Framework for accessing IPv6 content for IPv4-only clients
draft-rfvlb-behave-v6-content-for-v4-clients-01

B. Rajtar, I.Farrer, A.Vizdal, X. Li, C.Bao

IETF 87, Berlin, July/August 2013

Motivation

- As more content is becoming available over IPv6, but a lot of customers are still using IPv4-only clients

- The draft describes a method for IPv4-only customers to access IPv6 content

- Intended to be used purely for simple 'NAT46 friendly' protocols (e.g. HTTP)

- The problem is based on Scenario 4 of RFC6144 and the solution uses mechanisms described in RFC6145 for translation
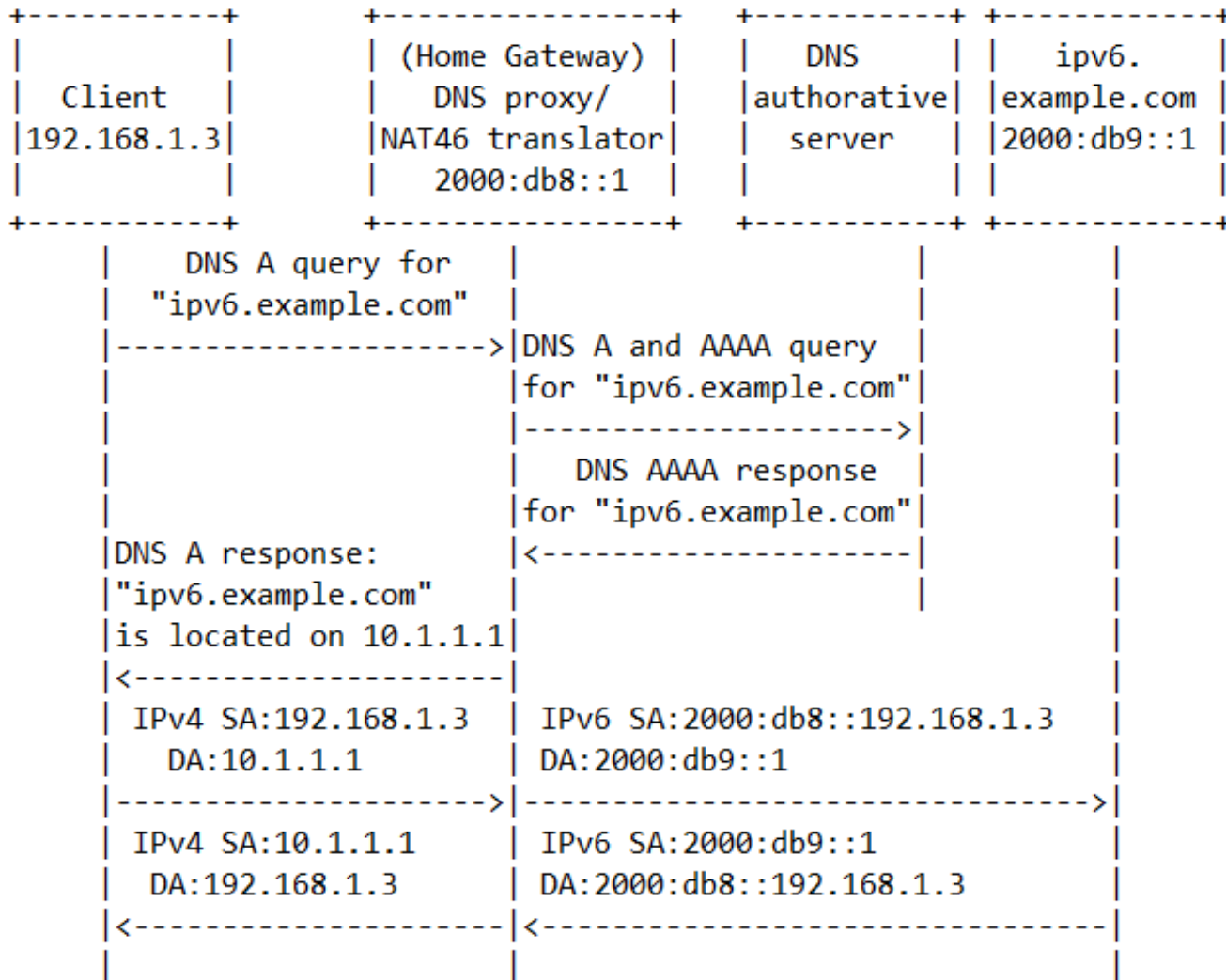
Benefits

