Identity and Locators in IPv6

IAB Meeting

IETF 60

August 2004

This is not a new discussion

- Big Internet discussion from 10 years ago
- Has anything changed in this debate over the past 10 years?

Agenda…

- How Multi-Homing WG has approached the problem
- What forms of approach are possible to create a useful ID / Locator split in IPv6
- Discussion on next steps

The Multi-Homing Motivation

- How do you create a service that's available 100% of the time?
  - Use a server architecture and location environment that uses sufficient resiliency to provide 100% availability

  and

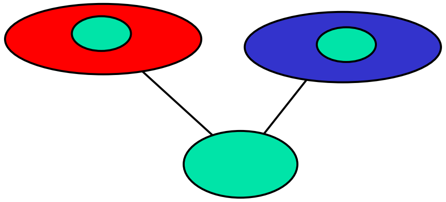
  - Connect to the Internet using a service provider than can provide 100% _guaranteed_ availability

100% Network Availability?

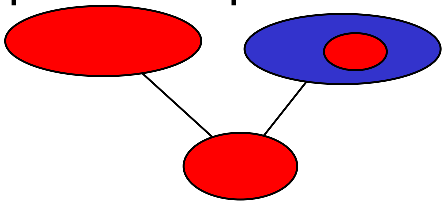
- Multiple connections to a single provider ?
  - BUT - there's a single routing state that is vulnerable to failure
- Multiple Connections to multiple providers ?
  - More attractive, potentially allowing for failover from one provider to another in the event of various forms of network failure

IPv4 Multihoming

- Either:
  - Obtain a local AS
  - Obtain PI space
  - Advertise the PI space to all upstream providers
  - Just follow routing

