v6ops
Internet-Draft
Intended status: Informational
Expires: February 15, 2019

J. Palet Martinez
The IPv6 Company
August 14, 2018

NAT64/464XLAT Deployment Guidelines in Operator and Enterprise Networks
draft-ietf-v6ops-nat64-deployment-02

Abstract

This document describes how NAT64 and 464XLAT can be deployed in an IPv6 network, whether cellular ISP, broadband ISP, or enterprise and the issues to be considered when having an IPv6-only access link, regarding: a) DNS64, b) applications or devices that use literal IPv4 addresses or non-IPv6 compliant APIs, and c) IPv4-only hosts or applications.

Status of This Memo

This Internet-Draft will expire on February 15, 2019.

Copyright Notice

Table of Contents

1. Introduction

NAT64 ([RFC6146]) describes a stateful IPv6 to IPv4 translation mechanisms, which allows IPv6-only hosts to communicate with IPv4-only servers using unicast UDP, TCP or ICMP, by means of IPv4 public addresses sharing (assigned to the translator), among multiple IPv6-only hosts.

The translation of the packet headers is done using the IP/ICMP translation algorithm defined in [RFC7915] and algorithmically translating the IPv4 addresses to IPv6 addresses following [RFC6052].

DNS64 ([RFC6147]), is in charge of the synthesis of AAAA records from the A records, so only works for applications making use of DNS, avoiding changes in both, the IPv6-only hosts and the IPv4-only server, so they can use a NAT64. As discussed in Section 5.5 of [RFC6147], a security-aware and validating host has to perform the DNS64 function locally.

However, the use of NAT64 and/or DNS64 present three issues:

a. Because DNS64 ([RFC6147]) modifies DNS answers, and DNSSEC is designed to detect such modifications, DNS64 ([RFC6147]) can potentially break DNSSEC, depending on a number of factors, such as the location of the DNS64 function (at a DNS server or validator, at the end host, ...), how it has been configured, if the end-hosts is validating, etc.

b. Because the need of using DNS64 ([RFC6147]) or an alternative "host/application built-in" mechanism for address synthesis, there is a major issue for NAT64 ([RFC6146]), as doesn't work when literal addresses or non-IPv6 compliant APIs are being used.

c. NAT64 alone, doesn't provide a solution for IPv4-only hosts or applications located within a network which are connected to a service provider IPv6-only access, as it was designed for a very specific scenario ([RFC6144], Section 2.1).

The same issues are true if part of, for example, an enterprise network, is connected to other parts of the same network or third party networks by means of IPv6-only connectivity. This applies to many other similar cases.

According to that, across this document, the use of "operator network" is interchangeable with equivalent cases of enterprise (or similar) networks.

An analysis of stateful IPv6/IPv6 mechanisms is provided in

[RFC6889].

This document looks into different possible NAT64 ([RFC6146]) deployment scenarios, including IPv4-IPv6-IPv4 and similar ones, which were not documented by [RFC6144], such as 464XLAT ([RFC6877]), in operator (broadband and cellular) and enterprise networks, and provides guidelines to avoid the above-mentioned issues.

Towards that, this document first looks into the possible NAT64 deployment scenarios (split in "known to work" and "known to work under special conditions"), providing a quick and generic comparison table among them. Then the document describes the issues that an operator need to understand on different matters that will allow to define what is the best approach/scenario for each specific network case. A summary provides some recommendations and decision points and then a clarification of the usage of this document for enterprise networks is provided. Finally, an annex provides an example of a broadband deployment using 464XLAT.

The target deployment scenarios in this document may be covered as well by other IPv4-as-a-Service transition mechanisms. It is out of scope of this document to provide a comparison among them.

[RFC7269] provides additional information about NAT64 deployment options and experiences.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. NAT64 Deployment Scenarios

Section 7 of DNS64 ([RFC6147]), provides 3 scenarios, looking at the location of the DNS64. However, since the publication of that document, there are new possible scenarios and NAT64 use cases that need to be considered now, despite they were specifically ruled out of the original NAT64/DNS64 work.

Consequently, the perspective in this document is to broaden those scenarios, including a few new ones. However, in order to be able to reduce the number of possible cases, we work under the assumption that typically, the service provider wants to make sure that all the customers have a service without failures. This means considering the worst possible case:

a. There are hosts that will be validating DNSSEC.

b. Literal addresses and non-IPv6 compliant APIs are being used.

c. There are IPv4-only hosts or applications beyond the IPv6-only link (e.g., tethering in cellular networks).

We use a common set of possible "participant entities":

1. An IPv6-only access network (IPv6).

2. An IPv4-only remote network/server/services (IPv4).

3. The NAT64 function (NAT64) in the service provider.

4. The DNS64 function (DNS64) in the service provider.

5. An external service provider offering the NAT64 and/or the DNS64 function (extNAT64/extDNS64).

6. 464XLAT customer side translator (CLAT).

We split the possible scenarios in two general categories:

1. Known to work.

2. Known to work under special conditions.

3.1. Known to Work

The scenarios in this category are known to work. Each one may have different pros and cons, and in some cases the trade-offs, maybe acceptable for some operators.

3.1.1. Service Provider NAT64 with DNS64

In this scenario, the service provider offers both, the NAT64 and the DNS64 functions.

This is probably the most common scenario, however also may have the implications related the DNSSEC.

This scenario also may fail to solve the issue of literal addresses or non-IPv6 compliant APIs, as well as the issue of IPv4-only hosts or applications inside the IPv6-only access network.

```
           +----------+        +----------+        +----------+
           |          |        |  NAT64   |        |          |
           |   IPv6   +--------+    +     +--------+   IPv4   |
           |          |        |  DNS64   |        |          |
           +----------+        +----------+        +----------+
```

Figure 1: NAT64 with DNS64

A totally equivalent scenario will be if the service provider offers only the DNS64 function, and the NAT64 function is provided by an outsourcing agreement with an external provider. All the considerations in the previous paragraphs of this section are the same for this sub-case.

```
                               +----------+
                               |          |
                               | extNAT64 |
                               |          |
                               +----+-----+
                                    |
                                    |
           +----------+        +----+-----+        +----------+
           |          |        |          |        |          |
           |   IPv6   +--------+  DNS64   +--------+   IPv4   |
           |          |        |          |        |          |
           +----------+        +----------+        +----------+
```

Figure 2: NAT64 in external service provider

As well, is equivalent to the scenario where the outsourcing agreement with the external provider is to provide both the NAT64 and DNS64 functions. Once more, all the considerations in the previous paragraphs of this section are the same for this sub-case.

```
                               +----------+
                               | extNAT64 |
                               |    +     |
                               | extDNS64 |
                               +----+-----+
                                    |
           +----------+             |             +----------+
           |          |             |             |          |
           |   IPv6   +-------------+-------------+   IPv4   |
           |          |                           |          |
           +----------+                           +----------+
```

Figure 3: NAT64 and DNS64 in external provider

One more equivalent scenario will be if the service provider offers the NAT64 only, and the DNS64 function is from an external provider with or without a specific agreement among them. This is a scenario already feasible today, as several "global" service providers provide free DNS64 services and users often configure manually their DNS. This will only work if both the NAT64 and the DNS64 are using the WKP (Well-Known Prefix) or the same NSP (Network-Specific Prefix). All the considerations in the previous paragraphs of this section are the same for this sub-case.

