Internet Area Working Group
Internet-Draft
Updates: 791, 2003, 2780, 4301,
4727, ietf-intarea-ipv4-id-update
(if approved)
Intended status: Standards Track
Expires: September 8, 2011

B. Briscoe
BT

March 7, 2011

Reusing the IPv4 Identification Field in Atomic Packets
draft-briscoe-intarea-ipv4-id-reuse-00

Abstract

This specification takes a new approach to extensibility that is both principled and a hack. It builds on recent moves to formalise the increasingly common practice where fragmentation in IPv4 more closely matches that of IPv6. The large majority of IPv4 packets are now 'atomic', meaning indivisible. In such packets, the 16 bits of the IPv4 Identification (IPv4 ID) field are redundant and could be freed up for the Internet community to put to other uses, at least within the constraints imposed by their original use for reassembly. This specification defines the process for redefining the semantics of these bits. It uses the previously reserved control flag in the IPv4 header to indicate that these 16 bits have new semantics. Great care is taken throughout to ease incremental deployment, even in the presence of middleboxes that incorrectly discard or normalise packets that have the reserved control flag set.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 8, 2011.

Table of Contents

Intended status: Standards Track? (to be removed before publication)

This draft defines a process and a protocol for enabling new protocols, including their progression from experimental track to standards track. A process specification cannot have lesser status than the protocols it enables. So if this specification were to start on the experimental track, it would not initially have sufficient status to enable standards track protocols.

In order for the IETF to consider whether this draft itself should be experimental or standards track, it has been written as if it is intended for the standards track. Otherwise the parts of the process for enabling standards track protocols would have had to have been written hypothetically, which would have been highly confusing. If the IETF decides this specification ought to start out on the experimental track, the standards track parts of the process will have to be edited out.

Appendix B discusses whether this draft itself would be better to start as experimental or standards track.

1. Introduction

The Problem: The extensibility provisions in IP (v4 and v6) have turned out not to be usable in practice. Hardware has been optimised for the common case, so packets using extensibility mechanisms (e.g. IPv4 options or IPv6 hop-by-hop options) are very likely to be punted to the software slow-path and consequently likely to be dropped whenever the software processor is busy [Fransson04, Cisco.IPv6Ext].

This specification takes a different approach to extensibility. Rather than flagging protocol extensions as 'extensions', it places extension headers where they will be ignored by pre-existing hardware. As code is added to routers to handle newly added extensions, the code can tell the machine where to look for the relevant header.

This approach recognises that extensions added after a protocol suite was first defined are different to options defined as a coherent part of the original protocol suite. Machines that have no code to understand a protocol extension that was added later do not need to punt a packet to the software processor merely to scan through chains of headers that it will not know how to process.

Having only settled on this approach long after the TCP/IP suite has been defined, it becomes necessary to find places in the existing protocol headers that are already ignored by existing machines. In an 'atomic' IPv4 packet, the Identification (IPv4 ID) field is one

such place that is redundant. This specification defines the process through which the 16 bits in this field can be returned to the IETF for use in future standards actions, at least within the constraints imposed by their original use for reassembly.

Background: [ipv4-id-update] proposes to update IPv4 to more closely match the approach taken to fragmentation in IPv6. It recommends that IPv4 sources send 'atomic' packets whenever possible. An atomic packet is one that has not yet been fragmented (MF=0 and fragment offset=0) and for which further fragmentation is inhibited (DF=1) [ipv4-id-update]. If fragmentation is necessary, it is only permitted at devices that control the uniqueness of the IP ID field, e.g., sources, tunnel ingresses (for the outer header), and the public side of NATs.

In practical scenarios, the IPv4 ID field is too small to guarantee uniqueness during the lifetime of a packet anyway [RFC4963]. Therefore it has become safer to disable fragmentation altogether and instead use an approach such as packetization layer path MTU discovery [RFC4821]. The large majority of IPv4 packets are now atomic.

Approach: This specification defines the IPv4 control flag that was previously reserved [RFC0791] as the Recycled flag (RC). An implementation can set RC=1 in an atomic packet to unambiguously flag that the IPv4 ID field is not to be interpreted as IP Identification, but instead it has the alternative semantics of an ID-Reuse field. By setting RC=1, IPv4 implementations can distinguish a value deliberately written into the ID-Reuse field from the same value that just happened to be written into the IP ID field of an atomic packet by a pre-existing implementation.

Thus, this specification effectively uses up the last bit in the IPv4 header in order to free up 16 other bits. However, there are some constraints on the use of these 16 bits due to their original use as the IP ID field (enumerated in Section 5.1). Of course the main constraint it that the bits are not available in non-atomic packets. But fragmentation is now used only rarely anyway, so it makes sense to see if the the Internet community can invent ways to use the 16 bits in the IPv4 ID field despite the constraints.

Frequently Asked Questions:

1. There are many cases where a non-compliant machine ignores Don't Fragment (DF=1) and fragments a packet anyway.

   One answer is that we cannot allow non-complaint behaviour to always block progress. Another answer is that we may be able to

detect and circumvent such non-compliant behaviour. For instance, if a non-compliant router fragments packets with DF=1, it may be possible to enhance path maximum transmission unit discovery (PMTUD) to find a lower segment size small enough to prevent the offending box from fragmenting packets.

2. {ToDo}

Document Roadmap: Section 3 defines the semantics of the updated IPv4 wire protocol and Section 4 defines intermediate node behaviour. Section 5 defines the process to be used for reassigning sub-fields of the IPv4 ID-Reuse field. Then Section 6 describes a way to circumvent problems likely to arise when deploying this new protocol. Finally, Section 7 enumerates the updates to pre-existing RFCs, before the tailpiece sections considering IANA, Security and draw conclusions.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Further terminology used within this document:

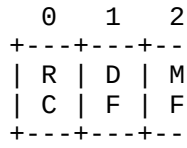
Atomic packet: A packet not yet having been fragmented (MF=0 and fragment offset=0) and for which further fragmentation has been inhibited (DF=1), or in the syntax of the C programming language ((DF==1) && (MF==0) && (Offset==0)) [ipv4-id-update].

Recycled (RC) flag: The control flag that was ’reserved’ in [RFC0791] (Figure 1). The flag positioned at bit 48 of the IPv4 header (counting from 0). Alternatively, some would call this bit 0 (counting from 0) of octet 7 (counting from 1) of the IPv4 header.

ID-Reuse field: Octets 5 and 6 (counting from 1) of the IPv4 header of an atomic packet (Figure 3). The field that would have been the IP Identification field if the packet were not atomic.

3. IPv4 Wire Protocol Semantics for Reusing the Identification Field

This specification defines the control flag that was defined as ’reserved’ in [RFC0791] as the Recycled (RC) flag (Figure 1).



The Recycled (RC) Flag was previously reserved.

Figure 1: The Control Flags at the Start of Byte 7 of the IPv4 Header

Figure 2 recaps the definitions of octets 5 to 8 (counting from 1) of the IPv4 header [RFC0791].

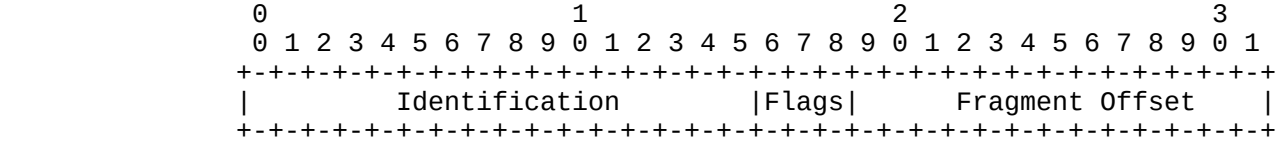
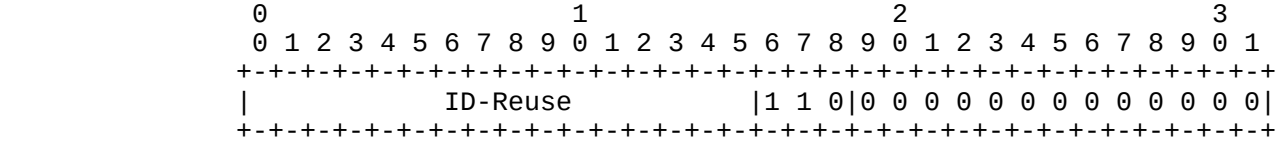


Figure 2: Recap of RFC791 Definition of Octets 5 to 8 of the IPv4 Header.

If an IPv4 implementation sets RC=1 on an atomic packet, octets 5 & 6 of the IPv4 header MUST be interpreted with the semantics of the ID-Reuse field, and MUST NOT be interpreted as the Identification field. Figure 3 shows how octets 5 & 6 are redefined as the ID-Reuse field when the packet is atomic, in the case where RC=1.



The Identification Field is redefined as the ID-Reuse Field when the Packet is Atomic and specifically when RC=1

Figure 3: Octets 5 to 8 of the IPv4 Header.

If the Recycled flag is cleared to RC=0 on an atomic packet, some sub-fields of octets 5 & 6 of the IPv4 header MAY be interpreted with the semantics of the ID-Reuse field, but only in the highly constrained circumstances defined in Section 6.2.

For the avoidance of doubt, the Recycled flag alone MUST NOT be assumed to indicate that the packet is atomic. Only the combination of ((DF==1) && (MF==0) && (Offset==0)) indicates that a packet is atomic. Then if the Recycled flag is also set, the ID field unambiguously has the semantics of the ID-Reuse field. If the Recycled flag of an atomic packet is cleared, its ID field only has

the semantics of the ID-Reuse field in specific limited circumstances.

It is expected that proposals to use the ID-Reuse field will each need a few bits, not the whole 16 bit field. Therefore this specification establishes a new IANA registry (Section 8) to record assignments of sub-divisions of the ID-Reuse field. In this way, it will be possible for new uses of different sub-divisions to be orthogonal to each other. The process for incrementally defining new sub-divisions is specified in Section 5.

If an IPv4 packet header has RC=1 but it is not atomic ((DF==0) || (MF==1) || (Offset !=0)), then all the fields of the IPv4 header are undefined and reserved for future use. If an implementation receives such a packet, it could imply:

- that some currently unknown attack is being attempted
- or that some future standards action has defined a meaning for this reserved combination of header values

Therefore, if an implementation receives a non-atomic packets with RC=1, it MUST treat the packet as if the Recycled flag were cleared to 0, but it MUST NOT change the Recycled flag to zero. It MAY log the arrival of such packets and/or raise an alarm. It MUST NOT always drop such packets, but it MAY drop them under a policy that can be revoked if it is established that the appearance of such packets is the result of a future standards action.

For convenience only, the above rules are summarised in Table 1. The semantics of octets 5 & 6 of the IPv4 header are tabulated for each value of the RC flag (rows) and for whether the packet is atomic or not (columns).

RC flag	Non-Atomic	Atomic
0	Identification	ID-Reuse (Limited)
1	Undefined	ID-Reuse

Table 1: The Dependence of the Semantics of Octets 5 & 6 of the IPv4 Header on whether the Packet is Atomic and on the RC Flag

4. Behaviour of Intermediate Nodes

4.1. End-to-End Preservation of ID-Reuse Semantics

If the source sets the RC flag to 1 on an atomic packet, another node MUST NOT clear the RC flag to zero. Otherwise the semantics of the ID-Reuse field would change (see the Security Considerations in Section 9 for discussion of the integrity of the ID-Reuse field). Note that intermediate nodes are already not expected to change an atomic packet to non-atomic, which otherwise would also risk changing the semantics of the ID-Reuse field.

If the source zeros the RC flag on an atomic packet, an intermediate node MAY change the RC flag to 1. At this time, no case is envisaged where an intermediate node would need to do this. However, as this behaviour preserves ID-Reuse semantics safely, it is not precluded in case it will prove useful (e.g. for sender proxies).

4.2. Tunnel Behaviour

This specification does not need to change the following aspects of IPv4-in-IPv4 tunnelling, which already provide the most useful semantics for the ID-Reuse field:

- For some time, it has been mandated that an atomic packet "MUST" be encapsulated by an atomic outer header [RFC2003] (although some implementations are broken in this respect).

- On decapsulation the outgoing header will naturally propagate the ID-Reuse field of the inner header.

However, compliant IPv4 encapsulation implementations SHOULD copy the ID-Reuse field when encapsulating an atomic IPv4 packet in another atomic IPv4 header, irrespective of the setting of the Recycled flag. It would be ideal but impractical to assert 'MUST' in this last clause, given it cannot be assumed that pre-existing IPv4-in-IPv4 encapsulators will propagate the ID-Reuse field to the outer header (see Section 5.1).

IPv6 packets without a fragmentation extension header are inherently atomic. Therefore, if an IPv4 header encapsulates an IPv6 packet, the encapsulator is already required to set the outer as atomic.

There is no direct mapping between the IPv4 ID-Reuse field as a whole and any IPv6 header field (main or extension), because the ID-Reuse field is merely a container for yet-to-be-defined sub-fields. However, sub-fields of the ID-Reuse field might be defined to provide a mapping for IPv6 extension headers that need to be visible in the

outer IPv4 header of a tunnel. The present specification cannot say anything in general about any such mappings or any associated tunnel behaviour. Any such behaviour will have to be defined when individual ID-Reuse sub-fields are specified.

5. Process for Defining Subdivisions of the ID-Reuse Field

When IPv4 was designed, then later IPv6, all the fields in the main IP header were initially defined together in a coordinated fashion. In contrast, the only practical way to define new uses for the bits in the ID-Reuse field will be to adopt a gradual addition approach, in which subsets of the bits or codepoints will have to be assigned on the merits of each request at the time.

Each new scheme will need to submit an RFC that requests a subdivision of the ID-Reuse field and assigns behaviours to the codepoints within this subdivision. A specification defining a new use of a subdivision of the ID-Reuse field MUST register this use with the IANA, which will maintain a registry for this purpose (Section 8).

Proposals to reuse the IP ID field could relate to other parts of the IPv4 header in the following different ways {ToDo: this list is not exhaustive}:

Orthogonal: Some new protocol proposals will need to apply whatever is in the rest of the packet, e.g. whether unicast or multicast, whatever the Diffserv codepoint and whatever else might have been added in the rest of the IP-Reuse field. Schemes that need to be orthogonal to other elements of the IPv4 protocol will require assignment of a number of bits as a dedicated sub-field of the ID-Reuse field.

Mutually exclusive: It might be impossible for two uses of the ID-Reuse field to both apply to the same packet. Such mutually exclusive schemes will only each require a range of codepoints within a sub-field.

Conditional: Some protocol proposals might only apply when other parts of the header satisfy certain conditions, e.g. only for multicast packets. The IANA will need to register these conditions so that the bits can still be assigned for other uses when the conditions do not apply.

To allow interworking between sub-fields that are being defined incrementally, every new protocol MUST assign the all-zeros codepoint of its sub-field to mean the new protocol is 'turned off'. This means that implementations of the new protocol will treat such

packets as they would have been treated before the new protocol was defined.

Implementations MUST also clear to zero any bits in the ID-Reuse field that are not defined at the time the implementation is coded.

Proposals to use sub-fields of ID-Reuse will have to be assessed in the order they arrive without knowing what future proposals they might preclude. To judge each proposal, at least the following criteria will be used:

Constraint satisfaction: Each proposal MUST either satisfy all the constraints in Section 5.1 below, or include measures to circumvent them.

General usefulness: Proposals that are not applicable to a broad set of services that can be built over the internetwork protocol SHOULD NOT warrant consuming the newly freed up IPv4 header space.

Parsimony: Burning up a large proportion of the remaining bits will count against a proposal.

Backward compatibility with prior uses of ID-Reuse: As more sub-fields of the ID-Reuse field become defined, each new proposal SHOULD ensure that it takes into account potential interactions with earlier standards actions or experiments defining other sub-fields.

Forward compatibility with potential uses of ID-Reuse: In addition, proposals that demonstrate sensitivity to potential future uses of the remaining sub-fields of the ID-Reuse field will be more likely to progress through the IETF's approval process.

Do no harm: Proposals that do no harm to existing uses of the Internet will be favoured over those that do more harm.

5.1. Constraints on Uses of the ID-Reuse Field

Atomic packets: The IPv4 ID field cannot be reused if the packet is not atomic, because then the IP ID field will need to be used for its original purpose: fragment reassembly.

IPsec interaction: The IP Authentication Header (AH) [RFC4302] assumes and requires the IPv4 ID field to be immutable, otherwise verification of authentication and integrity checks will fail. Any new use of bits in the ID-Reuse field MUST ensure the bits are immutable, at least between IPsec endpoints (whether transport or tunnel mode). It cannot be assumed that pre-existing IPsec

implementations will check the setting of the Recycled flag.

Note that the Recycled flag itself is considered mutable and masked out before calculating an authentication header [RFC4302] (see Section 9).

Tunnelling: Any new use of the ID-Reuse field in atomic packets cannot reliably assume that the ID-Reuse field will propagate unchanged into the outer header of an IPv4-in-IPv4 tunnel [RFC2003, RFC4301]. It is likely that an IPv4 tunnel ingress will encapsulate an atomic packet with another atomic outer header, as this behaviour was mandated in [RFC2003]. However it is known that some implementations are broken in this respect. It is possible that an IPv4 encapsulator might copy the IP ID field of an arriving atomic packet into the outer header. However this behaviour has never been required and therefore cannot be guaranteed for pre-existing tunnels.

Nonetheless, it can be assumed that the IPv4 ID field will be preserved through the inner header into the outgoing packet at the other end of the tunnel (even though this behaviour would not strictly have been necessary for an atomic packet).

Incremental deployment: Each new proposal will need to consider any detrimental effects from pre-existing IPv4 implementations, assuming that they are likely to act on atomic packets without first checking on the setting of the Recycled flag.

5.2. Process Example

For illustration purposes, imagine two RFCs have been published: an experimental track RFC called Experiment A (ExA) and a standards track RFC called Standard B (StB) and . Imagine they define respectively a use for bits bits 14 to 15 and 11 to 13 of the ID-Reuse field. Figure 4 shows example IANA registry entries for these imaginary sub-fields.

```
Protocol name:         StB
RFC:                   BBBB
Leftmost bit:          11
No. of bits allocated: 3
Sub-field defined if:  Atomic packet and RC=1

Protocol name:         ExA
RFC:                   AAAA
Leftmost bit:          14
No. of bits allocated: 2
Sub-field defined if:  Atomic packet and RC=1
```

Figure 4: Example IANA Registry of Sub-fields of the ID-Reuse Field

Figure 5 shows an example of how incremental specification of subdivisions of ID-Reuse would work.

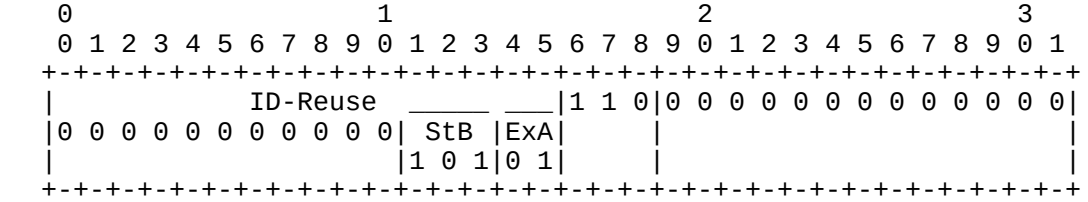


Figure 5: Example of Reuse of Octets 5 & 6 using RC=1

The bits shown in each row of Figure 5 define the semantics of the bits shown in the next row down, as follows:

- The top row identifies that the packet is atomic and the RC flag is 1. Therefore octets 5 & 6 of the IPv4 header are redefined as the ID-Reuse field.