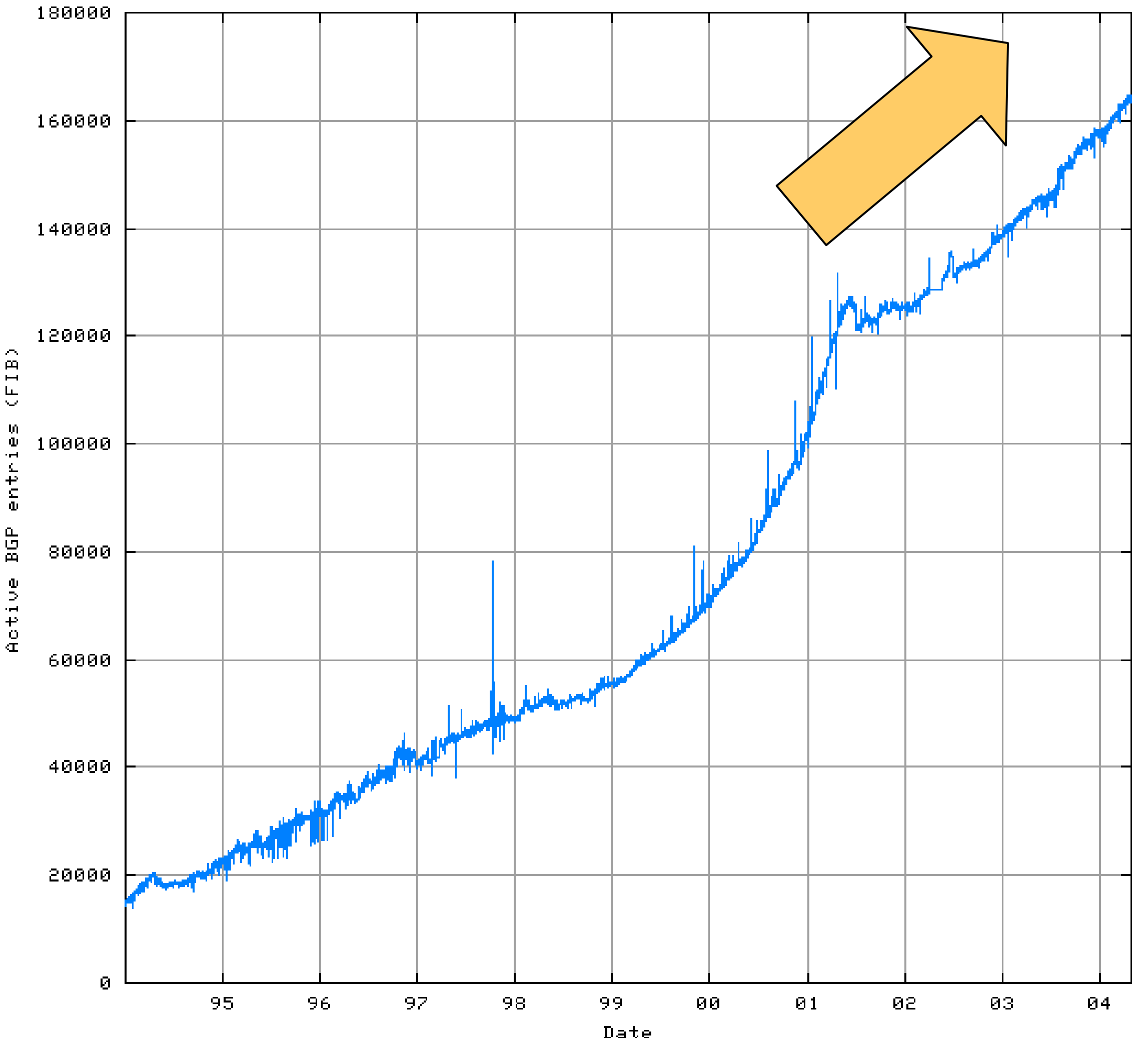


- Or:
  - Use PA space fragment from one provider
  - Advertise the fragment it to all other upstream providers
  - Just follow routing



Scaling Global Routing

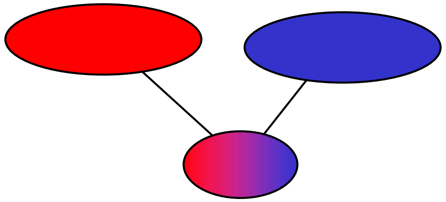
- Both approaches have obvious implications in terms of additional entries being added to the global routing system, with little (or no) control over route object propagation



