

Individual Submission
Internet-Draft
Expires: April 15, 2005

G. Huston
APNIC
October 15, 2004

Architectural Approaches to Multi-Homing for IPv6
draft-ietf-multi6-architecture-01.txt

Status of this Memo

This document is an Internet-Draft and is subject to all provisions of [section 3 of RFC 3667](#). By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she become aware will be disclosed, in accordance with [RFC 3668](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on April 15, 2005.

Copyright Notice

Copyright (C) The Internet Society (2004).

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.

Internet-Draft

Multi6 Architecture

October 2004

Document Revision Notes

The following changes have been made to the draft:

[draft-ietf-multi6-architecture-01:](#)

[Section 4.2:](#) Multi-homing: Mobility

New section added based on contribution from Marcelo Bagnulo.

[Section 5.2:](#) Persistent, Opportunistic and Ephemeral Identities

Additional text added about considerations if id/locator split in persistent identities and the requirements of multi-homing.

[Appendix A:](#)Notes on Various approaches

This section removed, to be replaced by a new WG document.

[draft-ietf-multi6-architecture-00:](#)

Notes: IPv6

Added text outlining the MIPv6 return routeability tests and the implications of this approach with multi-homing.

[Section 2:](#) The Multi-Homing Space

Added text on session initiation by the local host in the circumstance of degraded path connectivity.

[Section 4.2:](#) Multi-homing: Identity Considerations

Added text on session initiation by the local host in the circumstance of degraded path connectivity. Also added text to cover the case that the remote host may need to discover the need to perform a locator switch for the multi-homed host in ways other than direct notification from the local end.

[Section 4.4:](#) Multi-homing: Modified Protocol Element

Change "single endpoint-to-endpoint session" to "single endpoint to single endpoint communication".

[Section 4.5:](#) MModified Site-Exit and Host Behaviors

Change NAT analysis reference to the multi6-threats draft.

[Section 5.1](#) Endpoint Identity Structure

Added a qualification about unstructured identities and their utility as a resolution key.

Huston

Expires April 15, 2005

[Page 2]

Internet-Draft

Multi6 Architecture

October 2004

[Section 5.3.2](#) Locator Selection

Added this section which describes the considerations of traffic engineering in the context of locator selection.

[Section 5.3.3](#) Layering Identity

Added qualification about use of transport (session) identities for UDP.

[Section 6.1](#) Establishing Session State

Qualified the use of "transport" to be "identity protocol element", indicating that this may be transport, IP of a wedge layer, and edited the text to reflect multi-homing capabilities in the protocol stack. Added text on locator discovery and selection in the functional decomposition of session establishment.

[draft-huston-multi6-architectures-01:](#)

[Section 2](#): The Multi-Homing Space

Added text to include consideration of session initiation in the face of changes to the connectivity topology, and a note about the potential to consider traffic engineering across multiple paths.

[Section 3](#): Functional Goals and Considerations

Changed 'requirements' to 'goals'.

[Section 5.1](#) Endpoint Identity Structure

Added consideration of disambiguating locators and identities when identities are drawn from the same address space as locators. Added text about identities drawn from PA space and the problems this raises. Also added text about disambiguating DNS FQDN pseudo-anycast from DNS-based multi-homing with

equivalent locator sets.

[Section 5.2](#) Persistent, Opportunistic and Ephemeral Identities

New section added to the draft considering the implications of these three approaches to identity.

[Section 5.3.1](#) Triggering Locator Switches

Added section on ICMP triggers.

[Section 5.3.3](#) Layering Identity

New section added, considering the implications of placing endpoint identity functionality in the transport or internetwork protocol elements, or as a wedge element, conceptually placed

between these two elements.

[Section 6](#). Functional Decomposition of Multi-Homing Approaches

New section added.

Table of Contents

1.	Introduction	5
2.	The Multi-Homing Space	5
3.	Functional Goals and Considerations	7
4.	Approaches to Multi-Homing	8
4.1	Multi-Homing: Routing	9
4.2	Multi-Homing: Mobility	10
4.3	Multi-homing: Identity Considerations	12
4.4	Multi-homing: Identity Protocol Element	14
4.5	Multi-homing: Modified Protocol Element	15
4.6	Modified Site-Exit and Host Behaviors	16
5.	Approaches to Endpoint Identity	17
5.1	Endpoint Identity Structure	18
5.2	Persistent, Opportunistic and Ephemeral Identities	20
5.3	Common Issues for Multi-Homing Approaches	23
5.3.1	Triggering Locator Switches	24
5.3.2	Locator Selection	25
5.3.3	Layering Identity	26
5.3.4	Session Startup and Maintenance	28
5.3.5	Dynamic Capability Negotiation	30
5.3.6	Identity Uniqueness and Stability	30
6.	Functional Decomposition of Multi-Homing Approaches	31

