Network Working Group
Internet Draft
draft-crocker-multiaddr-analysis-01.txt
12dc.doc
Expires: <4-04>

D. Crocker
Brandenburg
InternetWorking

October 19, 2003

CHOICES FOR MULTIADDRESSESING

STATUS OF THIS MEMO

COPYRIGHT NOTICE

ABSTRACT

An IP Address serves the dual roles as references to a "place" on the Internet and to a host on the Internet, labeled "locator" and "identifier", respectively. Systems that use IP Addresses as identifiers cannot support dynamic changes in the mapping between the identifier and the locator. For a system to use a different IP Address pair, participants must initiate a new exchange. In the case of TCP, this means a new connection. In recent years, there have been efforts to overcome this limitation, through different approaches at different places in the Internet architecture. This paper reviews the basic requirements for

support of multiaddressing (mobility and multihoming), and the efforts to support them. Barriers to adoption, administrative overhead, and operational efficiency are of particular concern.

CONTENTS

1. INTRODUCTION

"We need a way for sites to be internally stable even when their relationship to the world around them

changes for whatever reason."

-- E. Lear

An IP Address serves as references to a "place" on the Internet and to a host on the Internet. These two roles are generally labeled "locator" and "identifier", respectively. Systems that use IP Addresses as identifiers typically cannot support dynamic changes in the mapping between the identifier and the locator. For example, TCP includes a single source/destination IP Address pair in its definition of a connection. Hence its transport association is tied to that pair. This is problematic for hosts that are multihomed or mobile. Both have access to multiple IP Addresses, but they are prevented from using more than one within an existing context, because the context is named by that pair. For a system to use a different IP Address pair, participants must initiate a new exchange. In the case of TCP, this means a new connection.

In recent years, there have been efforts to overcome this limitation, through different approaches at different places in the Internet architecture. Some approaches modify the IP infrastructure, with embedded redirection services. Some define transport enhancements to support a set of locators directly, and some define a layer between classic IP and classic transport. The primary goal of these multiaddressing efforts is to preserve established connections when an IP Address changes. Each of the existing proposals has notable limitations in functionality, implementation, deployment or use.

This paper reviews the basic requirements for support of multiaddressing (mobility and multihoming), and the efforts to support them. Barriers to adoption, administrative overhead, and operational efficiency are of particular concern. In addition, the analysis considers enhanced functionality that is possible from the use of multiaddressing, such as performance-based load-sharing, across the different locators available to a multihomed host.

1.1. Scenarios

What are the situations and concerns that affect design and use of a mechanism for the support of multiaddressing?

> Section 3 of [MOBMH], has an excellent discussion of these issues.
> It is included here by reference without section 3.2.
> Section 3.2 covers an interesting topic that appears to be independent of multiaddressing.
> The included text comprises the following sub-sections:

1.2. IETF Background

Historically, IETF focus on mobility has split between initial attachment configurations, into an otherwise static environment such as by using DHCP, versus forwarding mechanisms, such as by modifying the IP infrastructure with Mobile IP. Multihoming has largely been ignored, except in routing protocol work. Recent efforts are pursuing direct enhancements to transport or insertion of a mapping layer between IP and transport. There has also been adjunct activity, relevant to this topic.

The following summary of IETF activities draws on text from the Abstracts of documents for those activities. In addition, there is a useful analysis of the different architectural and protocol efforts is in Section 3, "Internet Stack Placement" in [NSRG]. Specification efforts are discussed in more detail in Section

> The Name Space Research Group [NSRG] considered modifications to the Internet architecture, including whether an additional level of naming is needed, above layer 3 but below the application layer. Purpose-Built Keys [PBK] specifies a template for the use of specially generated public/private key pairs, to provide assurance that successive messages in the communication come from the same source. This is accomplished without the use of external certification authorities. Hence it ensures authentic continuity during a session, but does not provide "global" or "absolute" authentication.
>
> Stream Control Transmission Protocol [SCTP] is a reliable transport protocol for multiplexed data streams. It includes modern mechanisms for safe initiation of a connection, as well as the necessary tools for reliability and congestion control. It also has a mechanism for communication access to multiple IP Addresses between the participation host pair. [TCP-MH] uses TCP options to support multihoming. Datagram Congestion Control Protocol [DCCP] is a proposal for a network-friendly, unreliable transport-level datagram delivery service.

Mobile IP work has divided between IPv4 and IPv6. [MIP4] and [MIP6] allow a node to continue using its "permanent" home address as it moves around the internet.

Host Identity Protocol [HIP] is used to establish a rapid authentication between two hosts and to provide continuity of communications between those hosts independent of the networking layer. The [LIN6] protocol defines a layer that supports multiple locators, between IPv6 and transport. Multiple Address Service for Transport [MAST] supports association of multiple IP Addresses during the life of any transport instantiation, by defining a layer between IP and transport. It operates only in the endpoints and works with IPv4 and IPv6.

1.3. Discussion Venue

Discussion and commentary are encouraged about the topics presented in this document. The preferred forum is the <mailto:multi6@ops.ietf.org> mailing list, for which archives and subscription information are available at <http://ietf.org/html.charters/multi6-charter.html>.

NOTE: The early drafts of a review document, like this, are certain to have significant errors. The author strongly requests guidance for clarifying and correcting any problematic text.

In particular, those working on the proposals and specifications discussed here are encouraged to provide corrections and additional text, to ensure accuracy.

1.4. Document History

-00 Derived from draft-crocker-mast-proposal-00. Extended discussions about alternative proposals and architectural issues, separated from the -proposal- draft.

-01 Substantial revisions to all sections. More complete review of efforts. More extensive terminology definitions. Section 3 renamed to "Considerations". Material that evaluates proposals is moved out of it, to the next section.

Later versions need cleaner separation of topics,

such as requirements and definition of what multi-addressing support really means in different situations.