- The middle row shows the bits assigned to Standard B and Experiment A by IANA. An implementer has to ensure that all the bits of the ID-Reuse field that are yet to be defined (bits 0-10) are cleared to zero.

- The bottom row shows that an implementation of ExA has set its 2-bit sub-field to codepoint 01 and an implementation of StB has set its 3-bit sub-field to codepoint 101. The meaning of each would be defined in the RFCs for ExA and StB respectively.

Imagine now that Experiment C (ExC) is defined later to use bits 0-7 of the ID-Reuse field. If the packet in Figure 5 is received by an implementation of ExC, then it will see only zeros in the ExC sub-field. Therefore the implementation of ExC will treat the packet as if ExC is turned off (as mandated in Section 5).

Similarly, the implementation of protocol StB can rely on being able to turn off Experiment A by setting bits 14 & 15 to zero.

6. Incremental Deployment of New Uses of the IPv4 ID Field

When implementations first set the Recycled flag to 1, they are likely to be blocked by certain middleboxes, either deliberately (e.g. firewalls that assume anomalies are attacks) or erroneously (e.g. having misunderstood the phrase "reserved, must be zero" in RFC791). It is also possible that broken 'normalisers' might clear RC to zero if it is 1, although so far no tests have found such broken behaviour.

To address this problem, Section 6.2 introduces a way to use a sub-field of ID-Reuse without having to set RC=1. In this approach, packet headers using the new protocol will be indistinguishable from an IPv4 header not using the new protocol. Therefore it will be possible to guarantee that middleboxes will not treat packets using the new protocol any differently from other IPv4 packets.

Many pre-existing IPv4 hosts cycle through all the values in the IP ID field even when sending atomic packets in which the IP ID field has no function. Therefore, these pre-existing IPv4 hosts will occasionally issue a packet that happens to look as if it is using a codepoint of a new protocol using the IP ID field. Without RC=1, there will be no way to distinguish the two.

	middlebox traversal	new protocol recognition
RC=0	Assured	Uncertain
RC=1	Uncertain	Assured

Table 2: Tradeoff between deterministic middlebox traversal and deterministic protocol recognition

Table 2 shows the tradeoff between using RC=0 or RC=1:

RC=0: If an implementation of a new protocol uses RC=0, its packets will traverse middleboxes, but it will suffer a small fraction of false positives when recognising which packets using the new protocol -- occasionally it will mistakenly assume a packet is using the new protocol when it is actually just random noise in the IP ID field from a pre-existing implementation.

RC=1: If an implementation of a new protocol uses RC=1, its packets may be black-holed by some middleboxes, but it will be certain which packets use the new protocol and which don't.

Nonetheless, a probabilistic protocol that can be deployed may be more useful than a deterministic protocol that cannot.

6.1. Process Example with RC=0

Figure 6 shows an example of how this approach would work with RC=0. For illustration purposes imagine, as in the previous example in Section 5.2, that an experimental track RFC has been published called Experiment A (ExA) that defines bits 14 to 15 of the ID-Reuse field for atomic packets with RC=1. Now imagine another experimental track RFC has been published called Experiment B (ExB) that defines a use for bits 11 to 13 of the ID-Reuse field, but does not require RC=1. In fact a packet is defined as complying with ExB whether RC=1 or RC=0 (i.e., RC=X, where 'X' means don't care). Figure 7 shows the IANA registry entries for these imaginary sub-fields.

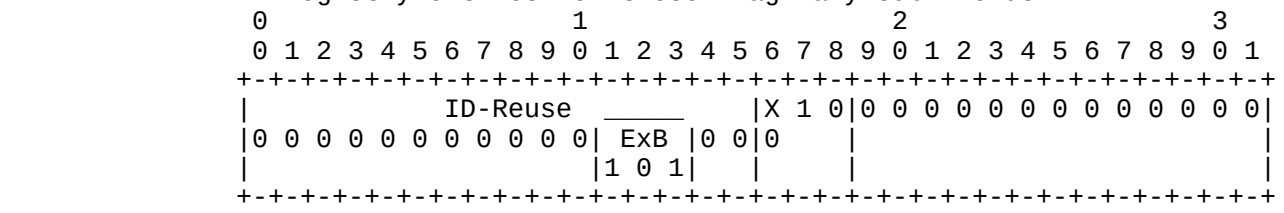


Figure 6: Example of Experimental Reuse of Octets 5 & 6 Without Requiring RC=1

The bits shown in each row of Figure 6 define the semantics of the bits shown in the next row down, as follows:

- The top row identifies that the packet is atomic. The RC flag is don't care ('X'), so RC does not have to be 1. Implementations can clear RC to 0 to traverse awkward middleboxes, but RC can be set to 1 otherwise.

- The middle row shows that an implementation of Experiment B (ExB) has set RC=0. It is also using the ID-Reuse field, so it clears all the bits to zero except those in its own sub-field (bits 11-13). It will have registered this experimental use with the IANA as shown in the top example of Figure 7.

- The bottom row shows that an implementation of ExB has set its 3-bit sub-field to codepoint 101, the meaning of which will have been defined in the RFC specifying the ExB protocol.

Note that, the process for using protocol ExB without RC=1 (Section 6.2) precludes an implementation from using the ExA protocol in the same packet -- any one packet can only be part of one RC=0 protocol at a time.

6.2. Process for Using the ID-Reuse Field Without Requiring RC=1

This approach SHOULD NOT be used unless the preferred approach (Section 5) is impractical due to middleboxes blocking packets with RC set to 1.

To follow this non-preferred approach, the registration with the IANA MUST specify that the sub-field of ID-Reuse is defined for 'RC=X', meaning "don't care", that is RC may be either set or cleared (for an example, see the final bullet of the imaginary registration details in Section 8). The RFC defining the relevant ID-Reuse sub-field MUST also make it clear that the sub-field is defined for either value of the Recycled flag (RC=X) in an atomic IPv4 packet.

This approach will not be feasible for all protocols; only those that satisfy the severe constraints laid down below. Otherwise, for protocols that cannot satisfy these prerequisite constraints, the preferred approach in Section 5 wth RC=1 will be the only option.

Once a sub-field of the ID-Reuse field has been registered with the IANA, implementations of the protocol can use any of the available codepoints within that sub-field in atomic packets without having to set RC=1, if and only if the following constraints can be satisfied:

1. New protocol implementations MUST NOT use RC=0 unless the treatments associated with all the new codepoints are generally benign to packets not taking part in the protocol. 'Benign' means the new protocol SHOULD do no more harm to other packets than previous implementations did. Using the term 'SHOULD' rather than 'MUST' does not completely rule out new protocol proposals that might sometimes introduce slightly more harm, but such proposals will need to give strong justifications

2. Implementations MUST clear all the other bits of the ID-Reuse field (except those in the new protocol's sub-field) to zero. Note that this is different to the approach with RC=1, where more than one sub-field at once can be non-zero

3. In addition the constraints in Section 5.1 must also be satisfied.

Constraint #1 is severe but necessary in order to ensure that a new protocol (e.g. ExB) does not harm atomic packets from pre-existing

IPv4 implementations. For example, a receiving implementation of ExB can assume that most packets with all zeros in bits 0-10 and 14-15 were deliberately set by another implementation of ExB. But many pre-existing implementations of IPv4 will be cycling (sequentially or randomly) through all the IPID values as they send out packets. Occasionally they will send out a packet that happens to look like it complies with protocol ExB. For the case of ExB with a 3-bit sub-field, such false positives will occur with probability 1 in 2^13 (˜0.01%). We term this the misrecognition probability.

If the new protocol were designed to do harm (e.g. to deprioritise certain packets against others) that would be fine for those packets intended to take part in the new protocol. But it would not be acceptable to harm even a small proportion of packets misrecognised as using the new protocol. This is why the RC=0 approach can only be allowed for a new protocol that is generally benign.

Constraint #2 is necessary in order to ensure the misrecognition probability remains low. If only one sub-field is allowed at one time, all the other bits in the ID-Reuse field will have to be zero. This ensures that a pre-existing IPv4 implementation cycling through all the IP ID values will collide less frequently with values used for each new protocol.

As already stated (Section 5), each new protocol MUST define the all-zeros codepoint of its sub-field to mean that the new protocol is 'turned off'.

This arrangement ensures that packets with an IPv4 ID of zero will never collide with a codepoint used by any ID-Reuse scheme, whether RC=0 or RC=1. All zeros was deliberately chosen as the common 'turned off' codepoint because some pre-existing implementations have used zero as the default IP ID for atomic packets.

In either case, whether the Recycled flag is set or not, a sub-field of the ID-Reuse field MUST be registered with the IANA, initially for experimental use, by referencing the relevant experimental track RFC. This will ensure that experiments with different sub-fields of the ID-Reuse field can proceed in parallel on the public Internet without colliding with each other. The referenced RFC MUST define a coherent process for returning the bits for other uses if the experimental approach does not progress to the standards track.

The same sub-field can be used with the same semantics as the experiment progresses, initially with the Recycled flag cleared to 0 and later set to 1. And the same protocol semantics can be used whether the proposal is experimental or standards track. Thus, the whole process is designed to:

1. allow initial experiments to use RC=0 to traverse non-compliant middleboxes (Section 6);

2. then, once sufficient middleboxes forward RC=1 packets, the experiment can either be continued with RC=1 (Section 5);

3. or the experiment can progress cleanly to the standards track, while still using the same sub-field but with RC=1;

4. or the experiment can be terminated without having wasted any header bits.

(Step 1 is only feasible if the extra constraints in Section 6.2 can be satisfied. If not, Step 2 will be the only feasible first step.)

For the avoidance of doubt, any use of ID-Reuse, whether experimental or not, is also subject to the general constraints already enumerated in Section 5.1.

7. Updates to Existing RFCs

Great care has been taken to ensure all the updates defined in this specifications are incrementally deployable.

The definition of the RC flag in Section 3 updates the status of this flag that was "reserved, must be zero" in [RFC0791]. The redefinition of the IP Identification field as the ID-Reuse field when an IPv4 packet is atomic also updates RFC791.

Updates to existing RFC791 implementations are only REQUIRED if they discard IPv4 packets with RC=1, or change RC from 1 to 0, both of which are misinterpretations of RFC791 anyway. Otherwise, there will be no need to update an RFC791-compliant IPv4 stack until new use(s) for the ID-Reuse field are also specified.

The recommendation in Section 4.2 to copy the ID-Reuse field when encapsulating an atomic IPv4 packet with another atomic IPv4 header updates IPv4-in-IPv4 encapsulation specifications [RFC2003] [RFC4301]. These updates to tunnels are likely to be recommended rather than essential for interworking, so they can be implemented as part of routine code maintenance.

The ability to redefine the IPv4 ID field of an atomic packet updates [ipv4-id-update], which states "The IP ID is not defined if the packet (datagram) is atomic". Nonetheless, octets 5 & 6 of an atomic packet still MUST NOT be interpreted with the semantics of the Identification field.

[RFC2780] provides the IANA with guidelines on allocating values in IP and related headers. The process defined in Section 5 and Section 6 update RFC2780, given ID-Reuse is effectively a new field in the IPv4 header.

[RFC4727] defines the processes for experimental use of values in fields in the IP header that are managed by the IANA. The processes defined in Section 5 and Section 6 update RFC4727 to include the new alternative use of the IPv4 ID field as an ID-Reuse field.

8. IANA Considerations

The IANA is requested to establish a new registry to record allocation of sub-divisions of the ID-Reuse field and to avoid duplicate allocations. The ID-Reuse field is an alternative use of the Identification field of the IPv4 header in atomic packets (Section 3). All 16 bits are available for assignment, either as sub-fields of bits or as sets of codepoints within a sub-field of bits. Each sub-division of the ID-Reuse field MUST be allocated through an IETF Consensus action. The registry MUST then record:

Protocol name: the name for the protocol, as used in the RFC defining it

RFC: the RFC that defines the semantics of the codepoints used by the protocol

Leftmost bit: the leftmost bit allocated, counting from bit 0 at the most significant bit (which is bit 32 of the IPv4 header, counting from 0)

No. of bits allocated: the width in bits of the allocated sub-field

Codepoint range (optional): The range of codepoints within the assigned sub-field of bits that the protocol uses

Sub-field defined if: the precondition for the sub-field to be defined (Section 5). Valid entries MUST include the condition that the packet is atomic and MUST specifiy valid values of the Recycled (RC) flag, either 'RC=1' or 'RC=X', where 'X' means don't care (Section 6).

Two example registrations are shown in Figure 7.

```
Protocol name:         ExB
RFC:                   BBBB
Most significant bit:  11
No. of bits allocated: 3
Codepoint range:       all
Sub-field defined if:  Atomic packet and RC=X

Protocol name:         ExA
RFC:                   AAAA
Most significant  bit: 14
No. of bits allocated: 2
Codepoint range:       all
Sub-field defined if:  Atomic packet and RC=1
```

Figure 7: Example IANA Registry of Sub-fields of the ID-Reuse Field

9. Security Considerations

Integrity Checking: This specification make the semantics of octets 5 & 6 of the IPv4 header (IP ID or ID-Reuse) depend on the setting of octets 7 & 8 (all the Control Flags and the Fragment Offset field). The IP Authentication Header (AH) [RFC4302] covers octets 5 & 6 but not octets 7 & 8. Therefore AH can assure the integrity of the bits in the ID-Reuse field, but it cannot verify whether or not the sender intended those bits to have the semantics of an ID-Reuse field.

Any security-sensitive application of the ID-Reuse field will therefore need to provide its own integrity checking of the status of the Control Flags and Fragment Offset. Such a facility would need to take into account that the present specification allows an intermediate node to set the Recycled flag, but not to clear it (Section 4.1).

Covert channels: It has always been possible to use bit 48 of the IPv4 header for a 1 bit per packet covert channel, for instance between a network protected by IPsec and an unprotected network. Bit 48 could be covertly toggled to pass messages because it had no function (so no-one would notice any affect on the main communication channel) and it was not covered by IPsec authentication. On the other hand, once alerted to the vulnerability, it has always been easy for an IPsec gateway to spot bit 48 being used as a covert channel, given bit 48 was meant to always be zero.

Now that bit 48 has been given a function, it will often no longer be possible for an attacker to toggle it without affecting the main data communication. However, whenever the main communication

does not depend on bit 48, it will be easier to for an attacker to toggle it covertly given it will no longer stand out as anomalous behaviour.

10. Conclusions

This specification builds on recent moves to make the approach to fragmentation in IPv4 more closely match that of IPv6. Already the fields that support fragmentation in the IPv4 header are usually redundant, but unfortunately they are non-optional.

This specification makes it possible to reuse the 16 bits of the IPv4 ID field when they are not needed for reassembly. The last unused bit in the IPv4 header is used in order to unambiguously flag that the IP ID field has new semantics. The bit is called the Recycled flag, because it allows the IP ID field to be recycled for new purposes when it would otherwise be redundant. Whenever the IP ID field has new semantics, it is termed the ID-Reuse field.

The process for redefining the semantics of sub-fields of this ID-Reuse field has been laid down, both for experimental and standards actions. Great care has been taken throughout to ease incremental deployment. The same sub-field can be used with the same semantics as an experiment evolves into a standards action. Initially it is even possible for certain experiments to leave the Recycled flag cleared to zero, in order to traverse any awkward middleboxes that incorrectly discard or normalise packets if the Recycled flag is set.

11. Acknowledgements

Rob Hancock originally pointed out that code to handle new protocols can tell the machine where to look for the relevant header. Dan Wing pointed out that codepoints, not just whole bits, could be assigned for protocols that are mutually exclusive.

Bob Briscoe is partly funded by Trilogy, a research project (ICT-216372) supported by the European Community under its Seventh Framework Programme.

Comments Solicited (to be removed by the RFC Editor):

Comments and questions are encouraged and very welcome. They can be addressed to the IETF Internet Area working group mailing list <int-area@ietf.org>, and/or to the author(s).

12. Outstanding Issues (to be removed when all resolved)

1. ...

13. References

13.1. Normative References

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC2003] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2780] Bradner, S. and V. Paxson, "IANA Allocation Guidelines For Values In the Internet Protocol and Related Headers", BCP 37, RFC 2780, March 2000.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, December 2005.

[RFC4727] Fenner, B., "Experimental Values In IPv4, IPv6, ICMPv4, ICMPv6, UDP, and TCP Headers", RFC 4727, November 2006.

[ipv4-id-update] Touch, J., "Updated Specification of the IPv4 ID Field", draft-ietf-intarea-ipv4-id-update-01 (work in progress), October 2010.

13.2. Informative References

[Cisco.IPv6Ext] Cisco, "IPv6 Extension Headers Review and Considerations", Cisco Technology White Paper , October 2006, <http://www.cisco.com/en/US/technologies/tk648/tk872/technologies_white_paper0900aecd8054d37d.html>.

[Fransson04] Fransson, P. and A. Jonsson, "End-to-end measurements on performance penalties of IPv4 options", Luleae University of Technology, Technical Report 2004:03, 2004, <http://pure.ltu.se/portal/files/1299598/

LTU-TR-0403-SE.pdf>.

[RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, March 2007.

[RFC4963] Heffner, J., Mathis, M., and B. Chandler, "IPv4 Reassembly Errors at High Data Rates", RFC 4963, July 2007.

Appendix A. Why More Bits Cannot be Freed (To be Removed by RFC Editor)

Given this specification uses the last unassigned bit in the IPv4 header, it is worth checking whether it can be used to flag a new use for more than the 16 bits in the IP ID field of atomic packets.

IHL: Ideally, the Internet header length field (4 bits) could be made redundant if the length of those IPv4 headers with bit 48 set were redefined to be fixed at 20 octets. Then a similar approach to IPv6 could be taken with the Protocol field redefined as a Next Header field and each extension header specifying its own length.

Unfortunately, although IPv4 options are rarely used and generally ignored, this idea would not be incrementally deployable. There are probably billions of pre-existing implementations of the IPv4 stack that will use the IHL field to find the transport protocol header, without ever looking at bit 48. If the IHL field were given any other semantics conditional on bit 48 being set, all these pre-existing stacks would break.

Header Checksum: Ideally, the Header Checksum (16 bits) could be made redundant in those IPv4 headers with bit 48 set. Then a similar approach to IPv6 could be taken where the integrity of the IP header relies on the end-to-end checksum of the transport protocol, which includes the main fields in the IP header.

Unfortunately, again, this idea would not be incrementally deployable. Pre-existing implementations of the IPv4 stack might verify the header checksum without ever looking at bit 48. And anyway IPv4 stacks on probably every pre-existing router implementation would update the checksum field without knowing to check whether bit 48 was set. Therefore if the field were used for any other purpose than a checksum, it would be impossible to predict how its value might be changed by a combination of pre-existing and new stacks.