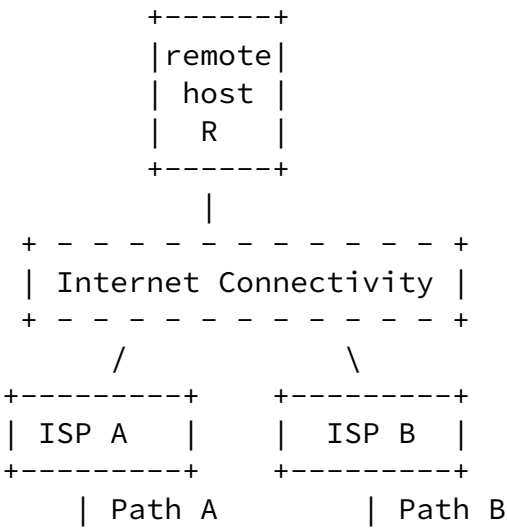
6.1	Establishing Session State	31
6.2	Rehoming Triggers	32
6.3	Rehoming Locator Pair Selection	32
6.4	Locator Change	32
6.5	Removal of Session State	32
7.	Security Considerations	32
8.	Acknowledgements	33
9.	Informative References	33
	Author's Address	34
	Intellectual Property and Copyright Statements	35

[1.](#) Introduction

The objective of this analysis 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

A simple formulation of the multi-homing environment is indicated in Figure 1.



that local attempts to initiate a communication with remote hosts should take into account the current connectivity state in undertaking locator selection and setting up initial locator sets.

In addition there is the potential consideration of being able to distribute traffic load across a number of network paths according to some pre-determined objective, as a form of traffic engineering.

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. Functional Goals and Considerations

[RFC 3582](#) [1] documents some goals that a multi-homing approach should attempt to address. These goals include:

- o redundancy

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

The reader is referred to [1] for a complete description of each of these goals.

In addition, [2] 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:

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

In evaluating various approaches, further consideration also include:

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

4. Approaches to Multi-Homing

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

- o Routing
Use the IPv4 multi-homing approach
- o Mobility
Use the IPv6 Mobility approach
- o New Protocol Element
Insertion of a new element in the protocol stack that manages a persistent identity for the session
- o Modify a Protocol Element
Modify the Transport or IP protocol stack element in the host in order to support dynamic forwarding locator change
- o 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

Internet-Draft

Multi6 Architecture

October 2004

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 goals for multi-homing approaches with one notable exception: scalability. 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

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: Mobility

Preserving established communications through movement is similar to preserving established communications through outages in multihomed sites as both scenarios require the capability of dynamically changing the locators used during the communication while maintaining, unchanged, the endpoint identifier used by ULP. Since MIPv6 protocol [\[3\]](#) already provides the required support to preserve established communications through movement, it seems worthwhile to explore whether it could also be used to provide session survivability in multihomed environments.

MIPv6 uses a preferred IP address, the Home Address (HoA), as a stable identifier for the mobile node (MN). This identifier is then dynamically mapped to a valid locator (Care-of Address, or CoA) that corresponds to the current attachment point within the network topology. When the MN is at the Home Network, the HoA is used both as locator and as identifier. When the MN is not at the Home Network, the HoA is used as an identifier, and the CoA is used as locator. A relaying agent (Home Agent) placed in the Home Network is used to forward packets addressed to the HoA to the current location, specified by the CoA. After each movement, the MN must inform its Home Agent of the new CoA, and optionally inform those entities with which it has established communications with (Correspondent Nodes, or CNs). The mapping information between the HoA and the current CoA is conveyed using Binding Update (BU) messages. When the BU message is exchanged between the MN and the Home Agent, it is possible to assume the existence of a pre-established Security Association that can be used to protect the binding information. However, when the BU

message is exchanged between the MN and the CN, it is not possible to assume the existence of such Security Association. In this case, it is necessary to adopt an alternative mechanism to protect the binding information contained in the message. The selected mechanism is called Return Routability procedure and the background for its design is detailed in [4]. The goal of the mechanism is to allow the CN to verify that the MN that is claiming that a HoA is currently located at a CoA is entitled to make such claim, which essentially means that the HoA was assigned to the MN, and that the MN is currently located at the CoA. In order to verify these updates the CN sends two different secrets, one to the claimed HoA and another one to the claimed CoA. If the MN receives both secrets, this means that the

Home Agent located at the Home Network has a trust relationship with the MN and it has forwarded the secret sent to the HoA, and that the MN is receiving packets sent to the CoA. So, by including authorization information derived from both secrets within the BU message, the MN will be able to prove to the CN that the claimed binding between the HoA and the CoA is valid.