Of course, if the external DNS64 is agreed with the service provider, then we are in the same case as in the previous ones already depicted in this scenario.

```
                            +----------+
                            |          |
                            | extDNS64 |
                            |          |
                            +----+-----+
                                 |
                                 |
        +----------+        +----+-----+        +----------+
        |          |        |          |        |          |
        |   IPv6   +--------+  NAT64   +--------+   IPv4   |
        |          |        |          |        |          |
        +----------+        +----------+        +----------+
```

Figure 4: NAT64; DNS64 by external provider

3.1.2. Service Provider offering 464XLAT, with DNS64

464XLAT ([RFC6877]) describes an architecture that provides IPv4 connectivity across a network, or part of it, when it is only natively transporting IPv6.

In order to do that, 464XLAT ([RFC6877]) relies on the combination of existing protocols:

1. The customer-side translator (CLAT) is a stateless IPv4 to IPv6 translator (NAT46) ([RFC7915]) implemented in the end-user device or CE, located at the "customer" edge of the network.

2. The provider-side translator (PLAT) is a stateful NAT64 ([RFC6146]), implemented typically at in the operator network.

3. Optionally, DNS64 ([RFC6147]), may allow an optimization: a single translation at the NAT64, instead of two translations (NAT46+NAT64), when the application at the end-user device

supports IPv6 DNS (uses AAAA RR).

Note that even in the 464XLAT ([RFC6877]) terminology, the provider-side translator is referred as PLAT, for simplicity and uniformity, in this document is always referred as NAT64.

In this scenario the service provider deploys 464XLAT with DNS64.

As a consequence, the DNSSEC issues remain, unless the if the host is doing the address synthesis.

464XLAT ([RFC6877]) is a very simple approach to cope with the major NAT64+DNS64 drawback: Not working with applications or devices that use literal IPv4 addresses or non-IPv6 compliant APIs.

464XLAT ([RFC6877]) has been used initially in IPv6-only cellular networks. By supporting CLAT, the end-user device applications can access IPv4-only end-networks/applications, despite those applications or devices use literal IPv4 addresses or non-IPv6 compliant APIs. It is also used for tethering purposes.

In addition to that, in the same example of the cellular network above, if the User Equipment (UE) provides tethering, other devices behind it will be presented with a traditional NAT44, in addition to the native IPv6 support, so clearly it allows IPv4-only hosts inside the IPv6-only access network.

Furthermore, as indicated in [RFC6877], 464XLAT can be used in broadband IPv6 network architectures, by implementing the CLAT functionality at the CE.

In order to understand all the communication possibilities, let's assume the following representation of two dual-stack peers:

```
                   +-------+     .-----.                     .-----.
                   |       |    /       \                   /       \
       .-----.     | Res./ |   /  IPv6-  \     .-----.     /  IPv4-  \
      / Local \    | SOHO  +--(   only    )---( NAT64 )---(   only    )
     /         \   |       |   \ Internet/\    `-----'     \ Internet/
    (   Dual-   )--+ IPv6  |    \       /  \              / \       /
     \  Stack  /   |  CE   |     `--+--'    \   .-----.  /   `--+--'
      \ Peer  /    | with  |        |        \ / Remote\/       |
       `-----'     | CLAT  |    +---+----+    /         \    +---+----+
                   |       |    |DNS/IPv6|   (   Dual-   )   |DNS/IPv4|
                   +-------+    |  with  |    \  Stack  /    +--------+
                                | DNS64  |     \ Peer  /
                                +--------+      `-----'
```

Representation of 464XLAT among two peers with DNS64

The possible communication paths, among the IPv4/IPv6 stacks of both peers, in this case, are:

a. Local-IPv6 to Remote-IPv6: Regular DNS and native IPv6 among peers.

b. Local-IPv6 to Remote-IPv4: DNS64 and NAT64 translation.

c. Local-IPv4 to Remote-IPv6: Not possible. It is not expected that services are deployed in Internet using IPv6-only, unless there is certainty that peers will also be IPv6-capable.

d. Local-IPv4 to Remote-IPv4: Regular DNS, CLAT and NAT64 translations.

The following figures show different choices for placing the different elements.

```
           +----------+        +----------+        +----------+
           |   IPv6   |        |  NAT64   |        |          |
           |     +    +--------+    +     +--------+   IPv4   |
           |   CLAT   |        |  DNS64   |        |          |
           +----------+        +----------+        +----------+
```

Figure 5: 464XLAT with DNS64

An equivalent scenario will be if the service provider offers only the DNS64 function, and the NAT64 function is provided by an outsourcing agreement with an external provider. All the considerations in the previous paragraphs of this section are the same for this sub-case.

```
                               +----------+
                               |          |
                               | extNAT64 |
                               |          |
                               +----+-----+
                                    |
                                    |
           +----------+        +----+-----+        +----------+
           |   IPv6   |        |          |        |          |
           |     +    +--------+  DNS64   +--------+   IPv4   |
           |   CLAT   |        |          |        |          |
           +----------+        +----------+        +----------+
```

Figure 6: 464XLAT with DNS64; NAT64 in external provider

As well, is equivalent to the scenario where the outsourcing agreement with the external provider is to provide both the NAT64 and DNS64 functions. Once more, all the considerations in the previous paragraphs of this section are the same for this sub-case.

```
                               +----------+
                               | extNAT64 |
                               |    +     |
                               | extDNS64 |
                               +----+-----+
                                    |
           +----------+             |             +----------+
           |   IPv6   |             |             |          |
           |     +    +-------------+-------------+   IPv4   |
           |   CLAT   |                           |          |
           +----------+                           +----------+
```

Figure 7: 464XLAT with DNS64; NAT64 and DNS64 in external provider

3.1.3. Service Provider offering 464XLAT, without DNS64

The major advantage of this scenario, using 464XLAT without DNS64, is that the service provider ensures that DNSSEC is never broken, even in case the user modifies the DNS configuration.

In this scenario, as in the previous one, there are no issues related to IPv4-only hosts inside the IPv6-only access network, neither to the usage of IPv4 literals or non-IPv6 compliant APIs.

Let's assume the representation of two dual-stack peers as in the previous case:

```
                   +-------+     .-----.                     .-----.
                   |       |    /       \                   /       \
       .-----.     | Res./ |   /  IPv6-  \     .-----.     /  IPv4-  \
      / Local \    | SOHO  +--(   only    )---( NAT64 )---(   only    )
     /         \   |       |   \ Internet/\    `-----'     \ Internet/
    (   Dual-   )--+ IPv6  |    \       /  \              / \       /
     \  Stack  /   |  CE   |     `--+--'    \   .-----.  /   `--+--'
      \ Peer  /    | with  |        |        \ / Remote\/       |
       `-----'     | CLAT  |    +---+----+    /         \    +---+----+
                   |       |    |DNS/IPv6|   (   Dual-   )   |DNS/IPv4|
                   +-------+    +--------+    \  Stack  /    +--------+
                                               \ Peer  /
                                                `-----'
```

Representation of 464XLAT among two peers without DNS64

The possible communication paths, among the IPv4/IPv6 stacks of both peers, in this case, are:

a. Local-IPv6 to Remote-IPv6: Regular DNS and native IPv6 among peers.

b. Local-IPv6 to Remote-IPv4: Because there is no DNS64, will fall-back to d. below.

c. Local-IPv4 to Remote-IPv6: Not possible. It is not expected that services are deployed in Internet using IPv6-only, unless there is certainty that peers will also be IPv6-capable.

d. Local-IPv4 to Remote-IPv4: Regular DNS, CLAT and NAT64 translations.