Need to add a chart that compares the proposals.

Need to incorporate the remainder of Marcelo's suggested changes.

Need to discuss enhancements made possible by multiaddressing support.

NOTE: D. Crocker has put forward the MAST proposal. That may have colored the perspectives in this discussion paper.

2. TERMINOLOGY

This paper discusses requirements and methods for enabling an endpoint to use multiple locators during single application associations. This topic does not yet have a stable, core set of terms in general use. The following definitions are intended to remedy that deficiency; they are taken from existing definitions, when available. Work on multiaddress enhances existing infrastructure capabilities. This work is uncovering ambiguities to terms that have been used. For multiaddressing, it is therefore confusing to use some common terms, notably "address". Hence they SHOULD NOT be used.

2.1. Recommended

Agent — refers to a third-party that is handling something on behalf of one or more other parties. The term indicates the separateness of the entity, as well as its key relationship to the other entities. In multiaddressing, it refers to an intermediary service that represents an endpoint, for the purposes of referral and/or relaying.

Association — refers to an established communication context between endpoints, such as a TCP connection.

Endpoint — refers to "the fundamental agent of end-end communication" [EID]. It is an end-system that participates in an association.

	Endpoints are distinguished from intermediate, infrastructure nodes and hosts.
Identifier	refers to a unique label for an endpoint. The label is used simply for distinguishing one endpoint from another. Because a locator is usually globally unique, it might be able to serve as an identifier. However this use will often suffer administrative and referential limitations as a global identifier for mobile endpoints. This is exemplified by the current problems experienced with the dual role of IP Addresses.
Initiator	refers to an endpoint that initiates contact with a target endpoint. In client/server architecture it is the client.
IP Address	specifies a topological network access point. The term is usually considered to specify an endpoint interface. However discussions about mobility are notably clarified by viewing the value as belonging to the network (interface) rather than to the endpoint.
Locator	refers to a "the name of a network attachment point" [SALT], usually in terms of the network's topology. Locators typically facilitate mapping into routes, such as by indicating a topological hierarchy. IP Addresses specify a topological network access point. They usually are considered to specify an endpoint interface. However discussions about mobility are enhanced by viewing the value as belonging to the network (interface) rather than to the endpoint.
Mobility	refers to an endpoint's having different locators over time. This may even include discontinuities, during which an endpoint has no valid locators. In addition, the nature of a transition from one locator to another may include overlapping availability of locators. Interestingly, this looks the same as multihoming. Mobility may be for a single endpoint or for the subnetwork to which the endpoint is attached. In the latter case, the endpoint connection is stable, with respect to its sub-network, but the sub-network propogates connectivity change information to the endpoint.

Multiaddressing	refers to an endpoint's having more than one locator available, over the lifetime of an association. It encompasses both multihoming and mobility. The core requirement for multiaddressing is preservation of established communications, across the use of different locators.
Multihoming	refers to the availability of multiple endpoint locators at the same endpoint, simultaneously. It is typically used to refer to multiple network attachments for a host, but works equally well for multiple upstream network attachments by the local network, when the different upstream locators are visible to the host. Interestingly, multihomed environments often must support dynamic changes, such as when adding a new upstream provider. Therefore, multihoming can include mobility features and mobility can include multihoming features. When needing to renumber a network, due to changes in up-stream service, the process can be operated as dynamic multihoming.
Path discovery	provides a sender with the means for learning about the locators from which they can send.
Path selection	is required when more than one locator is available to the sender. Although the sender is limited to specifying an locator, rather than a path, it appears that thinking of it as path selection aids consideration of solutions. In effect, it formulates the selection task as being similar to the job of routers. Route formulation is mature technology, so that this aspect of multiaddress processing will be tractable, if not straightforward.
Referral	permits an initiator to obtain a locator for a target, such as a client being referred to a server. A third-party process is required for referral, in the absence of an association. For existing associations, participating endpoints might be able to supply their own referrals. The primary Internet mechanism for referral has been the Domain Name Service (DNS). The DNS uses long, variable-length strings (names) and is

	tailored for large-scale referral with identifiers and locators (mappings) that change infrequently.
Referral Agent	refers to the function that maintains the mapping between a mobile node's identifier and its locator(s). [LIN6] calls this a Mapping Agent.
Rehoming	refers to an endpoint's changing an association from one locator to another,
Relaying	refers to the redirection of packets, on behalf of an endpoint. Other endpoints see a stable locator for the endpoint obtaining the relaying service.
Relaying Agent	refers to an agent that performs packet forwarding (relaying) on behalf of an endpoint. The Relaying Agent thereby persents a stable locator to the Internet, for the endpoint. For mobility, the agent resides on an endpoint's "home" network and relays datagrams to the endpoint's actual location on the Internet. The endpoint is modified to support this forwarding technique.
Rendezvous	refers to one endpoint making contact with an explicitly identified other endpoint.
Target	refers to an endpoint that receives contact from an Initiator endpoint. In a client/server architecture, this is the server.

2.2. Deprecated

Address	Refers to "the name of some network attachment point." [SALT] The term has become a problematic because addresses often are used for two, distinct functions. Hence the term should not be used by itself, for these discussions, except with reference to particular protocol specifications, such as "IP Address". Instead, use "identifer" and "locator", as appropriate.
Connection	A state of association between two endpoints.

Because the term is typically used for to refer to transport-layer state, discussions about multiaddressing should use the more general term "association", except with reference to particular protocol specification, such as "TCP Connection".

3. CONSIDERATIONS

The core requirement for multiaddressing is continuity of access within an association. However applications having this as a compelling requirement have not yet been evident on the Internet. Hence there ius some risk that proposed mechanisms to solve the requirement will not correctly ancitipate the details of the requirement.

This section is a general discussion of requirements, constraints and concerns. It does not attempt to offer a formal set of requirements or recommendations.