The lifetime of the binding that is created in the CN using authorization information obtained through the Return Routability procedure is limited to a 7 minutes, in order to prevent time-shifted attacks [4]. In a time-shifted attack, an attacker located along the path between the CN and the MN forges the Return Routability packet exchange. The result of such attack is that the CN will forward all the traffic addressed to the HoA to the CoA selected by the attacker. The attacker can then leave the position along the path but the effects of the attack will remain until the binding is deleted, shifting in time the effect of the attack. By limiting the lifetime of the binding in the CN, the effect of this attack is reduced to 7 minutes, because after that period a new Return Routability procedure is needed to extend the binding lifetime. It should be noted that the Return Routability procedure is vulnerable to Man-In-The-Middle attacks, since an attacker located along the path between the CN and the MN can forge the periodic Return Routability packet exchange.

The possible application of the MIPv6 protocol to the multihoming problem would be to use BU messages to convey information about alternative addresses to be used after the outage.

In this scenario, the multihomed host adopts the MN role and the host

outside the multihomed site adopts the CN role. When a communication is established between the multihomed host and the external host, the address used for initiating the communication is used as a HoA. The communication continues using this address as long as no outage occurs. If an outage occurs and the HoA becomes unreachable, an alternative address of the multihomed node is used as a CoA. In this case, the multihomed node sends a BU message to the external host, informing about the new CoA to be used for the HoA, so that the established communication can be preserved using the alternative address. However, such BU message has to be validated using authorization information obtained through the Return Routability procedure, which implies that the binding lifetime will be limited to a fixed period of no more than 7 minutes. The result is that the binding between the HoA and the new CoA will expire after this interval has elapsed, and then the HoA will be used for the communication. Since the HoA is unreachable because of the outage, the communication will be interrupted. It should be noted that it is not possible to acquire new authorization information by performing a new Return Routability procedure, because it requires communication

through the HoA, which is no longer reachable. Consequently, a mechanism based on the MIPv6 BU messages to convey information about alternative addresses will preserve communications only for 7 minutes.

The aspect of MIPv6 which appears to present issues in the context of multi-homing is the return routeability mechanism. In MIPv6 identity validity is periodically tested by return routeability of the identity address. This regular use of a distinguished locator as the identity token cannot support return reachability in the multi-homing context in the event of extended path failure of the path that is associated with the identity locator.

[4.3](#) 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 scalable 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 conceivably 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 packet 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 the local multi-homed host that 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 some form of 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 failures. 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 site continues 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's address prefix. In the event that the local host wishes to initiate a session at this point, then it may need to use an initial source locator that reflects the situation that the only viable destination address to use is the one that is based on ISP B's address prefix. It may be the case that the local host is not always aware of this return routeability constraint, or it may not be able to communicate this information directly to R, in which case R needs to discover or be passed this information in other ways.

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 is a fundamental change in the semantics of an IP address in the context of the role of the endpoint address within the end-to-end architectural model [5]. 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.

[4.4](#) 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.

[4.5](#) 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 single endpoint communication.

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.

[4.6](#) 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 security considerations, and any packet rewriting mechanism has to provide proper protection against the attacks described in [6], in particular against redirection attacks

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.

5. 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 remainder of the PDU 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. This is equivalent to generic tunneling approaches where the outer encapsulation of the transmitted packet contains location address information, while the next level packet header contains information that is to be exposed and used at the location endpoints, being, in this case, identity information.

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 to 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.

[5.1](#) 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 endpoint identity / locator 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 imply 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 prefixes, 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 distinguished identity 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. However, this reuse of the locator token space as identity tokens has the potential to create the anomalous situation where a particular locator value is used as an identity value by a different endpoint. It is not clear that the identity and locator contexts can be clearly disambiguated in every case, which is a major drawback to this particular approach.