Scaling

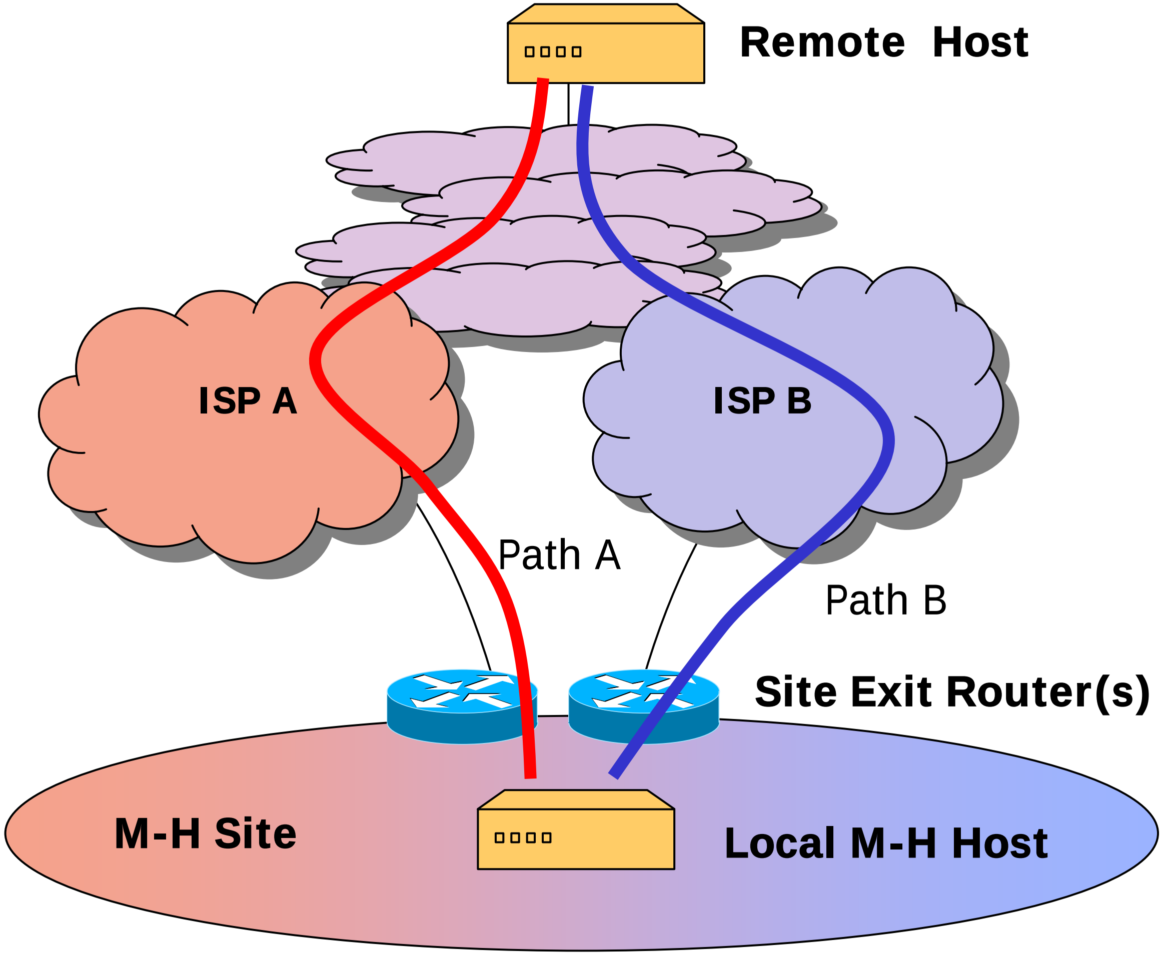
- There are potentially millions of sites that would see a benefit in multi-homing. It is commonly believed that the routing table cannot meet this demand
- Is there an alternative approach that can support multi-homing without imposing a massive load on the global routing system?
  - Change scope controls in routing
    - (ptomaine, grow)
  - Change the protocol architecture
    - (HIP, multi6)

What Multi-Homing would like…



- The multi-homed site uses 2 address blocks
  - One from each provider
- No additional routing table entry required

The Problem Space



Functional Goals

- RFC3582 enumerates the goals as:
  - Redundancy
  - Load Sharing
  - Traffic Engineering
  - Policy
  - Simplicity
  - Transport-Layer Surviveability
  - DNS compatibility
  - Filtering Capability
  - Scaleability
  - Legacy compatibility

- Also we need to think about::
  - Interaction with routing
  - Aspects of an ID/Locator split, if used
  - Changes to packets on the wire
  - Names, Hosts, endpoints and the DNS

i.e. Do everything, simply, efficiently and cheaply with no other impact !

Status of Multi6

- There appears to be no simple, secure, one-ended approach to this problem space
  - Both ends of the session need to be aware of the capability of binding multiple locators into a single session
  - This implies that multi-homing in V6 will remain, in the near future, a routing technique
- And the agenda for multi6 is, in reality, focussed on the practical issues of id/locator split protocol design (in practice, or a virtual split)
  - And the question is “is the scope of the multi6 effort sufficiently generic so as to provide useful outcomes for the general case of id/locator split functionality in IPv6?”