3.1. Mobility

Mobility is time-varying access to multiple locators for the same endpoint. Key parameters to mobility are scope of change, rate of change and source(s) of the change. Over what portion of the Internet topology might a change take place; how often will changes occur; and which of the participants will change their locators?

3.1.1. Rapid, Local Initiators

It is generally accepted that rapid, local changes should be handled by a layer below IP. These changes are therefore invisible to IP and above, so that associations are automatically preserved across change.

3.1.2. Rare, Distant Initiators

For initiator endpoints that are subject to occasional detachment and eventual reconnection, the current set of technologies is probably sufficient. These require reconfiguration, such as through DHCP, and establishing new associations. Applications wishing to retain association state, across these transitions, do so above the transport layer. They are capable of establishing new transport associations, as needed.

3.1.3. Periodic, Moderate

What is missing from the operational Internet is support for initiator and target systems that move over the course of minutes or hours and need to maintain existing transport associations or need to maintain their availability for new associations.

The difference between mobility prior to initial contact and mobility during an existing association is significant. In the latter case, the mobile host can use the association state when needing to inform the other endpoint about the change. Prior to an association -- or when both endpoints are mutually mobile -- an independent referral service is required.

The difference between initiator mobility and target mobility is also significant, with respect to initial contact. In particular the initiator needs to be able to obtain a valid locator for the target. Again, this requires a referral mechanism, such as having the routing system map from identifiers to routes, rather than locators to routes. Either it must be provided implicitly within the network or there must be an external "referral" mechanism. For static servers, the DNS already provides this referral quite well. However current DNS use does not support frequent locator changes over short periods. Hence enhancements are needed to support referral with a mobile target.

3.2. Multihoming

The Internet already supports a number of types of "indirect" multihoming.

3.2.1. Infrastructure

The core of dynamic packet-switched routing entails exploitation of alternative routes, so that the path between endpoints might vary considerable over the course of an association.

3.2.2. Site

For networks with multiple attachments to a backbone, external routing technology already permits propagation of alternate routing information. However it does not make these alternatives visible to endpoints.

Further a domain name may have multiple locator records that point to the same network. However there is no indication whether the same records are, instead, pointing to different, redundant systems; on the other hand the importance of this ambiguity is

not clear.

3.2.3. Endpoint

What is notably missing is a means for an existing association to directly use multiple paths, in particular when the paths terminate at one of the endpoints. Here, the fact that classic Internet transport services rely on single, specific IP Addresses is the barrier.

Endpoint support of multihoming can be useful for robustness and throughput. The former makes loss of a path transparent to the association. The latter increases the effective bandwidth for an association. In general, the former goal is dominating current work. At the least, using multiple paths for increased bandwidth ensures a high degree of out-of-order arrivals. This usually reduces target endpoint performance, rather than increasing it.

3.3. Security

The level of security built into IP is minimal. Some would say it is non-existent. However, classic transport services rely on having a significant degree of correlation between the IP Address in the source field of an IP datagram and the likelihood that the IP datagram came from that locator. The context of repeated exchanges between source and destination locators is taken as a validation of this correlation. Permitting the IP Address of a source to vary during an association is an invitation to connection hijacking, and related attacks. Hence, any support for multiple locators within an association must contain a strong anti-hijacking mechanism.

All other security concerns are independent of multiaddressing. They may already be handled by additional mechanisms, such as [IPSec] and [TLS]. There is no indication that any of these other mechanisms need to be changed, to support multiaddressing. Once there is an effort to design protection against hijacking, it is easy to consider adding more protections, such as privacy or, perhaps, other kinds of authentication. Although such mechanisms obviously would be useful, they are not essential to the basic requirements of multiaddressing. Further, they might be redundant with mechanisms provided elsewhere in the architecture.

Any effort related to multiaddress support, which goes beyond preventing hijacking, needs to have explicit discussion about its relationship to other security mechanisms and the need for attaching these additional capabilities to multiaddress support. As with any opportunity for adding features to a design effort,

there should be concern about causing unnecessary design complexity, delays to the specification effort, and difficulty in implementation.

3.4. Implementation

The software that supports IP and classic transport services is mature. Usually it is highly tuned and highly robust. Often it is also complex. Hence it can be risky to introduce modifications to one or more of these modules. On the other hand, attempting to introduce multiaddress support through additional modules runs the risk of being awkward and inefficient.

3.5. Deployment and Use

However difficult it is to have vendors make major modifications to mature software, it is far more difficult to deploy the changes to a global, installed base of hundreds of millions of platforms. Changes to support multiaddressing need to consider barriers to adoption by users and operators, both ISPs and enterprises.

What is the effort needed to deploy the changes? What is the effort needed to use it? How broad must the adoption be before users can obtain benefit? What dependencies do the changes have on existing or new services?

Making one new service depend upon the reliable performance of another new service greatly increases the riskiness of the effort. Making a change require modification to the Internet's infrastructure typically creates a long delay before it is useful. In particular, early adopters of the mechanism gain no immediate benefit from their efforts; this acts as a disincentive for adoption. Everyone waits for others to take the first step.

3.6. Matters of State

Support for multiple locators requires adding a conceptual layer of referential indirection. Beyond simple use of the DNS, endpoints currently use individual endpoint locators within an association. In order to use multiple locators, to refer to the same endpoint, some type of aggregation and mapping mechanism must be added. The mechanism defines a relationship between the referenced endpoint and a set of locators. Where should this state information be placed in the Internet architecture?

If the major lesson of the Internet is scaling, the major embodiment of that lesson is to place complexity in the edges, rather than the infrastructure. Generally, this does not mean

that there is a balanced debate between the choices. Rather, there is an assumption that a change should be made to the edges rather than the infrastructure. It is made in the infrastructure only when there is a clear agreement that doing otherwise will seriously reduce the utility of the change.