If identity values are to be drawn from the protocol's address space it would appear that the basic choice is to either draw these identity values from a different part of the address space, or use a distinguished or home address as both a locator and an identity. This latter option, that of using a locator as the basis of an endpoint identity on a locator, when coupled with a provider-aggregated address distribution architecture leads to the outcome of a multi-homed site using a provider-based address prefix as a common identity prefix. As with locator addresses in the context of a single-homed network, a change of provider connectivity implies a consequent renumbering of identity across the multi-homed site. If avoiding such forced renumbering is a goal here, there

would be a preference in drawing identity tokens from a pool that is not aligned with network topology. This may point to a preference from this sector to use of identity token values that are not drawn from the locator address space.

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, as the FQDN can be used in the context of session establishment, and a DNS query used to establish a set of initial locators. Of course it is also the case that there may not necessarily be a unique endpoint associated with a FQDN, and in such cases if there were multiple locator addresses associated with the FQDN via DNS RRs, shifting between locators may imply directing the packet to a different endpoint where there is no knowledge of the active session on the original endpoint.

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. If such unstructured tokens are used in a limited context then 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 the identity protocol element to confirm that two locators are part of the same mapped locator set for a remote endpoint. In this context the unstructured opportunistic endpoint identifier values are used in determining locator equivalence rather than in some form of lookup function.

[5.2](#) Persistent, Opportunistic and Ephemeral Identities

The consideration in the previous section highlights one of the major aspects of variance in the method of supporting a split between

identity and location information.

One form uses a persistent identity field, by which it is inferred that the same identity value is used in all contexts where this form of identity is required, in support of concurrent sessions, and in support of sequential sessions. This form of identity is intended to remain constant over time and over changes in the underlying connectivity. It may also be the case that this identity is completely distinct from network topology, so that the same identity is used irrespective of the current connectivity and locator addressing used by the site and the host. In this case the identity

is persistent, and the identity value can be used as a reference to the endpoint stack. This supports multi-party referrals, where if parties A and B establish a communication, B can pass A's identity to a third party C, who can then use this identity value to be the active party in establishing communication to A.

If persistent identifiers are to be used to initiate a session, then it follows that the identity is used as a lookup key to establish a set of locators that are associated with the identified endpoint. It is desirable that this lookup function be deterministic, reliable, robust, efficient and trustable. The implication of this is that such identities must be uniquely assigned, and experience in identity systems points to a string preference for a structured identity token space that has an internal hierarchy of token components. These identity properties have significant commonality with those of unicast addresses and domain names. The further implication here is that persistent structured identities also rely on the adoption of well-ordered distribution and management mechanisms to preserve their integrity and utility. Such mechanisms generally imply a significant overhead in terms of administrative tasks.

As noted in the previous section, an alternative form of identity is an unstructured identity space, where specific values are drawn from the space opportunistically. In this case the uniqueness of any particular identity value is not assured in all cases. The use of such identities as a lookup key to establish locators is also altered, as the unstructured nature of the space has implications relating to the efficiency of the lookup, and the authenticity of the lookup is weakened due to the inability to assure uniqueness of the identity key value. A conservative approach to unstructured identities limits their scope of utility, such as per-session identity keys. In this scenario the scope of the selected identity is limited to the parties who are communicating, and limited to the duration of the communication session. The implication of this limitation is that the identity is a session-level binding point to allow multiple locators to be bound to the session, and the identity cannot be used as a reference to an endpoint beyond the context of

the session. Such opportunistic identities with explicitly limited scope do not require the adoption of any well-ordered mechanisms of token distribution and management.

Another form of identity is an ephemeral form, where a session identity is a shared state between the endpoints, established without the exchange of particular token values that take the role of identity keys. This could take the form of a defined locator set, or the form of a session key derived from some set of shared attributes of the session, as two examples here. In this situation there is no form of reference or use of an identifier as a means of initiating a session. The ephemeral identity value has a very limited role in terms of allowing each end to reliably determine the semantic equivalence of a set of locators within the context of membership of a particular session.