It is clear that reusing fields other than the IPv4 ID would be fraught with incremental deployment problems. The reason the IPv4 ID field can be reused, is that an atomic packet already does not need

an Identification field, whether bit 48 is set or not. Setting bit 48 merely allows new implementations that understand ID-Reuse semantics to be certain the value in the ID-Reuse field was not written by an implementation that intended it to have Identification semantics.

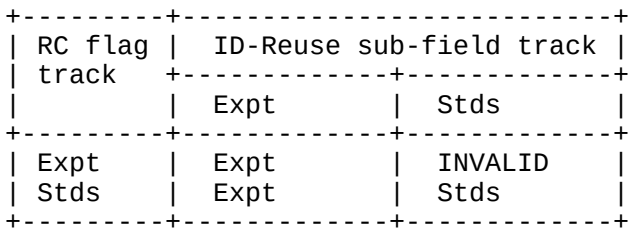
Appendix B. Experimental or Standards Track? (To Be Removed Before Publication)

This document defines a protocol (using the Recycled flag) to enable other protocols (using the ID-Reuse field). The Recycled flag protocol is currently written as if it is on the IETF standards track. Nonetheless it might be feasible to write it for the experimental track. This appendix discusses the pros and cons of each.

The Recycled flag uses up the last unused bit in the IPv4 header. The present specification defines a use for this last bit in the expectation that the Internet community will find ingeneous new use(s) for sub-fields of the ID-Reuse field, because then the Recycled flag will be needed to unambiguously indicate the new semantics. However, there is a risk that the last IPv4 header bit could be wasted, if no new uses for the IP ID field can be found within the constraints of its previous use for fragment reassembly, or if new experimental uses are proposed but none successfully proceed through to standards actions.

The risk of wasting the last bit would be mitigated if the definition of the Recycled flag itself was initially on the experimental track. Then, if some experimental use(s) of the ID-Reuse field did see widespread adoption, the RC flag protocol could progress to the standards track. On the other hand, if no ID-Reuse experiments happened, the RC flag could possibly be reclaimed for another use in the future. This would require all experiments with the RC flag to be confined in time, so that stray implementations of old experiments would not conflict with future uses of the flag.

Eventually, each specification for each sub-field of ID-Reuse might either progress on the experimental track or standards track. However, an enabler for standards track specifications cannot itself only be experimental. Therefore the RC flag protocol would have to be on the standards track, to enable standards track protocols as well as experimental. Figure 8 illustrates this need for the RC flag protocol to have sufficient rank for any protocols it enables.



The IETF track of the RC flag protocol in the present document (rows) and of any particular RFC specifying a sub-field of the ID-Reuse field (columns). The combination determines the status of any particular sub-field as shown at the intersection of the relevant row and column.

Figure 8: Validity of Combinations of IETF tracks for the RC flag and an ID-Reuse Subfield

One purpose of the present draft is to outline how new uses of ID-Reuse sub-fields can progress seamlessly from experimental track to standards track. Therefore, this draft is written as if it were on the standards track. Otherwise the processes for enabling standards track documents would have had to be written hypothetically, which would have been highly confusing. Nonetheless, no intent to prejudge that this document should be or will be on the standards track is implied.

If it were decided that the present draft should start on the experimental track, all the text about enabling standards track protocols would have to be edited out, or perhaps moved to a non-normative appendix.

Alternatively, the IETF might see some obvious new uses for sub-fields of the ID-Reuse field that would make it reasonable to fast-track the RC flag straight onto the standards track.

Author's Address

Bob Briscoe
BT
B54/77, Adastral Park
Martlesham Heath
Ipswich IP5 3RE
UK

Phone: +44 1473 645196
EMail: bob.briscoe@bt.com
URI: http://bobbriscoe.net/

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: September 8, 2011

Y. Wu
H. Ji
Q. Chen, Ed.
China Telecom
T. Tsou, Ed.
Huawei Technologies

March 7, 2011

IPv4 Header Option For User Identification In CGN Scenario
draft-chen-intarea-v4-uid-header-option-00

Abstract

In some application scenarios, it is necessary to be able to identify an user when CGN is deployed. This document defines a new IPv4 header option for host identification, which contains NAT mapping information, e.g. the internal source IP address before translation. Each time a NAT device performs translation on an IP packet, NAT mapping information will be added in the IP header. With the NAT mapping information, it will be easy to identify a host.

Table of Contents

1. Introduction

Some existing applications, e.g. web server, FTP server, etc, may need to perform operations based on the user's IP address, e.g., controlling the number of sessions, anti-virus filtering, traffic control against malicious attack, account security assistance, etc.

In the initial phase of IPv6 transition, CGNs are deployed to resolve the IPv4 public address depletion problem. Due to dynamic address mapping, some services and applications which require the knowledge of the source address will have problems. It is possible to query NAT log server or CGN to find out a user's source address [ID.draft-zhang-v6ops-cgn-source-trace], but this will impose high performance requirements on the NAT log server or CGN, and usually this kind of service is only available for law enforcement department of the operators themselves.

If the address mapping information is carried as an IPv4 header option, it will help those services and applications work, with minimum impact to the network.

An alternative solution is proposed by draft-wing-nat-reveal-option [draft-wing-nat-reveal-option]. The solution is based on TCP option; although quite some interesting applications are based on TCP, but there are still some scenarios it cannot cover, e.g., user traffic monitoring and analysis, and some UDP based applications.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

1.2. Terminology

The following terms are used in this document:

BNG: Broadband Network Gateway

CPE: Customer Premises Equipment

CGN: Carrier Grade NAT

UID: User Identification

UE: User Equipment

2. Motivating Scenarios

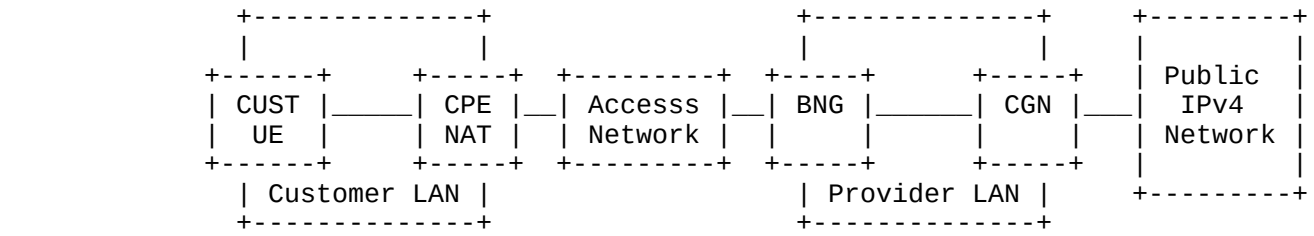


Figure 1: NAT444 Deployment

Dynamic IP address mapping in CGN will cause problems for services and applications which require knowledge of the source IP address. This section describes some typical scenarios where normal operations cannot be carried out without some mitigating measures such as those proposed in this document.

2.1. Limit The Number Of Sessions From An IP Address

Some download services need to limit the number of concurrent sessions from a same IP address. But if CGN is deployed, multiple users may be sharing the same IP address, so that such a mechanism will prevent some users from accessing services properly.

2.2. Anti-virus Filtering And Limiting Malicious Attack Traffic

Some existing traffic monitoring and analysis devices gather statistics and perform analysis, to enable anti-virus filtering based on the source IP address of packets. Some servers apply security policies based on source IP address to prevent malicious attacks [RFC4732]. For example, servers can refuse malicious users according to their source IP address to prevent drunk mail, malicious registration, etc. Deployment of CGN will impact the correct operation of traffic monitoring and analysis.

2.3. Account Security Assistance

Some existing services provide user account security guarantees by combining authentication and the user’s IP address. For example, the server can log the user’s IP address each time the user logs in, and if the user logs in with an IP address different from the last one or the most often used one, the server can inform the user, and may ask the user for extra authentication information. The deployment of CGN will stop this kind of assistance from working.

3. User Identification (UID) IPv4 Header Option

3.1. Option Format

The UID option consists of an option header and one or more instances of the NAT Mapping sub-option. The NAT Mapping sub-option is described in the next section. The UID option is illustrated in Figure 2.

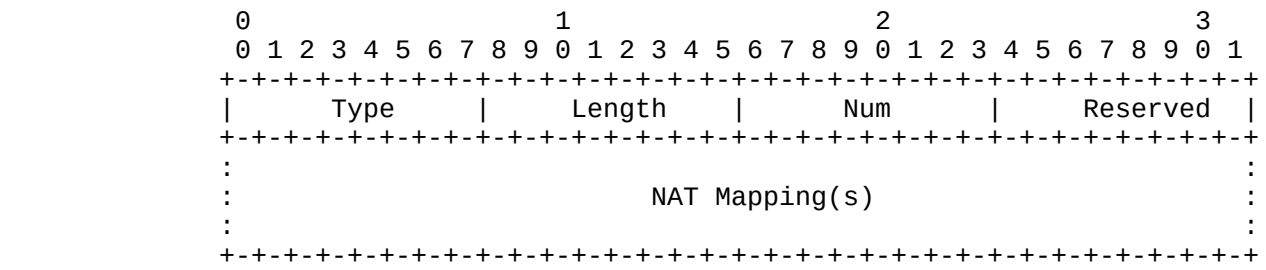


Figure 2: UID IPv4 Header Option Format

The fields of the option header are defined as follows:

Type:

The option type, which has the format specified in [RFC0791] and the following specific sub-field values:

Copied flag: 1 (copy into fragments)

Option class: 2 (debugging and measurement)

Option number: TBD.

Length:

Total length of the option in octets. As specified in [RFC0791], the length value includes the Type and Length octets in its count. Also as specified in [RFC0791], the maximum value of Length is 40 octets minus the length of any other IPv4 header options that are present.

Num:

The number of appended NAT Mapping sub-option instances.

Reserved: always 0.

3.2. NAT Mapping Sub-option

Each instance of the NAT Mapping sub-option records the source of the packet from the point of view of the NAT adding that instance. Depending on the scenario, that source can be identified by an IPv4 address, IPv6 address, or one of several types of tunnel plus host or context identifier, depending on whether DS Lite or Gateway-Initiated DS Lite is used. The format of the NAT Mapping sub-option is shown in Figure 3.

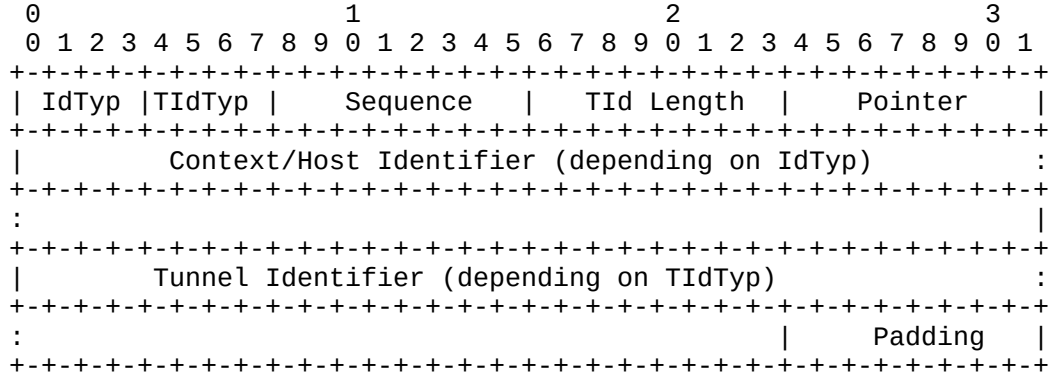


Figure 3: NAT Mapping Sub-option Format

The fields of the NAT Mapping sub-option are as follows:

IdTyp:

Type of context or host identifier. For native transport, this is either IPv4 address or IPv6 address. For DS Lite [ID.DS-Lite], it is always IPv4 address. For Gateway-Initiated DS Lite [ID.GI-DS-Lite], it is the type of the context identifier. This document specifies the following values for IdTyp:

00: reserved;

01: IPv4 address;

02: IPv6 address;

03: GRE key;

04: IPv6 Flow Label.

All other values are reserved.

TIdTyp:

Type of tunnel identifier. For native IP transport, this is NULL. For DS Lite, it is IPv6 address. For Gateway-Initiated DS Lite, it can be IPv4 or IPv6 address or MPLS VPN ID. Hence this document specifies the following values for TIdTyp:

00: NULL;

01: IPv4 address;

02: IPv6 address;

03: MPLS VPN ID.

All other values are reserved.

Sequence:

Sequence number of the NAT Mapping sub-option instance, indicating the order in which it was added to the option. The sequence number is assigned to the instance when it is created, and never changes after that. As a result, downstream entities can know if instances have been deleted because of lack of space if the first instance present in the option does not have a sequence number equal to 1.

TId Length:

Length of the tunnel identifier. This is equal to 0 if the TIdTyp is NULL, 4 if the TIdTyp is IPv4 address, 16 if the TIdTyp is IPv6 address, and 7 if the TIdTyp is MPLS VPN ID.

Pointer:

The sum of the lengths of the Context/Host Identifier field, the Tunnel Identifier field, and the Padding field, effectively pointing to the end of the sub-option instance.

Context/Host Identifier:

The source address of the incoming packet, for native transport. The source address of the decapsulated packet, for DS Lite. The context identifier value, for Gateway-Initiated DS Lite. The length of this field is 16 for an IPv6 address and 4 for all other types. A context identifier of type Flow Label MUST be

constructed by placing the Flow Label in the least significant bits of the word in network byte order and setting the most significant bits to zeroes.

Tunnel Identifier:

For native transport, this field is empty. For tunneled transport, it is the IPv4 or IPv6 source address in the outer header or the MPLS VPN ID of the tunnel.

Padding:

Always 0. Present only when needed to extend the Tunnel Identifier to a four-octet boundary (i.e., when the identifier is an MPLS VPN ID).

3.3. Procedures

3.3.1. Procedures At a NAT

If a NAT conforming to this specification receives a packet that it will forward as an IPv4 packet, then:

- if the incoming packet (after decapsulation if applicable) was an IPv6 packet, or if it was an IPv4 packet but contained no UID header option, and if sufficient space exists in the IPv4 header to permit it, the NAT MUST add the UID option containing a single instance of the NAT Mapping sub-option. The sequence number of the instance MUST be 1.

- if the incoming packet (after decapsulation if applicable) is an IPv4 packet containing the UID header option, the NAT MUST append an instance of the NAT Mapping sub-option to the existing sequence of instances. The sequence number of the new instance MUST be the sequence number of the preceding instance incremented by 1. For the settings of the remaining fields of the instance, see below. If the result is to cause the IPv4 header to exceed its limit of 60 octets [RFC0791], the NAT MUST delete the NAT Mapping sub-option with the lowest sequence number from the UID option. The NAT MUST repeat this action until the IPv4 header length does not exceed 60 octets. If as a result, no more sub-option instances remain in the UID option, the NAT MUST delete the option itself.

In either case, the remaining fields are set according to the particular transport mechanism in use.

3.3.2. Procedures At an Edge Device Or Firewall

Depending on local policy, edge routers or firewalls conforming to this specification MAY strip off the UID option on the outgoing interfaces if necessary, e.g., because the application server or end user may not be able to recognize the UID option, or because there may be potential interoperability issues in the communication between ISPs due to this option. In this case, the UID option is still useful for user traffic monitoring and analysis in the operator's network.

3.3.3. Procedures At Other Routers

Other routers along the packet path should pass the option along unchanged and copy it to fragments when fragmentation occurs, simply in conformity to [RFC0791]. For greater certainty, routers conforming to this specification MUST behave as just described.

4. Maximum Transmission Unit

Because IPv4 header options are inserted into packets, which will change the length of an IP packet, a NAT Device MUST modify the MTU value in an ICMP message accordingly when receiving or generating a ICMP Packet Too Big error message.

5. NAT configuration

There SHOULD be a configurable parameter on the NAT for the administrator to enable/disable use of the UID option.

6. Impact To Existing Devices

The UID option is in the IP header, and complies with the format defined in [RFC0791]. As mentioned in Section 3.3.3, any network devices that fully support [RFC0791] should handle the UID option without any change. User terminal devices do not have to support this option.

Resolving the user identification problem via the UID option protects the existing investment and does not require extra cost while being compatible with existing user and network devices. Obviously the consuming applications such as download services, traffic monitoring and analysis, and enhanced identification need to be modified to make use of the information provided by the UID option.

7. Security Considerations

TBD.

8. IANA Considerations

This document defines a new IPv4 option type, which shall be allocated by IANA. This requires IANA to set up a new registry for IPv4 options. The initial population of this registry consists of the options defined in [RFC0791], plus the new option added by this specification. [Need to determine if any other options have been defined. Registry format to be added later.]

9. Acknowledgements

To be completed.

10. References

10.1. Informative References

[RFC4732] Handley, M., Rescorla, E., and IAB, "Internet Denial-of-Service Considerations", RFC 4732, December 2006.

[draft-wing-nat-reveal-option]
Yourtchenko, A. and D. Wing, "Revealing hosts sharing an IP address using TCP option(work in progress)", August 2010.

10.2. Normative References

[ID.DS-Lite]
Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion (Work in progress)", March 2011.

[ID.GI-DS-Lite]
Brockners, F., Gundavelli, S., Speicher, S., and D. Ward, "Gateway Initiated Dual-Stack Lite Deployment(work in progress)", Oct 2010.

[ID.draft-zhang-v6ops-cgn-source-trace]
zhang, D., "Solution Model of Source Address Tracing for Carrier Grade NAT (CGN)(work in progress)", February 2011.

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

Authors' Addresses

Youming Wu
China Telecom
109, Zhongshan Ave. West, Tianhe District
Guangzhou 510630
P.R. China

Phone:
Email: wuym@gsta.com

Hui Ji
China Telecom
NO19.North Street,CHAOYANGMEN,DONGCHENG District
Beijing
P.R. China

Phone:
Email: jihui@chinatelecom.com.cn

Qi Chen (editor)
China Telecom
109, Zhongshan Ave. West, Tianhe District
Guangzhou 510630
P.R. China

Phone:
Email: chenqi.0819@gmail.com

Tina Tsou (editor)
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
P.R. China

Phone:
Email: tena@huawei.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 16, 2012

W. Dec
R. Asati
Cisco
C. Congxiao
CERNET Center/Tsinghua University
H. Deng
China Mobile
M. Boucadair
France Telecom

October 14, 2011

Stateless 4Via6 Address Sharing
draft-dec-stateless-4v6-04

Abstract

This document presents an overview of the characteristics of stateless 4V6 solutions, alongside a assessment of the issues attributes. The impact of translated or mapped tunnel transport modes is also presented in the broader context of other industry standard reference architectures and existing deployments.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 16, 2012.

