Things MULTI6 Developers should think about

draft-lear-multi6-things-to-think-about-03

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

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 November 22, 2004.

Copyright Notice

Copyright (C) The Internet Society (2004). All Rights Reserved.

Abstract

This document specifies a set of questions that authors should be prepared to answer as part of a solution to multihoming with IPv6. The questions do not assume that multihoming is the only problem of interest, nor do they demand a more general solution either.
1. Introduction

At the time of this writing there are some six separate solutions looking at the problem of multihoming within IPv6 and related problems, such as the locator/identifier split.

In order to sort through how proposed solutions compare against one another, and potentially, how they can borrow mechanisms and design decisions from one another, this document contains a list of pointed questions.

The purpose of these questions is to focus people who propose solutions on practical operational problems that both single-homed and multi-homed deployments may face.

Unless it is blatantly obvious, each question contains some reasoning as to why it is being asked. It is envisioned that no solution will answer every question with completeness, but that there will be tradeoffs to be made. The answers by the various designers of solutions will hopefully shed some light on which tradeoffs we as a community wish to make.

It would seem silly for people who have written out detailed answers to these questions to have to repeat the exercise. Therefore, a simple reference to existing documents will suffice, so long as the answer is complete. If it is not complete, then feel free to reference it and add what text is necessary to make the answer complete.

This document presumes a familiarity with RFC 3582 [2], and does not attempt to repeat the requirements work gathered there.

1.1 Differences between -02 and -03

Added questions relating to whether existing communications can be piggybacked. Added questions about how the mechanism aggregates if site-wide, or how information is gotten by parties if per-host. Several other additions.

1.2 Differences between -01 and -02

Security Considerations and a touchup in some of the words.

1.3 Differences between -00 and -01

In section 2.1.2.1, address whether rendezvous SHOULD be handled in a solution.
In section 2.2.3, clarified wording.

New subsection 2.4.3 on what if you do not use DNS

New subsections 2.3.4, 2.4.7, 2.4.8, 2.4.9, 2.4.14

Clarifications on 2.4.15, and perhaps elsewhere.
2. The Questions

2.1 Routing

2.1.1 How will your solution solve the multihoming problem?

That’s why we’re here. Remember, a reference is fine.

2.1.2 Uniqueness

2.1.2.1 Does your solution address mobility?

If so, how are rendezvous handled? Can your solution handle both locators changing at the same time? If so, please explain. Should it? If not, how will your solution interact with MOBILEIP-V6 [3] (MIPv6)

2.2 Identifiers and locators

2.2.1 Does your solution provide for a split between identifiers and locators?

2.2.2 What is the lifetime of a binding from an identifier to a locator?

2.2.3 How is the binding updated?

Will transport connections remain up when new paths become available or when old ones become unavailable? How does the end node discover these events?

2.3 On The Wire

2.3.1 At what layer is your solution applied, and how?

Is it applied in every packet? If so, what fields are used?

2.3.2 Why is the layer you chose the correct one?

Each layer has its benefits and tradeoffs. For instance, transport layer solutions would require that EVERY transport be modified, while IP layer solutions may entail expansion of the packet or a change to the pseudo-header (thus requiring changes to the transport layer).

2.3.3 Can your solution be aggregated and implemented site-wide?

Some mechanisms may be implemented per host. Others may be implemented at the site level. If yours is implemented site-wide does it aggregate well? If your solution is implemented at the host
level, how is necessary information gotten to the relevant parties of the communication?

2.3.4 Does your solution impact existing traffic engineering methods, such as MPLS-TE?

Traffic engineering allows for source-based traffic aggregation to a particular destination. How will your mechanism interact with such existing methods?

2.3.5 Does your solution expand the size of an IP packet?

Expanding the size of an IP packet may cause excessive fragmentation in some circumstances.

2.3.6 Will your solution add additional latency?

Latency is an important factor in many applications, including voice. Any substantial amount of additional latency, including session initiation would be highly undesirable.

2.3.7 Can multihoming capabilities be negotiated end to end during a connection?

If the proposal introduces additional overhead, can the information be somehow piggybacked on messages that are already used? This would be useful in order to keep connection setup constant. Please also indicate any drawbacks that might apply due to this piggybacking.

2.3.8 Do you change the way fragmenting is handled?

If you use a shim approach, do you fragment above or below the shim? How are fragments identified, so that they can be reassembled? If you use any additional names, do they need to be associated with fragments? If not, why not? If so, how will that happen?

2.3.9 Are there any changes to ICMP error semantics?

Do you create new codes? If so, why and what do they mean? Will a host that is not aware of your scheme see them?

2.4 Names, Hosts, Endpoints, or none of the above?

2.4.1 Please explain the relationship of your solution to DNS

If your solution uses new names for identifiers, please explain what mappings are defined, and how they are performed.
If there are any additional administrative requirements, such as new zones or RR types to manage, please explain them as well.