The latter two forms of identity represents a approach to identity that minimizes management overhead, and provides mechanisms that are limited in scope to supporting session integrity. This implies that support for identity functions in other contexts and at other levels of the protocol stack, such as within referrals, in the use of identities within an application's data payload, or as a key used to initiate a communication session with a remote endpoint would need to be supported by some other identity function. Such per-session limited scope identities imply that the associated multi-homing approaches must use existing mechanisms for session startup, and the adoption of a session-based identity and associated locator switch agility becomes a negotiated session capability. On the other hand, the use of a persistent identity as a session initiation key implies that identity is part of the established session state, and locator agility can be an associated attribute of the session, rather than a subsequent negotiated capability. In a heterogeneous environment where such identity capability is not uniformly deployed this would imply that if a session cannot be established with a split identity locator binding, the application should be able to back off to a conventional session startup by mapping the identity to a specific locator value and initiating a session using such a value. The reason why the application may want to be aware of this distinction is that if the application wishes to use self-referential mechanisms within the application payload, it would appear to be appropriate to use an identity-based self-reference only in the context of a session where the remote party was aware of the semantic properties of this referential tag.

In terms of functionality and semantics opportunistic identities form a superset of ephemeral identities, although their implementation is significantly different. Persistent identities support a superset of the functionality of opportunistic identities, and again the

implementations will differ.

In the context of support for multi-homing configurations, use of ephemeral identities that are used in the context of locator equivalence appears to represent a viable approach which allows a negotiated use of multiple locators within the context of communication between a pair of hosts in most contexts of multi-homing. However ephemeral identities offer little more in terms of functionality. They cannot be used in referential contexts, cannot be used to initiate communications, and provide limited means of support for various forms of mobility, and impose some constraints on the class of multi-homed scenarios that can be supported. Ephemeral identities are generated in the context of an established communication state, and the implication in terms of multi-homing is that at the two end points need to have discovered through existing mechanisms a viable pair of locators prior to generating an ephemeral identity binding. The implication is that there is some form of static 'home' for the end points which is discovered by conventional referential lookup.

The use of a persistent identity space that supports dynamic translation between an equivalent set of locators and one or more equivalent identity values offers the potential for greater flexibility in application, extending beyond the multi-homing configuration to various contexts of nomadism and mobility, as well as extending into service-specific functions, depending on how the mapping between identities and locators is managed. However it remains an open question as to the nature of secure mapping mechanisms that would need to be used in the more general context of identity to locator mapping, and it is also an open question as to how the mapping function would relate to viable endpoint-to-endpoint connectivity. It is a common aspect of identity realms that the most critical aspect of the realm is the nature of the resolution of the identity into some other attribute space.

It appears reasonable to observe that, within certain constraints, multi-homing does not generically require the overhead of the introduction of a fully distinct persistent identity space and the associated identity resolution functionality, and if the nature of the multi-homing space in this context is one of the use of a token to allow efficient detection of locator equivalence for session survivability, then ephemeral identities appear to be an adequate mechanism for this role.

[5.3](#) Common Issues for Multi-Homing Approaches

The above overview encompasses a very wide range of potential

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.

[5.3.1](#) 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.

- o 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

transport-level signaling, or by transport session timeouts, assuming an internal signaling mechanism between the transport stack element and the locator pool management stack element.

- o ICMP Triggers

Path failure within the network may generate an ICMP Destination Unreachable ICMP packet being directed back to the sender. Rather than sending this signal to the transport level as an indicator of session failure, the IP layer should redirect the notification identity module as a trigger for a locator switch.

- o Routing Triggers

Alternatively, in the absence 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.

- o 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.

[5.3.2](#) Locator Selection

The selection of a locator to use for the remote end is obviously constrained by the current state of the topology of the network, and the primary objective of the selection process is to select a viable locator that allows the packet to reach the intended destination point. The selection of a source locator can be considered as an indication of preference to the remote end of a preferred locator to use for the local end. However, where there are two or more viable locators that could be used, the selection of a particular locator may be influenced by a set of additional considerations.

The selection of a particular locator from a viable locator set implies a selection of one particular network path in preference to other viable paths. An implication of this selection process is that

the host is that path selection policy and, by inference, traffic engineering, is not constrained to a network-based operation of path manipulation through adjustment of forwarding state within network elements. There is an interaction between the locator selection process and traffic engineering functions. The use of an address selection policy table, as described in [RFC 3484](#) [7] are relevant to the selection process.

The element that performs the locator selection, either as a protocol element within the host or a selection undertaken at a site-exit router, also determines traffic policy, so the choice of using remote packet locator rewriting or host based locator selection shifts the policy capability from one element to the other.