It needs to be noticed that this scenario works while the local hosts/applications are dual-stack (which is the current situation), because the connectivity from a local-IPv6 to a remote-IPv4 is not possible without an AAAA synthesis. This aspect is important only when in the LANs behind the CLAT there are IPv6-only hosts and they need to communicate with remote IPv4-only hosts. However, doesn't look a sensible approach from an Operating System or application vendor perspective, to provide IPv6-only support unless, similarly to c. above, there is certainty of peers supporting IPv6 as well.

The following figures show different choices for placing the different elements.

```
           +----------+        +----------+        +----------+
           |   IPv6   |        |          |        |          |
           |    +     +--------+  NAT64   +--------+   IPv4   |
           |   CLAT   |        |          |        |          |
           +----------+        +----------+        +----------+
```

Figure 8: 464XLAT without DNS64

This is equivalent to the scenario where there is an outsourcing agreement with an external provider for the NAT64 function. All the considerations in the previous paragraphs of this section are the same for this sub-case.

```
                               +----------+
                               |          |
                               | extNAT64 |
                               |          |
                               +----+-----+
                                    |
           +----------+             |              +----------+
           |   IPv6   |             |              |          |
           |    +     +-------------+--------------+   IPv4   |
           |   CLAT   |                            |          |
           +----------+                            +----------+
```

Figure 9: 464XLAT without DNS64; NAT64 in external provider

3.2. Known to Work Under Special Conditions

The scenarios in this category are known to not work unless significant effort is devoted to solve the issues, or are intended to solve problems across "closed" networks, instead of as a general Internet access usage. In addition to the different pros, cons and trade-offs, which may be acceptable for some operators, they have implementation difficulties, as they are beyond the original expectations of the NAT64/DNS64 original intent.

3.2.1. Service Provider NAT64 without DNS64

In this scenario, the service provider offers a NAT64, however there is no DNS64 function support.

As a consequence, an IPv6 host in the IPv6-only access network, will not be able to detect the presence of DNS64 by means of [RFC7050], neither learning the IPv6 prefix to be used for the NAT64.

This can be sorted out as indicated in Section 4.1.1.

However, despite that, because the lack of the DNS64 function, the IPv6 host will not be able to obtain AAAA synthesized records, so the NAT64 becomes useless.

An exception to this "useless" scenario will be manually configure mappings between the A records of each of the IPv4-only remote hosts and the corresponding AAAA records, with the WKP (Well-Known Prefix) or NSP (Network-Specific Prefix) used by the service provider NAT64, as if they were synthesized by a DNS64.

This mapping could be done by several means, typically at the authoritative DNS server, or at the service provider resolvers by means of DNS RPZ (Response Policy Zones). The latest, may have implications in DNSSEC, if the zone is signed. Also, if the service provider is using a NSP, having the mapping at the authoritative server, will mean that may create troubles to other parties trying to use different NSP or the WKP, unless multiple DNS "views" are also being used at the authoritative servers.

Generally, the mappings alternative, will only make sense if a few set of IPv4-only remote hosts need to be accessed by a single network or reduced set of them, which support IPv6-only in the access, with some kind of mutual agreement for using this procedure, so it doesn't care if they become a trouble for other parties across Internet ("closed services").

In any case, this scenario doesn't solve the issue of literal addresses or non-IPv6 compliant APIs, neither it solves the problem of IPv4-only hosts within that IPv6-only access network.

```
        +----------+        +----------+        +----------+
        |          |        |          |        |          |
        |   IPv6   +--------+  NAT64   +--------+   IPv4   |
        |          |        |          |        |          |
        +----------+        +----------+        +----------+
```

Figure 10: NAT64 without DNS64

3.2.2. Service Provider NAT64; DNS64 in the IPv6 hosts

In this scenario, the service provider offers the NAT64, but not the DNS64. However, the IPv6 hosts have a built-in DNS64 function.

This may become common if the DNS64 function is implemented in all the IPv6 hosts/stacks, which is not the actual situation. At this way, the DNSSEC validation is performed on the A record, and then the host can use the DNS64 function so to be able to use the NAT64, without any DNSSEC issues.

This scenario fails to solve the issue of literal addresses or non-IPv6 compliant APIs, unless the IPv6 hosts also supports Happy Eyeballs v2 ([RFC8305], Section 7.1), which may solve that issue.

However, this scenario still fails to solve the problem of IPv4-only hosts or applications inside the IPv6-only access network.

```
        +----------+        +----------+        +----------+
        |   IPv6   |        |          |        |          |
        |    +     +--------+  NAT64   +--------+   IPv4   |
        |   DNS64  |        |          |        |          |
        +----------+        +----------+        +----------+
```

Figure 11: NAT64; DNS64 in IPv6 hosts

3.2.3. Service Provider NAT64; DNS64 in the IPv4-only remote network

In this scenario, the service provider offers the NAT64 only. The remote IPv4-only network offers the DNS64 function.

This is not common, and looks like doesn't make too much sense that a remote network, not deploying IPv6, is providing a DNS64 function and as in the case of the scenario depicted in Section 3.2.1, it will only work if both sides are using the WKP or the same NSP so, the same considerations apply. It can be also tuned to behave as in Section 3.1.1

This scenario still fails to solve the issue of literal addresses or non-IPv6 compliant APIs.

This scenario also fails to solve the problem of IPv4-only hosts or applications inside the IPv6-only access network.

```
        +----------+        +----------+        +----------+
        |          |        |          |        |   IPv4   |
        |   IPv6   +--------+  NAT64   +--------+    +     |
        |          |        |          |        |   DNS64  |
        +----------+        +----------+        +----------+
```

Figure 12: NAT64; DNS64 in the IPv4-only

3.3. Comparing the Scenarios

This section compares the different scenarios, including the possible variations (each one represented in the precedent sections by a different figure), looking at the following parameters:

a. DNSSEC: Are there host validating DNSSEC?

b. Literal/APIs: Are there applications using literals or non-IPv6 compliant APIs?

c. IPv4-only: Are there hosts or applications using IPv4-only?

d. Foreign DNS: Is the scenario surviving if the user changes the DNS? Note that this apply similarly to split DNS, DNS in VPNs and DNS privacy.

In the next table, the columns represent each of the scenario from the previous sections, by the Figure number. The possible values are:

- Scenario "bad" for that item.

+ Scenario "good" for that item.

Needs to be noted that in some cases "countermeasures", alternative or special configurations, may be available for the items designated as "bad", so this comparison is making a generic case, as a quick comparison guide. In some cases, a "bad" item is not necessarily a negative aspect, all it depends on the specific needs/characteristics of the network where the deployment will take place. For instance, in a network which has only IPv6-only hosts and apps using only DNS and IPv6-compliant APIs, there is no impact using only NAT64 and DNS64, but if the hosts may validate DNSSEC, that item is still relevant.