Copyright Notice

Table of Contents

1. Introduction

As network service providers move towards deploying IPv6 and IPv4 dual stack networks, and further on towards IPv6 only networks, a problem arises in terms of supporting residual IPv4 services, over an infrastructure geared for IPv6-only operations, and doing so in the context of IPv4 address depletion. This class of problem is referred to by the draft as the 4via6 problem, for which a stateless solution is desired driven by motivation as documented in [I-D.operators-softwire-stateless-4v6-motivation]. Solutions such as a 4rd [I-D.despres-softwire-4rd], [I-D.murakami-softwire-4v6-translation], and [I-D.xli-behave-divi-pd], as well as dIVI [I-D.xli-behave-divi] offer such stateless solutions, by using fully distributed NAPT44 functionality located on end user CPEs, which allows the network operators' core to remain effectively stateless in terms of NAT44. The solutions, collectively called Stateless4V6, rely on the same IPv4 address being used by multiple CPEs, each with a different TCP/UDP port range, and are derived from the Address+Port (A+P) solution space [I-D.ymbk-aplusp]. Differences between the solutions come down to the mode of transport (translation or mapped tunneling), and the mapping algorithm used. This document looks at the issues that have been claimed as applying to A+P technology, in the specific context of the referenced solutions, and also analyzes the two modes of transport.

2. Terminology

Stateless4V6 domain: A domain is composed out of an arbitrary number of 4V6 CE and Gateway nodes that share a mapping relationship between an operator assigned IPv6 prefix and one or more IPv4 subnets along with all the applicable TCP/UDP ports, all mapped into the IPv6 address space. An 4V6 system can have multiple domains.

Stateless4V6 CE: A CPE node that implements 4V6 functionality including NAPT44 which is provisioned by means of 4V6. The device interfaces to the SP network using native IPv6 and a IPv4-IPv6 adaptation service.

Stateless4V6 Gateway A Service Provider node that implements the stateless 46 adaptation functionality for interfacing between the SP's IPv6 domain and an IPv4 domain in delivering end user IPv4 connectivity beyond the domain.

IPv4 Address sharing The notion of attributing the same IPv4 address by multiple CEs in an 4V6 domain.

Port-set: A set composed of unique TCP/UDP ports (ranges) associated to a IPv4 address. A single 4V6 CE is expected to have a single port-set for each IPv4 address.

Port-set-id: A numeric identifier of a given port set that is unique in a given 4V6 domain. A port-set-id is used to algorithmically determine the port-set members. The port-set-id is conveyed to CEs as part the CE's IPv6 addressing information, ie it is part of IPv6 subnet or address of a given CE, and its format places no restriction on the use of SLAAC or DHCP addressing.

CE-index: A numeric value, composed of a full or partial IPv4 address and optionally a port-set-id, which uniquely identifies a given CE in an 4V6 domain.

3. Stateless 4V6 Technical and Architectual Overview

This section presents the architectural and technical overview of a stateless 46 solution, and evidenced in whole or in part by various stateless 4via6 solution proposals such as 4rd, dIVI. Figure 1 depicts the overall architecture with two IPv4 user networks connected via 4via6 CPEs that share an IPv4 address. The goal of the system is to allow IPv4 user connectivity to the Public IPv4 network, across an operator's IPv6 network.

A key characteristic of the system, and a major differentiator with respect to previous solutions, is that translation state is only (ever) present on the CE, with the rest of the system performing stateless transport. This stateless transport applies to both the mapped-tunnel and translated modes, as described in the dedicated sections.

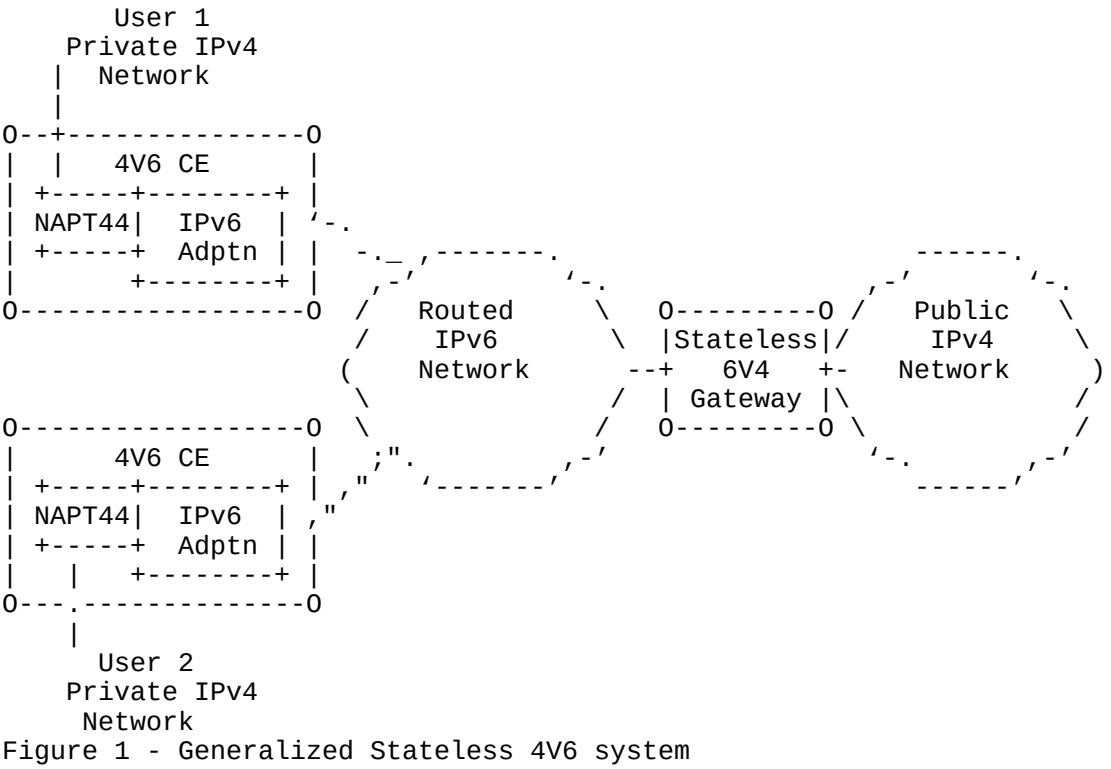


Figure 1 - Generalized Stateless 4V6 system

On IPv4 network user side, the routed IPv6 service provider network is demarcated with a 4V6 CE. The CPE externally has only a native IPv6 interface to the SP network, and a native IPv4 interface towards the end user network.

The IPv4 Internet is demarcated from the operator IPv6 network with one or more operator managed stateless 6V4 gateways that contain an IPv6 adaptation function (not detailed in the diagram) matching the one in the CE. Note: The stateless 6v4 gateway can be integrated into any existing network element (eg a core router, or an IP Edge).

Internally, the 4V6 CE is modelled as having a port restricted NAPT44 function coupled with a stateless IPv6 adaptation function that is able to ferry the end-user's IPv4 traffic across the IPv6 network, besides deriving 4V6 provisioning info from it. The NAPT44 function derives its IPv4 address, which may be shared with that of other users, and its unique Layer 4 (TCP/UDP) port range from the IPv6 address/prefix by means of an 4V6 algorithm and a port indexing schema. Any IPv4 ALG functionality that the CPE may support, remain unaffected. The CPE is expected to act as a DNS resolver proxy, using native DNS over IPv6 to the SP network.

Two forms of the IPv6 adapatation function are: i) 4v6 stateless tunneling ii) 4v6 stateless translation, each described in further in this document.

The service provider is assumed to be operating all the necessary provisioning and accounting infrastructure to support a regular IPv6 deployment. Similarly, the network operator is assumed to have the ability to assign an IPv6 prefix or IPv6 address to a CPE, and log such an address assignment.

End user host's DO NOT implement any of the 4V6, or other address sharing technologies, nor are they addressed directly with a shared IPv4 address. End user IPv4 hosts connected to the CPE receive unique private addresses assigned by the CPE, and it is the CPE that is directly addressed by the shared IPv4 address.

Although tangential to the discussion of stateless 4V6, it is useful to note that the CPE is expected to have a native IPv6 interface to the end user network, with any of the end user IPv6 hosts (single or dual stack) receiving IPv6 addresses from an IPv6 delegated prefix issued to the CPE.

3.1. IPv4 address and algorithmic port indexing

At the heart of the 4V6 solution, irrespective of mode of transport, lies the algorithm described in the specific solution drafts that allows the mapping of a shared IPv4 address and a TCP/UDP given port-set to a single IPv6 prefix or address. Notably, the 4V6 system allows both the shared IPv4 address use, as well as full non-shared IPv4 address use, all subject to the 4V6 domain configuration.

The S46 domain information required to compute the IPv4 address and correct port set is retrieved from the 4V6 prefix advertised to the CE, and pre-configured or statelessly acquired domain information.

3.2. 4V6 CE IPv6 Address and domain info

As presented in Section 2, IPv6 address of an 4V6 CE is composed out of the SP advertised IPv6 4V6 prefix, containing the CE-index, and an algorithmically computed appendix to complete the 128-bit address. This IPv6 address is *in addition* to any other IPv6 interface address that the CE configures or is configured with, including a SLAAC address from the 4V6 prefix or any IPv6 address source. One characteristics of the resulting IPv6 prefix or address is that it is for all intents and purposes a regular IPv6 prefix address that can be assigned to any regular IPv6 host.

The IPv6 4V6 interface is reserved for the 4V6 application and the

4V6 IPv6 adaptation function will exclusively use this IPv6 address. This is because the 4V6 system supports stateless communication between the 4V6 CE and the 4V6 gateway only by means of packets sent to/from this address.

3.3. IPv6 Adaptation Function

The IPv6 adaptation function plays a key role in the 4V6 system, in statelessly allowing the IPv4 user payload to be transported across an IPv6 (only) network. Two modes of such a function are currently proposed and presented in the following subsections

3.3.1. 4V6 Stateless Tunneling Mode

This type of IPv6 adaptation function is adopted and described in [I-D.despres-softwire-4rd].

The 4V6 gateway operates in the IPv4->IPv6 direction by mapping all or part of the IPv4 destination address and the port Index derived from the UDP/TCP payload into an IPv6 CE destination address. The resulting packet is sent using IPv4inIPv6 encapsulation to the CE, sourced from the 4V6's gateway IPv6 address, where the original IPv4 packet is extracted and passed to the stateful NAPT44 function.

The 4V6 CE operates in the IPv4->IPv6 direction, for traffic bound to the IPv4 internet, by encapsulating the IPv4 packet in an IPv6 header using IPv4inIPv6 encapsulation, and sending the resulting packet to the (well known) unicast address of the 4V6 gateway. There the IPv4 packet is extracted and forwarded.

The the original IPv4 packet addressing information is only partially visible on the IPv6 data plane, and the original Layer 4 information is only visible as part of the encapsulated IPv4 payload packet.

The figure below illustrates the CE model of a 4v6 Mapped Tunnel mode.

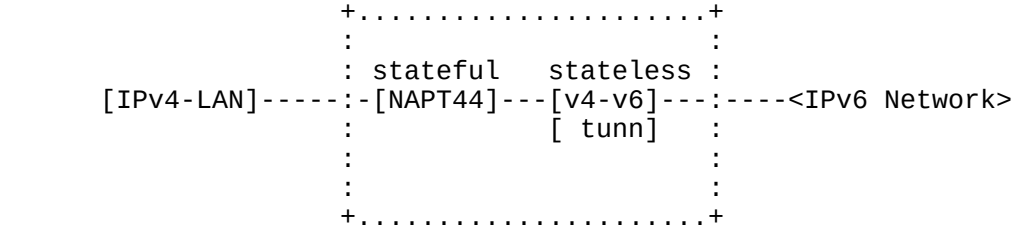


Figure 2 - 4v6 CE model with Tunnel mode

3.3.2. 4V6 Stateless Translation mode

This type of IPv6 adaptation function is adopted and described in [I-D.murakami-softwire-4v6-translation], I-D.xli-behave-divi-pd, and[I-D.xli-behave-divi] The 4V6 translation mode transport operates by means of stateless NAT46 [RFC6145] extended to map the the TCP/UDP port index algorithmically derived from received IPv4 packets into an IPv6 address suffix, in the IPv6 header, besides the full IPv4 mapped representation of the original IPv4 address information. The resulting packet is then sent across the IPv6 domain as an IPv6 packet - this IPv6 packet, besides mapping the original original IPv4 address information into a determinate IPv6 format, also places the Layer 4 and packet content directly after the IPv6 header, as any regular IPv6 with TCP/UDP packet. This IPv6 packet is thus capable of being processed by regular IPv6 network elements or servers in the IPv6 domain. At either end of the IPv6 domain, the IPv4 packet header is statelessly recreated, by the 4v6 CE or gateway, again using exactly the same NAT64 process as in [RFC6145].

The figure below illustrates the IPv6 4v6 Stateless Translation model of a 4v6 CE.

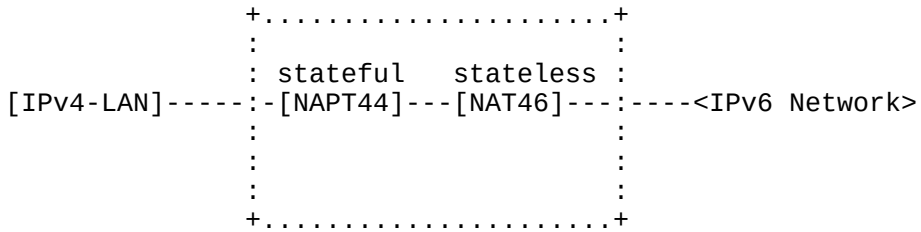


Figure 3 - 4v6 CE model with stateless NAT64

4. Comparison of 4V6 transport modes

This section presents the an overview of the similarities and differences between an IPv4-IPv6 translation based 4V6 transport mode and one that utilizes IPv4-in-IPv6 tunnelling. The comparison takes into consideration a wider deployment view composed of functionality that is known to be in common use today.

4.1. General Characteristics of 4V6 modes

The following table presents a comparison of the 4V6 transport modes, in terms of the base technology, and constrains, including also IPv4.

Item	4V6 Translation mode	4V6 Tunnel Mode
Base Technology	Port restricted NAPT44 with modified stateless NAT64 on CPE and Gateway	Port restricted NAPT44 with IPv4 in IPv6 mapped tunneling on CPE and Gateway
Location of stateful NAPT44 function	CPE	CPE
IPv4 Forwarding paradigm	L3 + L4 lookup	L3 + L4 lookup
IPv6 Addressing Constraints	CE uses 4V6 suffix.	CE uses 4V6 suffix.
Type of IPv6 prefix/address announcement method supported	ICMPv6 (SLAAC), DHCPv6 (both IA_NA and IA_PD)	ICMPv6 (SLAAC), DHCPv6 (both IA_NA and IA_PD)
Can the 4V6 IPv6 prefix be used by non 4V6 devices?	Yes	Yes
IPv4 addressing constraints	Fixed sharing ratio per IPv4 address.	Fixed sharing ratio per IPv4 address.
TCP/UDP Port range constraint	Ports are statically allocated	Ports are statically allocated
Requires ALG64 or DNS64	No	No
Requires IPv6 DNS on CPE	Recommended	Recommended
4V6 CE Parameter provisioning methods (assuming suitable protocol extensions)	ICMPv6, Stateless DHCPv6, TR69	ICMPv6, Stateless DHCPv6, TR69.

IPv6 Domain Routing to CE based on:	Regular closest IP match to CE-IPv6 subnet	Regular closest IP match to CE-IPv6 subnet
IPv6 Domain Routing to 4V6 Gateway based on	IPv6 4V6 domain aggregate route	4V6 Gateway unicast/anycast address
IPv4 Header Checksum recalculation required	Yes	No
Supports non TCP/UDP Protocols	No*	No*
ICMPv4 Limitations	No ICMPv4 from "outside the domain". Internal ICMPv4-v6 translation as per [RFC6145]	No ICMPv4 from "outside the domain".
ICMPv5 identifier NAT/Markup needed	Yes	Yes
Supports IPv4 fragmentation (without additional state)	No	No
Requires IPv6 PMTU discovery/configuration	Yes	Yes
Supports IPv4 Header Options	No - as per NAT64 [RFC6145]	Yes (use of source route option is constrained)
TCP/UDP Checksum recalculation	Yes - depending on suffix, as per NAT64 [RFC6145]	No
Supports UDP null checksum	Yes/Configurable - as per NAT64 [RFC6145]	Yes
Transparency to DF bit	Yes, configurable as per [RFC6145]	Yes

Supports IPv4 Fragmentation	Partial (no fragments from outside the domain)	Partial (no fragments from outside the domain)
Transparency to IPv4 TOS	Yes, configurable as per [RFC6145]	Yes
Overhead in relation to original average payload on IPv6 of a) ~550 bytes b) 1400 bytes).	a) 0% b) 0%	a) 4.36% b) 1.71%
Supports non-shared IPv4 usage (ie whole IPv4 address assignment to a single device)	Yes	Yes
Can support IPv4 to IPv6 host communication (for traffic not requiring ALGs)	Yes - As per [RFC6145] stateless NAT64 specification	No
Changes to network element provisioning tool(s)**	Yes - Mapping IPv4 to IPv6 addresses	Yes - Enabling IPv4inIPv6 functionality

* Without specific ALGs. Non UDP/TCP protocols, like ICMP, can be supported with specific ALGs.

**Network (feature) provisioning tools/applications need to be 4V6 aware. With the translation technique, the tool needs to enable the operator to map IPv4 addresses to IPv6 addresses as disctated by the 4V6 domain. With the tunneling technique, the tool needs to allow the operator to enable IPv4 (inIPv6) functionality and modify its characterstics.

4.2. Mobile SP Architecture and 4V6 Applicability

This section presents the applicability and comparison of the 4V6 modes to current 3GPP architectures used by Mobile SP for delivering all sorts of mobile services.

4.2.1. 3GPP overview

The 3rd Generation Partnership Project (3GPP) is a collaboration between groups of telecommunications associations, whose scope is to develop a globally applicable mobile phone systems and architectures based on service requirements. 3GPP standards are structured as Releases, each of which incorporates numerous individual standard documents. Currently, 3GPP Release 7 is the latest release in common practical deployment, with Release 8 being readied for deployment. Releases 9 and 10 are finalized, and work is underway on Release 11.

One of the major service requirement drivers of recent and ongoing 3GPP releases is the realization of services that deliver specific QoS, or user charging goals, all based on a policy system (eg tiered data rate or volume plans). Technically this translates to the Policy and Charging Control (PCC) framework, which in turn attributes specific functionality to nodes in the 3GPP architecture, such as the PDN-Gw and the PCRF. This functionality comprises both data-plane features (eg IP flow classification) as well as the interfaces/ protocols between nodes (eg Diameter, and its specific 3GPP applications).

The 3GPP specifications allow both IPv4 and IPv6 traffic to be handled, and subject to operator defined handling and charging polices by means of applying suitable user traffic filters. Such filters are currently defined to be either IPv4 or IPv6, are applicable to user plane traffic, and are used in a variety of critical roles including the signalling of PDP contexts/EPC Bearers, as well as PCC signalling and interaction with applications.

The following table illustrates the impact of the 4V6 translation and tunnel transport modes respectively on the 3GPP architecture including PCC interfaces. In assessing the impact of these 4V6 transport modes a number of additional assumptions are taken:

- The 3GPP system supports native IPv6 user traffic, as say per either of the E-UTRAN Release 8 or 9 specifications, using the relevant EPS bearer or PDP functionality.