If hosts perform this policy determination, then a more fine grained outcome may be achievable, particularly if the anticipated traffic characteristics of the application can be signalled to the locator selection process. A further consideration here appears to be that hosts may require additional information if they are to make locator address selection decisions based on some form of metric of relative load currently being imposed on select components of a number of end-to-end network paths. These considerations raise the broader issue of traffic engineering being a network function entirely independent of host function or an outcome of host interaction with the network. IN the latter case there is also the consideration of whether the host is to interact with the network, and, if so, how is

this interaction is to be signalled to hosts.

[5.3.3](#) Layering Identity

The consideration of triggering locator switch highlights the observation that differing information and context is present in each layer of the protocol stack. This impacts on how identity / locator bindings are established, maintained and expired.

These impacts include questions of what amount of state is kept, by which element of the protocol stack, at what level of context (dynamic or fixed, and per session or per host). It also includes considerations of state maintenance, such as how stale or superfluous state information is detected and removed. Does only one piece of code have to be aware of this identity/locator binding or do multiple transport protocols have to be altered to support this functionality? If so, are such changes common across all transport protocols, or do different protocols require different considerations in their treatment of this functionality?

It is noted that the set of approaches considered here include proposals to place this functionality within the IP layer, with the

end-to-end transport protocol layer and as a shim between the IP and transport protocol layer.

Placing this identity functionality at the transport protocol layer implies that the identity function can be tightly associated with a transport session. In this approach session startup can trigger the identity / locator initial binding actions and transport protocol timeouts can be used as triggers for locator switch actions. Session termination can trigger expiration of local identity / locator binding state. Where per-session opportunistic identity token values are being used, the identity information can be held within the overall session state. In the case of persistent identity token values the implementation of the identity can also choose to use per-session state, or may choose to pool this information across multiple sessions in order to reduce overheads of dynamic discovery of identity / locator bindings for remote identities in the case of multiple sessions to the same remote endpoint.

One of the potential drawbacks of placing this functionality within

the transport protocol layer is that it is possible that each transport protocol will require a distinct implementation of identity functionality. This is a considerable constraint in the case of UDP, where the UDP transport protocol has no inherent notion of a session state.

An alternative approach is to use a distinct protocol element placed between the transport and internet layers of the protocol stack. The advantage of this approach is that it would offer a consistent form of mapping between identities and locators for all forms of transport protocols. However this protocol element would not be explicitly aware of sessions and would either have to discover the appropriate identity / locator mapping for all identity-addressed packets passed from the transport protocol later, irrespective of whether such a mapping exists and whether this is part of a session context, or have an additional mechanism of signaling to determine when such a mapping is to be discovered and applied. At this level there is also no explicit knowledge of when identity / locator mapping state is no longer required, as there is no explicit signaling of when all flows to and from a particular destination has stopped and resources consumed in supporting state can be released. Also, such a protocol element would not be aware of transport level timeouts, so that additional functionality would need to be added to the transport protocol to trigger a locator switch at the identity protocol level. Support of per-session opportunistic identity structure is more challenging in this environment, as the transport protocol layer is used to store and manipulate per-session state.

It is also possible to embed this identity function within the

internet protocol layer of the protocol stack. As noted in the previous section, per session information is not readily available to the identity module, so that opportunistic per-session identity values would be challenging to support in this approach, as well as determining when identity / locator state information should be set up and released. It would also appear necessary to signal transport level timeouts to the identity module as a locator switch trigger. Some attention needs to be given in this case to synchronizing locator switches and IP packet fragmentation, and consideration of IPSEC is necessary in this case, in order to avoid making changes to the address field in the IP packet header that trigger a condition at the remote end where the packet is not recognizable in the correct

context.

[5.3.4](#) Session Startup and Maintenance

The next issue is 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 similar. 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 dynamically added to the session through the actions of the endpoint identity protocol stack element, or both.

If opportunistic identities are used where the identity is not a fixed discoverable value but one that is generated in the context of

a session then additional actions must be performed at session startup. In this case there is still the need for defined locators that are used to establish a session, but then an additional step is required to generate session keys and exchange these values in order to support the identity equivalence of multiple locators within the

ensuing session. This may take the form of a capability exchange and an additional handshake and associated token value exchange within the transport protocol if an in-band approach is being used, or it may take the form of a distinct protocol exchange at the level of the identity protocol element, performed out-of-band from the transport session.

Some approaches are capable of a further distinction, namely that of initial session establishment and that of establishment of additional shared state within the session to allow multiple locators to be treated as being bound to a common endpoint identity. It is not strictly necessary that such additional actions be performed at session startup, but it appears that such actions need to be performed prior to any loss of end-to-end connectivity on the selected initial locator, so that any delay in this additional state exchange does increase the risk of session disruption due to connectivity changes.

