Network Working Group
Internet-Draft
Expires: June 10, 2004

E. Lear
Cisco Systems

December 11, 2003

Things MULTI6 Developers should think about draft-lear-multi6-things-to-think-about-00

Status of this Memo

Copyright Notice

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.

This document contains only some useful questions. There are others that should be added. If you know of one, please email the author, as he has assuredly missed many.

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.

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? 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?

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 Does your solution expand the size of an IP packet?

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

2.3.4 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?

2.3.5 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 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].

2.4.3 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?

2.4.4 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?

2.4.5 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.

2.4.6 What application/API changes are needed?

Will old code just work with the new mechanism?

2.4.7 Is this solution backward compatable with "old" IP version 6?

Can it be deployed incrementally? Please describe how.

Does your solution impose requirements on non-multihomed/non-mobile hosts?

2.4.8 Is your solution backward compatable with IPv4?

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

2.4.9 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?

2.4.10 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-Sitelocal (whatever that will be)?

2.4.11 Are there any layer 2 implications to your proposal?

While Ipv6 has a simplified approach to layer 2, perhaps you unsimplifiied it. If so, please provide details.

2.4.12 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.

2.5 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. However, any additional 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 acknoledge 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.

Normative References

[1] Hinden, R. and S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture", RFC 3513, April 2003.

[2] Abley, J., Black, B. and V. Gill, "Goals for IPv6 Site-Multihoming Architectures", RFC 3582, August 2003.

[3] Johnson, D., Perkins, C. and J. Arkko, "Mobility Support in IPv6", draft-ietf-mobileip-ipv6-24 (work in progress), July 2003.

[4] Nordmark, E. and T. Li, "Threats relating to IPv6 multihoming solutions", draft-nordmark-multi6-threats-00 (work in progress), October 2003.

Informative References

[5] Kitamura, H., "A SOCKS-based IPv6/IPv4 Gateway Mechanism", RFC 3089, April 2001.

Author's Address

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

EMail: lear@cisco.com

Intellectual Property Statement

Full Copyright Statement

Acknowledgment

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