- The 4V6 gateway functionality is not part of the 3GPP core architecture (given that currently it is not scoped by a 3GPP Release). Instead, the 4V6 gateway is taken to be a stand alone component in the 3GPP network operator’s core reachable via the SGi interface.

The above system, in the context of 3GPPs E-UTRAN architecture as defined in [E-UTRAN] is shown in Figure 2

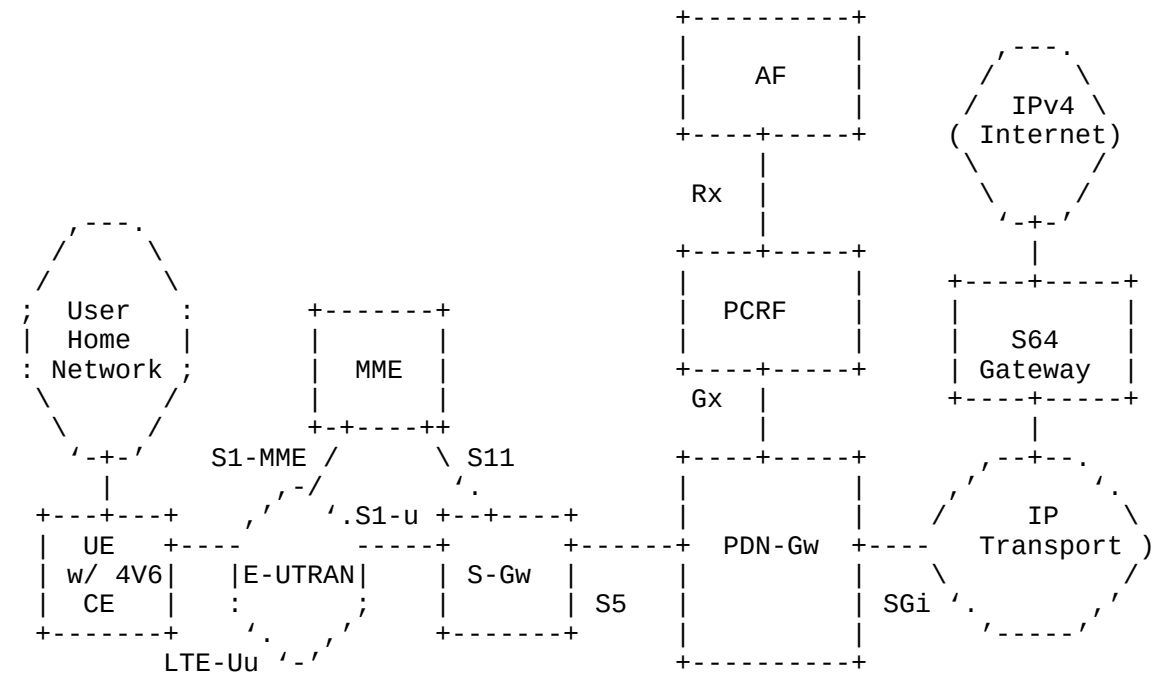


Figure 2 - 3GPP Architecture with 4V6

The main 3GPP system components, and terms are summarized as follows (the reader is referred to [E-UTRAN for a more detailed definition]:

UE The User Equipment, typically a phone or a 3G/4G capable Home Router (shown to incorporate 4V6 functionality)

E-UTRAN Evolved Universal Terrestrial Radio Access Network. The Radio Access network, composed on E-NodeB elements.

MME Mobility Management Entity. Responsible for user authentication, PDN/SGw selection. Does not interact with the user data plane

S-Gw Serving Gateway (function). Responsible for handling local mobility, (some) traffic accounting, traffic forwarding, bearer establishment.

PDN-Gw Packet Data Network Gateway (function). Responsible for per user IP traffic handling, incl. address assignment, filtering, QoS, accounting.

PCRF Policy And Charging Rules Function. Responsible for authorizing and applying policy rules, as well as binding them to user bearers.

Bearer The bearer represents a virtual connection, typically that between a UE and a PDN-Gw. The bearer is specified as an IP Fliter (in terms of IP address, port numbers) and is the object of policy rules. 3GPP, depending on Release and document, defines many terms that are used to refer to the same notion: PDP context, EPS Bearer.

AF Application Function. A functional element offering (higher level) applications that require dynamic policy and/or charging control over the user plane (bearer) behaviour. The AF can be seen as bridging the gap between applications and how they affect the IP data plane of a user.

S5 It provides user plane tunnelling and tunnel management between SGW and PDN-GW, using GTP or PMIPv6 as the network based mobility management protocol.

S1-u Provides user plane tunnelling and inter eNodeB path switching during handover between eNodeB and SGW, using the GTP-U protocol

SGi It is the interface between the PDN-GW and the packet data network. Packet data network may be an operator external public or private packet data network or an intra operator packet data network.

Gx Bearer and flow control interface between the user data-plane element (PDN-Gw) and the Policy System. A Diameter based interface with a suite of 3GPP applications

4.2.2. 3GPP and 4V6 modes

4V6 translated traffic appears for all intents and purposes as regular IPv6-user traffic to the 3GPP system and packet processing functions (eg the PDN-Gw). Hence, and based on the stated assumptions, any such 4V6 traffic can be handled using existing native IPv6 functionality defined by the core 3GPP specifications.

In contrast, 4V6 tunneled traffic requires additional data plane processing to get to the "real" user IPv4 payload and apply the desired functions. Such additional processing is currently not part of the functionality covered by the 3GPP specifications. In view of this, and solely in relation to the 4V6 tunnel transport mode, two alternative hypotheses need to be placed in order to complete the comparison

i) that such IPv4 in IPv6 processing functionality will be supported as part of the existing EPS bearer functionality defined in E-UTRAN, perhaps as a dedicated EPS bearer (ie an additional virtual interface per subscriber). Or, that;

ii) a new 46 EPS bearer type (ie interface type) identification and signalling will be defined by the 3GPP architecture, which formalizes the v4inv6 relationship between the IPv4-user payload and the v6-user layers.

An apparent benefit of approach (ii) would be in allowing the system to clearly distinguish and expose to other systems v4-user traffic versus v6-user traffic, which is composed of v4inv6 and regular v6 traffic that a UE may generate. The former approach (i) is more convoluted given the ambiguity in distinguishing, and representing such a combination of v6-user and v6-user-bearer and v4-user traffic, all while keeping coherence in terms of the policy system. These two options are designated with ** in the table below.

Item	4V6 Translation Mode	4V6 Mapped Tunnel Mode
User Data Plane at the PDN-Gw (as per section 5.1.2 in [EUTRAN])	IPv6 over GTP-U over UDP over IP	IPv4 over IPv6 over GTP-U over UDP over IP
Gx (Diameter)	No discernible impact	Impacted: no way to express v4 over v6 in TFT Filter and Flow Descriptors
Rx (Diameter)	No discernible impact	Impacted: no way to express v4 over v6 in Media-Component-Description and, Flow-Description-AVP
S5 (GTP)	No impact	Impacted with new PDP/EPS Bearer type*
New 46 Bearer definition	Not required	Possibly required**

Secondary interface (dedicated bearer or secondary PDP) for 46 traffic	Not required	Possibly required**
PDN-Gw	No impact	New TFT capability, IP Gate functionality, changes to Gx, and likely changes to S5/S7 related to signalling the new bearer
SGw	No Impact	No discernible impact
PCRF	No impact for IPv6. Feature to map IPv4-IPv6 addresses needed only in case of IPv4-only applications.	Impacted for both IPv6 and IPv4-only applications and Gx applications utilizing flow control/charging
AF Application Function	No discernible impact	Flow based application control impacted
UE	4V6 application	4V6 application
LTE-Uu	No discernible impact	Likely changes required to support signalling of EPS bearer or PDP type
Lawful Intercept	No discernible impact	New rules for tunnel support

*A new PDP Type or EPS bearer signalling has a broader 3GPP system wide impact not fully covered here.

As the table illustrates, the 4V6 tunnel transport model appears to affect a significant number of 3GPP elements, when the intent if realize a full suite of services. This observation appears to apply to any other carrier inserted tunneling technology (eg DS-lite). Hence, a substantial investment in 3GPP standard terms and in the evolution of deployed systems appears to be required.

In contrast the 4V6 translation mode bears none to no discernible impact on existing 3GPP Release 8/9 specifications and their deployments, while allowing the operator to realize the full set of services on 4V6, alongside any native IPv6 traffic, allowed for by these architecture. Hence, little beyond the addition of 4V6 components operating using translation mode appears to be required.

4.3. Cable SP Architectures & 4V6 Applicability

Cable SPs (commonly referred to as Multi System Operators (MSOs)) usually deliver video, data, and voice service over the cable and fiber access to residential and commercial customers. Many MSOs offer SLAs with various services by exploiting QoS not only in their IP/MPLS network, but also their access network.

The cable access network (now synonymous with Hybrid Fiber Coax (HFC)) is commonly enabled with Data Over Cable Service Interface Specifications (DOCSIS, a CableLabs standard) to facilitate the implementation of packet based services. In this paradigm, the HFC/ DOCSIS access bandwidth is typically shared among a number of customers, hence, ensuring optimal service quality & experience per customer becomes extremely important for MSOs' success.

Cable SPs/MSOs ensure the optimal service quality of various advanced & real-time multimedia services (such as IP telephony, multimedia conferencing, interactive gaming etc.) by utilizing "PacketCable" framework to enforce QoS on the HFC/DOCSIS access.

The next sub-section 4.3.1 provides a brief introduction to PacketCable, section 4.3.2 explains a key PacketCable construct - Classifier, and section 4.3.3 tabulates the impact of 4V6 modes to PacketCable enabled DOCSIS/IP services.

4.3.1. PacketCable Introduction

PacketCable,a CableLabs standard, defines a framework for ensuring the Quality of Service (QoS) on the HFC/DOCSIS Access. PacketCable specifications (e.g. PacketCable 1.0, PacketCable Multi Media [PCMM], PacketCable Dynamic QoS [PC-DQOS], PacketCable 2.0) specify interoperable interface specifications for executing QoS, Admission Control, Accounting, Policy, and Security functions on Cable Modem (CM) and Cable Modem Termination System (CMTS), as/when needed. They all require DOCSIS 1.1 or later versions.

The PacketCable framework is also critically important for MSOs to comply with government regulations for things such as E911 when they offer voice/telephony services, Lawful Intercept (LI) etc.

The figure below illustrates one of PacketCable variants i.e. PCMM [PCMM] architecture, as an example, that defines a set of IP-based interfaces (referred to as pkt-mm-1 through 12) pertaining to core QoS and policy management capabilities.

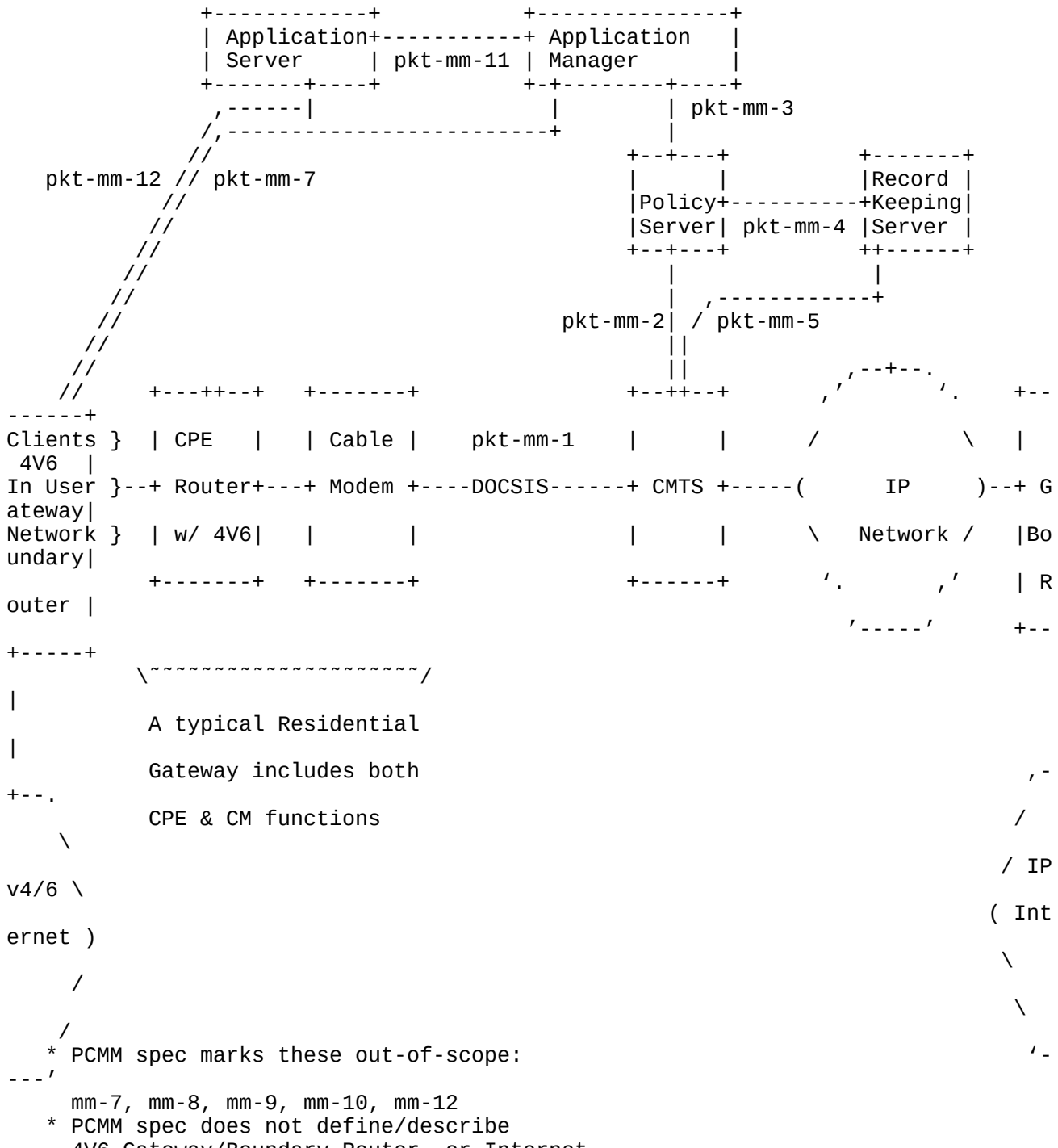


Figure 3 - PacketCable Multimedia Architecture (with 4V6)

4.3.2. PacketCable Construct - Classifier

PacketCable framework fundamentally relies on Cable Modem (CM) and Cable Modem Termination System (CMTS) to first qualify and then classify the appropriate IP traffic between them, for effective QoS enforcement. The framework requires the usage of "Classifier" for both qualification (in control plane) and classification (in data plane).

Taking PCMM specification [PCMM] again as an example, PCMM mandates the usage of classifier in the control plane (i.e. 'Upstream Packet Classification Encoding' in pkt-mm-1 interface (DOCSIS) , whereas 'Multimedia Classifier Object' in pkt-mm-2 and pkt-mm-3 interfaces (COPS)) for conveying the attributes of an IP flow belonging to an application (telephony, say), and subsequently its usage in the data plane i.e. filter matching on the IP packets' layer2/3/4 headers prior to QoS treatment.

The PCMM specification mandates the 'classifier' to include Source and Destination IP addresses, DSCP/TOS, IP Protocol, Source and Destination ports for an IPv4 traffic flow received by the CMTS, and similarly, Source and Destination IP addresses, TC, Next Header, Source and Destination ports for an IPv6 traffic flow received by the CMTS.

Similar to PCMM, PacketCable DQOS specification [PC-DQOS] also mandates the usage of classifier in the control plane (DSx messaging). In particular, PC-DQOS mandates the classifier definition to have 'protocol' (or next header) in IP header to be 17 (=UDP) along with specific Source and Destination ports (and Source and Destination IP addresses, optionally) so as to accommodate voice RTPoUDPoIP traffic.

In summary, the CMTS (and CM) construct their data-plane filter based on the 'classifier' information.

4.3.3. 4V6 Modes Impact on PacketCable

In 4V6 Tunnel mode, the 4V6 tunneled traffic requires additional data plane processing to get to the "real" user IPv4 payload and apply the desired functions. Such additional processing is currently not part of the functionality covered by the PacketCable specifications, nor part of compliant implementations.

In 4V6 Translation Mode, the 4V6 translated traffic appears for all intents and purposes as regular IPv6-user traffic to the PacketCable framework (both control plane and data plane). Hence, it is likely that any such 4V6 traffic can be handled using native IPv6

functionality e.g. classifier as defined by the PacketCable specifications and supported by CMTS and CM.

Taking PCMM specification as an example, it is worth noting that PCMM already allows for (and mandates) a minimum of four classifiers to be included in Gate-set. Hence, a Policy Server can communicate (via pkt-mm-2) both IPv4 and IPv6 classifier to the CMTS, which can use IPv6 classifier for constructing its data-plane filters (for DownStream processing), and convey IPv4 classifier to the CM via DOCSIS messages (pkt-mm-1) for any Upstream Processing. So, the 4V6 Translation Mode would work out in current implementations/deployment reasonably well.

Separately, it is likely that the CPE Router would be engaged in serving IPv4 multicast content to IPv6 receivers (and vice versa) in future, requiring 'translation' function.

In summary, while 4V6 Translation mode can work with the existing PacketCable framework, 4V6 Tunnel mode can not.

5. Overview of potential issues and discussion

This section summarizes the issues attributed to an A+P, or port restricted scheme, along with a discussion of applicability to the assumed system and possible resolutions. The summary of issues stem from [I-D.thaler-port-restricted-ip-issues] and associated discussions.

5.1. Notion of Unicast Address

5.1.1. Overview

The issue, referred to as the "definition of a unicast address", relates to the notion that in a shared IPv4 address system, multiple hosts will be visible as having a single IPv4 address outside of the system. This issue is a general characteristic of any NAPT44 based solution [I-D.ietf-intarea-shared-addressing-issues], including DS-Lite. However, a more specific aspect of this issue in the context of an address sharing system is the possibility that a single host having multiple interfaces will be assigned the same IPv4 address (with different port ranges) on each of its interfaces. It may also be that multiple hosts sharing an address find themselves on the same Layer 2 segment. Either would impede hosts from working within the notion of known host IP stack and protocol implementations.

5.1.2. Discussion

A number of the characteristics of the 4via6 solution architecture cause the issues not to be applicable, key of which is that there is no expectation for any kind of end hosts to be part of the shared IPv4 address system.

In the stateless 4via6 system, CPE nodes are assigned with a shared IPv4 address+port range by means of the unique IPv6 address, containing the embedded IPv4 address + port index, of that CPE node. The CPE node is in addition enabled to be running the port restricted NAPT44 function from the IPv6 derived address, a key characteristic of the solution. On the IPv6 plane, the IPv6 address of the CPE is practically indistinguishable from any "regular" IPv6 address, and in fact any host that is not aware of it conveying an embedded IPv4 address would be able to use this just like any other regular IPv6 address, ie the 4via6 solution uses standard IPv6 addressing. In terms of the IPv4 dimension, since the shared address and port index are never used to address native IPv4 nodes or hosts, but instead uniquely assigned to a single NAPT44 function that is part of the CPEs, all legacy or other IPv4 hosts are not exposed to the presented issues.

