Dual Stack Hosts Using "Bump-In-the-Host" (BIH)

draft-huang-behave-bih-00

Bill Huang (CMCC)

Hui Deng (CMCC)

Teemu Savolainen (Nokia) - presenting

Behave WG meeting @ IETF#78

26-July-2010

Earlier work

- RFC 2767 Dual Stack Hosts using the "Bump-In-the-Stack" Technique (BIS), February 2000, *Informational*

- RFC3338 Dual Stack Hosts Using "Bump-in-the-API" (BIA), October 2002, *Experimental*

- BIS and BIA intented to increase number of applications that can make use of IPv6 networks
- BIS and BIA make it possible to update network and services without dependency to application updates
- BIS and BIA both come with limitations common and well-known for protocol translation -> not general purpose tools
- BIA avoids some of the problems of BIS by intercepting packets sooner
  - E.g. no protocol translation needed, no messing with DNS

The original problem still exist

- At ~2000 there were less IPv6 enabled applications than today, but there still *are* and *will be* IPv4-only applications
  - E.g. custom made corporate applications
- Need remains to allow class of IPv4-only applications to work over IPv6 accesses; especially applications that:
  1. Do name resolution
  2. Initiate communications
  3. Do not pass IP addresses in payloads

BIA and BIS are partially obsolete

- BIA uses IPv4 addresses already used for other purposes (0.0.0.1-0.0.0.255)
  - 0.0.0.0/8 is used in many places to **indicate an interface index** (except for 0.0.0.0). See e.g. RFC1724 and some socket APIs in Windows & FreeBSD
  - IANA has reserved 0.0.0.0/8 for self-identification in RFC5735
  - => BIA should use RFC1918 instead
- Both assume dual-stack internet access, lacking explicit support for IPv6-only access
- Reverse DNS lookup is not documented in either

BIS + BIA = BIH

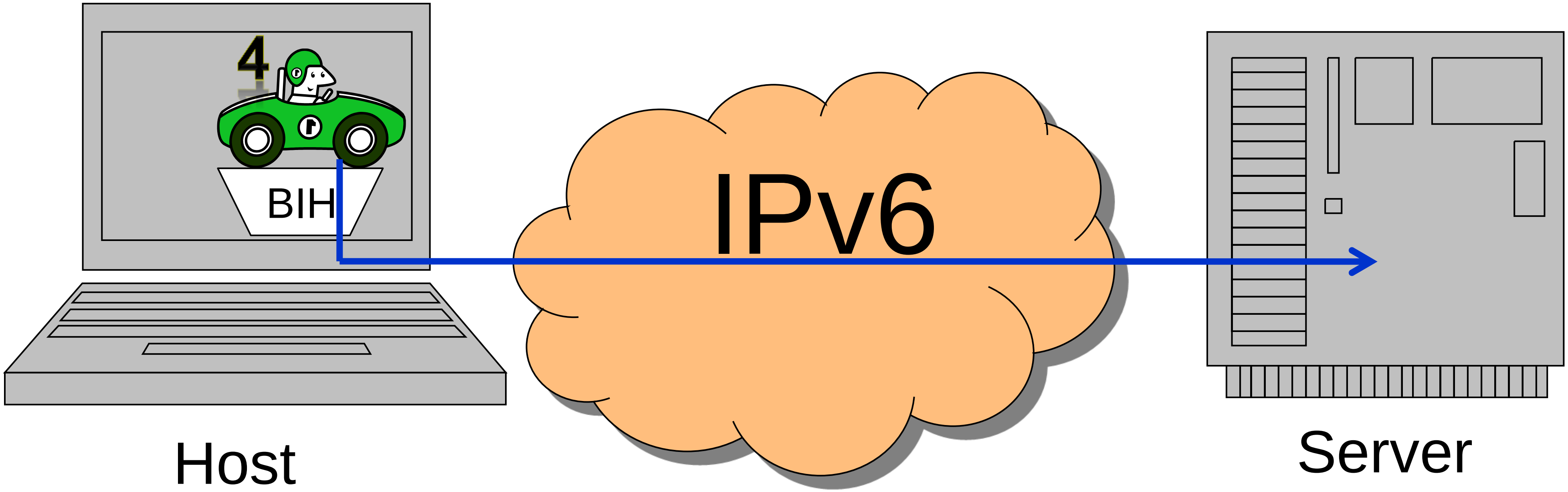
- BIS and BIA RFCs are essentially the same: there is a Bump (somewhere) In the Host = BIH
- No sense to update both documents independently
- draft-huang-behave-bih-00 merges and updates both
- As a result BIH has two implementation options:
  - Socket layer (aka BIA)
  - Network layer (aka BIS)