This methodology can even be applied to some infrastructure changes. A change that will clearly have an infrastructure impact might be introduced incrementally, via endpoint modifications. Two major examples of this are DNS and MIME. Both were added to operational, infrastructure services -- the IP Internet and the Internet Mail service, respectively -- but were added in a fashion that made no immediate changes to existing services. Rather, edge systems independently chose to adopt the changes. Any two endpoints wishing to exploit the change, for interacting with each other, immediately benefited from the adoption. Over time, adoption became sufficiently broad-based to make the change effectively part of the infrastructure service. Although the IP network works well without the DNS, end-user utility of the Internet, without the DNS, would be nil. Similarly the ability to use attachments has become a fundamental part of the Internet Mail experience.

Addition of support for multiaddressing faces a similar type of choice. Should the change be made above the transport layer, in the transport layer, in the IP layer, or perhaps between IP and transport? How is the aggregation established and how is it maintained? Do IP (or TCP, or...) packets contain the mappings or are they maintained in the endpoints or, perhaps, in the IP infrastructure?

The answers to these questions need to be determined by their effect on barriers to adoption, operational overhead, and administrative convenience.

3.7. Endpoint Identifiers

Historically, IP Addresses have served the dual role of network interface locator and endpoint identifier (EID). Adding support for multiaddressing serves to highlight the need for splitting these two roles. IP Addresses work quite well as network interface locators. However their topological dependence makes them work poorly as identifiers, in the richer world of multiaddressing.

Does an EID need to be assigned by a registry or can it be dynamically computed? Does it need to be publicly visible across the Internet or can it be kept private to individual associations? Does it need to be used frequently, such as in every datagram, or is it needed only for specific transactions,

such as initiating or recovering an association?

It is appealing to define an EID to be publicly registered and carried in every datagram. This provides the maximum amount of decoupling from locatoring and appears to offer an especially clean modification to the transport layer interface. Transport header calculation "merely" needs to switch to use of the EID, rather than the locator. With sufficiently strong protection against hijacking, this approach probably will make the locator irrelevant to the transport layer.

However there still must be a mapping between EID and locators, so the IP service knows where to send the datagram. Hence, the state information of an EID/locator "routing table" must reside somewhere. Unless the IP infrastructure is modified to support EIDs directly, this state information is most probably in the endpoints.

Having a public EID means that a new, global registration service must be developed and operated. Some believe network operators will not mind this additional work; others disagree.

Having an EID in every datagram means that the string must be as short as possible. Even then it will add significant overhead to datagram header size. However given the need to process multiaddressing, having the EID in every datagram probably will not alter datagram processing overhead, in the endpoints, from any other approach to using EIDs.

If an EID is used only occasionally, one candidate is a domain name. Domain names already have an administrative structure, and they are well engrained into Internet use. Their length is not a problem, when they are need only periodically. One objection to using domain names is that they are already used in a number of ways that do not suit the role of EID. It is unclear how the fact that domain names serve multiple roles prevents their serving the role of EID. One observation is that the ability of a domain name to map to multiple IP Addresses makes it problematic to ensure that the name will later map to the same IP Address as was initially selected. At the least, resolution of this concern requires careful specification and even more careful administration.

3.8. Signaling

How does an endpoint learn its own locators, so that it can inform another endpoint, during an association? The notable challenge is when a NAT modifies the locator an endpoint uses directly, to a different locator that is visible to the rest of the network.

How does an endpoint communicate that available set of locators to another endpoint?

DNS is currently useful for registering essentially static sets. More dynamic or tailored communication requires a signaling exchange between endpoints. This can be done through a distinct signaling protocol, such as is done with [MAST], or inline -- that is, as a sub-exchange -- within an existing protocol, such as is done with [TCP-MH].

3.9. Operation Through NATs

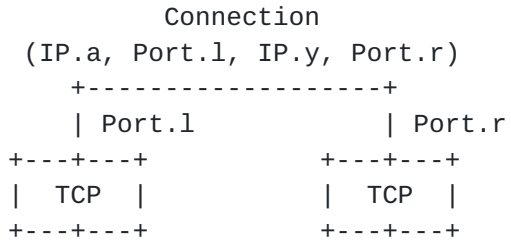
A Network Locator Translation [NAT] device maps between one set of locators, and another. In typical cases, locators from the interior of a network are mapped to different ports on a single, public locator on the outside of the network. This mapping task must be performed with knowledge of transport protocol details because it must adjust transport headers, as well as IP-level locators. Stateless NATs are likely to work with most multihoming solutions and some mobility solutions. The NAT will simply do its usual task of replacing IP Addresses and adjusting dependent headers of common transport protocols, accordingly.

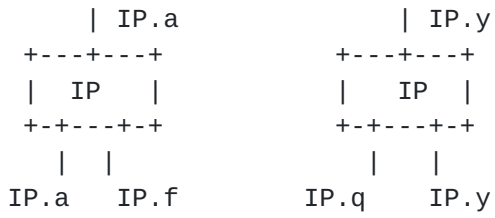
However, there is the basic question of whether a multiaddressed initiator correctly knows its own locators. Typically it will not. Given the prevalence of NATs, a solution to multiaddressing needs to deal with this scenario. Some solutions require that NATs be upgraded to support the solution. This is another example of an infrastructure dependency.

4. INTERNET STACK PLACEMENT PROPOSALS

From a purely technical standpoint, multiAddressesing can be supported through a number of different mechanisms. This section discusses the possible venues within the Internet stack, and existing efforts that are pursuing these choices.