Item / Figure	1	2	3	4	5	6	7	8	9	10	11	12
DNSSEC	-	-	-	-	-	-	-	+	+	+	+	+
Literal/APIs	-	-	-	-	+	+	+	+	+	-	-	-
IPv4-only	-	-	-	-	+	+	+	+	+	-	-	-
Foreign DNS	-	-	-	-	+	+	+	+	+	-	+	-

Figure 13: Scenario Comparison

As a general conclusion, we should note that if the network must support applications using literals, non-IPv6-compliant APIs, or IPv4-only hosts or applications, only the scenarios with 464XLAT, or equivalent built-in local address synthesis features, will provide a solution. Further to that, those scenarios will also keep working if the user changes the DNS setup. Clearly also, depending on if DNS64

is used or not, DNSSEC may be broken for those hosts doing DNSSEC validation.

4. Issues to be Considered

This section reviews the different issues that an operator needs to consider towards a NAT64/464XLAT deployment, as they may bring to decision points about how to approach that deployment.

4.1. DNSSEC Considerations and Possible Approaches

As indicated in Section 8 of [RFC6147] (DNS64, Security Considerations), because DNS64 modifies DNS answers and DNSSEC is designed to detect such modifications, DNS64 can break DNSSEC.

If a device connected to an IPv6-only WAN queries for a domain name in a signed zone, by means of a recursive name server that supports DNS64, and the result is a synthesized AAAA record, and the recursive name server is configured to perform DNSSEC validation and has a valid chain of trust to the zone in question, it will cryptographically validate the negative response from the authoritative name server. This is the expected DNS64 behavior: The recursive name server actually lies to the client device. However, in most of the cases, the client will not notice it, because generally, they don't perform validation themselves and instead, rely on the recursive name servers.

A validating DNS64 resolver in fact, increase the confidence on the synthetic AAAA, as it has validated that a non-synthetic AAAA for sure, doesn't exists. However, if the client device is NAT64-oblivious (most common case) and performs DNSSEC validation on the AAAA record, it will fail as it is a synthesized record.

The best possible scenario from DNSSEC point of view is when the client requests the DNS64 server to perform the DNSSEC validation (by setting the DO bit to 1 and the CD bit to 0). In this case, the DNS64 server validates the data thus tampering may only happen inside the DNS64 server (which is considered as a trusted part, thus its likelihood is low) or between the DNS64 server and the client. All other parts of the system (including transmission and caching) are protected by DNSSEC ([Threat-DNS64]).

Similarly, if the client querying the recursive name server is another name server configured to use it as a forwarder, and is performing DNSSEC validation, it will also fail on any synthesized AAAA record.

All those considerations are extensively covered in Sections 3, 5.5

and 6.2 of [RFC6147].

The ideal solution to avoid DNSSEC issues, will be that all the signed zones also provide IPv6 connectivity, together with the corresponding AAAA records, which is out of the control of the operator needing to deploy NAT64.

An alternative solution, which was the one considered while developing [RFC6147], is that validators will be DNS64-aware, so could perform the necessary discovery and do their own synthesis. That was done under the expectation that it was sufficiently early in the validator-deployment curve that it would be ok to break certain DNSSEC assumptions for networks who were really stuck in a NAT64/ DNS64-needing world.

Previous data seems to indicate, that the figures of DNSSEC broken by using DNS64 will be around 1.7% ([About-DNS64]).

As already indicated, the scenarios in the previous section, are in fact somehow simplified, looking at the worst possible case (or saying it in a different way: "trying to look for the most perfect approach"), because breaking DNSSEC will not happen if the end-host is not doing validation, which is the case today in 1.7% of the cases. So, a decision point for the operator must depend on "do I really care for that percentage of cases or can I provide alternative solutions for them?". Some possible solutions may be taken, as depicted in the next sections.

4.1.1. Not using DNS64

The ideal solution will be to avoid using DNS64, but as already indicated this is not possible in all the scenarios.

However, not having a DNS64, means that is not possible to heuristically discover the NAT64 ([RFC7050]) and consequently, an IPv6 host in the IPv6-only access network, will not be able to detect the presence of the NAT64, neither to learn the IPv6 prefix to be used for it, unless it is configured by alternative means.

The learning of the IPv6 prefix could be solved by means of adding the relevant AAAA records to the ipv4only.arpa. zone of the service provider recursive servers, i.e., if using the WKP (64:ff9b::/96):

```
               ipv4only.arpa.  SOA     . . 0 0 0 0 0
               ipv4only.arpa.  NS      .
               ipv4only.arpa.  AAAA    64:ff9b::192.0.0.170
               ipv4only.arpa.  AAAA    64:ff9b::192.0.0.171
               ipv4only.arpa.  A       192.0.0.170
               ipv4only.arpa.  A       192.0.0.171
```

An alternative option to the above, is the use of DNS RPZ (Response Policy Zones).

One more alternative, only valid in environments with PCP support (for both the hosts or CEs and for the service provider network), to follow [RFC7225] (Discovering NAT64 IPv6 Prefixes using PCP).

Other alternatives may be available in the future, such as DHCPv6 options.

It may be convenient to support at the same time several of the approaches described, in order to ensure that clients with different ways to configure the NAT64 prefix, obtain it. This is also convenient even if DNS64 is being used.

4.1.2. DNSSEC validator aware of DNS64

In general, DNS servers with DNS64 function, by default, will not synthesize AAAA responses if the DNSSEC OK (DO) flag was set in the query. In this case, as only an A record is available, it means that the CLAT will take the responsibility, as in the case of literal IPv4 addresses, to keep that traffic flow end-to-end as IPv4, so DNSSEC is not broken. However, this will not work if a CLAT is not present as the hosts will not be able to use IPv4 (scenarios without 464XLAT).

4.1.3. Stub validator

If the DO flag is set and the client device performs DNSSEC validation, and the Checking Disabled (CD) flag is set for a query, as the DNS64 recursive server will not synthesize AAAA responses, the client could perform the DNSSEC validation with the A record and then may query the network for a NAT64 prefix ([RFC7050] or [RFC7225]) in order to synthesize the AAAA ([RFC6052]). This allows the client device to avoid using the CLAT and still use NAT64 even with DNSSEC.

If the end-host is IPv4-only, this will not work if a CLAT is not present (scenarios without 464XLAT) or the client is able to locally do the address synthesis.

Some devices/OSs may implement, instead of CLAT, a similar function by using Bump-in-the-Host ([RFC6535]), implemented as part of Happy

Eyeballs v2 (Section 7.1 of [RFC8305]). In this case, the considerations in the above paragraphs are also applicable.

4.1.4. CLAT with DNS proxy and validator

If a CE includes CLAT support and also a DNS proxy, as indicated in Section 6.4 of [RFC6877], the CE could behave as a stub validator on behalf of the client devices, following the same approach described in the precedent section (Stub validator). So, the DNS proxy actually lies to the client devices, which in most of the cases will not notice it unless they perform validation themselves. Again, this allow the client devices to avoid using the CLAT and still use NAT64 with DNSSEC.

Once more, this will not work without a CLAT (scenarios without 464XLAT).

