# IP/ICMP Translation Algorithm (IIT)

Xing Li, Congxiao Bao, Fred Baker
2008-11-17

Abstract

- This document specifies an update to the Stateless IP/ICMP Translation Algorithm described in RFC 2765. The algorithm translates between IPv4 and IPv6 packet headers (including ICMP headers).

- The specification addresses both a stateful and a stateless mode.

  – In the stateless mode, translation information is carried in the address itself, permitting both IPv4->IPv6 and IPv6->IPv4 session establishment.

  – In the stateful mode, translation state is maintained between IPv4 address/port pairs and IPv6 address/port pairs, enabling IPv6 systems to open sessions with IPv4 systems.

- The choice of operational mode is made by the operator deploying the network. This document confines itself to the actual translation.

Scenario

IPv4-only — **Xlate** / **DNS** — **IPv6-only**

IPv4 packets   IPv6 packets
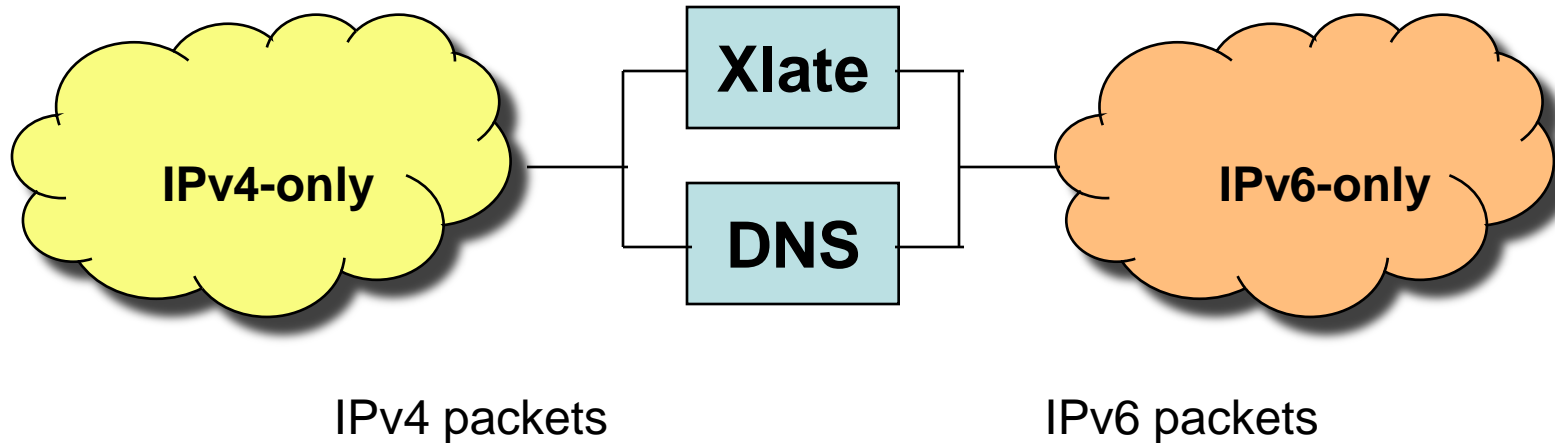
- The IPv4 packets arrived in the IP/ICMP translator will be translated to IPv6 packets.
  - The translator translates the packet headers from IPv4 to IPv6 and translate the addresses in those headers from IPv4 addresses to IPv6 addresses.
