the same end-point. During WIMP session establishment, WIMP introduces a separate header into the data packets, between the IP and TCP/UDP headers that contains information about the WIMP session. The WIMP session establishment packets can optionally be piggy-backed on data packets. WIMP does not introduce a separate header into all IPv6 packets. Instead, once a WIMP session is established, the IPv6 FlowID is used to hold an identifier for the WIMP host-pair context associated with a given packet.

WIMP is intended to provide a solution to some of the security concerns, particularly regarding connection hijacking, that have been raised for some other endpoint identity / locator separation mechanisms.

Reviewers of WIMP have raised some questions of this approach, particularly concerning the use of an optional header while operating

below IP fragmentation. The piggy-backing mechanism requires that the packets not be fragmented, but it doesn't explain how upper layers will become aware of the MTU limitations on those packets and/or how this mechanism would interact with Path MTU discovery. Like HIP, WIMP

makes no provision to handle application-level referrals and does not contain a mechanism for global endpoint identifier to locator mapping. It has also been pointed out that it is interesting to consider whether the WIMP approach to security, hash chaining, could be applied to other endpoint identity / locator separations mechanisms, such as NOID.

Summary:
- New Protocol Stack Element
- Layer 3 (Inserted in the upper part of IP, below IPSEC and fragementation / reassembly
- Identity values based on hash of FQDN
- Identity peering protocol

Current IETF Documents:
- draft-ylitalo-multi6-wimp

A.5 Host-Centric IPv6 Multihoming

Host-Centric Multihoming is, in some ways, the simplest way to address the IPv6 site multihoming problem. The concept is that every host in the multihomed site is configured with multiple prefixes that correspond to different service providers. Each host configures addresses within those prefixes and selects among those addresses when connecting to a remote host. This configuration is automated using Router Renumbering and IPv6 Address Autoconfiguration. However, this simple solution Layer 3 (inserted in the upper part of IP, below IPSEC and fragementation / reassembly has several practical limitations and drawbacks, and this draft attempt to address them.

In particular, the Host-Centric Multihoming proposal attempts to address the "site exit issue". Hosts cannot control the exit path that their packets will take from the local site, so hosts with multiple addresses may use a source IP address from one ISP on packets that end-up being routed through a different ISP. In many cases, the ISPs will run ingress filtering and will discard those packets.

One solution to the site exit problem is to changes the ISP ingress filters to accept all of the source address prefixes that are used within the site. Other approaches are to perform source-based routing within the site, to deploy a single site-exit router or to structure the network so that all exit routers are connected to a single DMZ network that employs source-based routing. A virtual DMZ can be

constructed by configuring a mesh of tunnels between all exit routers, tunneling packets to the correct exit router based on source address. Each of these solutions has operational drawbacks and/or introduces inefficiencies.

This proposal suggests another solution to the site exit problem called "source address discovery". Based on Path MTU discovery, this mechanism involves adding extra information to the ICMP Destination Unreachable message that the packet was discarded due to an ingress filter. This extra information indicates what address prefix should be used to pass the ingress filter. Rather than adding a field to the ICMP message, this extra information is communicated via the source address that the route Layer 3 (Inserted in the upper part of IP, below IPSEC and fragementation / reassembly).

It also proposes a "superior" alternative called "exit router discovery", which allows hosts to specify which exit router will be used for each packet. Instead of sending ICMP error messages when ingress filtering causes packets to be discarded, the exit router will send the equivalent of a redirect message and future packets with the same source/destination address pair will be tunneled to the indicated exit router. This mechanism involves tunneling to a site-exit anycast address that is derived from the sites' prefixes. The draft primary focuses on the specification of this "superior" approach, largely ignoring some pertinent questions such as: Will residential and enterprise-level IPv6 routers reall support anycast routing?

One important thing to note about the host-centric multihoming solution is that it doesn't appear to provide any ability for transport connections to survive a change in the topology that causes a host to become unreachable at an address that is currently used as a connection end-point. It also does not offer any support for legacy applications that do application-level referrals, requiring that a full set of locators be exchanged as part of the referral.

A.6 Summaries of Selected ID/LOC Separation Documents

This section summarizes a set of selected ID/Loc separation documents. The selection includes documents that appear to be active, and this section provides a very short summary of each one. The first sub-section lists documents that are new or updated since IETF 58 and the second sub-section lists older documents that remain active. The documents in each sub-section are listed alphabetically by draft filename.

