shim6

kurtis@kurtis.pp.se

2006-04-24

Background

- This is more or less my personal take...
  - But I happen to be the co-chair of shim6 with Geoff Huston
- I made a similar presentation at RIPE51 and Geoff at Nanog and me again at APRICOT
- Thanks to Erik Nordmark for contributions!

The problem

- Some end sites want multiple connections to different upstreams for
  - Resilience
  - Renumbering avoidance
- This does not *HAVE* to imply multiple upstream providers
  - But it can

The problem

- Routing system constraints
  - In order for the multiple upstreams to forward traffic to the end-site, a unique identifier is needed for the longest-prefix-match algorithm
- In IPv4 this is either of
  - PI address block
  - “more specific” PA
  - Multiple addresses on each node

Effects of Multihoming

- Leads to "uncontrolled" growth of the routing table
  - Can lead to problems in the future
- Would be better if each end-user/site could get a block from each provider
  - And be able to use both prefixes as source addresses in case of failures
  - Today this does not work due to inbound-filtering at the ISPs

The IETF effort

- The multi6 WG was tasked with inventorying possible solutions
  - And benchmarking/selecting a solution
  - Selected an architecture based on separating locator / identifier
- Work on protocol is moved to the shim6 WG

The SHIM6 Solution

- host-based solution (rather than host and router)
- network layer (rather than transport)
- discoverable negotiated capability
- no new identifier space

The SHIM6 Approach

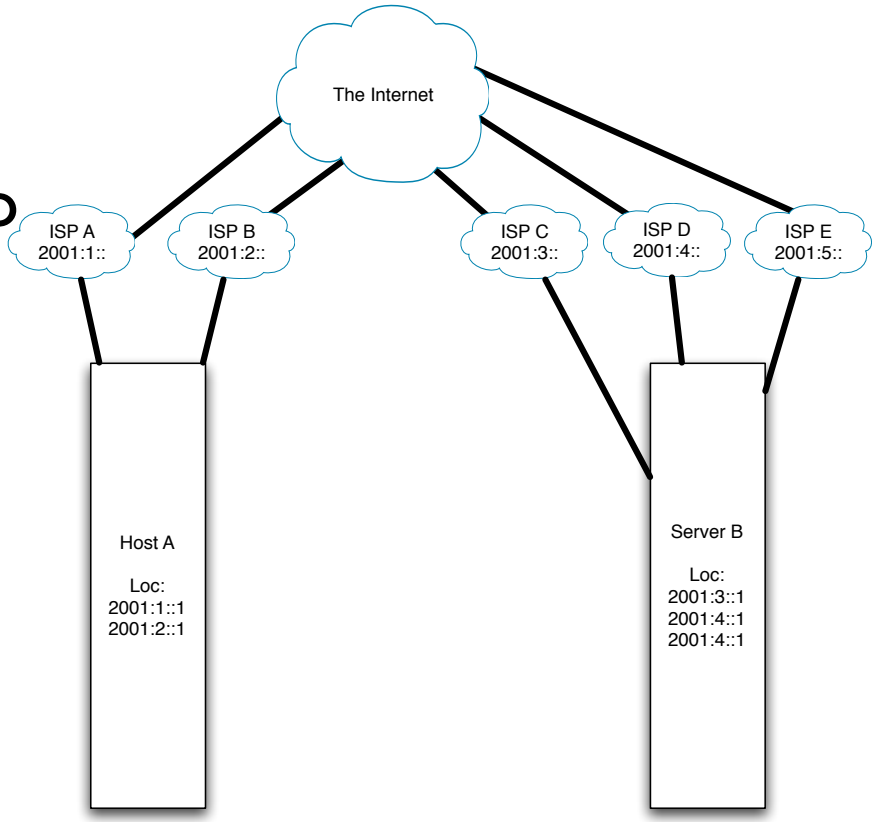
- A functional module at layer 3 (IP)
- The initial locator is the upper layer identifier (RFC3484/RFC3484bis selection)
- Subsequent negotiation to enable the Shim6 module for an upper layer identifier pair
- The Shim6 module translates upper layer identifiers into the currently active forwarding layer locators
- The upper layer identifier pair plus a context value forms the shared shim6 state identifier
- An IPv6 end-to-end header is used to signal SHIM6 context

shim6 - protocol

- Current thinking is that the base header will look remarkably like a HIP header
  - but it is *NOT!*
- Some issues are still TBD but we have come a far way....

A case study...

- Let's assume two hosts that try to communicate
  - Client A and Server B



A case study...