4.1.5. ACL of clients

In cases of dual-stack clients, stub resolvers should send the AAAA queries before the A ones. So, such clients, if DNS64 is enabled, will never get A records, even for IPv4-only servers, and they may be in the path before the NAT64 and accessible by IPv4. If DNSSEC is being used for all those flows, specific addresses or prefixes can be left-out the DNS64 synthesis by means of ACLs.

Once more, this will not work without a CLAT (scenarios without 464XLAT).

4.1.6. Mapping-out IPv4 addresses

If there are well-known specific IPv4 addresses or prefixes using DNSSEC, they can be mapped-out of the DNS64 synthesis.

Even if this is not related to DNSSEC, this "mapping-out" feature is actually, quite commonly used to ensure that [RFC1918] addresses (for example used by LAN servers) are not synthesized to AAAA.

Once more, this will not work without a CLAT (scenarios without 464XLAT).

4.2. DNS64 and Reverse Mapping

When a client device, using a name server configured to perform DNS64, tries to reverse-map a synthesized IPv6 address, the name server responds with a CNAME record pointing the domain name used to reverse-map the synthesized IPv6 address (the one under ip6.arpa), to the domain name corresponding to the embedded IPv4 address (under in-

addr.arpa).

This is the expected behavior, so no issues to be considered regarding DNS reverse mapping.

4.3. Using 464XLAT with/without DNS64

In the case the client device is IPv6-only (either because the stack is IPv6-only, or because it is connected via an IPv6-only LAN) and the remote server is IPv4-only (either because the stack is IPv4-only, or because it is connected via an IPv4-only LAN), only NAT64 combined with DNS64 will be able to provide access among both. Because DNS64 is then required, DNSSEC validation will be only possible if the recursive name server is validating the negative response from the authoritative name server and the client is not performing validation.

However, when the client device is dual-stack and/or connected in a dual-stack LAN by means of a CLAT (or has the built-in CLAT), DNS64 is an option.

1. With DNS64: If DNS64 is used, most of the IPv4 traffic (except if using literal IPv4 addresses or non-IPv6 compliant APIs) will not use the CLAT, so will use the IPv6 path and only one translation will be done at the NAT64. This may break DNSSEC, unless measures as described in the precedent sections are taken.

2. Without DNS64: If DNS64 is not used, all the IPv4 traffic will make use of the CLAT, so two translations are required (NAT46 at the CLAT and NAT64 at the PLAT), which adds some overhead in terms of the extra NAT46 translation, however avoids the AAAA synthesis and consequently will never break DNSSEC.

Note that the extra translation, when DNS64 is not used, takes place at the CLAT, which means no extra overhead for the operator, and no perceptible impact for a CE in a broadband network, while it may have some impact in a battery powered device. This cost for a battery powered device, is possibly comparable to the cost when the device is doing a local address synthesis (see Section 7.1 of [RFC8305]).

4.4. Manual Configuration of Foreign DNS

When clients, in a service provider network, use DNS servers from other networks, for example manually configured by users, they may support or not DNS64, so the considerations in Section 4.3 will apply as well.

Even in the case that the external DNS supports DNS64 function, we

may be in the situation of providing incorrect configurations parameters, for example un-matching WKP or NSP, or a case such the one described in Section 3.2.3.

A similar situation may happen in case of split DNS scenarios, for example, when using a VPN that forces all the DNS queries thru the VPN, ignoring the DNS64.

Having a CLAT and using an external DNS without DNS64, ensures that everything will work, so the CLAT must be considered as an advantage against user configuration errors.

However, it needs to be reinforced, that if there is not a CLAT (scenarios without 464XLAT), an external DNS without DNS64 support, will not only guarantee that DNSSEC is broken, but also disallow any access to IPv4-only networks, so will behave as in the Section 3.2.1.

4.5. DNS Privacy

If clients use mechanisms for DNS privacy, such as DNS over TLS ([RFC7858]), DNS over DTLS ([RFC8094]), DNS queries over HTTPS ([I-D.ietf-doh-dns-over-https]) or DNS over QUIC ([I-D.huitema-quic-dnsoquic]), as they may provide different results to the same query, it must be expected equivalent effects as described in Section 4.4.

4.6. Well-Known Prefix (WKP) vs Network-Specific Prefix (NSP)

[RFC6052] (IPv6 Addressing of IPv4/IPv6 Translators), Section 3, discusses some considerations which are useful to decide if an operator should use the WKP or an NSP.

Taking in consideration that discussion and other issues, we can summarize the possible decision points as:

a. The WKP MUST NOT be used to represent non-global IPv4 addresses. If this is required, because the network to be translated use non-global addresses then an NSP is required.

b. The WKP MAY appear in inter-domain routing tables, if the operator provides NAT64 to peers, however special considerations related to BGP filtering are then required and IPv4-embedded IPv6 prefixes longer than the WKP MUST NOT be advertised in BGP. An NSP may be a more appropriate option in those cases.

c. If several NAT64s use the same prefix, packets from the same flow may be routed to different NAT64s in case of routing changes. This can be avoided either by using different prefixes for each

NAT64, or by ensuring that all the NAT64s coordinate their state. Using an NSP could facilitate that.

d. If DNS64 is required and users may change their DNS configuration, and deliberately choose an alternative DNS64, most probably alternative DNS64 will use by default the WKP. If an NSP is used by the NAT64, the users will not be able to use the operator NAT64.

4.7. IPv4 literals and old APIs

A hosts or application using literal IPv4 addresses or older APIs, behind a network with IPv6-only access, will not work unless a CLAT (or equivalent function) is present.

A possible alternative approach is described as part of Happy Eyeballs v2 Section 7.1 ([RFC8305]), or if not supporting HEv2, directly using Bump-in-the-Host ([RFC6535]), and then a DNS64 function.

Those alternatives will solve the problem for and end-hosts, however, if that end-hosts is providing "tethering" or an equivalent service to others hosts, that need to be considered as well. In other words, in a case of a cellular network, it resolves the issue for the UE itself, but may be not for hosts behind it.

Otherwise, 464XLAT is the only valid approach to resolve this issue.

4.8. IPv4-only Hosts or Applications

An IPv4-only hosts or application behind a network with IPv6-only access, will not work unless a CLAT is present. 464XLAT is the only valid approach to resolve this issue.

4.9. CLAT Translation Considerations

As described in Section 6.3 of [RFC6877] (IPv6 Prefix Handling), if the CLAT can be configured with a dedicated /64 prefix for the NAT46 translation, then it will be possible to do a more efficient stateless translation.

However, if this dedicated prefix is not available, the CLAT will need to do a stateful translation, for example performing stateful NAT44 for all the IPv4 LAN packets, so they appear as coming from a single IPv4 address, and then in turn, stateless translated to a single IPv6 address.

One possible setup, in order to maximize the CLAT performance, is to

configure the dedicated translation prefix. This can be easily achieved automatically, if the broadband CE or end-user device is able to obtain a shorter prefix by means of DHCPv6-PD ([RFC3633]) so, the CE can use a /64 for that. This is also possible when broadband is provided by a cellular access.

The above recommendation is often not possible for cellular networks, when connecting smartphones (as UEs), as they don't use DHCPv6-PD ([RFC3633]) an instead a single /64 is provided for each PDP context and use /64 prefix sharing ([RFC6877]). So, in this case, the UEs typically have a build-in CLAT client, which is doing a stateful NAT44 before the stateless NAT46.