- The IPv6 packets arrived in the IP/ICMP translator will be translated to IPv4 packets.
  - The translator translates the packet headers from IPv6 to IPv4 and translate the addresses in those headers from IPv6 addresses to IPv4 addresses.
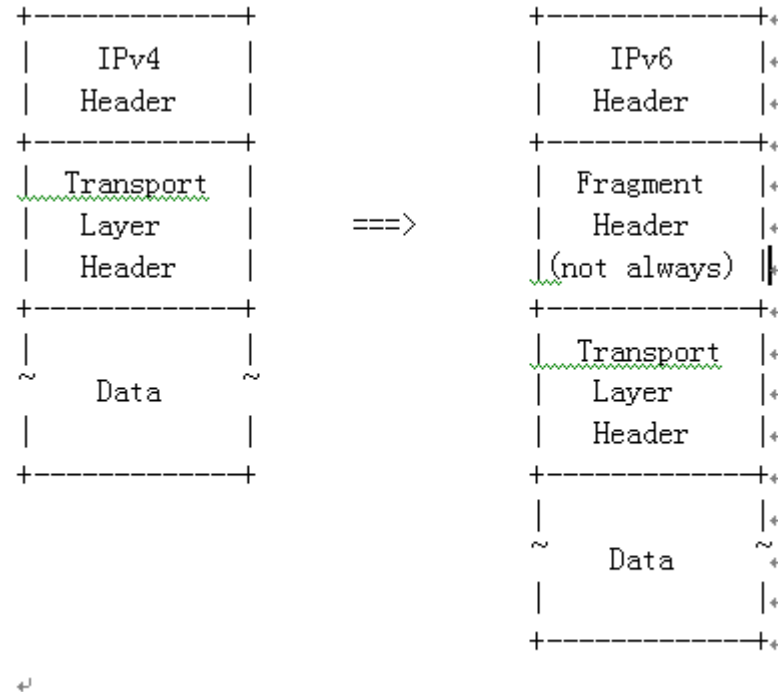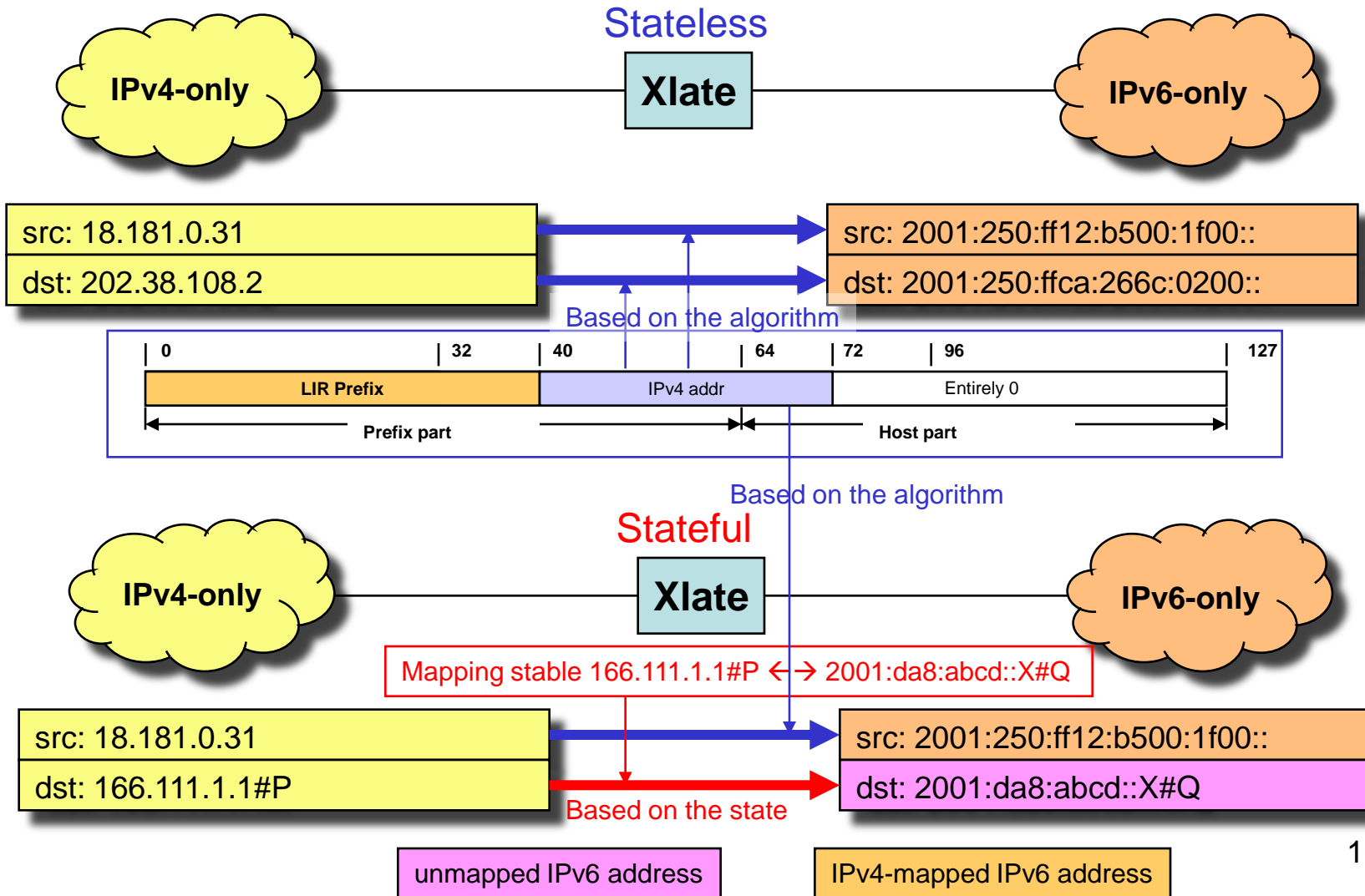
Translating from IPv4 to IPv6


```
+-------------+                  +-------------+
|    IPv4     |                  |    IPv6     |
|   Header    |                  |   Header    |
+-------------+                  +-------------+
|  Transport  |                  |  Fragment   |
|   Layer     |      ===>        |   Header    |
|   Header    |                  | (not always)|
+-------------+                  +-------------+
|             |                  |  Transport  |
|             |                  |   Layer     |
~    Data     ~                  |   Header    |
|             |                  +-------------+
+-------------+                  |             |
                                 ~    Data     ~
                                 |             |
                                 +-------------+
```