The current architecture for transport use of IP Addresses makes a direct linkage to a specific IP Addresses pair:

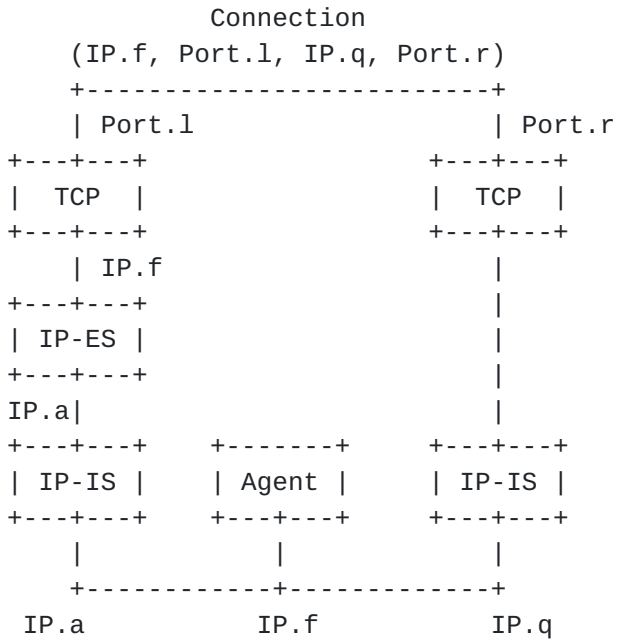




This example shows each host being multihomed. However a given association must choose a single IP Addresses, at each end, and bind the connection to it.

4.1. IP Infrastructure

In the classic Internet infrastructure model, a datagram contains topological references to the source and destination network interfaces. The network knows nothing about higher-level issues, such as whether two interfaces are attached to the same endpoint. This design derives from the explicit desire to keep the Internet infrastructure as simple as possible, by putting as much functionality as possible into the endpoints rather than in the Internet's switching devices.



4.1.1. Dynamic DNS

For maintaining target availability during an association, DNS dynamic update [DNSDYN] is a plausible mechanism to obtain new locator information. Although this would probably not be helpful for transitions during an association, it could suffice for referrals to establish initial rendezvous. However it is not widely deployed and the typical DNS record lifetime settings and

client caching behaviors suggest that existing DNS use is better tailored for changes over days, rather than hours. Separately the core role of DNS for Internet infrastructure operations suggests avoiding major changes to its operational model. Supporting potentially high volumes of rapid changes probably require very different software and administration than are used for the current DNS.

4.1.2. MIP

The Mobile IP [MIP4, MIP6] efforts provide an encapsulation-based forwarding service. A "home" relaying agent intercepts datagrams using an original destination IP Addresses, and then forwards the datagram to the target's current IP Addresses. An optimization supported in the IPv6 version permits direct transmission to the new venue, if initial use of the home agent. The encapsulation tunnels the original IP datagram inside a new one. Direct exchanges carry both locator and endpoint identifier values. [HOWIE] provides an interesting discussion of MIPv6 adoption and use issues.

A major benefit to this approach is that use of the home agent requires no direct support by the non-mobile endpoint. A multiaddressed endpoint must be modified to support this capability, but the other side of an association need not be modified. This places the requirement for change onto systems with the incentive to use it.

Conceptually, the biggest problem with this approach is that it attempts to take topology-related information -- the IP Addresses -- and use it as the basis for contacting an endpoint non-topologically. Operationally, the biggest problems with this approach are that it requires adoption by an infrastructure service, and forwarding services are inefficient and often complex.

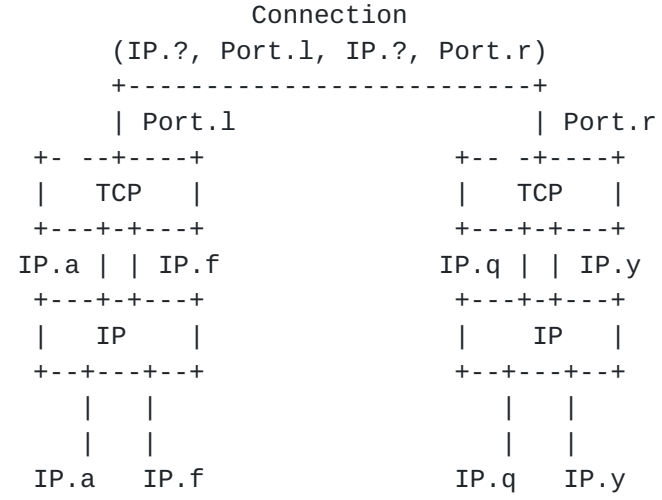
This approach changes the infrastructure and changes the IP datagram. Hence it changes multiple, different aspects of the Internet architecture, with each change potentially constituting a significant barrier to adoption, reliability or efficiency.

4.2. Transport-Level

Recent transport protocols, such as [SCTP], [TCP-MH] and the proposal for [DCCP], use multiple IP Addresess directly in the transport association. These efforts have primarily focused on multihoming, with the time-varying nature of mobility being ignored or retrofitted.TCP-MH notably uses TCP options for inline signaling of multihoming information between the endpoints; its

current specification appears to have weak protection against hijacking but this can be remedied.

A transport-level approach has the benefit of placing the necessary functionality only in end-systems and avoiding possible locators translation problems. It also has the considerable benefit of leaving the IP infrastructure unchanged. Given the complexity and robustness of that infrastructure, as well as the considerable time and effort that was needed to achieve its stability, any design that avoids changing the infrastructure is to be commended.



The fact that the functionality is applicable across all transport services suggests that there might be benefit in having IP multiAddressesing functionality reside in a single architectural module, separate from any specific transport service. In any case these new transport protocol efforts cannot affect the considerable installed base of services using older transport protocols, such as TCP and UDP.

Given that multiAddressesing is directly visible to the transport level, it is not clear how to formally define a connection. Are "virtual" locators used? Is one of the locators used? Is a separate, formal "identifier" used?