This raises a further question of whether the identity / locator split is a capability negotiation performed per session or per remote end, or whether the use of a distinguished identity value by the upper level application to identify the remote end triggers the identity / locator mapping functionality further down in the protocol stack at the transport level, and that this is performed without any further capability negotiation within the session.

Within the steps related to session startup there is also the consideration that the passive end of the connection follows a process where it may need to verify the proposed new address contained in the source address of incoming packets before using it as a destination address for outgoing packets. It is not necessarily the case that the sender's choice of source address reflects a valid path from the receiver back to the source. While using this offered address appears to offer a low overhead response to connection attempts, if this response fails the receiver may need to discover the full locator set of the remote end through some locator discovery mechanism in order to establish whether there is a viable locator that can use a forwarding path that reaches the remote end.

Alternatively, the passive end would use the initially offered locator and if this is successful leave it to the identity modules in each stack to exchange information to establish the current complete locator set for each end. This approach implies that the active end

of a communication needs to cycle through all of its associated locators as source addresses until it receives a response or exhausts its locator set. While this may extend the time to confirm that no path exists to the remote end, it has the potential to improve the characteristics of the initial exchange against denial of service attacks that could force the remote end to engage in a high volume of spurious locator lookups.

[5.3.5](#) Dynamic Capability Negotiation

The common aspect of these approaches is that they all involve changes to the end-to-end interaction, as both ends of the communication need to be aware of this separation. The implication is that this form of support for multi-homing is relatively sweeping in its scope, as the necessary changes to support multi-homing extend beyond changes to the hosts and/or routers within the multi-homed site and encompass changes to the IPv6 protocol itself. It would be prudent when considering these changes to evaluate associated mechanisms that allow the communicating endpoints to discover each other's capabilities and only enable this form of split identity / locator functionality when it is established that both ends can support it.

It is a corollary of this form of negotiated capability that it is not strictly necessary that only one form of functionality can be negotiated in this way. If the adoption of a particular endpoint identity / locator mapping scheme is the outcome of a negotiation between the endpoints then it would be possible to negotiate to use one of a number of possible approaches. There is some interaction between the approach used and the form of endpoint identity, and some care needs to be taken that any form of acceptable outcome of the endpoint identity capability negotiation is one that allows the upper level application to continue to operate.

[5.3.6](#) Identity Uniqueness and Stability

When considering the properties of long-lived identities, it is reasonable to assume that the identity assignment is not necessarily one that is permanent and unchangeable. In the case of structured identity spaces the identity value reflects a distribution hierarchy. There are a number of circumstances where a change of identity value is appropriate. For example, if an endpoint device is moved across administrative realms of this distribution hierarchy it is likely that the endpoint's identity value will be re-assigned to reflect the new realm. It is also reasonable to assume that an endpoint may have more than one identity at any point in time. [RFC 3014](#) [8] provides a rationale for such a use of multiple identities.

If an endpoint's identity can change over time, and if an endpoint can be identified by more than one identity at any single point in time, then some further characteristics of endpoint identifiers should be defined. These relate to the constancy of an endpoint identity within an application, and the question of whether a transport session relies on a single endpoint identity value, and, if so, whether an endpoint identity can be changed within a transport session, and under what conditions the old identity can continue to be used following any such change. If the endpoint identity is a long-lived reference to a remote endpoint, and if multiple identities can exist for a single unique endpoint, then the question arises as to whether applications can compare identities for equivalence, and whether it is necessary for applications to recognize the condition where different identities refer to the same endpoint. These identities may be used within applications within a single host, or may be identified being used on applications on different hosts.

[6.](#) Functional Decomposition of Multi-Homing Approaches

The following sections provide a framework for the characterization of multi-homing approaches through a decomposition of the functions associated with session establishment, maintenance and completion in the context of a multi-homed environment.

[6.1](#) Establishing Session State

What form of token is passed to the transport layer from the upper level protocol element as an identification of the local protocol stack?

What form of token is passed to the transport layer from the upper level protocol element as an identification of the remote session target?

What form of token is used by the upper level protocol element as a self-identification mechanism for use within the application payload?

Does the identity protocol element need to create a mapping from the upper level protocol's local and remote identity tokens into an identity token that identifies the session? If so, then is this translation performed before or after the initial session packet exchange handshake?