5. Summary of Deployment Recommendations for NAT64

It can be argued that none of the possible transition mechanisms is perfect, and somehow, we may consider that actually this is a good thing as a way to push for the IPv6 deployment, or otherwise, it may be further delayed, with clear undesirable effects for the global Internet.

However, for an operator, being in business means minimizing the adverse transition effects, and provide the most perfect one reasonably balanced with cost (CAPEX/OPEX), and at the same time looking for a valid long-term vision.

NAT64/464XLAT has demonstrated to be a valid choice in several scenarios (IPv6-IPv4 and IPv4-IPv6-IPv4), with hundreds of millions of users, offering different choices of deployment, depending on each network case, needs and requirements.

Depending on those requirements, DNS64 may be a required function, while in other cases the adverse effects may be counterproductive. Similarly, in some cases NAT64, together with DNS64, may be a valid solution, when for sure there is no need to support hosts or applications which are IPv4-only (Section 4.7 and Section 4.8). However, in other cases the limitations they have, may suggest the operator to look into 464XLAT as a more complete solution.

Service providers willing to deploy NAT64, need to take into account the considerations of this document in order to better decide what is more appropriate for their own specific case.

In the case of broadband managed networks (CE provided or suggested/ supported by the operator), in order to fully support the actual user needs (IPv4-only devices and applications, usage of literals and old APIs), they should consider the 464XLAT scenario and in that case, must support the customer-side translator (CLAT).

If the operator offers DNS services, in order to increase performance by reducing the double translation for all the IPv4 traffic, they may support DNS64 and avoid, as much as possible, breaking DNSSEC. In this case, if the DNS service is offering DNSSEC validation, then it must be in such way that it is aware of the DNS64. This is considered the simpler and safer approach, and may be combined as well with the other possible recommendations described in this document:

- DNS infrastructure MUST be aware of DNS64 (Section 4.1.2).

- Devices running CLAT SHOULD follow the indications in Section 4.1.3 (Stub validator). However, this may be out of the control of the operator.

- CEs SHOULD include a DNS proxy and validator (Section 4.1.4).

- Section 4.1.5 (ACL of clients) and Section 4.1.6 (Mapping-out IPv4 addresses) MAY be considered by each operator, depending on their own infrastructure.

This "increased performance" approach has the disadvantage of potentially breaking DNSSEC for a small percentage of validating end-hosts versus the small impact of a double translation taking place in the CE. If CE performance is not an issue, which is the most frequent case, then a much safer approach is to not use DNS64 at all, and consequently, ensure that all the IPv4 traffic is translated at the CLAT (Section 4.3).

If DNS64 is not used, at least one of the alternatives described in Section 4.1.1, must be followed.

The operator need to consider that if the user can modify the DNS configuration (which most probably is impossible to avoid), and instead of configuring a DNS64 choose an external regular DNS (non-DNS64), a scenario with only NAT64 will not work with any IPv4-only remote host, while it will continue working in the case of 464XLAT (Section 4.4). Same effects are to be expected if DNS privacy protocols are being used by customers (Section 4.5).

Similar considerations need to be taken regarding the usage of a NAT64 Well-Known vs Network-Specific Prefix (Section 4.6), in the sense of, if using DNS64, they must match and if the user can change the DNS configuration, they will, most probably, not.

In broadband networks, it is recommended that CEs supporting CLAT, support DHCPv6-PD ([RFC3633]) and internally reserve one /64 for the stateless NAT46 translation. The operator must ensure that the

customers get allocated prefixes shorter than /64 in order to support this optimization. One way or the other, this is not impacting the performance of the operator network.

As indicated in Section 7 of [RFC6877] (Deployment Considerations), operators may follow those suggestions in order to take advantage of traffic engineering requirements.

In the case of cellular networks, the considerations regarding DNSSEC may appear as out-of-scope, because UEs OSs, commonly don't support DNSSEC, however applications running on them may do, or it may be an OS "built-in" support in the future. Moreover, if those devices offer tethering, other client devices may be doing the validation, hence the relevance of a proper DNSSEC support by the operator network.

Furthermore, cellular networks supporting 464XLAT ([RFC6877]) and "Discovery of the IPv6 Prefix Used for IPv6 Address Synthesis" ([RFC7050]), allow a progressive IPv6 deployment, with a single APN supporting all types of PDP context (IPv4, IPv6, IPv4v6), in such way that the network is able to automatically serve all the possible combinations of UEs.

One last consideration is that many networks may have different scenarios at the same time, for example, customers requiring 464XLAT, others not requiring it, customers requiring DNS64, others not, etc. In general, the different issues and approaches described in this document can be implemented at the same time for different customers or parts of the network, so not representing any problem for complex cases.

Finally, if the operator chooses to provide validation for the DNS64 prefix discovery, it must follow the advice from Section 3.1. of [RFC7050] (Validation of Discovered Pref64::/n).

6. Deployment of NAT64 in Enterprise Networks

The recommendations of this document can be used as well in enterprise networks, campus and other similar scenarios, when the NAT64 (and/or DNS64) are under the control of that network, and for whatever reasons, there is a need to provide "IPv6-only access" to any part of that network or it is IPv6-only connected to third party networks.

An example of that is the IETF meetings network itself, where a NAT64 and DNS64 are provided, presenting in this case the same issues as per Section 3.1.1. If there is a CLAT in the IETF network, then there is no need to use DNS64 and it falls under the considerations

of Section 3.1.3. Both scenarios have been tested and verified already in the IETF network itself.

Next figures are only meant to represent a few of the possible scenarios, not pretending to be the only ones that are feasible.

The following figure provides an example of and IPv6-only enterprise network connected with dual-stack to Internet and using local NAT64 and DNS64.

```
       +----------------------------------+
       |       Enterprise Network         |
       | +----------+        +----------+ |       +----------+
       | |   IPv6   |        |  NAT64   | |       |   IPv4   |
       | |   only   +--------+    +     | +-------+     +    |
       | |   LANs   |        |  DNS64   | |       |   IPv6   |
       | +----------+        +----------+ |       +----------+
       +----------------------------------+
```

Figure 14: IPv6-only enterprise with NAT64 and DNS64

The following figure provides an example of dual-stack enterprise network connected with dual-stack to Internet and using CLAT without DNS64.

```
       +----------------------------------+
       |       Enterprise Network         |
       | +----------+        +----------+ |       +----------+
       | |   IPv6   |        |          | |       |   IPv4   |
       | |     +    +--------+  NAT64   | +-------+     +    |
       | |   CLAT   |        |          | |       |   IPv6   |
       | +----------+        +----------+ |       +----------+
       +----------------------------------+
```

Figure 15: Dual-stack enterprise with CLAT without DNS64

Finally, the following figure provides an example of an IPv6-only provider with NAT64, and a dual-stack enterprise network by means of their own CLAT without DNS64.

```
         +----------------------------------+
         |       Enterprise Network         |
         | +----------+        +----------+ |        +----------+
         | |   IPv6   |        |          | |  IPv6  |          |
         | |     +    +--------+   CLAT   | +--------+   NAT64  |
         | |   IPv4   |        |          | |  only  |          |
         | +----------+        +----------+ |        +----------+
         +----------------------------------+
```

Figure 16: Dual-stack enterprise with CLAT without DNS64

7. Security Considerations

This document does not have any new specific security considerations.