4.2.1. SCTP

Stream Control Transmission Protocol (SCTP) provides reliable delivery of multiple message streams. It supports multihoming, with the exchange of multiple locators for each SCTP association endpoint, at the start of the association. Current specification use the additional locators to provide reliability if the primary locator ceases to be useful. Enhancements are being discussed to support mobility.

Basic security for SCTP control exchange relies on the usual reliability of mapping a locator to an Internet route. An ephemeral association identifier is also used, to prevent "blind" attacks on the association.

4.2.2. TCP-MH

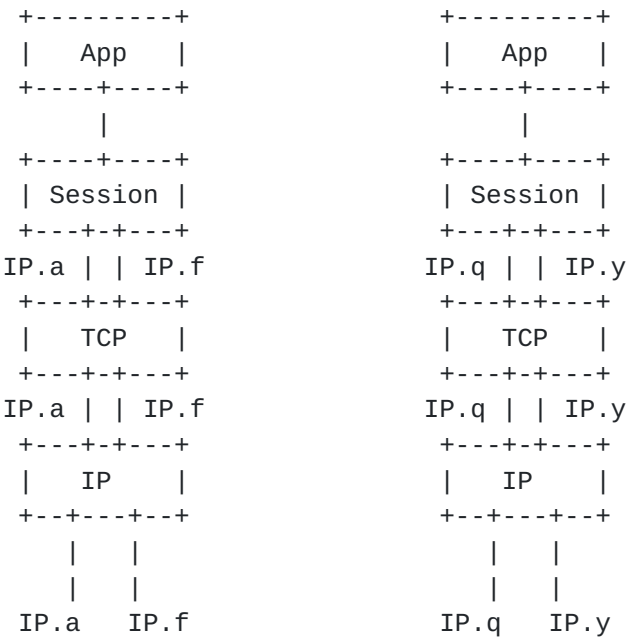
TCP-MH uses a TCP option to support multihoming, by specifying multiple IPv4 and/or IPv6 locator pairs that can be used during the same TCP connection. A simple transaction serial number is used to prevent hijacking. TCP-MH manages its pool of locators with individual locator add/delete operations. This style of exchange can lose synchronization between the copies of locator lists maintained by the two endpoints.

4.2.3. DCCP

Datagram Congestion Control Protocol (DCCP) establishes an transport connection for the unreliable delivery of datagrams. It has an option to support multihoming. An ephemeral identifier is used to prevent hijacking.

4.3. Session-Level

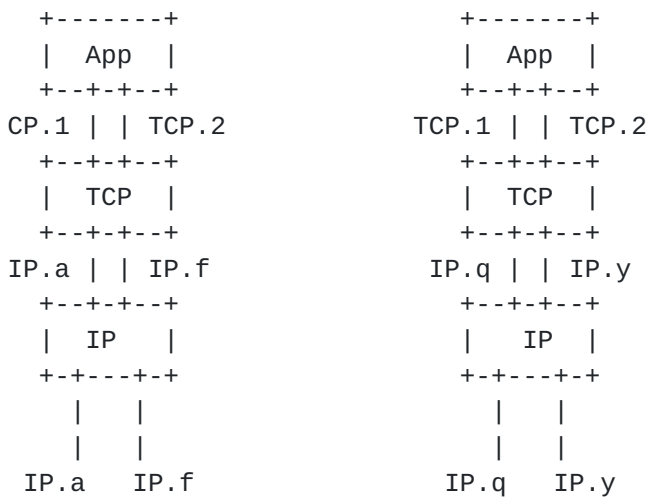
The session layer provides functionality above transport and below the application. In effect it is a way of institutionalizing application-level support. The merit of placing multiAddressesing support at the session layer is that it can use multiple transport services.



The problem with this approach is that a full session layer typically replicates substantial portions of the transport service, in order to ensure reliability and in-order data sequencing. This makes the session-level approach notably complicated and inefficient.

4.4. Application-Level

Applications often provide themselves with enhanced infrastructure support services, to compensate for limitations in the lower protocol, or to optimize functionality and performance according to the peculiarities of the specific application. A typical example is with reliable data transfer, when using an unreliable datagram service. The obvious difficulty with this approach is that it burdens each new application with re-creating these (common) underlying services.



There well might be some benefit in permitting applications to discover details about multiaddressing capabilities, and possibly in permitting them to specify some details of their use, through an enhanced API. However the prevalence of multiAddressesing dictates its support in lower layers.

4.5. IP Endpoint

A recent approach to multiAddressesing defines a new "convergence" layer that exists only in the endpoint systems (hosts) and operates between classic IP and the transport layer. Hence these capabilities are invisible to the IP relaying infrastructure and can be invisible to the transport layer. However they may specify new or modified adjunct infrastructure services, especially to obtain full rendezvous capabilities.

This type of approach can be viewed as using a "shim" or "wedge"

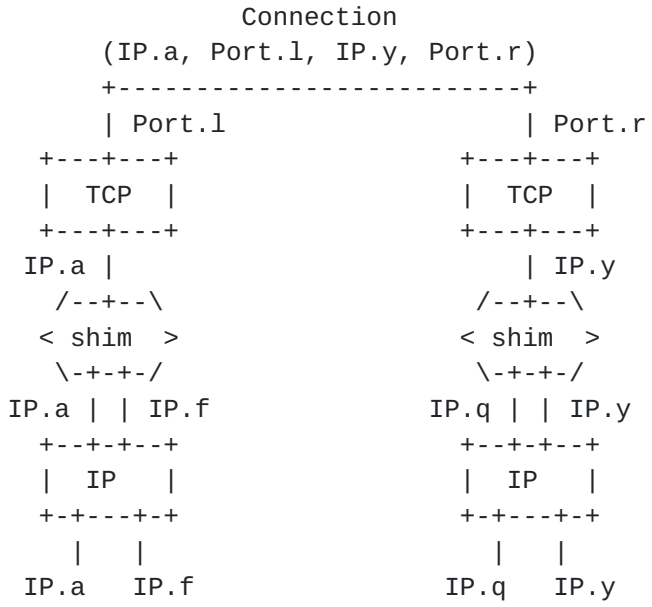
partial-layer, between IP and transport, or it can be viewed as partitioning IP, between a lower, relaying module that is common to all IP nodes, versus an upper module that performs IP-related functions specific to endpoints.