A.6.1 New or Updated Documents Since IETF58

- TLC-FM: Transport Layer Common Framework for Multihoming
  draft-arifumi-multi6-tlc-fm
  This draft proposes a transport-layer mechanism for ID /Locator mapping. There is a conceptual layer within the transport layer that provides support for common multihoming functions. It is compatible with the use of Mobile IPv6 (MIP6) to provide mobility support.
  In TLC-FM, like SCTP, the ID consists of a collection of locators that may be used to reach a given host. It employs transport-level clues (such as TCP retransmissions) to decide when to switch locators. For UDP connections, ICMP error messages or application-level hints are necessary.
  This mechanism is not well enough specified to fully evaluate it, but it doesn't appear to offer any support for application-level referrals.

- Multi-Homing Tunnel Broker (MHTB)
  draft-bagnulo-multi6-mhtb
  This document defines an enhancement to RFC 3178, IPv6 Multihoming Support at Site Exit Routers, to reduce the administrative overhead of maintaining a configured tunnel for each multihoming association. However, this draft does not address another major drawback of the RFC 3178 approach, that it does not protect against the complete failure of one or more connected ISPs.

- Framework for Common Endpoint Locator Pools (CELP)
  draft-crocker-celp
  Dave Crocker and Avri Doria's CELP draft, reviewed in the previous section.

- Multi-Homing: the SCTP Solution
  draft-coene-multi6-sctp
  One confusing question about the direction of this work is why SCTP is being discussed as a "solution" to site multihoming, when a clear requirement for a site multihoming solutions is the ability to support existing TCP-based and UDP-based applications. This document isn't really a proposal, though --

it consists of answers to the questions posted in Eliot Lear's "Things MULTI6 Developers Should Think About" draft, and does not discuss how SCTP does (or doesn't) address the requirements outlined in the Multi6 requirements RFC.
An interesting thing about this proposal is that it claims that SCTP is not an ID/Loc separation mechanism, however in some academic sense it actually is. The ID is the group of available IP addresses, and the locator is whichever address is currently

being used for communication. SCTP also experiences the same complexities as other proposals (AKA NOID, CELP) that use a pool of locators as the ID -- How do you choose which locator to use? And how do you detect when a member of the pool becomes invalid for use as a locator? So, while it isn't actually a solution for site multihoming, SCTP may provide some useful experiences and mechanisms that may apply to a class of possible solutions.

- Host Identity Protocol (HIP) Rendezvous Mechanisms
  draft-eggert-hip-rendezvous-00.txt
  This is an overview draft that discusses the concept of HIP rendezvous mechanisms to improve the applicability of HIP for mobility and multihoming. This is a survey document that outlines the problem and discusses different type of solutions to the problem.

- Host-Centric IPv6 Multihoming
  draft-huitema-multi6-hosts
  Draft by Christian Huitema and others, described above.

- Things MULTI6 Developers Should Think About
  draft-lear-multi6-things-to-think-about
  Eliot Lear's efforts to collect a set of practical questions that should be considered for all MULTI6 protocols.

- Host Identity Protocol (HIP)
  draft-moskowitz-hip
  This is the base protocol specification for HIP. Along with the HIP architecture, these documents form the basis of the HIP work.

- Consideration on HIP Based IPv6 Multi-Homing

draft-nikander-multi6-hip
Pekka Nikander's document that submits HIP as a solution for the MULTI6 problem space.

- 8+8 Addressing for IPv6 End to End Multihoming
  draft-ohta-multi6-8plus8

- Threats Relating to Transport Layer Protocols Handling Multiple Addresses
  draft-ohta-multi6-threats

- Multihoming Using IPv6 Addressing Derived from AS Numbers
  draft-savola-multi6-asn-pi
  This draft provides a mechanism for organizations that have been assigned a 16-bit AS number to use that number to

auto-generate a globally routable, provider-independent address prefix.

- Problem Statement: HIP Operation over Network Address Translators
  draft-stiemerling-hip-nat
  Summarizes the problems with running HIP and IPsec-based data transmission across NATs.

- Operational Approach to Achieve IPv6 Multihomed Network
  draft-toyama-multi6-operational-site-multihoming
  This document proposes to support site multihoming in IPv6 by assigning additional /32 prefixes and AS numbers to "groups" of providers who will provide multihomed /48 prefixes to their mutual customers.
  It is currently unclear to the reviewer how/if this proposal would work and/or scale since it seems to involve two different providers advertising the same /32 and the same AS number into the default free zone. It requires some type of peering "to share prefix assignments" between ISPs, and the diagram shows some type of connection between the ISPs, but I don't know what the details of that connection are.
  It also has the potential to more quickly exhaust the AS number space and to result in a substantially larger number of routes in default free routers, since the number of "groups" could scale exponentially with the number of providers.

- Crypto Based Host Identifiers (CBHI)
  draft-van-beijnum-multi6-cbhi
  This draft defines a crytographic mechanism for generating host identifiers. It is intended for use with other protocols that require host identifiers, such as ODT (see below).