Going beyond the ascribed issue however, it appears desirable to have the 4via6 CPEs that are to be part of the shared system to be able to provide a hint to the network operator in terms of their special capability. Such a hint can be a DHCPv6 Option Request Option, which would be useful to allow the DHCPv6 sub-system to also inform the CPE of any other stateless 4via6 system parameters. A largely similar ORO option is currently being defined as part of [I-D.ietf-softwire-ds-lite-tunnel-option]

Recommendation: Define a suitable DHCPv6 ORO for conveying the 4via6 capability of a CPE.

5.2. Implementation on hosts

5.2.1. Overview

The issue, as presented, relates to the need for modifications on end hosts or devices to support a port constrained mechanism and the overall impossibility of realizing such modifications. Furthermore, host applications that attempt to bind to specific ports that are not part of the allowed port range will fail to do so and may also require modifications.

5.2.2. Discussion

As presented in Section 3 the solution assumes the use of a dedicated CPE implementing the 4via6 functionality within a port constrained mode and NAPT44. Granted, CPE nodes will require to implement new functionality such as the IPv6 adaptation function, that is likely alongside introducing native IPv6 support. However, any and all existing end user IPv4 devices (eg PCs, etc) will not affected. Nor are such devices expected to behave in any way different from that of today, where they typically obtain a private rfc1918 address and multiplexed by a CPE using a NAPT44 function.

In summary, the assumed 4via6 solution requires a specific 4via6 CPE but does not require any IPv4 host stack changes.

5.3. 4V6 address and impact on other IPv6 hosts

5.3.1. Overview

The issue relates to the question of whether the operation of a regular IPv6, non 4V6 capable, host would be adversely impacted should it be assigned or auto-configured with an address from an S64 address or prefix pool.

5.3.2. Discussion

The 4V6 prefix is for all intents and purposes a regular IPv6 prefix, and as such can be announced/assigned to any IPv6 host which in turn can used derived addresses like any other IPv6 address. Thus, an 4V6 IPv6 domain can address non-4V6 devices, leaving such devices to operate as native IPv6.

There is however a restriction on the 4V6 CE devices. As described in Section 2, a 4V6 CE constructs itself the full 128 bit address from the concatenation of the IPv6 prefix, 4V6 domain information acquired statelessly, and a pre-determined or algorithmic interface-id. By definition, only one 4V6 CE can use the same IPv4 address and port index. Hence, while there is no exact limitation on the number of non 4V6 hosts that can be addressed from an 4V6 prefix, there is a limit of one 4V6 CE per 4V6 prefix. Using a 4V6 prefix to address network segments without 4V6 devices does diminish the efficiency of the IPv4 address sharing mechanism, in terms of using up port ranges onto segments that will not use them. This is naturally a deployment consideration which an operator can optimize.

5.4. Impact on 4V6 CE based applications

5.4.1. Overview

It has been claimed that applications implemented on the CE itself, eg a DNS resolver-client, may be impacted by the 4V6 functionality. In particular, a concern is that such applications would either need to be specially engineered to issue socket calls or extensive IP stack modifications made to support them.

5.4.2. Discussion

By definition the 4V6 CE is an IPv6 capable device, and any IPv6 capable applications will be able to use the native IPv6 stack (note: IPv6 interface selection, is discussed in section 5.5). As such, the concern raised does not apply to applications that can be expected to support IPv6, and instead only to IPv4-only applications running on the 4V6 CE.

The shared IPv4 address is intended to be used only by the 4V6 CE function. This shared IPv4 address does not need to be assigned to an interface on the 4V6 CE and thus a target for potential applications. Any such applications running on the 4V6 will use any of the other (likely private) IPv4 address on the CE, which then will be routed to the 4V6 function this is applied post routing for the packets generated by these applications.

5.5. 4V6 interface

5.5.1. Overview

A 4V6 CE will have a "self configured" 4V6 IPv6 interface address, alongside any other SLAAC or DHCPv6 derived addresses, potentially from the same prefix. This particular 4V6 address may be subject to specific filtering rules or restrictions by the operator, besides usage and filtering restrictions on the 4V6 CE. Also, for the 4V6 system to operate as intended, the 4V6 application on the CE must be restricted to using the specific 4V6 address when sourcing 4V6 packets. Also, the 4V6 CE needs to be set-up to correctly forward IPv4 traffic to the 4V6 application.

5.5.2. Discussion

While the method of creating the interface is implementation specific, the generic operating model that is envisaged is for the 4V6 application to create the 4V6 interface as a virtual interface with an IPv4 unnumbered address. The application would then install a default IPv4 route pointing to this virtual interface, which would

be effectively see the 4V6 application acting as a network appliance on the forwarded traffic. In terms of IPv6 behaviour, the 4V6 application is expected to be set up to specify the use (binding) to the 4v6 IPv6 virtual interface.

5.6. Non TCP/UDP port based IP protocols - ICMP)

5.6.1. Overview

This issue relates to the inability of using regular ICMP messages to "ping" an end-host that has been addressed with a shared IPv4 address. The issue can be generalized one applicable to any IP protocol that is not TCP/UDP port based, and also in terms of the ability of using such protocols from end hosts that are assigned a shared IPv4 address.

5.6.2. Discussion

The inability to ping a CPE from the IPv4 Internet is shared by other IPv4 address sharing mechanisms such as DS-Lite. Thus, the issue is no better or worse in the case of the stateless 4via6 solution. The same can be said of end user hosts using other non UDP/TCP port based protocols from behind a NAPT44 function, ie they will not function irrespective of address sharing or not.

As discussed in [I-D.ietf-intarea-shared-addressing-issues], all IP address sharing solutions break protocols which do not use transport numbers. A mitigation solution is to utilize specific ALGs. For ICMP in particular, a mitigation solution would be to rewrite the "Identifier" and perhaps "Sequence Number" fields in the ICMP request, treating them as if they were port numbers.

As a conclusion, this issue can be partially mitigated, likely at par to centralized NAT solutions.

5.7. Provisioning and Operational Systems

5.7.1. Overview

The general claim of this issue is that a service providers' provisioning and accounting systems would need to [radically] evolve to deal with the notions of shared IPv4 addresses and port range constrains.

5.7.2. Discussion

The stateless 4via6 solution relies on a fully operational IPv6 network, which on the IPv6 plane fundamentally does not differ from a

regular IPv6 network, and the stateless 4via6 solution may be seen as an IPv6 application - devices connecting to the network, need unique IPv6 addresses which the network is able to provide. In the 4via6 solution it happens that these unique IPv6 addresses embed an IPv4 address. Hence, additional system enhancements that the stateless 4via6 solution requires, over and above those simply needed to deploy and operate an IPv6 network, lie in the domain of supporting the provisioning of the IPv6 adaptation functionality of the CPEs. This may require the operator to use DHCPv6, or other provisioning methods such as IPv6CP, TR-69, in order to configure any relevant 4via6 service parameters to a CPE.

From an IPv4 perspective, an operator will likely want to have a management system capable of the assignment of IPv4 addresses to the shared pool, and tuning the re-use factor. In this, the solution exhibits no grossly different characteristics than those of any system with an operator managed NAT44 function where similar management capabilities need to be introduced.

One additional aspect of the stateless 4via6 solution needs to be highlighted. On a par basis this solution requires less per subscriber management, accounting and logging capabilities than centralized NAPT44 alternatives such as DS-Lite, due to the following:

- The assignment of an IPv6 address that embeds a deterministic IPv4 address and port range removes the need for the operator to perform any NAPT44 binding logging, ie the task of determining which user had a given IPv4 address and port at a given time is simply a matter of determining who had the corresponding IPv6 address, rather than collecting large amounts of dynamic binding data.

- There is no need for the operator to manage NAPT44 binding data access and retention.

- Given the stateless nature of the 4via6 solution, all subscriber CPEs in an operator’s domain can share exactly the same 4via6 service configuration, i.e. The operator does not need to be concerned with managing on a per user basis specific AFTR assignment and/or load balancing such users and throughout ensuring symmetric traffic flows throughout.

- The location of the NAPT44 function on the user’s CPE, allows easy and direct management of the port mappings by the end user removing a need for the operator to introduce PCP [I-D.wing-softwire-port-control-protocol] (or similar) protocols in on AFTRs, and on CPE devices. In effect the end user can

retains control of any bindings, which could be via today's GUI, or UPnP IGDv2, or even PCP.

- o As and when needed, a stateless 4via6 solution readily supports the assignment of an unshared IPv4 address, and full port control by the end user. A similar capability with centralised NAPT44 solutions involve onerous management of per subscriber configurations on the operator's AFTR.

5.8. Training & Education

5.8.1. Overview

The issue claims a concern with the need for developers and support staff to be trained & educated in dealing with a port constrained systems.

5.8.2. Discussion

There appear to be at least two levels of looking at this issue in the stateless 4via6 context. On one level, it is perfectly true that developers and support staff will need to be trained with running/ supporting a native IPv6 network, that is now a basis of the solution. This however is an inherent aspect of deploying an IPv6 network and applications. On another level, support and developers need to familiarized with the NAPT44 characteristics of the system, that are not different from those already known about such systems today. More specifically, there appears to be no such thing as a port unconstrained carrier grade NAPT44 system, in either tomorrow's stateless 4via6 or AFTR guises, or today's residential CPE NAPT44 implementations that have an inherent hard set translation limit (often 1024 translation, corresponding to a usage of 1024 ports). That application developers should be trained to be reasonably conservative in the usage of ports is thus not an issue of the stateless 4via6 solution, but pretty much of any NAPT44 based solution, even those in use today.

Another useful observation here is that the stateless 4via6 solution, actually allows an operator to retain existing troubleshooting procedures, given which today encompass CPE based NAPT44, rather than changing them radically to an AFTR. Furthermore, it is possible to alleviate any port-range constrains for users by allocating more generous port ranges without the need to manage such users configuration on active core network devices (eg AFTR).

5.9. Security and Port Randomization

5.9.1. Overview

Preserving port randomization [RFC6056] may be more or less difficult depending on the address sharing ratio (i.e., the size of the port space assigned to a CPE). Port randomization may be more difficult to achieve with a stateless solution than stateful solution. The CPE can only randomize the ports inside be assigned a fixed port range.

5.9.2. Discussion

The difference in the random port selection range may be significant in practice and using port-restricted systems without any measures (like random port selection in draft-bajko-pripaddrassign-03) is one of the trade-offs of the mechanism. It should be however noted that even full port unrestricted systems, today, rarely implement random port selection from the full port range, as such the difference is largely theoretical, again viewed from today’s perspective. Only with a longer term prospect of devices/hosts adopting random port selection according to RFC 6056 the NAT-based port-restricted mechanisms, will degrade security to a certain extent.

5.10. Unknown Failure Modes

5.10.1. Overview

The issue purports that a system with a port constraints introduces new unknown failure modes, not known with NAT44 or NAPT44 systems, and in general is more complex than such a system.

5.10.2. Discussion

This claim does not appear to have objective technical arguments that can be discussed. A restricted port range system, such as the one assumed in this document, does not appear to have any more or less complexity than any of the other NAPT44 solutions against which the same issue has not been levelled. That is a statement that can be made in consideration of each of those alternative solution network design (eg elaborate routing rules or topologies) and feature implementation complexities, which appear to be no better than that of a stateless 4via6 address port range system. Ultimately, system complexity is something best left adjudicated by the operators choosing to deploy one or the other of these IP based transition solutions.

5.11. Possible Impact on NAT66 use & design

5.11.1. Overview

The notion of a shared address with a constrained port range is seen as possibly bearing influence on use in future schemes involving NAT66, where IPv6 address sharing is in general deemed not to be desired (ie there is good reason to avoid PAT66).

5.11.2. Discussion

The authors do not propose, nor expect to see the IP address sharing characteristic applying to future NAT66/PAT66 discussions and specification. However, having said that it is useful to take a humble step back and consider the general aspect of causality in this context. The direct cause that brought about IPv4 shared address solutions to the fore was a shortage/exhaustion of a limited IPv4 address resource, alongside a failure of the community to migrate IPv4 networks to IPv6 in a timely manner. At the time of writing it is hard to imagine the same occurring with respect to IPv6 address resources, and hopefully the same set of causes will not be allowed to re-occur. This appears to be the only way to ensure that IPv6 address sharing effect does not come to be, as opposed to precluding such notions within the context of today’s IPv4 world where the causality is rather clear.

5.12. Port statistical multiplexing and monetization of port space

5.12.1. Overview

An issue attributed to 4V6 solutions is that due to their characteristic of assigning a fixed amount of ports to participating system nodes, the overall pool of ports cannot be dynamically/ statistically multiplexed.

A corollary of this claimed issue is the claim that port range constraints will lead to monetization by service providers of such port ranges, for example by charging users based on the number of ports assigned or creating some bronze, silver, gold type of port based service categories.

5.12.2. Discussion

The 4via6 address shared solution indeed limits the ability to "overload" ie statistically multiplex amongst users, the ports available of a given public IPv4 address. This can be seen as a trade off vs dynamic allocation and the need to log (large amounts) of NAT bindings. Furthermore, the solution is meant to be

fundamentally a transitional one for supporting legacy IPv4 users till full migration to IPv6 can occur. As an example, even with a static allocation of ~1000 ports per shared IP user, it allows an operator to effectively multiply by ~64 the current IPv4 unrealizable address space. To put it into a network growth perspective, it allows an operator to support for some 10 years a steady 50% annual increase in users, without requiring new IPv4 addresses. This is likely an alluring (if unlikely) prospect for most, but it demonstrates the fact that even with static port allocations, IPv4 address sharing can go a long way for many operators.

CGN-based solutions, because they can dynamically assign ports, provide better IPv4 address sharing ratio than stateless solutions (i.e., can share the same IP address among a larger number of customers). For Service Providers who desire an aggressive IPv4 address sharing, a CGN-based solution is more suitable than the stateless. However, in case a CGN pre-allocates port ranges, for instance to alleviate traceability complexity it also reduces its port utilization efficiency.

5.13. Readdressing

5.13.1. Overview

Due to the port range encoding being part of the CPE's IPv6 address, any change in the range requires a re-configuration of the CPEs 4via6 address. This is said to be an issue given the impact that IP address changes have on existing traffic flows, as well as general IPv6 network routing

5.13.2. Discussion

It is true that under the assumed notions of the stateless 4via6 solution, IPv6 re-addressing is required to effect a change in terms of the shared IPv4 address or ports. Such changes can and are likely best done using dynamic address configuration methods such as DHCPv6, or alternatively out of band management tools, eg TR-69, especially when the 4via6 address can be derived from a delegated prefix. Using these, the impact of the address change does not translate to a neither a classic IPv6 host renumbering problem nor an unmanageable network renumbering problem. On the CPE, the change only affects the 4via6 address of the CPE and not any end user IPv6 hosts behind the CPE (which would likely continue to derive their IPv6 addresses from an unchanged delegated prefix). On the service provider network side, the change, if any, represents a network renumbering case which the operator can be reasonably expected to handle within their network numbering plan, especially given that the IPv6-prefix of the an IPv4-in-IPv6 address is summarizable.

An addressing change will impacting any existing IPv4 flows that are being NAT'ed by the CPE. This is also analogous to the today's practice of IPv4 address changes espoused by some operators, which while not being commendable, is established in the market. Nevertheless, as a means of alleviating such an impact it appears desirable for the solutions to investigate the viability of mechanisms that could allow for more graceful addressing changes.

To facilitate IPv6 summarization and operator appears to have two 4V6 deployment choices. When encoding IPv4 addresses in lower order address space bits that are subject to summarization,the operator would need to assign a modest dedicated IPv6 prefix (such as a /64) as an 4V6 IPv6 addressing sub-domain. Alternatively, without resorting to a separate 4V6 addressing sub-domain, an operator could allow for the IPv4 address embedding to be embedded in a high-order portion of the IPv6 domain address space, one that closely follows the IPv6 domain prefix. These two valid address subnetting and deployment options deserve better description in the solution specifications.

5.14. Ambiguity about communication between devices sharing an IP address.

5.14.1. Overview

A regular IPv4 destination based routed system inherently does not allow two devices to communicate while sharing the same IPv4 address, even if with different ports. Similarly, such a system does not allow on the basis of a IPv4 source address alone to perform address spoofing prevention. These two issues naturally render regular IPv4 based routed networks incapable of supporting a shared address solution.

5.14.2. Discussion

In terms of the IPv4 data plane of the 4via6 solution, the CPE and the stateless gateway components need to be modified in terms of their IPv4 forwarding behaviour. The CPE's NAPT44 function, must be capable of sending traffic towards the IPv6 adaptation function when the traffic is addressed to its (shared) IPv4 address but a different port than the one assigned to the CPE. Similarly, the CPE's NAPT44 function must be capable of receiving traffic addressed from its (shared) IPv4 address but a different port than the one assigned to it.

On the IPv6 data plane the stateless 4via6 solution does not suffer from the issue by the nature of relying on regular IPv6 forwarding. Address-spoofing security can be realized on regular IPv6 devices

plane, in a way which effectively does not allow a CPE to send IPv6 traffic from a source IPv6 address that it has not been assigned. The spoofing of IPv4 addresses can be prevented in this manner in 4via6 solution relying on translation (dIVI). Tunneling 4via6 solutions (4rd) require IPv6+IPv4 source address validation to be performed at the CPE and stateless gateway, by the IPv6 adaptation function.

The conceptual IPv6 adaptation function has many of its core principles already defined either as part of IPinIP tunneling or stateless NAT64 drafts. However additional work, such as defining the port indexing schemes, is needed and is at the heart of what needs to be covered in the individual solution drafts that fall under the stateless 4via6 family. Throughout, no legacy IPv4 end-systems are expected to implement these techniques.

5.15. Other

5.15.1. Abuse Claims

Because the IPv4 address is shared between several customers, and in order to meet the traceability requirement discussed in Section 12 of [I-D.ietf-intarea-shared-addressing-issues], Service Providers must store the assigned ports in addition to the IPv4 address.

If the remote server does not implement the recommendation detailed in [I-D.ietf-intarea-server-logging-recommendations], the Service Provider may be obliged to reveal the identity of all customers sharing the same IP address at a given time.

5.15.2. Fragmentation and Traffic Asymmetry

In order to deliver a fragmented IPv4 packet to its final destination, among those having the same IPv4 address, a dedicated procedure similar to the one defined in Section 3.5 of [RFC6146] is required to reassemble the fragments in order to look at the destination port number.

When several stateless IPv4/IPv6 interconnection nodes are deployed, and because of traffic asymmetry, situations where fragments are not handled by the same stateless IPv4/IPv6 interconnection node may occur. Such context would lead to session breakdowns. As a mitigation, a solution would be to redirect fragments towards a given node which will be responsible for implementing the procedure documented in [RFC6146]. The redirection procedure is stateless.

As a conclusion, this issue can be mitigated.