The remainder of this sub-section considers these architectural views and then discusses the three IP Endpoint proposals.

4.5.1. Choosing an IP Endpoint Model

4.5.1.1 Shim Model

For the Shim, or wedge, approach, a portion of functionality is intercepted and modified by the shim module:



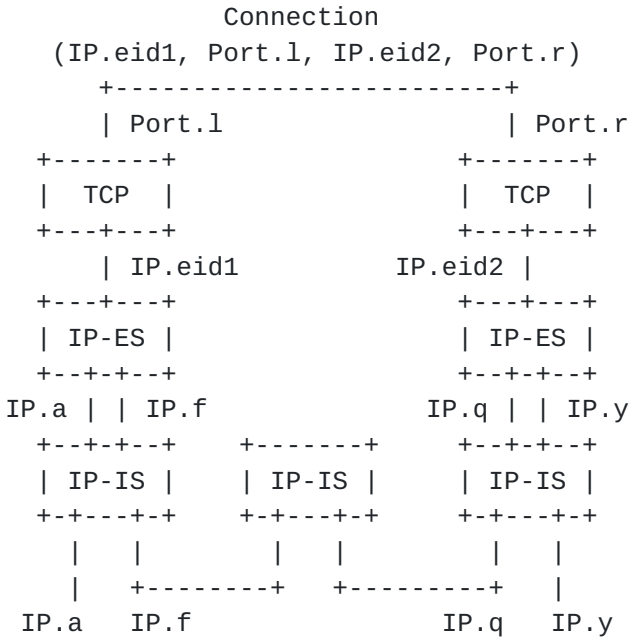
The transport layer is presented with the same service -- including the same apparent "addressing" model -- as previously. However the address is actually used as an identifier that maps to a set of locators. In the shim model, the identifier may be one of the valid locators.

4.5.1.2 IP/Transport Convergence Layer Model

Rather than viewing this type of service as being ad hoc, it can be seen as an example of IP-level services that reside only in the endpoint. That is, there is a distinction between the relaying activities in every "intermediate" system (IP-IS), versus IP functions that are needed only in the end-systems at the endpoints (IP-ES). For multiAddressesing, the architectural impact is embodied by using an Endpoint IDentifier [EID] in the

interface between IP-ES and the transport layer, rather than using an endpoint locators. Significantly, the EID might be private to the endpoint(s), rather than needing to be globally registered.

IPSec is another example of and IP-ES service. Note that this architectural change also must affect the upper-layer access to DNS, since DNS Address records must be converted to EIDs.



4.5.2. Host Identity Protocol (HIP)

HIP works with IPv4 and IPv6. Also, it:

* Creates a new, globally unique name space
* Uses strong, cryptographically based protocol details, overloading some HIP functionality with security functionality
* Is tied significantly to [IPSEC]
* Creates a new DNS RR entry
* Requires a Rendezvous (referral) server for mobility support
* Requires that NATs be aware of HIP

Many of the HIP features are appealing, such as the cleanliness of the architectural model depicted in Section 4 of the HIP architecture document. Were the Internet stack being created now, HIP well might be an excellent approach. However retrofitting this approach into the existing, deployed Internet entails serious barriers to adoption, such as its dependence on IPSec.

In general, addition of a DNS SRV record can be useful for achieving efficient rendezvous, with or without mobility. It permits participants to know whether a service is supported by its partner, without requiring a probe packet. While beneficial, this enhancement to DNS data structures is not required for multihoming or client (initiator) mobility.

4.5.3. LIN6

LIN6 defines a clean separation between identifier and locator, all within an IPv6 Address format. It defines a new, globally unique 64-bit endpoint identifier that is used by upper layers. This is then mapped to one or more IPv6 IP-layer locators.

The LIN6 specification also provides for the referral function (mapping agent), using DNS for basic name resolution and a separate, dynamically updated service to provide accurate information about rapidly changing locators.

LIN6 differs from HIP in that it:

* Is limited to IPv6 but integrates into IPv6 numbering
* Adds a transient "presence" service to DNS lookup, for dynamic locator mapping

4.5.4. MAST

MAST is a control protocol for the exchange of IP Addresses notification and authorization, to use additional IP Addresses in a given host-pair context.

The primary MAST exchange transmits:

* A list of current IP Addresses supported by the sender

Support exchanges:

* Establish a host-pair context
* Establish relevant authentication between the pair

MAST takes a more modest approach than HIP or LIN6. It does not define a new identifier space, has a simpler specification, permits easier implementation and adoption, and works equally with IPv4 and IPv6. MAST has the unusual characteristic of permitting its application to a transport association to begin after the association is underway.

Referral to a mobile target is provided as an adjunct function. Initial referral and rendezvous identification reliy on domain names. For mobile endpoints, dynamic locator information is

obtain through an associated presence service. New locator information is communicted during an association by the control protocol; security relies on use of a statistically unique, emphemeral identifier.

MAST differs from the list of HIP requirements in that it:

* Uses a name space that is transient and local to the host-pair
* Treats referral as an adjunct requirement and has no special requirements on DNS, in the base service
* Is transparent to NATs

MAST differs from LIN6 requirements in that it:

* Uses a name space that is transient and local to the host-pair
* Treats referral as an adjunct requirement
* Works with IPv4, as well as IPv6

5. SECURITY CONSIDERATIONS

This is a discussion paper and specifies no actions. Hence it has no security impact, except in terms of generally discussing security issues for the IP architecture.