Agenda…

- How Multi-Homing has approached the problem
- What forms of approach are possible to create a useful ID / Locator split in IPv6
- Discussion on next steps

ID / Locator Split

- The IP protocol architecture has made a number of simplifying assumptions
  - Your IP address is the same as your identity (who)
  - Your IP address is the same as your location (where)
  - Your IP address is used to forward packets to you (how)

- If you want multi-homing to work then your identity (who) must be dynamically mappable to multiple locations (where) and forwarding paths (how)
  - “its still me, but my location address has changed”

Benefits:

- Allow indirection between identity and location
- Provide appropriate authentication mechanisms for the right function
- Allow location addresses to reflect topology and provider hierarchies without overload of identity semantics
- Allow identities to be persistent across location change (mobility, re-homing)

Generic Approaches:

- Insert a new level in the protocol stack

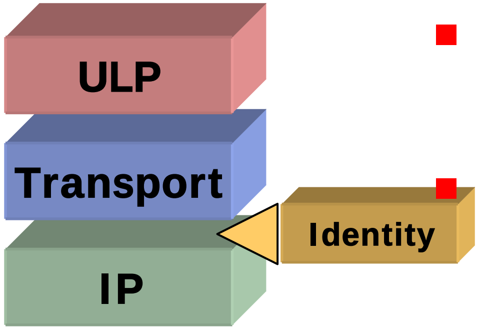
  New protocol element

- Modify the Transport or IP layer of the protocol stack in the host

  Modified protocol element

- The difference is subtle, but it relates to the persistence, scope and functionality of the identity binding.