Figure 3: IPv4-to-IPv6 Translation

8

4→6 Address translation

- Source Address:
  - In both the stateless mode and the stateful mode, the IPv6 address is the IPv4-mapped IPv6 address, which embeds the IPv4 address as specified in [Framework].
- Destination Address:
  - In stateless mode, which is to say that if the IPv4 destination address is within the range of the stateless translation prefix, the IPv6 address is the IPv4-mapped IPv6 address, which embeds the IPv4 address as specified in [Framework].
  - In stateful mode, which is to say that if the IPv4 destination address is within the range of the stateful translation prefix, the IPv6 address and transport layer destination port corresponding to the IPv4 destination address and destination port are derived from the database reflecting current session state in the translator.

4→6 Address translation examples

Stateless

IPv4-only —— **Xlate** —— **IPv6-only**

| src: 18.181.0.31 | → | src: 2001:250:ff12:b500:1f00:: |
| dst: 202.38.108.2 | → | dst: 2001:250:ffca:266c:0200:: |

Based on the algorithm

0	32	40	64	72	96	127
LIR Prefix		IPv4 addr		Entirely 0		

Prefix part ←——————→ Host part

Based on the algorithm

Stateful

IPv4-only —— **Xlate** —— **IPv6-only**

Mapping stable 166.111.1.1#P ←→ 2001:da8:abcd::X#Q

| src: 18.181.0.31 | → | src: 2001:250:ff12:b500:1f00:: |
| dst: 166.111.1.1#P | → | dst: 2001:da8:abcd::X#Q |

Based on the state

unmapped IPv6 address   IPv4-mapped IPv6 address

10

Knowing when to Translate

- The translator is assumed to know the address mapping algorithm and the IPv4 prefix range of the stateless translation in the stateless mode or the pool (s) of IPv4 address in the stateful mode.

- If the translator is implemented in a router providing both translation and normal forwarding, and the address is reachable by a more specific route without translation, the router should forward it without translating it.