5.15.3. Multicast Services

IPv4 service continuity must be guaranteed during the transition period, including the delivery of multicast-based services such as IPTV. Because only an IPv6 prefix will be provided to a CPE, dedicated functions are required to be enabled for the delivery of legacy multicast services to IPv4 receivers. This is critical since many of the current IPTV contents are likely to remain IPv4-formatted and there will remain legacy receivers (e.g., IPv4-only Set Top Boxes (STB)) that can't be upgraded or be easily replaced.

This issue is similar to the one encountered in the stateful case, and the same solution can be used to mitigate the issue (e.g., [I-D.qin-softwire-dslite-multicast]).

As a conclusion, this issue can be solved.

6. Conclusion

As per the discussion in this document, the authors believe that the set of issues specifically attributed to A+P based such as the stateless 4via6 solution with characteristics as per Section 3, either do not apply, or can be mitigated. In several aspects, a stateless 4V6 solution represents a reasonable trade off compared to alternatives in areas such as NAT logging, ease as of deployment and operations, all of which are actually facilitated by such a solution.

In terms of the 4V6 transport mode, both translation and mapped tunnel appear to be share the same key characteristics, but applicable to different contexts. The mapped tunnel mode appears desirable when the operator has no expectations of applying any more elaborate traffic based services, and/or concerned about the loss of IP Options or the use of NAT64 technology. The translation based approach appears particularly attractive to operators who are concerned about integrating traffic into a more elaborate suite of services based on regular IPv6 data-plane functionality, as opposed to specific IPinIP data plane functionality.

7. IANA Considerations

This document does not raise any IANA considerations.

8. Security Considerations

This document does not introduce any security considerations over and

above those already covered by the referenced solution drafts.

9. Contributors and Acknowledgements

The authors thank Dan Wing, Nejc Skoberne, Remi Depres, Xing Li, Jan Zorz, Satoru Matsushima, Mohamed Boucadair, Qiong Sun, and Arkadiusz Kaliwoda for their reviews and draft input.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

10.2. Informative References

[I-D.bajko-pripaddrassign]
Bajko, G., Savolainen, T., Boucadair, M., and P. Levis, "Port Restricted IP Address Assignment", draft-bajko-pripaddrassign-03 (work in progress), September 2010.

[I-D.despres-softwire-4rd]
Despres, R., "IPv4 Residual Deployment across IPv6-Service networks (4rd) A NAT-less solution", draft-despres-softwire-4rd-00 (work in progress), October 2010.

[I-D.ietf-intarea-server-logging-recommendations]
Durand, A., Gashinsky, I., Lee, D., and S. Sheppard, "Logging recommendations for Internet facing servers", draft-ietf-intarea-server-logging-recommendations-04 (work in progress), April 2011.

[I-D.ietf-intarea-shared-addressing-issues]
Ford, M., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Issues with IP Address Sharing", draft-ietf-intarea-shared-addressing-issues-05 (work in progress), March 2011.

[I-D.ietf-softwire-ds-lite-tunnel-option]
Hankins, D. and T. Mrugalski, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Option for Dual- Stack Lite", draft-ietf-softwire-ds-lite-tunnel-option-10 (work in progress), March 2011.

[I-D.ietf-softwire-dual-stack-lite]
Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", draft-ietf-softwire-dual-stack-lite-11 (work in progress), May 2011.

[I-D.murakami-softwire-4v6-translation]
Murakami, T., Chen, G., Deng, H., Dec, W., and S. Matsushima, "4via6 Stateless Translation", draft-murakami-softwire-4v6-translation-00 (work in progress), July 2011.

[I-D.operators-softwire-stateless-4v6-motivation]
Boucadair, M., Matsushima, S., Lee, Y., Bonness, O., Borges, I., and G. Chen, "Motivations for Stateless IPv4 over IPv6 Migration Solutions", draft-operators-softwire-stateless-4v6-motivation-02 (work in progress), June 2011.

[I-D.qin-softwire-dslite-multicast]
Wang, Q., Qin, J., Boucadair, M., Jacquenet, C., and Y. Lee, "Multicast Extensions to DS-Lite Technique in Broadband Deployments", draft-qin-softwire-dslite-multicast-04 (work in progress), June 2011.

[I-D.thaler-port-restricted-ip-issues]
Thaler, D., "Issues With Port-Restricted IP Addresses", draft-thaler-port-restricted-ip-issues-00 (work in progress), February 2010.

[I-D.vixie-dnsext-dns0x20]
Vixie, P. and D. Dagon, "Use of Bit 0x20 in DNS Labels to Improve Transaction Identity", draft-vixie-dnsext-dns0x20-00 (work in progress), March 2008.

[I-D.wing-softwire-port-control-protocol]
Wing, D., Penno, R., and M. Boucadair, "Pinhole Control Protocol (PCP)", draft-wing-softwire-port-control-protocol-02 (work in progress), July 2010.

[I-D.xli-behave-divi]
Bao, C., Li, X., Zhai, Y., and W. Shang, "dIVI: Dual-Stateless IPv4/IPv6 Translation", draft-xli-behave-divi-03 (work in progress), July 2011.

[I-D.xli-behave-divi-pd]
Li, X., Bao, C., Dec, W., Asati, R., Xie, C., and Q. Sun, "dIVI-pd: Dual-Stateless IPv4/IPv6 Translation with Prefix Delegation", draft-xli-behave-divi-pd-01 (work in progress), September 2011.

[I-D.ymbk-aplusp]
Bush, R., "The A+P Approach to the IPv4 Address Shortage", draft-ymbk-aplusp-10 (work in progress), May 2011.

[RFC5961] Ramaiah, A., Stewart, R., and M. Dalal, "Improving TCP's Robustness to Blind In-Window Attacks", RFC 5961, August 2010.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, October 2010.

[RFC6056] Larsen, M. and F. Gont, "Recommendations for Transport-Protocol Port Randomization", BCP 156, RFC 6056, January 2011.

[RFC6145] Li, X., Bao, C., and F. Baker, "IP/ICMP Translation Algorithm", RFC 6145, April 2011.

[RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, April 2011.

Authors' Addresses

Wojciech Dec
Cisco
Haarlerbergweg 13-19
1101 CH Amsterdam
The Netherlands

Email: wdec@cisco.com

Rajiv Asati
Cisco
Raleigh, NC
USA

Phone:
Fax:
Email: rajiva@cisco.com
URI:

Congxiao Bao
CERNET Center/Tsinghua University
Room 225, Main Building, Tsinghua University
Beijing, 100084
CN

Phone: +86 10-62785983
Fax:
Email: congxiao@cernet.edu.cn
URI:

Hui Deng
China Mobile
Beijing,
CN

Phone:
Fax:
Email: denghui@chinamobile.com
URI:

Mohamed Boucadair
France Telecom
France

Phone:
Fax:
Email: mohamed.boucadair@orange-ftgroup.com
URI:

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: March 16, 2014

K. Fleischhauer, Ed.
O. Bonness
Deutsche Telekom AG

September 12, 2013

draft-fleischhauer-ipv4-addr-saving-05
On demand IPv4 address provisioning in Dual-Stack PPP deployment scenarios

Abstract

Today the Dual-Stack approach is the most straightforward and the most common way for introducing IPv6 into existing systems and networks. However a typical drawback of implementing Dual-Stack is that each node will still require at least one IPv4 address. Hence, solely deploying Dual-Stack does not provide a sufficient solution to the IPv4 address exhaustion problem. Assuming a situation where most of the IP communication (e.g. always-on, VoIP etc.) can be provided via IPv6, the usage of public IPv4 addresses can significantly be reduced and the unused public IPv4 addresses can under certain circumstances be returned to the public IPv4 address pool of the service provider. New Dual-Stack enabled services can be introduced without increasing the public IPv4 address demand, whereas IPv6 will be the preferred network layer protocol. This document describes such a solution in a Dual-Stack PPP session network scenario and explains the protocol mechanisms which are used.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 16, 2014.

Copyright Notice

Table of Contents

1. Abstract

The Dual-Stack approach as defined in [RFC4213] provides the most straightforward and most common way for introducing IPv6 [RFC2460] into existing systems and networks. However, an inherent drawback of usual Dual-Stack deployment scenarios according to [RFC4213] section 2 is that network nodes will still require at least one IPv4 [RFC0791] address. A primary concern for most operators whose IPv6 deployment strategy relies upon the deployment of Dual-Stack architectures is hence focused on the ability to rationalize the usage of its global IPv4 address blocks while encouraging the use of IPv6.

Assuming now a situation where most of the IP communication (e.g. always-on, VoIP, etc.) can be provided via IPv6, the usage of public IPv4 addresses can be reduced significantly and the operators need mechanisms and solutions in order to release unused IPv4 address resources of Dual-Stack nodes and reallocate them later on, on demand. This document describes how such a solution can be deployed in a Dual-Stack PPP session scenario and details the protocol mechanisms of the solution which are also thought as contribution to [BBF-TR-242]. Furthermore it should be mentioned at this point that the sketched solution approach can also serve as general IPv4 sun setting approach for Dual-Stack PPP sessions, since it provides the possibility to return unused IPv4 addresses of Dual-Stack PPP sessions and transforming them into pure single stack IPv6 PPP sessions.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119]

2. Problem Statement and Purpose of IPv4 address efficiency

The Broadband Forum describes in [BBF-TR-187] a target IPv4/IPv6 Dual-Stack Architecture. TR-187 builds on the capabilities of

existing protocols such as Point-to-Point Protocol (PPP) [RFC1661] and Layer 2 Tunnelling Protocol (L2TP) [RFC2661] to provide IPv6 service in addition to today's IPv4 service. These protocols allow the parallel usage of IPv4 and IPv6 within a single PPP respectively L2TP session. Usually in such a scenario the service provider assigns both, a global IPv4 address and also IPv6 address/prefix parameter, to the CPE deployed in the customer's premises for the whole duration of the PPP session. Because of the potential parallel usage of IPv4 and IPv6 within such a Dual-Stack PPP scenario a public IPv4 address is always provisioned, also in the (future) case where it is assumed that most (or even all) of the communication is running on top of IPv6. This document extends the sketched Dual-Stack deployment scenario for PPP and L2TPv2 with a mechanism that allows a temporary assignment and a release of an unused IPv4 address. This approach covers also situations where the IPv4 address may only be provided on-demand later on, after initiating the Dual-Stack PPP session with an IPv6 context only. For a service provider using this mechanism it is assumed that a valuable increase of IPv4 address efficiency due to a time based sharing of complete IPv4 addresses can be achieved.

Basically, the mechanism is also applicable to cable and mobile networks. The corresponding DOCSIS and 3GPP standards may be adapted as a follow-on work to this draft later on.

2.1. Illustrative service provider use case

In order to illustrate the applicability and usefulness of the proposed "On demand IPv4 address provisioning" mechanism an illustrative network operator use case is provided in this section. Let's assume a network access and service provider which is offering Dual-Stack services via a single PPP connection to its customers, hence assuming a PPP encapsulation scheme. Independently of the nature and the number of services subscribed by the customer, (Single, Play, Double Play etc.), all customers should be produced and provisioned in the same way in order to keep the network operation costs and the network complexity as low as possible. Let's assume furthermore that the above mentioned network access and service provider has already migrated its VoIP service to IPv6, so that all Single play VoIP customers only need IPv6 connectivity and have no need for an IPv4 context within their Dual-Stack PPP session. However, the standard Dual-Stack PPP connection set-up today assumes the triggering of the IPCP negotiation phase, as well as an IPv6CP negotiation independently of the real need for IPv4 and/or IPv6 connectivity, so that after a successful Dual-Stack PPP connection establishment the PPP client site is provisioned with a complete set of IPv6 and IPv4 connection parameters. As a consequence in our example, the whole Single Play VoIP customer base of the network

access and service provider has also been provisioned with public IPv4 addresses, although these customers will never need IPv4 Internet connectivity during the whole lifetime of their PPP session. Hence a huge amount of not required and therefore unused IPv4 addresses has been wasted, that should be better kept in the provider address pools and delegated to other customers that really need IPv4 connectivity. In order to allow a more dynamic and on-demand provisioning of IPv4 parameters within Dual-Stack PPP sessions, a new mechanism is needed, that requests and also releases IPv4 addresses on-demand when they are really needed during the PPP session lifetime. Such a mechanism is proposed and described within this document.

(An additional advantage of such an on-demand IPv4 address releasing and provisioning mechanism consists in the fact that a straight-forward to operate and dynamic change in the customer profiles (e.g. upgrade of Single Play customers to Double Play services and vice versa) becomes possible with only minor changes to the customer service profile in the AAA platform of the service provider - no changes in the CPE or BRAS/NAS port configuration are needed. Besides that, this dynamic on-demand IPv4 address provisioning and releasing approach allows it to share one public IPv4 address in a timely sequential fashion between a bunch of customers.)

The following sections describe the basic network architecture and the "On demand IPv4 address provisioning" mechanisms in more details.

2.2. Architecture and Communication in a PPP Dual-Stack environment

Assuming a Dual-Stack network access via PPP, terminal devices can communicate via IPv4 and/or IPv6 transport, depending on their own and their IP communication partner capabilities. The actual usage of IPv4 or IPv6 or both protocols depends on the capabilities of

- the IP communication endpoints (e.g. protocol stack, applications, configuration of the preferences etc.),
- the network deployment itself (e.g. access network based on PPP, backbone network, Internet) and also on
- the used communication services (like e.g. VoIP over IPv6).

The last two points are mainly left to the responsibility of the network and service providers. The approach, sketched in this document, is based on the operational scenario that the customer starts a Dual-Stack PPP session in "IPv6-only" mode first and "adds" IPv4 later on only in the case that applications or services explicitly require IPv4 connectivity. When IPv4 connectivity is not

needed during the whole duration of PPP network connectivity then a continuous provisioning of a global IPv4 address to the customer device (e.g. end system, CPE etc.) is not necessary. Therefore mechanisms are needed to provision and release public IPv4 addresses for Dual-Stack PPP sessions dynamically and on-demand.

The goal of the solution sketched in this document, is to limit and decrease the public IPv4 address pool size of the PPP network access provider and hence to better rationalize the usage of the remaining IPv4 address blocks. Assuming that always-on services are reachable via IPv6, a Dual-Stack-capable PPP connected customer side device should in any case request IPv4 address parameters only on demand, when the need for establishing IPv4 connectivity has been detected and there is a need to forward IPv4 traffic towards the PPP WAN interface (e.g. of a CPE). Following this above sketched network scenario it is sufficient, when initially only IPv6 address parameters are provisioned to the PPP customer endpoint (e.g., end systems, CPE).

This means as a consequence that a customer device does not initially start a complete Dual-Stack PPP session but an IPv6-only PPP session. The IPv4 part of the complete Dual-Stack is initiated later on only in the case that IPv4 connectivity is explicitly requested.

Figure 1 below illustrates the network architecture of a PPP Dual-Stack environment for providing Internet access to residential customers.

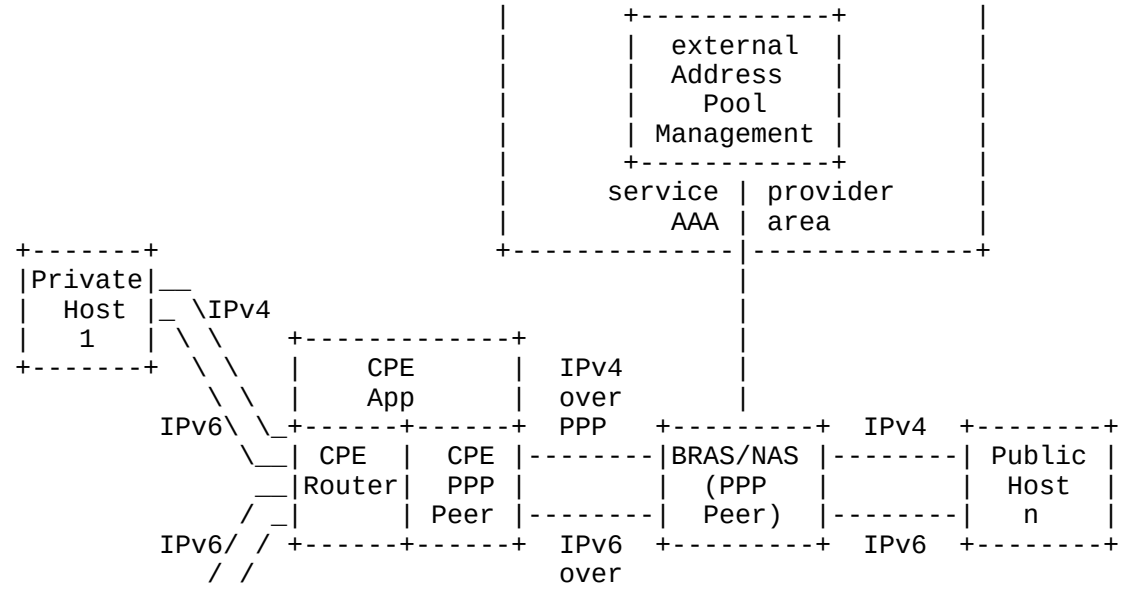


Figure 1: PPP Dual-Stack architecture

This abstract network topology consists of 3 major components:

1. Private Internet (aka. Customer LAN)

2. Public Internet (including access and service provider network)

3. Service Provider AAA area

The focus of this draft is mainly directed to the access network of the service provider as part of the Public Internet, where in our scenario PPP is used between the CPE and the provider Network Access Server (BRAS, NAS) in order to provide public Internet access to the customer.

The Service Provider's AAA area is a network which consists of several systems that interact with the Network Access Servers and provide AAA functionalities. Such Service Provider AAA functionalities also include management of the public IPv4 and public IPv6 address and prefix pools inside the BRAS/NAS and can also be integrated directly into the BRAS/NAS.

2.3. The advantage of the dynamic IPv4 address assigning feature

The dynamic IPv4 address assigning approach, sketched in this document, is based on the operational approach that the customer CPE initiates a PPP session based on IPv6 and adds IPv4 later on only if certain IPv4 applications or services explicitly require IPv4 connectivity. A particular public IPv4 address can therefore be assigned consecutively to different customers for the lifetime of their IPv4 PPP connection and has not to be bound to a single customer for the whole lifetime of the Dual-Stack PPP session. This consecutive assignment of public IPv4 addresses allows from a provider perspective a less complex IPv4-to-IPv6 migration in comparison to other IPv4-to-IPv6 migration approaches that are based on Carrier Grade NATs in service provider network (like e.g. Dual-Stack lite (like e.g. Dual-Stack lite [RFC6333]) or shared IPv4 addresses, since no additional network devices have to be deployed

and operated and the complete solution is based on simple extensions to already existing infrastructure components and processes. The customer will be provisioned with a public IPv4 address only in the case when global IPv4 connectivity is really needed and will not be provisioned with an IPv4 address by default when the Dual-Stack PPP session is initiated. Furthermore, a provisioned IPv4 address can be released (e.g., after a certain time interval) in case the CPE detects that there is no need any more for global IPv4 connectivity. In other words, when global IPv4 connectivity is not needed during the lifetime of the Dual-Stack PPP session then a (continuous) provisioning of a public IPv4 address to the CPE is not necessary and the provisioning of a public IPv4 address can be done on-demand and dynamically.

Hence, one of the main achievements of this mechanism is to limit and decrease the pool size for public IPv4 addresses at the service provider site.