- First step is for the host A to look-up the name of server B in the DNS. Host A will then get three AAAA's as reply.
  - No change from today
- Host A will based on RFC3484 do source/destination address selection and connect to one AAAA. If that fails, it will try another
  - The AAAA's can point to the same or different servers BTW...
    - No change from today

A case study...

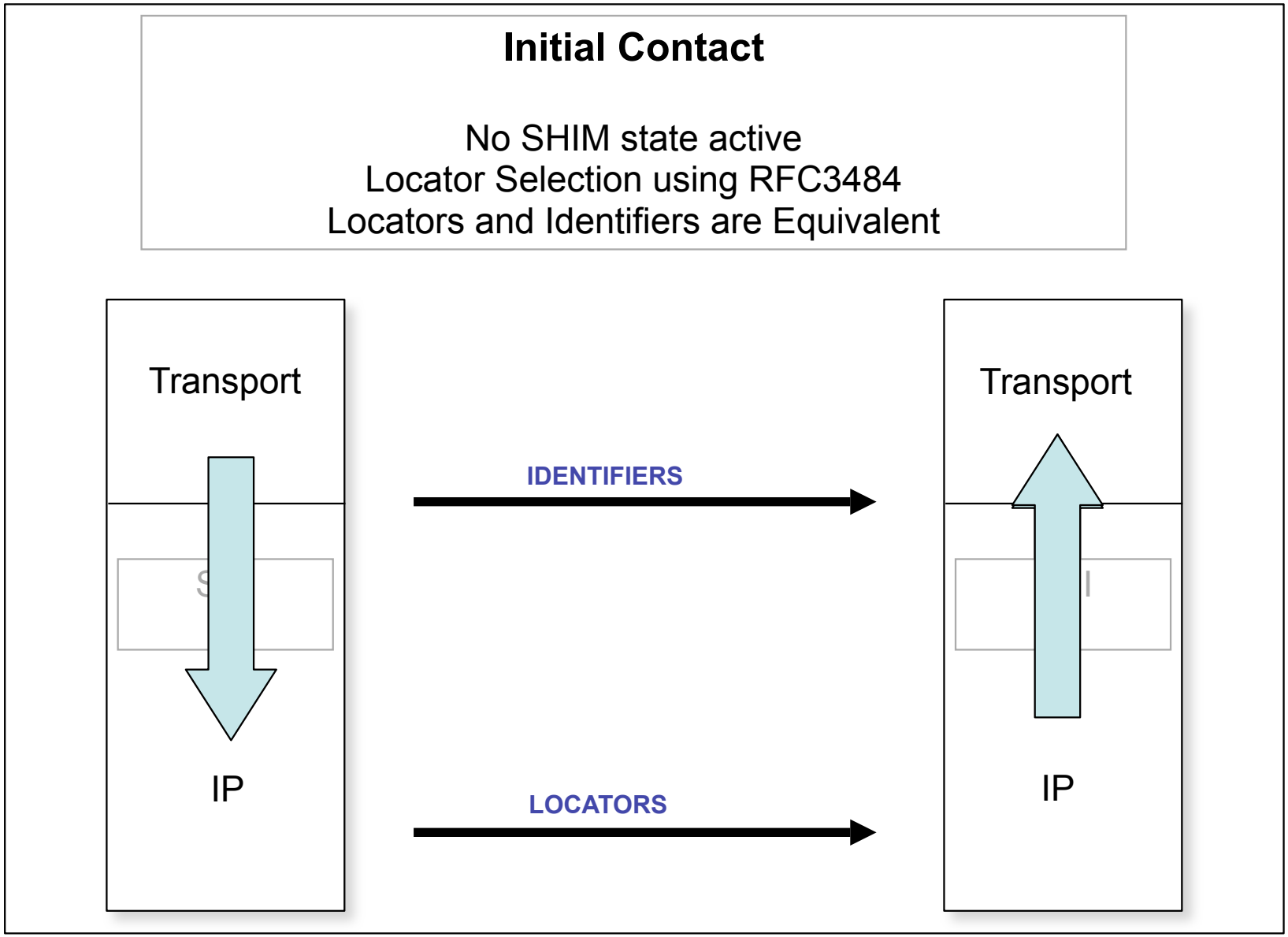
- BTW DNS, isn't really required
  - No change from today
- Let's assume an application uses TCP (but could be any transport protocol) and now establishes one or more sessions with the server
  - So far nothing new