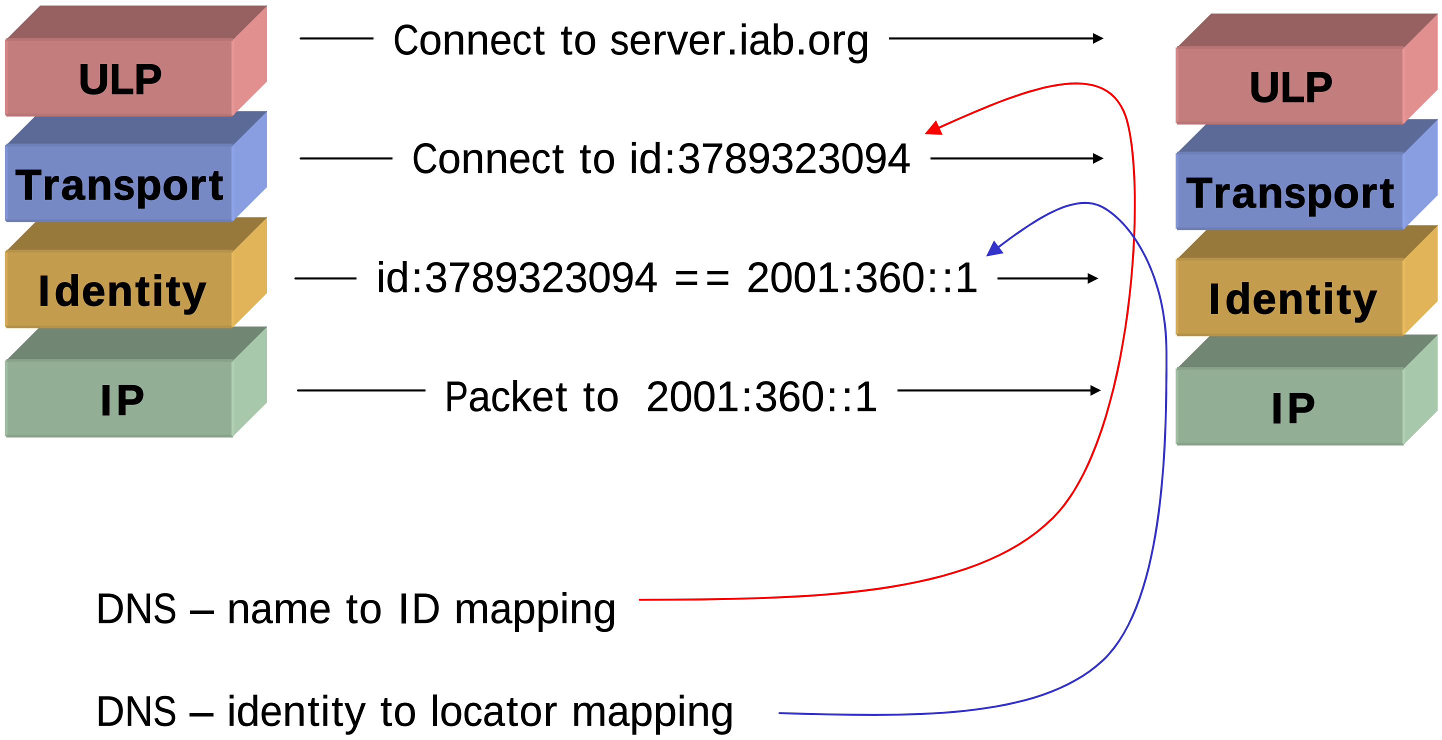
Identity Protocol Element



- Define an Identity Protocol element that:
  - presents an identity-based token to the upper layer protocol
  - Allows multiple IP address locators to be associated with the identity
  - Allows sessions to be defined by an identity peering, and allows the lower levels to be agile across a set of locators
  - Most likely to be placed at layer 3.5 (Transport / IP interface), allowing the transport layer to peer using identity tokens and the IP layer to form packets based on current locators
- Is this layer 3.6 (session) or layer 3.4 (host)?

Identity Protocol Element

A basic example scenario of host-based persistent identity



DNS – name to ID mapping

DNS – identity to locator mapping

Proposals for an Identity Token Space

Hierarchically Structured Space

- Use identity tokens lifted from a protocol’s “address space”
  - DNS, Appns, Transport manipulate an “address”
  - IP functions on “locators”
  - Stack Protocol element performs mapping
- FQDN as the identity token
  - Is this creating a circular dependency?
  - Does this impose unreasonable demands on the properties of the DNS?
- Structured token
  - What would be the unique attribute of a novel token space that distinguishes it from the above?

Unstructured

- Unstructured token
  - Allows for self-allocation of identity tokens (opportunistic tokens)
  - How to map from identity tokens to locators using a lookup service?

Identity Structures

- Persistent structured "address" that is a host-based identity (that may or may not have locator significance)
  - Can perform id locator mapping (bi-directionally) via a structured search mechanism

or

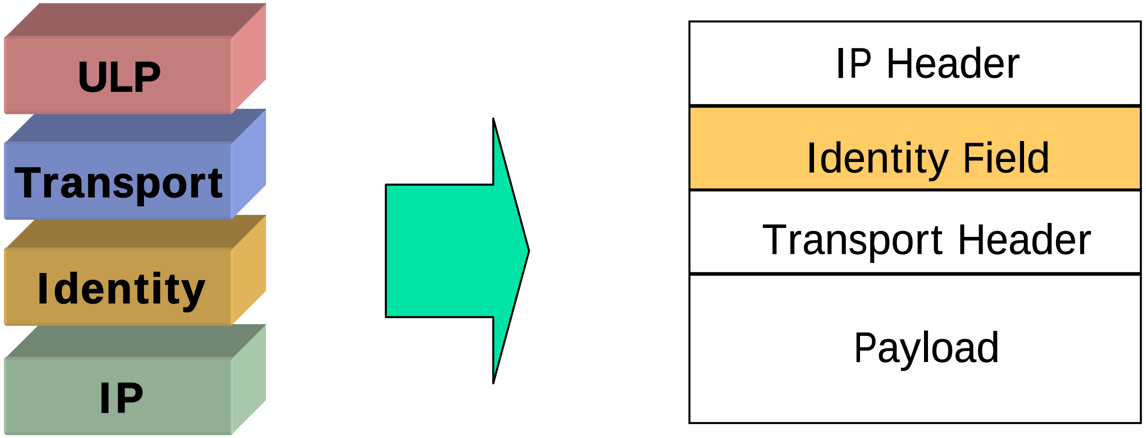
- Opportunistic self-generated bit sequence used in the context of session-based identity
  - Is used in the context of dynamic binding of additional locators to an existing session

