multi6 design team

results and open issues

Tony Li (chair),
Erik Nordmark, Sean Doran,
Pekka Savola, Mike O'Dell,
Dave Katz, Brian Carpenter,
Iljitsch van Beijnum

multihoming

- Connect to...
  a. The same ISP more than once (ignore this, should be invisible from the outside)
  b. More than one ISP
- Expecting...
  - Redundancy/failover
  - Load sharing
  - Provider independence???

how it's done in IPv4

- Announce provider independent address block over BGP to ISPs: RIR /20 policy in the way
- Annouce more specific out of ISP block: filtering/ISP dependency issues
- Multiaddressing: doesn't really work
- NAT: doesn't do anything useful by itself, only makes multiaddressing a bit easier

IPv6 scalability

- Current IPv4 routing table: 122k prefixes with less than 25k multihomers
- BGP with PI or PA could lead to a 10M - 1G routing table eventually
- Even with enough memory routers can't handle this as processing scales < linear
- So for long-term scalability we need multiaddressing with provider aggregation

m-addressing != m-homing

- Reachability dictated by (source address / exit ISP, destination address) tuple
- Preferably be able to set source/exit and destination independently
- In practice:
  - Source address must match exit ISP due to ISP ingress filtering
  - Exit ISP depends on destination
- So this needs work

the socket API

- Socket API expects a session to have one 128-bit source and one 128-bit destination
- Breaking the API is not the way to get multihoming off the ground; IPv4 to IPv6 was (is) bad enough
- So: application can only supply one address, network needs multiple addresses
- "Any problem in computer science can be solved with another layer of indirection"

loc/id separation

- Identifiers are stable, for use by transport protocols and applications
- Locators are subject to change, used to navigate packets through the network
- Traditionally the IP address has always served both functions. But now:

what's in packets?

- Tunneling: 128 bit id in inner header, 128 bit loc in outer header
- "Small": both ± 64 bit id and ±64 bit loc in address field
- "Big": sometimes 128 bit id, sometimes128 bit loc in address field

tunnel

- Pro:
  - Simple!
  - Implement anywhere
- Con:
  - Problem when tunnel endpoint != end host, possible detours
  - 40 bytes of overhead in each packet
  - ICMP/firewall/PMTU issues

"small"

- Pro:
  - Host doesn't need to know own address
- Con:
  - Work with unaggregatable MAC name-space or break autoconfiguration
  - Can't trust incoming id-loc association
  - If not break, certainly bend transports
  - Changes to both hosts and routers

"big"

- Pro:
  - No per-packet overhead
  - Implement in either hosts or middleboxes
- Con:
  - Need additional mechanism to find identifier for first incoming packet
  - Need to keep state to find identifiers for subsequent incoming packets

common mechanisms

- Use unspecified distributed database to find locators for an identifier
- Source locator in incoming packets is used as default destination locator
- But: source is ultimately responsible for selecting a destination locator that works
- Work at IP level, *not* per session

ISP ingress filtering

- As we have several valid source addresses, we can have routers rewrite them, but:
  - Do we want this?
  - How to differentiate between multi-homed and legacy traffic
- ICMP message?
- NAROS?

consensus

- Consensus in the design team on:
  - Locator/identifier separation approach
    - Be agnostic about where in the site which part of multihoming processing happens
    - Don't trust incoming loc/id associations
    - Source is responsible for selecting the right destination locator

in practice

- Application looks up names, gets identifier and opens session
- Transport protocol also uses identifier
- Sender maps source and dest identifiers to locators
- Receiver maps locators back to identifiers

open issues

- "Big", "small" or tunnel?
- Where does id and loc space come from?
  - Overlap id with regular IPv6
  - Overlap loc with regular IPv6
  - Overlap all
  - Overlap none
- Better path selection
- Interdomain multicast, IPsec

Such as
2001::/16
or 3ffe::16

questions?

- One at a time please!