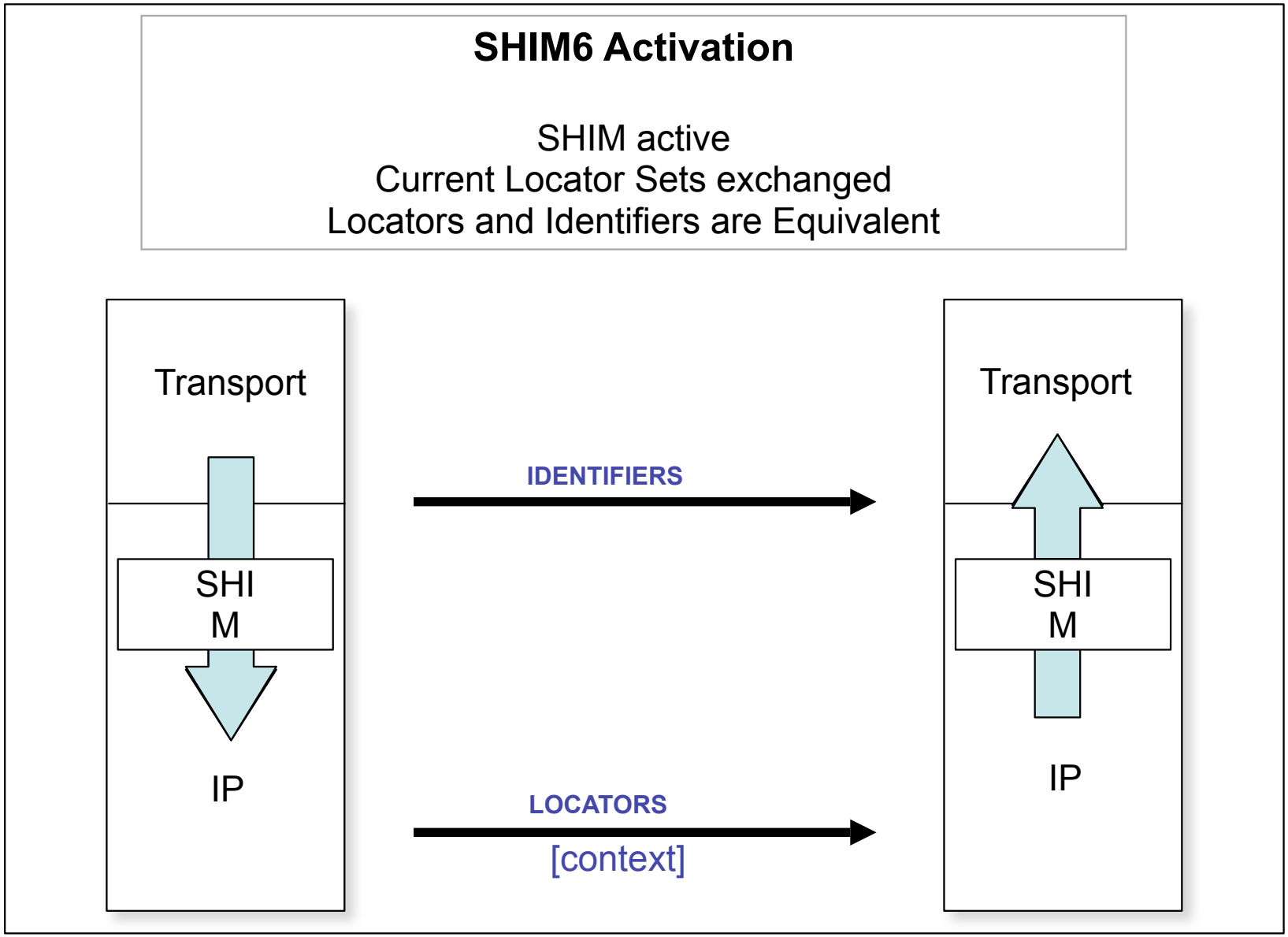
A case study...

- At some point in time shim6 at either end of the communication determines that some heuristic applies (e.g. number of packets between the pair of IP addresses; NOTE: Not the per TCP connection)
  - Shim6 will now perform a 4-way handshake to do context establishment
  - Now either side have the alternative IPv6 address for the other side plus some information to verify that these addresses belong to the other host

A case study...