or

- Trying to mesh these two approaches in some manner

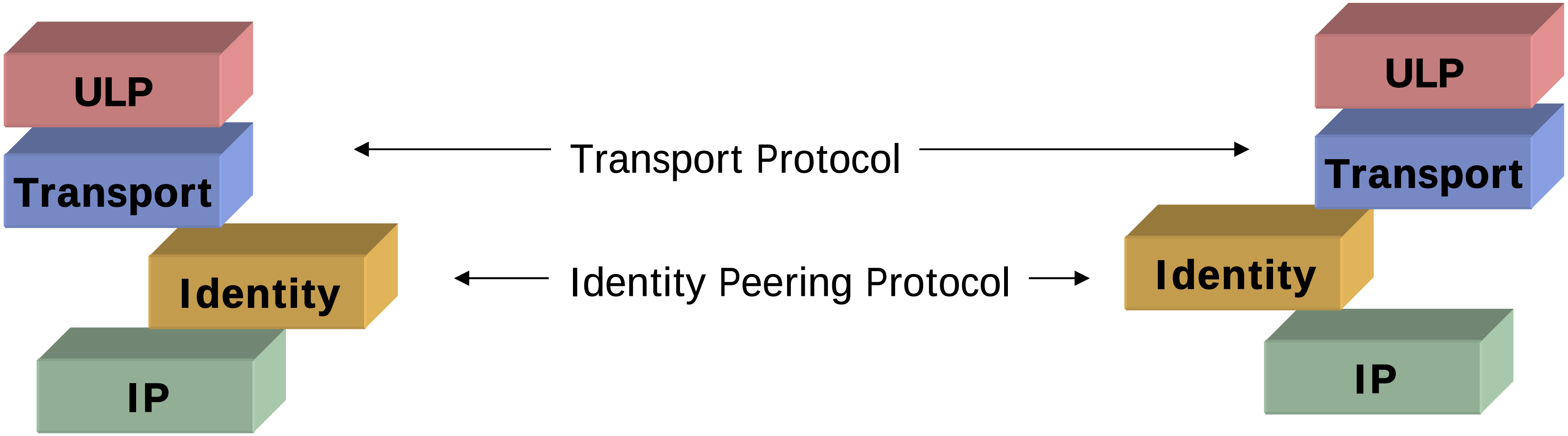
Protocol Element Implementation

- "Conventional"
  - Add a wrapper around the upper level protocol data unit and communicate with the peer element using this "in band" space