- IPv4-only clients have access to content available only on IPv6

- More demand for IPv6 content

- In case of softwires, less capacity needed for tunnel concentrators

Functional Elements

- IPv4 Client

- DNS46 proxy – Caching name server which proxies DNSv4 queries from the client as DNSv4&6 queries.

- NAT46 translator – IPv4/IPv6 translator.

- IPv6-only server

Algorithm Description

```
+----------+        +----------------+        +----------+ +------------+
|          |        | (Home Gateway) |        |   DNS    | |   ipv6.    |
|  Client  |        |   DNS proxy/   |        |authorative| |example.com |
|192.168.1.3|       |NAT46 translator|        |  server  | |2000:db9::1 |
|          |        |   2000:db8::1  |        |          | |            |
+----------+        +----------------+        +----------+ +------------+
     |      DNS A query for    |                   |            |
     |    "ipv6.example.com"   |                   |            |
     |------------------------>|DNS A and AAAA query|            |
     |                         |for "ipv6.example.com"|          |
     |                         |-------------------->|            |
     |                         |  DNS AAAA response  |            |
     |                         |for "ipv6.example.com"|          |
     |DNS A response:          |<--------------------|            |
     |"ipv6.example.com"       |                   |            |
     |is located on 10.1.1.1|                      |            |
     |<------------------------|                   |            |
     | IPv4 SA:192.168.1.3   | IPv6 SA:2000:db8::192.168.1.3     |
     |   DA:10.1.1.1         | DA:2000:db9::1                    |
     |------------------------>|------------------------------->|
     | IPv4 SA:10.1.1.1      | IPv6 SA:2000:db9::1              |
     |   DA:192.168.1.3      | DA:2000:db8::192.168.1.3         |
     |<------------------------|<-------------------------------|
     |                         |                   |            |
```

5

Algorithm Description

1.   The customer types in "ipv6.example.com" into his web browser and initiaties the request for the web page.

2.   The client operating system initiates a DNS query for "ipv6.example.com". Since the client uses IPv4, the query is for an A record.

3.   The DNS proxy receives the A record query and assumes the client is not IPv6 capable. Therefore, it initiates a DNS query for A and AAAA records for "ipv6.example.com" to the authorative DNS server.

4.   If a DNS response is received with only an AAAA record, the DNS proxy assumes that the server is IPv6-only.

5.   As a response to the client, the proxy returns a fake A record for "ipv6.example.com" pointing at an un-used IPv4 address from the private address space.

6.   The private IPv4 address and the resolved IPv6 address of "ipv6.example.com" must be kept in the translation table of the NAT46 translator.

7.   All IPv4 traffic from the client to "ipv6.example.com" will be translated to IPv6 as described in [RFC6145]. The destination address of the translated IPv6 packet will be the resolved AAAA record of "ipv6.example.com", while the source IPv6 address will be created according to [RFC6052].

8.   Return IPv6 traffic will be translated by the same device as the outgoing traffic, using IPv6 to IPv4 translation analogous to the previous step.

Differences from NAT-PT

- DNS-ALG not needed
- No full state keeping needed
- Based on configurable policy, not all protocols are NAT aware
- Intended for NAT-agnostic protocols (i.e. basic web browsing, FTP, etc.)

Next steps

- Find interested parties and reviewers for further development
- Extend to function with Dual stack servers (incl. Happy eyeballs interworking)
- Agree the draft across the WG
- Align work with other similar drafts
- Possibly call for WG adoption