- On Demand Tunneling for Multihoming (ODT)
  draft-van-beijnum-multi6-odt
  This draft discusses an automatic tunnelling-based solution for multihoming.

- Weak Identifier Multihoming Protocol (WIMP)
  draft-ylitalo-multi6-wimp
  WIMP proposal, described above.

A.6.2 Older Documents that Remain Active/Interesting

- RFC 3582: Goals for IPv6 Site-Multihoming Architectures

- Choices for Multiaddressing
  draft-crocker-mast-analysis

- What's In a Name: Thoughts from the NSRG
  draft-irtf-nsrg-report

- A Roadmap for Multihoming in IPv6
  draft-kurtis-multi6-roadmap

- Host Identity Protocol (HIP) Architecture
  draft-moskowitz-hip-arch-05.txt

- End-Host Mobility and Multi-Homing with Host Identity Protocol (HIP)
  draft-nikander-hip-mm

- Threats Relating to IPv6 Multihoming Solutions
  draft-nordmark-multi6-threats-00.txt

- Multihoming without IP Identifiers (NOID)
  draft-nordmark-noid
  Erik Nordmark's NOID specification, described above.

A.6.3 Related Multi-Homing drafts, Status unknown

This is a list of ID/Loc separation and/or MULTI6 documents, listed alphabetically by draft name.

- Extension Header for Site-Multi-homing Support
  draft-bagnulo-multi6-mhexthdr

- Application of the MIPv6 Protocol to the Multi-Homing Problem
  draft-bagnulo-multi6-mnm

- Multiple Address Service for Transport (MAST): An Extended Proposal
  draft-crocker-mast-proposal

- NAROS : Host-Centric IPv6 Multihoming with Traffic Engineering
  draft-de-launois-multi6-naros

- Application and Use of the IPv6 Provider Independent Global Unicast Format
  draft-hain-ipv6-pi-addr-use

- Simple Dual Homing Experiment
  draft-huitema-multi6-experiment-00.txt

- Host-Centric IPv6 Multihoming
  draft-huitema-multi6-hosts

- IPv4 Multihoming
  draft-ietf-multi6-v4-multihoming
  This documents how multi-homing is supported at present in the IPv4 protocol domain.

- Multihoming in IPv6 by Multiple Announcement of Longer Prefixes
  draft-kurtis-multihoming-longprefix

- Multihoming using 64-bit Crypto-based IDs
  draft-nordmark-multi6-cb64

- Strong Identity Multihoming using 128-bit Identifiers (SIM/

CBID128)
draft-nordmark-multi6-sim

- IPv6 Address Assignment and Route Selection for End-to-End Multihoming
  draft-ohira-assign-select-e2e-multihome

- Hierarchical IPv6 Subnet ID Autoconfiguration for Multi-Address Model Multi-Link Multihoming Site
  draft-ohira-multi6-multilink-auto-prefix-assign

- Hierarchical IPv6 Subnet ID Autoconfiguration for Multi-Address Model Multi-Link Multihoming Site
  draft-ohira-multi6-multilink-auto-prefix-assign

- The Architecture of End to End Multihoming
  draft-ohta-e2e-multihoming-05.txt

- 8+8 Addressing for IPv6 End to End Multihoming
  draft-ohta-multi6-8plus8-00.txt

- Threats Relating to Transport Layer Protocols Handling Multiple Addresses
  draft-ohta-multi6-threats-00.txt

- Multihomed ISPs and Policy Control
  draft-ohta-multihomed-isps-00.txt

- GAPI: A Geographically Aggregatable Provider Independent Address Space to Support Multihoming in IPv6
  draft-py-multi6-gapi

- Multi Homing Translation Protocol (MHTP
  draft-py-multi6-mhtp-01.txt

- Multihoming Using IPv6 Addressing Derived from AS Numbers
  draft-savola-multi6-asn-pi-01.txt

- IPv6 Site Multihoming: Now What?
  draft-savola-multi6-nowwhat

- Operation of NOID Multihoming Protocol on ISATAP Nodes
  draft-templin-isnoid

- LIN6: A Solution to Multihoming and Mobility in IPv6
  draft-teraoka-multi6-lin6

- Operational Approach to achieve IPv6 multihomed network
  draft-toyama-multi6-operational-site-multihoming-00.txt

- Two Prefixes in One Address
  draft-van-beijnum-multi6-2pi1a-00.txt

- Crypto Based Host Identifiers
  draft-van-beijnum-multi6-cbhi-00.txt

- Provider-Internal Aggregation based on Geography to Support Multihoming in IPv6
  draft-van-beijnum-multi6-isp-int-aggr-01.txt

- On Demand Tunneling For Multihoming
  draft-van-beijnum-multi6-odt-00.txt

Intellectual Property Statement

Full Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.