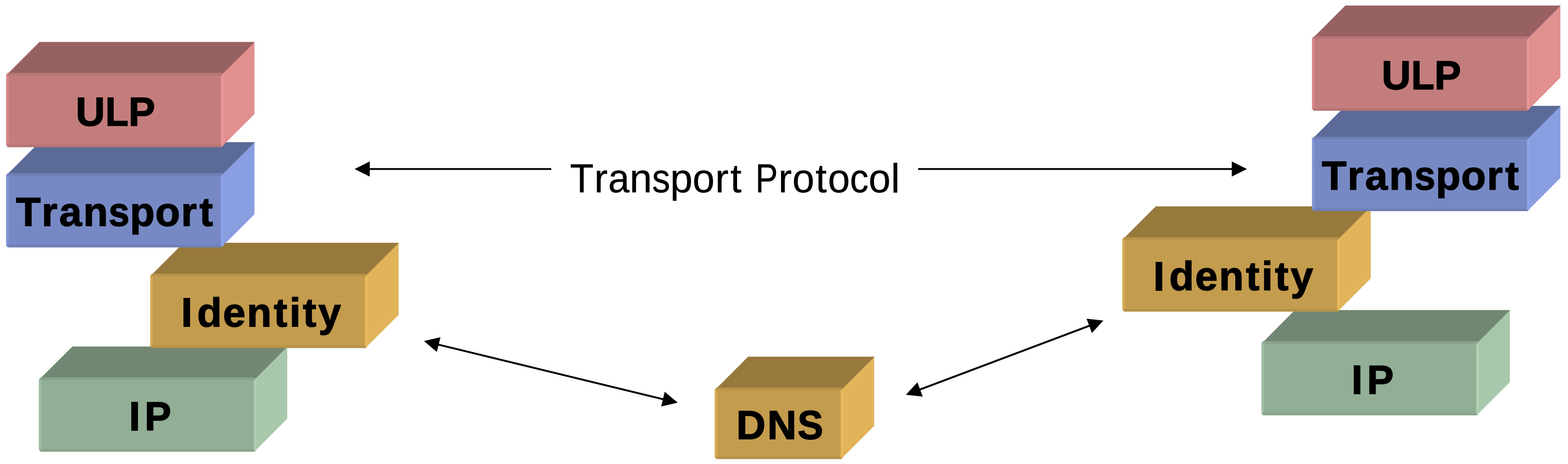
Protocol Element Implementation

- "Out of Band"
  - Use distinct protocol to allow the protocols element to exchange information with its peer

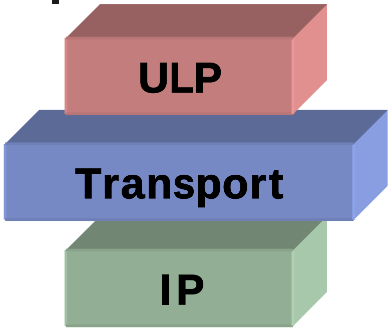


Protocol Element Implementation

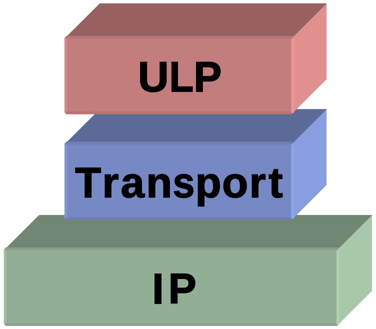
- "Referential"
  - Use a reference to a third party point as a means of peering (e.g. DNS Identifier RRs)



Identity Protocol Element Proposals

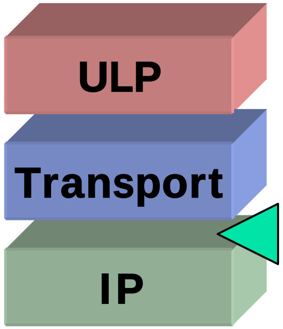


- Alter the Transport Protocol to allow a number of locators to be associated with a session
  - e.g. SCTP



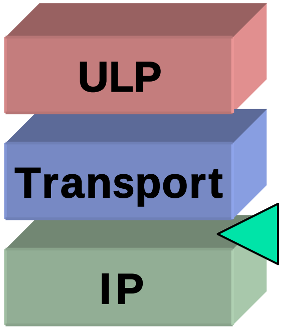
- Alter the IP protocol to support IP-in-IP structures that distinguish between current-locator-address and persistent-locator-address
  - i.e. MIP6

Identity Protocol Element Proposals



- HIP:
  - Shim between Transport and IP layer
  - Presents a stable identity to the transport layer (cryptographic hash of local identity key)
  - Allows multiple locators to be bound to the identity, and communicates this binding to the remote end (HIP protocol)
  - Allows the local host to switch source locators in the event of network failure to ensure session surviveability. The crytographic function is used to determine if the new locator is part of an already established session. (“same key, same session”)