BIH applicability statement

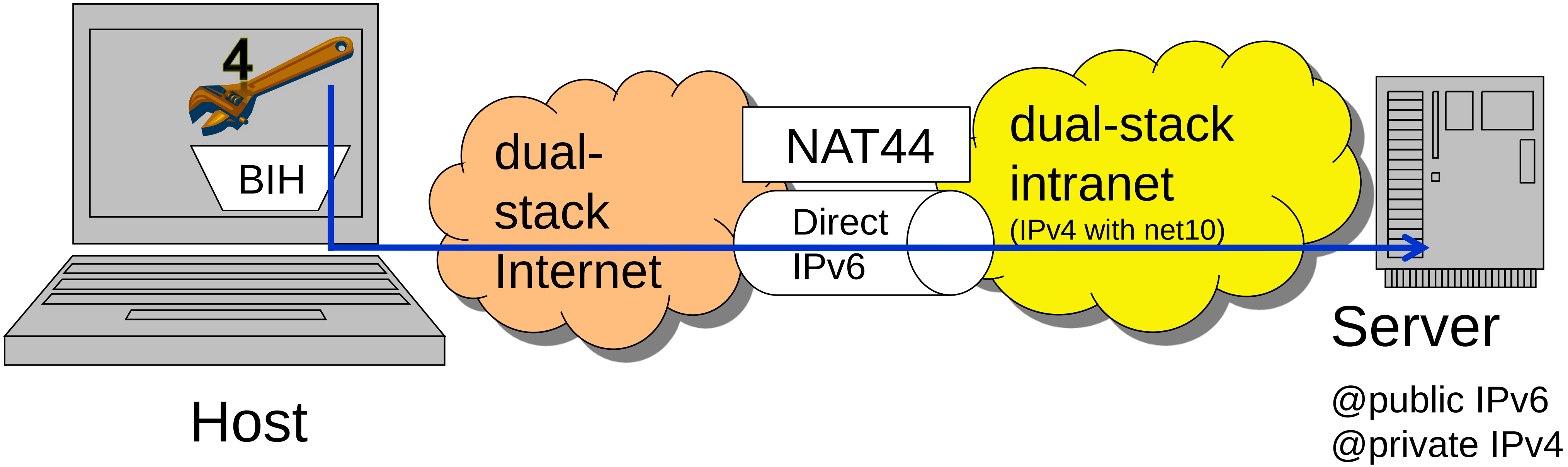
- BIH is not meant as a generic IPv6 transition tool

- BIH is targeted to help the class of applications that:
  1. Do name resolution
  2. Initiate communications
  3. Do not pass IP addresses in payloads
  4. Cannot be updated to IPv6 (in timely manner)

BIH in basic deployment scenario



BIH in more advanced scenario



- In this scenario, an IPv4-only application (e.g. when roaming) needs to connect to a server in a dual-stack intranet numbered with public IPv6 and private IPv4 addresses
- Static NAT44 mapping could do, but with BIH the application can talk to server directly over IPv6 without NAT44 tuning

BIH architecture models illustrated

```
+----------------------------------------------+
| +------------------------------------------+ |
| |                                          | |
| |            IPv4 applications             | |
| |                                          | |
| +------------------------------------------+ |
| +------------------------------------------+ |
| |           Socket API (IPv4, IPv6)        | |
| +------------------------------------------+ |
| +-[ API translator]------------------------+ |
| | +-----------+ +---------+ +------------+ | |
| | | Ext. Name | | Address | | Function   | | |
| | | Resolver  | | Mapper  | | Mapper     | | |
| | +-----------+ +---------+ +------------+ | |
| +------------------------------------------+ |
| +--------------------+ +-------------------+ |
| |                    | |                   | |
| |    TCP(UDP)/IPv4   | |   TCP(UDP)/IPv6   | |
| |                    | |                   | |
| +--------------------+ +-------------------+ |
+----------------------------------------------+
```

```
+-------------------------------------------------------------+
|  +-------------------------------------------------------+  |
|  | IPv4 applications                                     |  |
|  +-------------------------------------------------------+  |
|  +-------------------------------------------------------+  |
|  | TCP/IPv4                                              |  |
|  |   +---------------------------------------------------+  |
|  |   |  +-----------+    +---------+  +---------------+     |
|  |   |  | extension |    | address |  |  translator   |     |
|  |   |  | name      |    | mapper  |  +---------------+     |
|  |   |  | resolver  |    |         |  +---------------+     |
|  |   |  |           |    |         |  |     IPv6      |     |
|  +---+  +-----------+    +---------+  +---------------+     |
|  +-------------------------------------------------------+  |
|  |                  Network card drivers                 |  |
|  +-------------------------------------------------------+  |
+-------------------------------------------------------------+
+-------------------------------------------------------------+
|                        Network cards                        |
+-------------------------------------------------------------+
```

- No changes to BIS and BIA !
  - Except BIS’s Name Resolver is now called extension name resolver as in BIA (just sync)

Components 1/2

- Function mapper
  - Intercepts IPv4 socket calls and uses IPv6 instead
- Extension Name Resolver
  - Creates synthetic IPv4 addresses representing IPv6 destinations
  - May use destination’s true IPv4 address if available
  - Catches reverse DNS lookup queries done for synthetic IPv4 addresses
  - *Note: Can be implemented in user space by setting 127.0.0.1 as host’s DNS server address (loopback)*

Components 2/2

- Address mapper
  - Manages local IPv4 address allocation (RFC1918)
  - Manages mappings between locally generated or true IPv4 addresses and IPv6 addresses
- Translator
  - Uses newly defined protocol translator for IPv4->IPv6 (draft-ietf-behave-v6v4-xlate)

Proposal for behave WG

- Include the Bump-in-the-Host in the new charter
- Make it Standards Track document
- Be clear that this is not a general solution, but a point solution for those scenarios and applications that benefit of this