8. IANA Considerations

This document does not have any new specific IANA considerations.

Note: This section is assuming that https://www.rfc-editor.org/errata/eid5152 is resolved, otherwise, this section may include the required text to resolve the issue.

9. Acknowledgements

The author would like to acknowledge the inputs of Gabor Lencse, Andrew Sullivan, Lee Howard, Barbara Stark, Fred Baker, Mohamed Boucadair and TBD ...

Conversations with Marcelo Bagnulo, one of the co-authors of NAT64 and DNS64, as well as several emails in mailing lists from Mark Andrews, have been very useful for this work.

Christian Huitema inspired working in this document by suggesting that DNS64 should never be used, during a discussion regarding the deployment of CLAT in the IETF network.

10. ANNEX A: Example of Broadband Deployment with 464XLAT

This section summarizes how an operator may deploy an IPv6-only network for residential/SOHO customers, supporting IPv6 inbound connections, and IPv4-as-a-Service (IPv4aaS) by using 464XLAT.

Note that an equivalent setup could also be provided for enterprise customers. In case they need IPv4 inbound connections, several mechanisms, depending on specific customer needs, allow that.

Conceptually, most of the operator network could be IPv6-only

(represented in the next pictures as "IPv6-only Internet"). This part of the network connects the IPv6-only subscribers (by means of IPv6-only access links), to the IPv6 upstream providers, as well as to the IPv4-Internet by means of the NAT64 (PLAT in the 464XLAT terminology).

The traffic flow from and back to the CE to services available in the IPv6 Internet (or even dual-stack remote services, when IPv6 is being used), is purely native IPv6 traffic, so no special considerations about it.

Looking at the picture from the DNS perspective, there are remote networks with are IPv4-only, and typically will have only IPv4 DNS (DNS/IPv4), or at least will be seen as that from the CE perspective. At the operator side, the DNS, as seen from the CE, is only IPv6 (DNS/IPv6) and has also a DNS64 function.

In the customer LANs side, there is actually one network, which of course could be split in different segments, and the most common setup will be those segments being dual-stack (global IPv6 addresses and [RFC1918] for IPv4, as usual in any regular residential/SOHO IPv4 network today). In the figure it is represented as tree segments, just to show that the three possible setups are valid (IPv6-only, IPv4-only and dual-stack).

```
         .-----.    +-------+     .-----.                   .-----.
        / IPv6- \   |       |    /       \                 /       \
       (  only   )--+ Res./ |   /  IPv6-  \    .-----.    /  IPv4-  \
        \ LANs  /   | SOHO  +--(   only    )--( NAT64 )--(   only    )
         `-----'    |       |   \ Internet/    `-----'    \ Internet/
         .-----.    | IPv6  |    \       /                 \       /
        / IPv4- \   |  CE   |     `--+--'                   `--+--'
       (  only   )--+ with  |        |                         |
        \ LANs  /   | CLAT  |    +---+----+                +---+----+
         `-----'    |       |    |DNS/IPv6|                |DNS/IPv4|
         .-----.    +---+---+    |  with  |                +--------+
        / Dual- \       |        | DNS64  |
       (  Stack  )------|        +--------+
        \ LANs  /
         `-----'
```

Figure 17: CE setup with built-in CLAT with DNS64

In addition to the regular CE setup, which will be typically access-technology dependent, the steps for the CLAT configuration can be summarized as:

1. Discovery of the PLAT (NAT64) prefix: It may be done using

[RFC7050], or in those networks where PCP is supported, by means of [RFC7225], or other alternatives that may be available in the future (such as DHCPv6 options).

2. If the CLAT allows stateless NAT46 translation, a /64 from the pool typically provided to the CE by means of DHCPv6-PD [RFC3633], need to be set aside for that translation. Otherwise, the CLAT is forced to perform an intermediate stateful NAT44 before the a stateless NAT46, as described in Section 4.9.

The operator network need to ensure that the correct responses are provided for the discovery of the PLAT prefix, as well as it is highly recommended follows [RIPE-690], in order to ensure that multiple /64s are available including the one needed for the NAT46 translation.

The operator need to understand other issues, described across this document, in order to take the relevant decisions. For example, if several NAT64 are needed in the context of scalability/high-availability, an NSP should be considered (Section 4.6).

More complex scenarios are possible, for example, if a network offers multiple NAT64 prefixes, destination-based NAT64 prefixes, etc.

If the operator decides not to provide DNS64, then this setup turns into the one in the following Figure. This will be also the setup that, if the user has changed the DNS and consequently is not using the operator DNS64, it will be seen from the perspective of the CE.

```
      .-----.    +-------+     .-----.                   .-----.
     / IPv6- \   |       |    /       \                 /       \
    (  only   )--+ Res./ |   /  IPv6-  \    .-----.    /  IPv4-  \
     \ LANs  /   | SOHO  +--(   only    )--( NAT64 )--(   only    )
      `-----'    |       |   \ Internet/    `-----'    \ Internet/
      .-----.    | IPv6  |    \       /                 \       /
     / IPv4- \   |  CE   |     `--+--'                   `--+--'
    (  only   )--+ with  |        |                         |
     \ LANs  /   | CLAT  |    +---+----+                +---+----+
      `-----'    |       |    |DNS/IPv6|                |DNS/IPv4|
      .-----.    +---+---+    +--------+                +--------+
     / Dual- \       |
    (  Stack  )------|
     \ LANs  /
      `-----'
```

Figure 18: CE setup with built-in CLAT without DNS64

In this case the discovery of te PLAT prefix need to be arranged as

indicated in Section 4.1.1.

In this case the CE doesn't have a built-in CLAT, or the customer can choose to setup the IPv6 operator-managed CE in bridge mode (and optionally use its own external router), or for example there is an access technology that requires some kind of media converter (ONT for FTTH, CableModem for DOCSIS, etc.), the complete setup will look as in the next figure. Obviously, there will be some intermediate configuration steps for the bridge, depending on the specific access technology/protocols, which should not modify the steps already described in the previous cases for the CLAT configuration.

```
                    +-------+     .-----.                   .-----.
                    |       |    /       \                 /       \
                    | Res./ |   /  IPv6-  \    .-----.    /  IPv4-  \
                    | SOHO  +--(   only    )--( NAT64 )--(   only    )
                    |       |   \ Internet/    `-----'    \ Internet/
                    | IPv6  |    \       /                 \       /
                    |  CE   |     `--+--'                   `--+--'
                    | Bridge|        |                         |
                    |       |    +---+----+                +---+----+
                    |       |    |DNS/IPv6|                |DNS/IPv4|
                    +---+---+    +--------+                +--------+
                        |
         .-----.    +---+---+
        / IPv6- \   |       |
       (  only   )--+ IPv6  |
        \ LANs  /   | Router|
         `-----'    |       |
         .-----.    | with  |
        / IPv4- \   | CLAT  |
       (  only   )--+       |
        \ LANs  /   |       |
         `-----'    |       |
         .-----.    +---+---+
        / Dual- \       |
       (  Stack  )------|
        \ LANs  /
         `-----'
```

Figure 19: CE setup with bridged CLAT without DNS64

It should be avoided that several routers (i.e., the operator provided CE and a downstream user provided router) enable simultaneously routing and/or CLAT, in order to avoid multiple NAT44 and NAT46 levels, as well as ensuring the correct operation of multiple IPv6 subnets, so it is suggested to use HNCP ([RFC8375]).