2.4.2 If you have separate locators and identifiers, how will they be mapped?

Does the mapping work in both directions? How would someone debugging a network determine which end stations are involved?

2.4.3 If you are not using DNS...

Please describe the mechanism you are using.

2.4.3.1 Please describe authentication/authorization

How are bindings authenticated and authorized. What technology do you build on for this mechanism?

2.4.3.2 Is your mechanism hierarchical?

Please describe the hierarchical breakdown.

2.4.4 Please explain interactions with "2-faced" DNS

2-faced DNS is used so that hosts behind a NAT get one address for internal hosts, while hosts outside the NAT get another. Similar mechanisms are used for application layer gateways, such as SOCKS [5].

2.4.5 Does your solution require centralized registration?

For instance, if you are using the DNS, what will be the top level domain, and how will the name space distribute through it?

Also, how will the centralized registration be managed?

2.4.6 Have you checked for DNS circular dependencies?

If you are using the DNS in your solution, is it required for connectivity? What happens if the DNS fails? Can communication between the DNS resolver and the server make use of your solution? What about between the application and the resolver?

2.4.7 How does a host know its identity?

If you are establishing a new identity, how does the host learn it?
2.4.8 What if a DNS server itself is multihomed?

If a link fails or a service is dropped, how will it impact DNS? Again are there any dependency loops? Perhaps diagram out your dependencies to make sure.

2.4.9 What additional load will be placed on DNS servers?

Can the load be distributed? Remember that DNS is optimized for READ operations.

2.4.10 Any upstream provider support required?

If so, please describe.

2.4.11 What application/API changes are needed?

Will old code just work with the new mechanism? For instance, what about code that uses gethostbyname()? Will getaddrinfo() need to change? What about other API calls?

2.4.12 Is this solution backward compatible with "old" IP version 6?

Can it be deployed incrementally? Please describe how. Does your solution impose requirements on non-multihomed/non-mobile hosts? What happens if someone plugs in a normal IPv6 node?

2.4.13 Is your solution backward compatible with IPv4?

How will your mechanism interact with 6to4 gateways and IPv4 hosts?

2.4.14 Can IPv4 devices take advantage of this solution?

Can the same mechanism somehow be used on the existing network? N.B. this is NOT a requirement, but more a consideration.

2.4.15 What is the impact of your solution on different types of sites?

How are single homed sites impacted?

How are small multihomed sites impacted?
How does it scale for large multihomed sites?

What about ad-hoc sites such as an IETF event?

2.4.16 How will your solution interact with other middleboxes?

What are the implications for firewalls? What are the interactions with NAT? What are the interactions with web caches? What complications are introduced with your solution? For instance, are there implication for ingress filters? If so, what are they?

2.4.17 Are there any implications for scoped addressing?

Please see RFC 3513 [1]. How does your mechanism interact with multicast?

How does your solution interact with link-local addressing

How does your solution interact with Son-Of-Site local (whatever that will be)?

2.4.18 Are there any layer 2 implications to your proposal?

While IPv6 has a simplified approach to layer 2, perhaps you unsimplified it. If so, please provide details.

2.4.19 Referrals

How will your solution handle referrals, such as those within FTP?

It must be possible for existing applications to continue to work. Referrals exist within various other protocols, such as so-called "peer to peer" applications.

2.4.20 What new information should applications be aware of?

If there are new bindings, what does the application need to take advantage of them? How does the application present a "connection"?

2.5 Legal Stuff

Are you introducing a namespace that might involve mnemonics? Doing so might introduce trademark concerns. If so, how do you plan to address such concerns?

Are there any organizations required to manage a new name space? If so, please describe what they are and how the method will scale.
3. Security Considerations

How secure should a multi6 solution be? This is a reasonable question for each solution to answer. The author opines that the worst case should be no worse than what we have today. For example, would a multi6 solution open up a host on either end of a communication to a time-based attack? Any such risks should be clearly stated by the authors. Considerable time should be spent on threat analysis. Please see [4] for more details.
4. Acknowledgments

The author wishes to acknowledge everyone in the multi6 group and elsewhere that is putting forward proposals. It is easy to ask questions like the ones found in this draft. It is quite a bit harder to develop running code to answer them. Marcelo Bagnulo, Kurt Erik Lindqvist, Patrik Faltstrom, Brian Carpenter, and Iljitsch van Beijnim provided input to this document.
Normative References


Informative References


Author’s Address

Eliot Lear
Cisco Systems
170 W. Tasman Dr.
San Jose, CA  95134-1706
US

EMail: lear@cisco.com
Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any intellectual property 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; neither does it represent that it has made any effort to identify any such rights. Information on the IETF’s procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication 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 implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

Full Copyright Statement

Copyright (C) The Internet Society (2004). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assignees.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS 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.

Acknowledgment

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