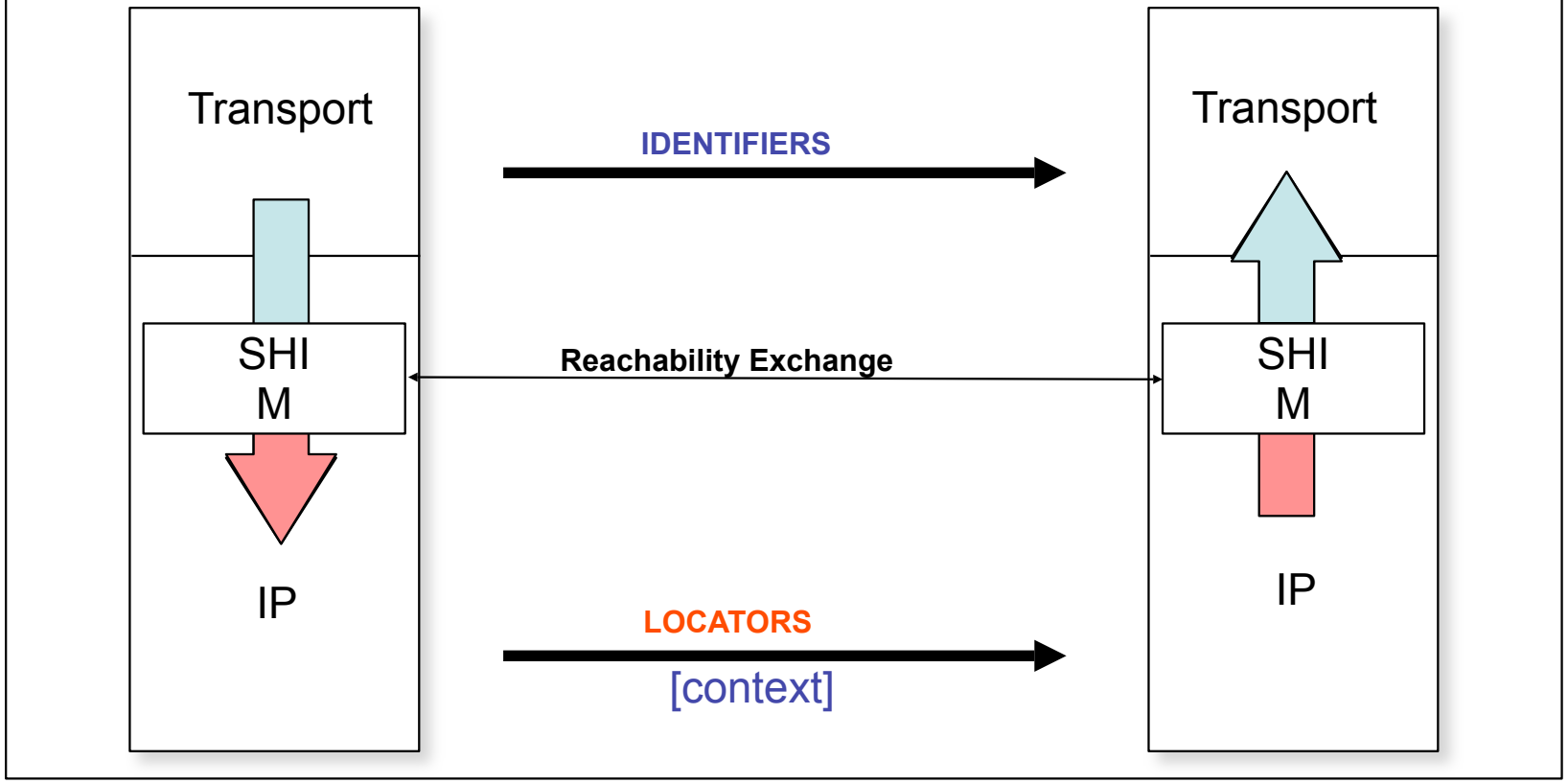
- After this shim6 can fail over traffic to another pair of locators (IPv6 addresses)
- Shim6 uses a mostly passive failure detection mechanism, and an exchange to find which locator pair is working after a detected failure


Initial Contact
No SHIM state active
Locator Selection using RFC3484
Locators and Identifiers are Equivalent
Transport
IP
IDENTIFIERS
LOCATORS
Transport
IP