An excellent discussion about the types of attacks that are relevant to multiaddressing mechanisms is contained in [WEAK]. Notably it discusses the use of association-specific ephemeral keys, without needing a global certificate service.

APPENDIX

A.1. Acknowledgements

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

Marcelo Bagnulo has contributed extensively to this draft. He would be listed as a co-author, but he was not given an opportunity to review it, due to the impending IETF I-D deadline.

Commenters on this text include: Marcelo Bagnulo, Fred Baker, Iljitsch van Beijnum, Vint Cerf, Spencer Dawkins, Robert Honore, James Kempf , Eugene Kim, Eliot Lear, Pekka Nikander, Erik

Nordmark, Tim Shepard, Randall R. Stewart, Fumio Teraoka, and Bob Hinden.

Networking history scholars may note that some terms used in the paper echo from the ancient times of OSI. Apologies are offered. Alas, that effort produced some useful architectural references and terminology, in spite of its problematic protocol specification work. Use of those terms, here, is a matter of pragmatics, not religion. The Internet community lacks broad use of some necessary terminology. Those objecting to any of the terms used here are encouraged to offer others, and to get community support for them.

A.2. References

Non-Normative

[DCCP] Kohler, E., M. Handley, S. Floyd, J. Padhye, "Datagram Congestion Control Protocol (DCCP)", draft-ietf-dccp-spec-04.txt, 30 June 2003

[DNSDYN] Vixie, P., Thomson, S., Rekhter, Y., Bound, J., Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC2136, April 1997

Wellington , B., "Secure Domain Name System (DNS) Dynamic Update", RFC 3007, November 2000

[EID] Chiappa, J.N., "Endpoints and Endpoint Names: A Proposed Enhancement to the Internet Architecture", <http://users.exis.net/~jnc/tech/endpoints.txt>, 1999

[ETCP] Zhang, B., Zhang, B., Wu, I., "Extended Transmission Control Protocol (ETCP) Project--Extension to TCP for Mobile IP Support", <http://www.cs.ucla.edu/~bzhang/etcp/report.html>

[HIP] Moskowitz, R., "Host Identity Protocol Architecture", < http://www.ietf.org/internet-drafts/draft-moskowitz-hip-arch-03.txt >

Moskowitz, R., "Host Identity Protocol", <ietf-id: draft-moskowitz-hip-07>

[HOWIE] Howie, D., "Consequences of using MIPv6 to Achieve Mobile Ubiquitous Multimedia", http://www.mediateam.oulu.fi/publications/pdf/38

4.pdf

[IPSEC] Kent, S. and R. Atkinson, "Security Architecture for the Internet Protocol", RFC 2401, November 1998

[LIN6] Teraoka, F., Ishiyama, M., Kunishi, M., "LIN6: A Solution to Mobility and Multi-Homing in IPv6", draft-teraoka-ipng-lin6-02.txt, 24 June 2003

[MOBMH] Nikander, P., "End-Host Mobility and Multi-Homing with Host Identity Protocol", <http://www.ietf.org/internet-drafts/draft-nikander-hip-mm-00.txt>

[NAT] Egevang, K., and P. Francis, "The IP Network Locators Translator (NAT)", RFC1631, May 1994

[NSRG] Lear, E., Droms, R., "What's In A Name: Thoughts from the NSRG", draft-irtf-nsrg-report-10.txt, September 2003

[MAST] Crocker, D., "Multiple Locators Service for Transport (MAST):
An Extended Proposal", draft-crocker-mast-proposal-00.txt, September 13,2003

[MIP4] Perkins, C., "IP Mobility Support for IPv4", RFC 3344, August 2002

MIP4 Working Group, <http://www.ietf.org/html.charters/mip4-charter.html>

[MIP6] Johnson, D., Perkins, C., Arkko, J., "Mobility Support in IPv6", draft-ietf-mobileip-ipv6-24.txt, June 30, 2003

Bagnulo, M., Garcia-Martinez, A., Soto, I., "Application of the MIPv6 protocol to the multi-homing problem", draft-bagnulo-multi6-mnm-00, February 25, 2003

MIP6 Working Group, <http://www.ietf.org/html.charters/mip6-charter.html>

[PBK] Bradner, S., Mankin, AS., Schiller, J., "A Framework for Purpose-Built Keys (PBK)", draft-bradner-pbk-frame-06.txt, June 2003

[SALT] Saltzer, J., " On the Naming and Binding of Network Destinations", RFC 1498, A ugust 1993

[SCTP] L. Ong, and J. Yoakum "An Introduction to the Stream Control Transmission Protocol (SCTP)", RFC 3286, May 2002

R. Stewart, R., Xie, Q., Morneault, K., Sharp, C., Schwarzbauer, H., Taylor, T., Rytina, I., Kalla, M., Zhang, L., Paxson, V., "Stream Control Transmission Protocol", RFC 2960, October 2000

R. Stewart, et al, "Stream Control Transmission Protocol (SCTP) Dynamic Locators Reconfiguration", draft-ietf-tsvwg-addip-sctp-07.txt, February 26, 2003

[TCP-MH] Matsumoto, A. Kozuka, M., Fujikawa, K., Okabe, Y., "TCP Multi-Home Options", draft-arifumi-tcp-mh-00.txt, 10 Sep 2003

[TLS] Dierks, T., C. Allen , "The TLS Protocol Version 1.0", RFC 2246, January 1999.

[WEAK] Arko, J., Nikander, P., "Weak Authentication: How to Authenticate Unknown Principals without Trusted Third Parties", , <http://www.tml.hut.fi/~pnr/publications/cam2002b.pdf>

A.3. Author's Adress

Dave Crocker
Brandenburg InternetWorking
675 Spruce Drive
Sunnyvale, CA 94086 USA

tel: +1.408.246.8253
dcrocker@brandenburg.com

A.4. Full Copyright Statement