How does the session initiator establish that the remote end of the session can support the multi-homing capabilities in its

protocol stack? If not, does the multi-homing capable protocol element report a session establishment failure to the upper level protocol, or silently fall back to a non-multi-homed protocol operation?

How do the endpoints discover the locator set available for each other endpoint (locator discovery)?

What mechanisms are used to perform locator selection at each end for the local selection of source and destination locators?

What form of mechanism is used to ensure that the selected site exit path matches the selected packet source locator?

[6.2](#) Rehoming Triggers

What triggers are used to identify that a switch of locators is desirable?

Are the triggers based on the end-to-end transport session and/or [http://bgp.potaroo.net/ietf/idref/](http://bgp.potaroo.net/ietf/idref/draft-ietf-multi6-things-to-think-abou)

[draft-ietf-multi6-things-to-think-abou](http://bgp.potaroo.net/ietf/idref/draft-ietf-multi6-things-to-think-abou) on notification of state changes within the network path from the network?

What triggers can be used to indicate the direction of the failed path in order to trigger the appropriate locator repair function?

[6.3](#) Rehoming Locator Pair Selection

What parameters are used to determine the selection of a locator to use to reference the local endpoint?

If the remote endpoint is multi-homed, what parameters are used to determine the selection of a locator to use to reference the <http://bgp.potaroo.net/ietf/idref/draft-ietf-multi6-things-to-think-abou> remote endpoint?

Must a change of an egress site exit router be accompanied by a change in source and / or destination locators?

How can new locators be added to the locator pool of an existing session?

[6.4](#) Locator Change

What are the preconditions that are necessary for a locator change?

How can the locator change be confirmed by both ends?

What interactions are necessary for synchronization of locator

change and transport session behavior?

6.5 Removal of Session State

How is identity / locator binding state removal synchronized with session closure?

What binding information is cached for possible future use?

7. 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.

Huston

Expires April 15, 2005

[Page 32]

Internet-Draft

Multi6 Architecture

October 2004

It is not proposed to enumerate these considerations in detail within this document, but to reference a distinct document that describes the security considerations of this domain [6].

8. Acknowledgements

The author acknowledges the assistance from the following reviewers: Brian Carpenter, Kurtis Lundqvist, Erik Nordmark, Iljitsch van Beijnum, Marcelo Bagnulo and Joe Touch.

9 Informative References

- [1] Abley, J., Black, B. and V. Gill, "Goals for IPv6 Site-Multihoming Architectures", [RFC 3582](#), August 2003.
- [2] Lear, E., "Things MULTI6 Developers should think about", Work in progress: Internet Drafts [draft-ietf-multi6-things-to-think-about-00.txt](#), June 2004, <http://bgp.potaroo.net/ietf/idref/draft-ietf-multi6-things-to-think-about/>.
- [3] Johnson, D., Perkins, C. and J. Arkko, "Mobility Support in IPv6", [RFC 3775](#), June 2004.
- [4] Nikander, P., Arkko, J., Aura, T., Montenegro, G. and E. Nordmark, "Mobile IP version 6 Route Optimization Security Design Background", Work in progress: Internet Drafts [draft-ietf-mip6-ro-sec-01.txt](#)<http://bgp.potaroo.net/ietf/idref/>

[draft-ietf-multi6-things-to-think-abou](#), July 2004,
<<http://bgp.potaroo.net/ietf/idref/draft-ietf-mip6-ro-sec/>>.

- [5] Saltzer, J., Reed, D. and D. Clark, "End-to-End Arguments in System Design", ACM TOCS Vol 2, Number 4, pp 277-288, November 1984,
<<http://web.mit.edu/Saltzer/www/publications/endtoend/endtoend.txt>>.
- [6] Nordmark, E. and T. Li, "Threats relating to IPv6 multi-homing solutions", Work in progress: Internet Drafts
[draft-ietf-multi6-multihoming-threats-01.txt](#), July 2004, <<http://bgp.potaroo.net/ietf/idref/draft-ietf-multi6-multihoming-threats/>>.
- [7] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", [RFC 3484](#), February 2003.
- [8] Narten, T. and R. Draves, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", [RFC 3041](#), January 2001.

Huston

Expires April 15, 2005

[Page 33]

Internet-Draft

Multi6 Architecture

October 2004

Author's Address

Geoff Huston
APNIC

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of

such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2004). This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

Acknowledgment

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