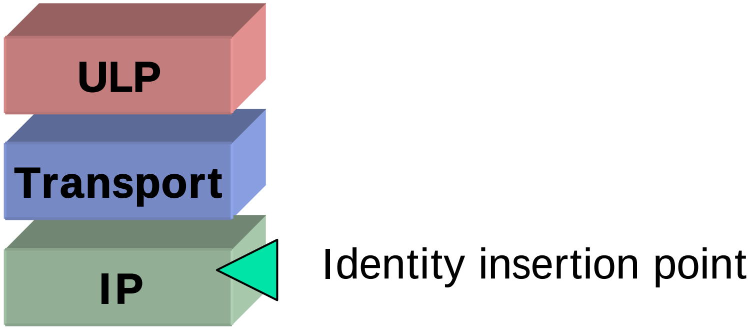
Identity Protocol Element Proposals



- NOID +
- SIM (CBID 128) +
- CB64:
  - Addition of an identifier shim layer to the protocol stack.
  - The identifier / locator mapping may be contained in the DNS (NOID) or may be contained within a protocol exchange (SIM), or a hybrid approach (CB64)

Identity Protocol Element Location

- It appears that the proposals share a common approach:
  - Above the IP forwarding layer (Routing)
  - Below IP fragmentation and IPSEC (IP Endpoint)



Common Issues

- Picking the ‘best’ source locator

  (how do know what destination works at the remote end?)

  - Use each locator in turn until a response is received
  - Use a identity peering protocol to allow the remote end to make its own selection from a locator set
- Picking the ‘best’ destination locator
  - Longest match
  - Use each in turn
- Picking the ‘best” source / destination locator pair
  - As these may be related choices

Common Issues

- Detecting network failure

  (How does a host know that its time to use a different source and/or destination locator?)

  - Heartbeat within the session
  - Modified transport protocol to trigger locator change
  - Host / Router interaction to trigger locator change
  - Application timeframe vs network timeframe
  - Failure during session startup and failure following session establishment

Common Issues

- Network layer protocol element
  - How do you know a session is completed?
    - The concept of session establishment and teardown is a transport concept, not an IP level concept
  - What do you need to do to bootstrap?
    - Are there ‘distinguished’ locators that you always need to use to get a session up?

Common Issues

- Session Persistence
  - Use one locator as the “home” locator and encapsulate the packet with alternative locators
  - Set up the session with a set of locators and have transport protocol maintain the session across the locator set
    - Optionally delay the locator binding, or allow the peer dynamic change of the locator pool
  - Use a new peering based on an identity protocol element and allow locators to be associated with the session identity

Common Issues

- Identity / Locator Binding domain
  - Is the binding maintained per session?
    - In which case multiple sessions with the same endpoints need to maintain parallel bindings
  - Is the binding shared across sessions?
    - In which case how do you know when to discard a binding set?

Common Issues

- Bilateral peer applications vs multi-party applications
  - What changes for 3 or more parties to a protocol exchange?
- Application hand-over and referral
  - How does the remote party identify the multi-homed party for third party referrals?

Security Considerations

- Major agenda of study required!
- Worthy of a separate effort to identify security threats and how to mitigate these threat

Agenda…

- How Multi-Homing has approached the problem
- What forms of approach are possible to create a useful ID / Locator split in IPv6
- Discussion on next steps

Open Questions

- Id/Loc questions
  - Is the specification of a structured identity space coupled with changes to the IPV6 protocol stack a case of solution overkill?
  - What additional infrastructure service overheads are required to distribute a structured identity space?
  - Is there an existing identity space that could be used for this purpose?
  - Is the identity point the device or the protocol stack?
  - Is per-session opportunistic identity a suitably lightweight solution?
  - Is this just multi-homing or a more generic id/locator discussion?

Open Questions

- Applications and Identities
  - Is a self reference within an application the identity value?
  - If so, then can opportunistic id values be used in this context?
  - Should an application be aware of the presence distinction between id and locator, and alter its self-identification according to the capability of the current session?
  - How does this apply to UDP?

Properties of an /Locator Split

- Properties of a structured identity space
  - Creating yet another managed token space for a set of structured stack identities may be overkill
- Properties of opportunistic keys
  - The lack of persistence may make initial key association vulnerable to attack
  - Lack of support for referral function
  - Continuation of overloaded semantics of IPv6 addresses
- Should a coherent architecture support a range of identity types?