- In general, however, if the IPv4 destination field contains an address that falls in these configured sets of prefixes the packet needs to be translated to IPv6.
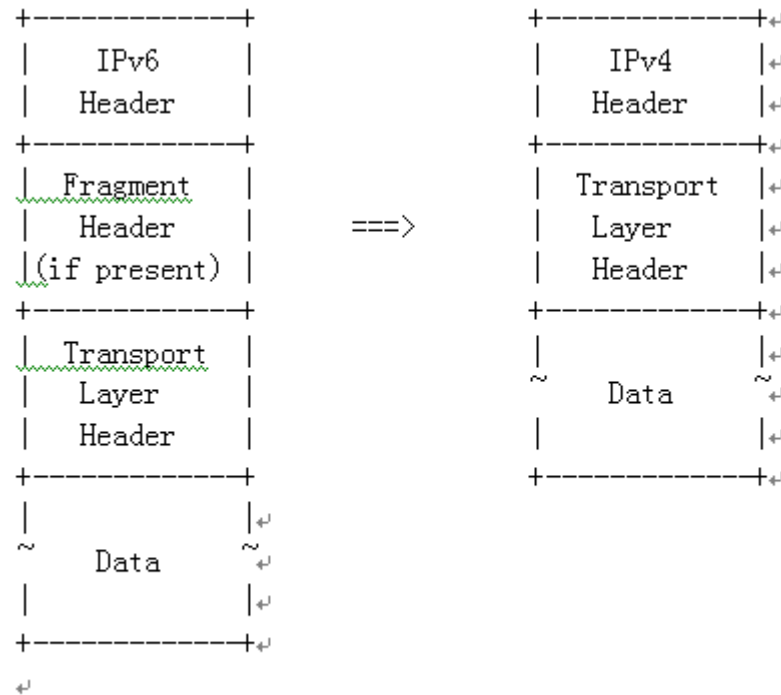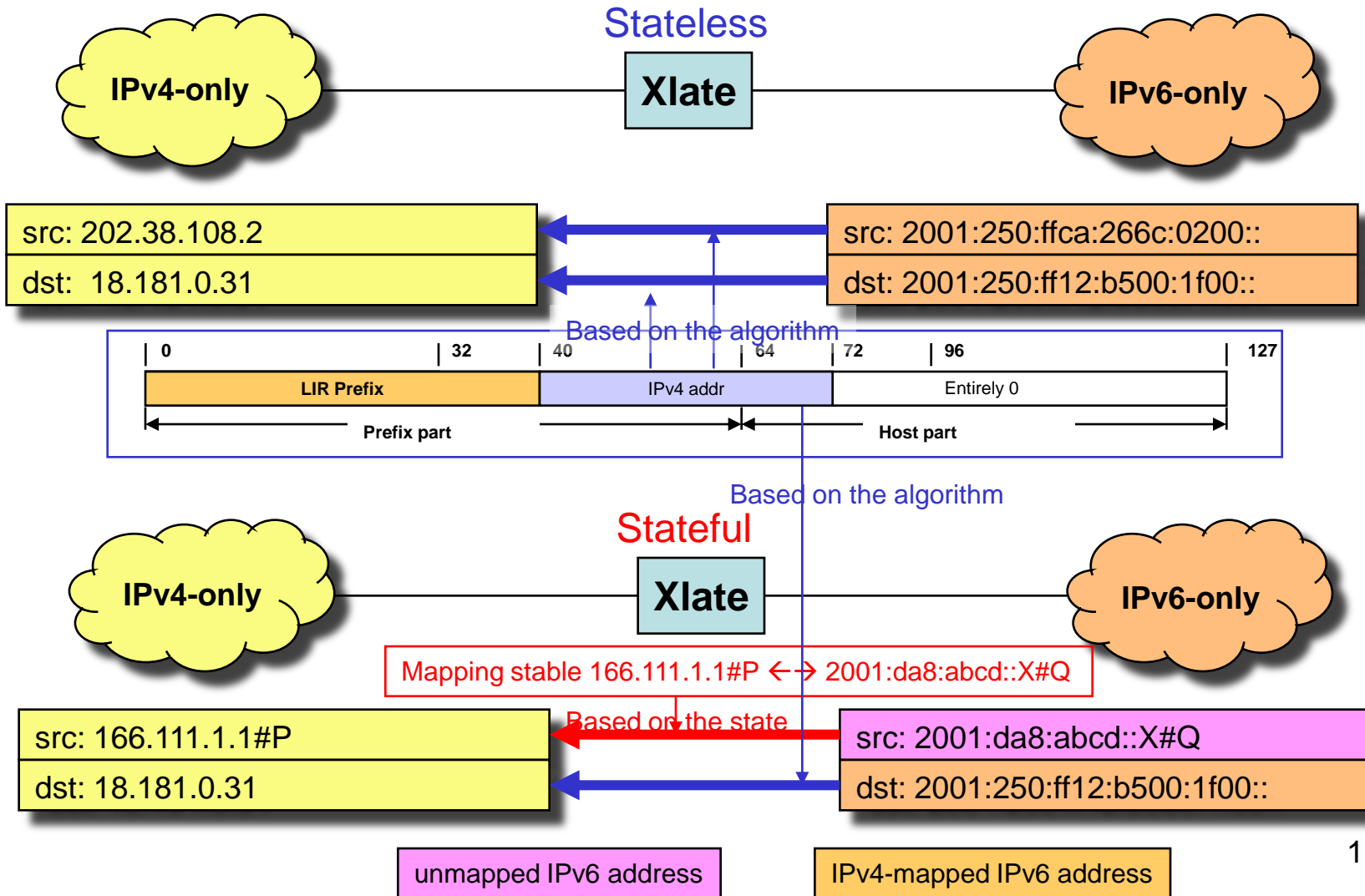
Translating from IPv6 to IPv4

```
+------------+                          +------------+
|    IPv6    |                          |    IPv4    |
|   Header   |                          |   Header   |
+------------+                          +------------+
|  Fragment  |                          |  Transport |
|   Header   |        ===>              |   Layer    |
| (if present)|                         |   Header   |
+------------+                          +------------+
|  Transport |                          |            |
|   Layer    |                          ~    Data    ~
|   Header   |                          |            |
+------------+                          +------------+
|            |
~    Data    ~
|            |
+------------+
```