Note that the procedure described here for the CE setup, can be simplified if the CE follows [I-D.ietf-v6ops-transition-ipv4aas].

11. ANNEX B: CLAT Implementation

A CLAT CE implementation basically requires support of [RFC7915] for the NAT46 functionality, [RFC7050] for the PLAT prefix discovery (and/or [RFC7225] for PCP), and if stateless NAT46 is supported, mechanisms to ensure that multiple /64 are available, such as DHCPv6-PD [RFC3633].

There are several OpenSource implementations of CLAT, such as:

Android: https://github.com/ddrown/android_external_android-clat.

Linux: https://github.com/toreanderson/clatd.

OpenWRT: https://github.com/openwrt-routing/packages/blob/master/nat46/files/464xlat.sh.

VPP: https://git.fd.io/vpp/tree/src/plugins/nat.

12. ANNEX C: Benchmarking

Several documents provide references to benchmarking, for example in the case of DNS64, [DNS64-Benchm].

13. ANNEX D: Changes from -00 and -01

Section to be removed for WGLC. Significant updates are:

1. Text changes across all the document.

14. References

14.1. Normative References

[RFC1918] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, DOI 10.17487/RFC1918, February 1996, <https://www.rfc-editor.org/info/rfc1918>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3633] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, DOI 10.17487/RFC3633, December 2003, <https://www.rfc-editor.org/info/rfc3633>.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, DOI 10.17487/RFC6052, October 2010, <https://www.rfc-editor.org/info/rfc6052>.

[RFC6144] Baker, F., Li, X., Bao, C., and K. Yin, "Framework for IPv4/IPv6 Translation", RFC 6144, DOI 10.17487/RFC6144, April 2011, <https://www.rfc-editor.org/info/rfc6144>.

[RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, DOI 10.17487/RFC6146, April 2011, <https://www.rfc-editor.org/info/rfc6146>.

[RFC6147] Bagnulo, M., Sullivan, A., Matthews, P., and I. van Beijnum, "DNS64: DNS Extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", RFC 6147, DOI 10.17487/RFC6147, April 2011, <https://www.rfc-editor.org/info/rfc6147>.

[RFC6535] Huang, B., Deng, H., and T. Savolainen, "Dual-Stack Hosts Using "Bump-in-the-Host" (BIH)", RFC 6535, DOI 10.17487/RFC6535, February 2012, <https://www.rfc-editor.org/info/rfc6535>.

[RFC6877] Mawatari, M., Kawashima, M., and C. Byrne, "464XLAT: Combination of Stateful and Stateless Translation", RFC 6877, DOI 10.17487/RFC6877, April 2013, <https://www.rfc-editor.org/info/rfc6877>.

[RFC6889] Penno, R., Saxena, T., Boucadair, M., and S. Sivakumar, "Analysis of Stateful 64 Translation", RFC 6889, DOI 10.17487/RFC6889, April 2013, <https://www.rfc-editor.org/info/rfc6889>.

[RFC7050] Savolainen, T., Korhonen, J., and D. Wing, "Discovery of the IPv6 Prefix Used for IPv6 Address Synthesis", RFC 7050, DOI 10.17487/RFC7050, November 2013, <https://www.rfc-editor.org/info/rfc7050>.

[RFC7225] Boucadair, M., "Discovering NAT64 IPv6 Prefixes Using the Port Control Protocol (PCP)", RFC 7225, DOI 10.17487/RFC7225, May 2014, <https://www.rfc-editor.org/info/rfc7225>.

[RFC7915] Bao, C., Li, X., Baker, F., Anderson, T., and F. Gont, "IP/ICMP Translation Algorithm", RFC 7915, DOI 10.17487/RFC7915, June 2016, <https://www.rfc-editor.org/info/rfc7915>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8305] Schinazi, D. and T. Pauly, "Happy Eyeballs Version 2: Better Connectivity Using Concurrency", RFC 8305, DOI 10.17487/RFC8305, December 2017, <https://www.rfc-editor.org/info/rfc8305>.

[RFC8375] Pfister, P. and T. Lemon, "Special-Use Domain 'home.arpa.'", RFC 8375, DOI 10.17487/RFC8375, May 2018, <https://www.rfc-editor.org/info/rfc8375>.

14.2. Informative References

[About-DNS64]
J. Linkova, "Let's talk about IPv6 DNS64 & DNSSEC", 2016, <https://blog.apnic.net/2016/06/09/lets-talk-ipv6-dns64-dnssec/>.

[DNS64-Benchm]
G. Lencse and Y. Kadobayashi, "Benchmarking DNS64 Implementations: Theory and Practice", Computer Communications (Elsevier), vol. 127, no. 1, pp. 61-74, DOI 10.1016/j.comcom.2018.05.005, September 2018.

[I-D.huitema-quic-dnsoquic]
Huitema, C., Shore, M., Mankin, A., Dickinson, S., and J. Iyengar, "Specification of DNS over Dedicated QUIC Connections", draft-huitema-quic-dnsoquic-05 (work in progress), June 2018.

[I-D.ietf-doh-dns-over-https]
Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", draft-ietf-doh-dns-over-https-12 (work in progress), June 2018.

[I-D.ietf-v6ops-transition-ipv4aas]
Palet, J., Liu, H., and M. Kawashima, "Requirements for IPv6 Customer Edge Routers to Support IPv4 Connectivity as-a-Service", draft-ietf-v6ops-transition-ipv4aas-07 (work in progress), August 2018.

[RFC7269] Chen, G., Cao, Z., Xie, C., and D. Binet, "NAT64 Deployment Options and Experience", RFC 7269, DOI 10.17487/RFC7269, June 2014, <https://www.rfc-editor.org/info/rfc7269>.

[RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <https://www.rfc-editor.org/info/rfc7858>.

[RFC8094] Reddy, T., Wing, D., and P. Patil, "DNS over Datagram Transport Layer Security (DTLS)", RFC 8094, DOI 10.17487/RFC8094, February 2017, <https://www.rfc-editor.org/info/rfc8094>.

[RIPE-690]
RIPE, "Best Current Operational Practice for Operators: IPv6 prefix assignment for end-users - persistent vs non-persistent, and what size to choose", October 2017, <https://www.ripe.net/publications/docs/ripe-690>.

[Threat-DNS64]
G. Lencse and Y. Kadobayashi, "Methodology for the identification of potential security issues of different IPv6 transition technologies: Threat analysis of DNS64 and stateful NAT64", Computers & Security (Elsevier), vol. 77, no. 1, pp. 397-411, DOI 10.1016/j.cose.2018.04.012, August 2018.

Author's Address

Jordi Palet Martinez
The IPv6 Company
Molino de la Navata, 75
La Navata - Galapagar, Madrid 28420
Spain

Email: jordi.palet@theipv6company.com
URI: http://www.theipv6company.com/