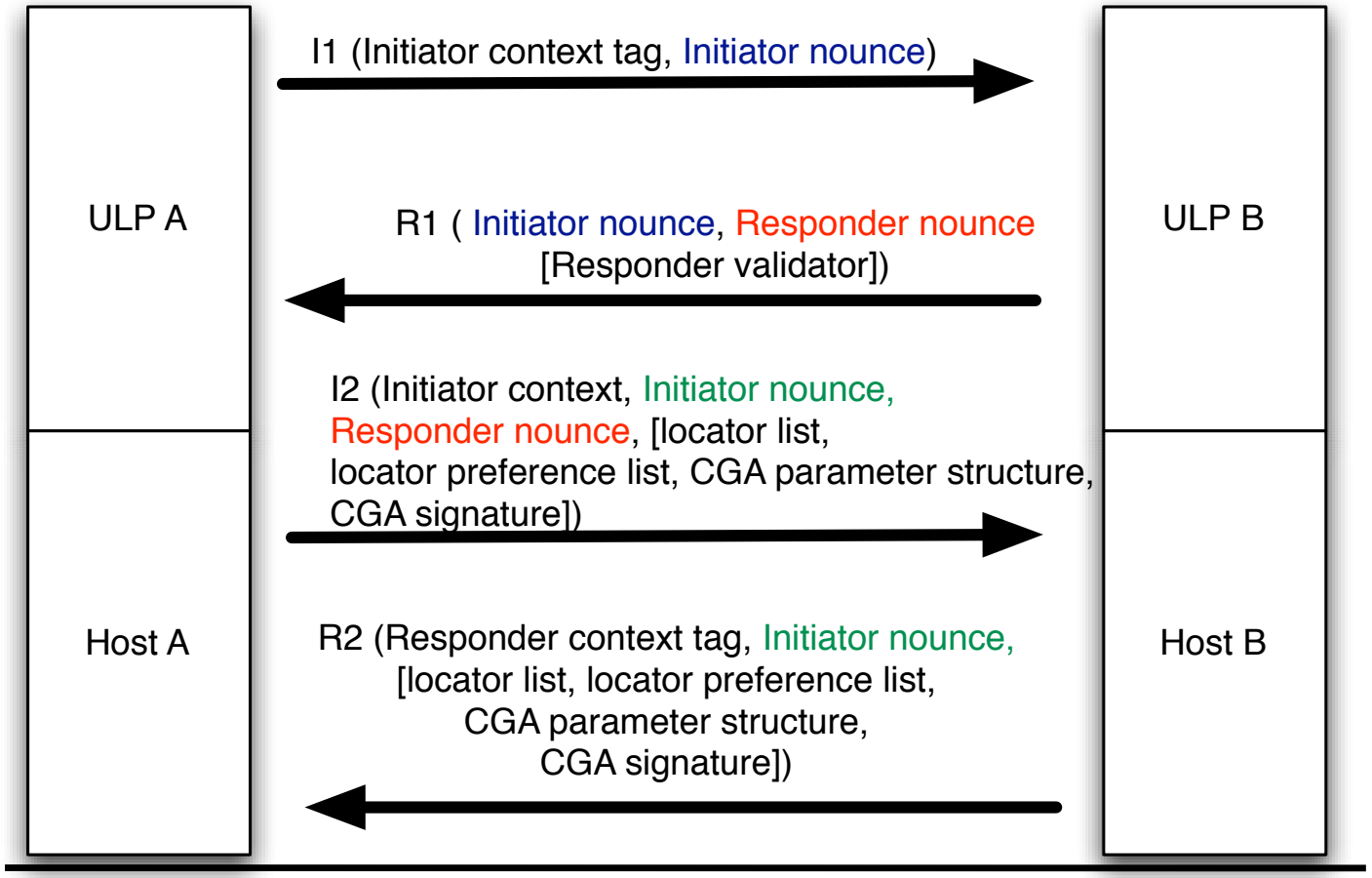

SHIM6 Activation
SHIM active
Current Locator Sets exchanged
Locators and Identifiers are Equivalent
Transport
SHI
M
IP
IDENTIFIERS
LOCATORS
[context]
Transport
SHI
M
IP

SHIM6 Locator Failure and Recovery

Detect locator failure
Explore for functioning locator pair
Use new locator pair – preserve identifier pair



Shim6 - protocol



2006-04-24

SHIM6 Control Elements

- initial handshake (4-way) and locator set exchange
- locator list updates
- explicit locator switch request
- keepalive
- reachability probe exchange
- Context recovery exchange (after one end has lost the context state)

SHIM6 WG Approach

- base protocol specification
  - protocol exchange and packet formats
  - address specification: CGA and HBA
  - functional decomposition
- refinements
  - upper layer signalling
  - traffic engineering hooks
  - contactless shim6
  - failure detection refinements
  - ingress filtering / source address path selection

?

Failure detection

Detecting (potential) failures

- Rely on Forced Bidirectional Communication
  - When there are no packets received and no packets transmitted, the shim does nothing
  - If packets are received but not transmitted (by the ULP) in some time interval (10s), the shim sends a keepalive to peer
  - If packets are transmitted and no packets are received, then there might be a problem (since the peer always sends if it is receiving)
- After detecting a failure, uses Probe messages to try the different locators in the locator set until it finds one which is working