Figure 5: IPv6-to-IPv4 Translation

15

6→4 Address translation

- Source Address:
  - In stateless mode, IPv6 packets that are translated always have an IPv4-mapped IPv6 address. Thus the IPv4 address is derived from the IPv4-mapped IPv6 address as specified in [Framework].
  - In stateful mode, IPv6 packets that are translated have an unmapped IPv6 address. The IPv4 source address and transport layer source port corresponding to the IPv6 source address and source port are derived from the database reflecting current session state in the translator as described in [nat64] .
- Destination Address:
  - In both the stateless mode and the stateful mode, IPv6 packets that are translated always have an IPv4-mapped IPv6 address. Thus the IPv4 address is derived from the IPv4-mapped IPv6 address as specified in [Framework].

6→4 Address translation examples

Stateless

IPv4-only — **Xlate** — IPv6-only

| src: 202.38.108.2 | src: 2001:250:ffca:266c:0200:: |
| dst: 18.181.0.31 | dst: 2001:250:ff12:b500:1f00:: |

Based on the algorithm

0	32	40	64	72	96	127
LIR Prefix		IPv4 addr		Entirely 0		

Prefix part — Host part

Based on the algorithm

Stateful

IPv4-only — **Xlate** — IPv6-only

Mapping stable 166.111.1.1#P ←→ 2001:da8:abcd::X#Q

Based on the state

| src: 166.111.1.1#P | src: 2001:da8:abcd::X#Q |
| dst: 18.181.0.31 | dst: 2001:250:ff12:b500:1f00:: |

unmapped IPv6 address   IPv4-mapped IPv6 address

17

Knowing when to Translate

- If the translator is implemented in a router providing both translation and normal forwarding, and the address is reachable by a more specific route without translation, the router should forward it without translating it.

- Otherwise, when the translator receives an IPv6 packet with an IPv4-mapped destination IPv6 address the packet will be translated to IPv4.

Host operation

- Interaction of IPv4-mapped IPv6 addresses with RFC3484 address selection
  - IPv6 systems should select source and destination addresses that are as similar as possible.
    - IPv6-only systems with IPv4-mapped IPv6 addresses to connect from their IPv4-mapped IPv6 address when communicating with IPv4-only systems. This is important, because it promotes stateless translation operation.
    - The IPv4-mapped IPv6 address systems may also find the IPv4-mappded IPv6 address pair "most similar" when communicating with other systems with IPv4-mapped IPv6 addresses.

New testing code of Xlate
http://www.ivi2.org/IVI/

IVI source code download

The IVI IPv4/IPv6 packet translation implementation as a Linux kernel patch is available below.
- IVI v0.5 kernel patch for Linux kernel 2.6.12
- IVI v0.5 kernel patch for Linux kernel 2.6.18

The IVI A/AAAA DNS proxy implementation is available below.
- IVIDNS v0.1 C code

For installing and configuration, please follow the instructions in the source code packages.

IVI test servers

- Access single-stack IPv6 server [2001:250:ffca:2672:100::] via IPv4
- Access single-stack IPv6 sever [2001:250:ffca:2672:100::] via IPv6
- Access IPv4 server (202.38.114.129) cross single-stack IPv6 network

---

IVI references

- IVI Update to SIIT and NAT-PT
- Prefix-specific and Stateless Address Mapping (IVI) for IPv4/IPv6 Coexistence and Transition

---