A similar effect in limiting and decreasing the IPv4 address demand can also be reached by using separate PPP sessions for IPv4 and IPv6. But in that case the following problems occur:

- For each additional PPP session additional AAA parameters have to be created and handled which leads to an extension of AAA domains and more complex processes.

- Each additional PPP session will require additional resources on the PPP endpoints (e.g. for handling additional customer credentials) also in devices that act as PPP intermediate agents.

- Accounting and controlling of traffic classes on an access line or customer base will be impeded or at least complicated.

Because of these reasons the introduction of an additional PPP session for IPv6 as additional network layer protocol on an access line with an additional PPP session is not recommended.

From a strategic perspective the dynamic IPv4 address assigning approach complements a Dual-Stack based IPv6 migration strategy for service provider access networks which may consist the following stages:

1. Implementation of IPv6 in the access network based on the Dual-Stack approach.

2. Completing the IPv6 introduction for all services which are under the control of the service provider.

3. Implementation of the dynamic IPv4 address assigning mechanism.

4. Monitoring the IPv4 usage and analyzing opportunities for stage 5.

5. Implementation of IPv6-only access products.

It is possible to realize stage 2 also at an earlier or later point in time. To reach a maximum effectiveness regarding IPv4 address efficiency it is recommended to keep this sequence.

3. Specification

As defined in RFC 2661 [RFC2661] PPP and L2TP provide the following main functionalities:

1. A method for encapsulating datagrams over serial links.

2. A Link Control Protocol (LCP) for establishing, configuring, and testing the data-link connection.

3. (Optional) Authentication Protocol for one or both peers.

4. A family of Network Control Protocols (NCPs) for establishing and configuring different network-layer protocols.

For provisioning of IPv4 or IPv6 communication parameters (e.g. addresses, DNS resolver) as network-layer protocols only the NCPs Internet Protocol (Version 4) Control Protocol (IPCP) RFC 1661 [RFC1661] and Internet Protocol (Version 6) Control Protocol (IPv6CP) RFC 2472 [RFC2472] are used. Whereas IPCP is responsible for configuring, enabling, and disabling the IPv4 protocol modules on both ends of the point-to-point link, IPv6CP is responsible for configuring, enabling, and disabling the IPv6 protocol modules on both ends of the point-to-point link. Once one of the two network-layer protocols has been configured, datagrams belonging to this network-layer protocol can be sent over the PPP link. Both NCP protocol mechanisms act independently of each other (see also requirement WLL-3 in [RFC6204]) and can be used to establish and pull-down IPv4 and IPv6 connection contexts within a Dual-Stack PPP session independently.

As an example, an implementation that wishes to close a dedicated NCP connection (e.g., IPCP or IPv6CP) SHOULD transmit a Terminate-Request to the peer. Upon reception of a Terminate-Request, a Terminate-Ack MUST be transmitted to the sender of the Terminate-Request. The PPP session itself and the other NCP connection inside the PPP session will remain existent. Only in the case that both NCP connections are closed, the Dual-Stack PPP session will be terminated.

3.1. Definition of the participating elements and their functionalities

This chapter identifies the network elements that are involved in the message flows to enable the on-demand IPv4 address provisioning functionality and describes their functionalities related to this mechanism.

- Customer Edge Router (CER a.k.a. CPE) / End System

Within the context of this document the CPE/End System is any device implementing a Dual-Stack PPP stack and acting as a PPP client with respect to the PPP server (e.g. BRAS/NAS) in the service provider network in order to achieve connectivity to the service provider network. In the case of a Customer Edge Router (CPE) this is a node (e.g. intended for home or small office usage) which forwards IPv4 and IPv6 packets that are not explicitly addressed to itself between the Local Area Network and WAN interface. The CPE itself can be abstracted into three functional blocks, one that carries the PPP session (e.g. a standalone DSL modem), one that is operating simply as a local router which includes the NAPT44 function and any IPV6 PD/ ND, DHCPv6 and DHCP for both stacks and one which includes the local CPE functionalities (e.g., DNS forwarder/cache, VoIP SIP agent). The PPP interface of this device is also called WAN (Wide Area Network) interface [RFC6204]. In the case of IPv4 an additional Network Address Translation (NAT) functionality is implemented on the router part. So within the Local Area Network private IPv4 addresses can be used as defined in [RFC1918]. Therefore the demand for global IPv4 connectivity of such a Customer Edge Router will be triggered either by local applications on the CPE or by receiving IPv4 packets on its customer network facing interfaces that are addressed to the public Internet.

In the case of an end system, this is a node that intends to communicate with other nodes by sending IPv4 and/or IPv6 packets. On an end system, the IPv4 connectivity demand can only be triggered by local protocols and own applications. However, in both cases (CPE or end system) an IPv4_idle_timer is implemented on the upstream (WAN) interface in order to detect IPv4 packets passing the WAN interface (incoming/ outgoing) and to measure the related IPv4 idle time when no IPv4 packet has been sent or received.

- Network Access Server (NAS a.k.a. BRAS)/Layer 2 Network Server (LNS)

The Network Access Server (NAS) (a.k.a. Broadband Remote Access Server BRAS) is a device providing local Dual- Stack PPP connectivity to the Service Provider access network and acting as a PPP server to the PPP client on the Customer Edge Router or customer end system. Within a RFC 2661 architecture the PPP server within the service provider network is the L2TP Network Server (LNS). The IPv4 address pool management can be provided locally on the BRAS/NAS/LNS or remotely. In the case of a local address pool management no additional information exchange to an external address pool management system is needed in order to assign or release IPv4 addresses. In the case of an external address pool management an information exchange between the BRAS/NAS/LNS and the address pool management system is required.

- External Address Pool Management

External Address Pool Management is used in the case when no local Address Pool Management system is implemented in the BRAS/NAS/LNS. In this case it is necessary that the BRAS/NAS/LNS communicates with an External Address Pool Management System for signaling assignment or release of IPv4 addresses. RADIUS as specified in [RFC2865] or DIAMETER as specified in [RFC3588] can be used as protocol between BRAS/NAS/LNS and the External Address Pool Management System.

3.2. Assigning IPv4 address parameter on-demand after establishing PPP session with IPv6 connectivity

A PPP client implementation wishing to establish a PPP connection MUST transmit a NCP Configure-Request to the PPP server. If every Configuration Option received in a NCP Configure-Request is recognizable and all values are acceptable, then the PPP server implementation MUST transmit a NCP Configure-Ack to the initiator of the NCP Configure-Request.

Applied to the above sketched Dual-Stack PPP session use case the configuration and enabling of the IPv6 protocol module will be done immediately after a successful LCP data link configuration (and maybe successful authentication phase) of the PPP session. Assuming that this IPv6CP configuration exchange has been successfully completed, the PPP session is now established and operational containing an IPv6-only network layer connection.

Separately from that, the IPv4 protocol module can (later on and dynamically on-demand) be configured and enabled using IPCP. However this SHALL only be done in the case that an IPv4 connectivity demand

has been detected on the PPP customer end system or CPE (PPP client). Therefore the BRAS/NAS MUST not initiate the negotiation of IPCP.

The following diagram illustrates the corresponding IPCP (and accounting) message exchange that is needed to configure the IPv4 protocol modules of an existing (Dual-Stack) PPP session on-demand.

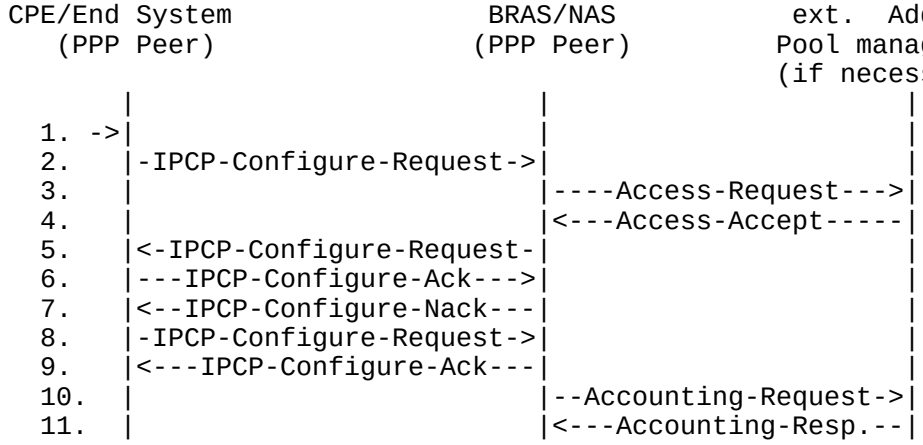


Figure 2: Message flow for assigning IPv4 address parameter

In the above diagram, the CPE/End System is triggered (1) to set up IPv4 connectivity via an already existing PPP session. The CPE/End System detects that there is no context (incl. a public IPv4 address) for its WAN interface available and starts the negotiation of the required IPv4 address and protocol parameters by sending an IPCP Configure-Request to the BRAS/NAS (2). The BRAS/NAS will request the corresponding IPv4 connectivity parameters (e.g. IPv4 address, DNS resolver address) from a local (e.g. within the BRAS/NAS) or remote database representing the Address Pool Management System(e.g. via RADIUS/DIAMETER) (3, 4). After this the PPP peers use the standard IPCP procedures to finalize the IPv4 address parameter negotiation (5, 6, 7, 8, 9). After a successful provisioning of the IPv4 address parameter the CPE/End system has full global IPv4 connectivity and can proceed with the IPv4 communication (in parallel to IPv6). In case of an external Address Pool Management, the BRAS/NAS will send an Accounting-Request message (10) to the external Address Pool Management System in order to signal the successful negotiation of the IPv4 address parameter. The external Address Pool Management System will answer with an Accounting-Response (11) message.

3.3. Releasing unused IPv4 address parameters

A PPP client implementation according to this draft wishing to close a dedicated NCP connection (e.g., IPCP or IPv6CP) SHOULD transmit a Terminate-Request to the peer. Upon reception of a NCP Terminate-Request, a Terminate-Ack MUST be transmitted to the sender of the Terminate-Request.

In the PPP Dual-Stack session scenario discussed here, the generation of the Terminate-Request message for the IPCP part of the PPP Dual-Stack session MUST be triggered by an IPv4 traffic idle timer within the PPP client when no IPv4 traffic has been detected on the upstream interface for a time interval longer than Initial_IPv4_Idle_Time. As long as there is still an ongoing IPv6 connection within the PPP session, the PPP session MUST be kept open. Equivalently, when no IPv6 connectivity is detected the IPv6CP session can be terminated again by sending an IPv6CP Terminate-Request and accepting this by a Terminate-Ack. Afterwards the link layer connectivity and hence the whole PPP connection can be terminated by exchanging the LCP Terminate-Request and Terminate-Ack messages.

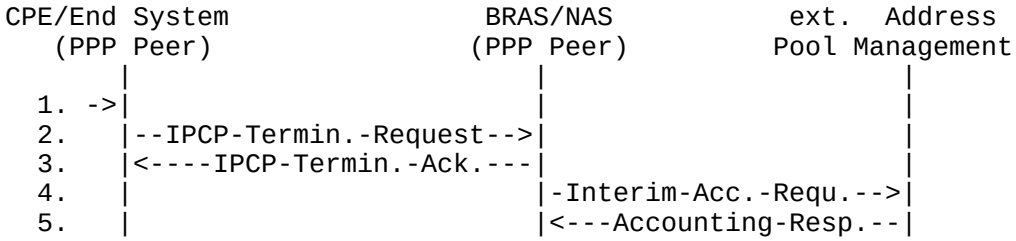


Figure 3: Message flow for releasing IPv4 address parameter

The termination of an IPCP connection within a Dual-Stack PPP session is illustrated in figure 3 above.

For this sample message flow it is assumed that there is still an IPv6CP connection active inside the Dual-Stack PPP session. After the expiration of the IPv4 traffic idle timer (1) the CPE/End system sends an IPCP terminate request to the peer (2). The request will be answered with an Terminate-Ack message (3). The IPv4 address can be returned to the local address pool (e.g. within the BRAS/NAS) or to the remote IPv4 address pool by sending Interim-Accounting messages (4, 5) (e.g. via RADIUS/DIAMETER).

3.4. Timer Considerations

IPv4_Idle_Timer

The IPv4_Idle_Timer on the upstream interface of the PPP client has to be started immediately after a successful establishment of the

IPCP session within the PPP connection and MUST count down starting from the Initial_IPv4_Idle_Time value to 0. When the upstream interface of the PPP client discovers incoming / outgoing IPv4 traffic then the IPv4_Idle_Time MUST be reset to the Initial_IPv4_Idle_Timer value. When the IPv4_Idle_Timer reaches the value 0 sending a Terminate-Request message MUST be triggered by a the PPP client (e.g., end system, CPE). The Initial_IPv4_Idle_Time value MUST be configurable to adopt the mechanism due to the needs of the applications which are using IPv4 and with respect to an optimization of the IPv4 address saving potential.

4. Potential for optimization

The efficiency of the "On demand IPv4 address provisioning" mechanism can be measured in the ratio of IPCP/RADIUS/DIAMETER signalling traffic to the amount of the saved global IPv4 addresses. Hence different options to optimize the efficiency of the proposed solution are possible, by suppressing unnecessary signalling load and blocking forbidden IPv4 connectivity requests.

4.1. Avoiding unnecessary load on BRAS/NAS and AAA

Unnecessary signaling load between PPP peers as well as between BRAS/NAS and external Address Pool Management can for instance occur when a IPv6-only customer requests IPv4 address parameters. This can be prevented by restricting the usage of a Dual-Stack CPE for IPv6-only customers to IPv6 only and/or by administratively refusing the IPCP configure requests of such an IPv6-only customer inside the BRAS/NAS.

The former case is more or less a business and customer relationship related issue which needs no engineering concepts.

This case can be solved by answering an IPCP Configure Request message from a IPv6-only customer with a LCP reject message as described in chapter 5.7 of [RFC1661]. The field Rejected-Protocol of the LCP reject message contains the value 0x8021 for IPCP and the Rejected-Information field contains a copy of the IPCP packet which is being rejected. Due to [RFC1661] upon reception of a Protocol-Reject, the implementation of the IPv4 capable CPE of the IPv6-only customer MUST immediately stop sending packets of the indicated protocol at the earliest opportunity. So the transmission of unnecessary IPCP and RADIUS messages during the running PPP session can be prevented.

Another opportunity to reduce IPCP signaling load and the corresponding signalling overhead between BRAS/NAS and external Address Pool Management is the definition of default IPv4 traffic idle timer values for always-on applications that are sending

periodic messages (see chapter 3.3). The value of this IPv4 traffic idle timer should be chosen a few seconds larger than the interval between periodic messages of always-on applications. Such an approach avoids problems for these applications when IPv4 is used and optimizes IPv4 address release and address assign message exchange. Very short and periodic IPv4 address renewal cycles can be avoided by such an approach.

4.2. Reducing IPv4 traffic on external interfaces

The easiest way to reduce IPv4 traffic demand (and hence the need for public IPv4 addresses) is to shift applications from usage of IPv4 to IPv6. In using the Dual-Stack approach which is a prerequisite of the mechanism described in this draft, no differences regarding the service level of both protocols are expected. Each service can be provided with the same quality level independently of the chosen version of the Internet Protocol.

But regarding applications on end systems the Internet access provider has only very limited influence. However for applications and services running on the CPE itself (e.g. VoIP User Agent) the internet access provider should be able to define and require their IPv6 readiness.

An additional point is the preferred usage of IPv6 on all external (WAN) interfaces in the case when the CPE acts as a relay and caches on behalf of certain protocols (e.g. DNS). When on a LAN interface a request message for such a protocol is received via IPv4 and a relaying to the external WAN interface is needed IPv6 should be the preferred network protocol. Such a requirement has already been defined for relaying/caching devices in [BBF-TR-124-i2] (section LAN.DNSv6, item 6).

5. Impacts on user experience and operation

5.1. Impacts on user experience and Happy Eyeballs implementations

In order to mitigate delays in end-to-end establishment in unstable Dual-Stack environments I [RFC6555] describes a mechanism to optimize the communication establishment for connection-oriented transports (e.g., TCP, SCTP). The IPv6 connectivity can be impaired for instance due connection failure to the IPv6 Internet, broken 6to4 or Teredo tunnels, or broken IPv6 peering. After making a connection attempt on the preferred address family (e.g. IPv6) and failing to establish a connection within a certain time period, a "Happy Eyeballs" implementation will decide to initiate a second connection attempt in parallel using the same or the other address family. It is recommended that the non-winning connections be abandoned, even

though they could -- in some cases -- be put to reasonable use. In the case of IPv6 connectivity problems a Dual-Stack host will hence use IPv4; in the case of IPv4 connectivity problems a Dual-Stack host will use IPv6 for reaching a certain destination.

In a Dual-Stack environment according to this document it is assumed that the IPv6 connectivity (at least in the access network) is not impaired. Nevertheless it is possible that the network path between access area and IPv6 destination is broken. In this case a fast fall-back to IPv4 is needed. In a Dual-Stack environment are, according to this draft, in general 3 states regarding IPv4 and IPv6 connectivity of interest:

1. Neither IPv4 nor IPv6 connectivity is given (PPP link is dead),

2. Only IPv6 connectivity is established and

3. IPv4 and IPv6 connectivity is established.

In the first case the "Happy Eyeball" scenario is not relevant.

In the second case a fast IPv4 fall-back has to be realized by triggering and using the mechanism described in chapter 3.2. Depending on the architecture scenario (IP address pool management inside or outside the BRAS/NAS) and the CPE and BRAS/NAS performance capabilities a delay of about hundred milliseconds for establishing the IPCP session has to be considered. In the case that meanwhile the communication is not established via IPv6 this will be done via IPv4. If the "Happy Eyeball" algorithm caches connection establishment successes/failures, this additional IPCP establishment delay could lead to wrong assumptions regarding the quality of the IPv6 and IPv4 connectivity. However, in following connection attempts using "Happy Eyeball" this can be corrected, because IPv4 connectivity is already established and no additional delay will be added.

The third case corresponds to a native Dual-Stack architecture, so no additional considerations are needed.

5.2. Operational impacts

As described above the used mechanisms for dynamically assigning / releasing IPv4 addresses do not need new PPP, IPCP, IPv6CP or RADIUS protocol elements. Therefore it can be assumed that an implementation of the proposed mechanisms on the distinct network elements can be realized easily. Nevertheless depending on the service provider IPv6 migration strategy and schedule it is possible that this mechanism is not everywhere in a PPP service provider

deployment active or passive supported. When a service provider allows the customer the usage of CPEs of their own choice it is possible that an IPv4 address releasing CPE will be connected to a non compatible BRAS/NAS in the service provider network. In this case the message flow initiated from the CPE could lead to IPv4 connectivity problems. In order to avoiding this, a CPE implementation according to this draft MAY provide capabilities to switch on/off the above described functionality in order to fall back to a support of an IPv6-only or a "standard" Dual-Stack service.

6. Acknowledgements

The author and contributors also wish to acknowledge the assistance and feedback of the following individuals or groups.

Tina Tsou

Alain Durand

Sven Schmidtke

Dan Wing

Vernon Schryer

Mark Townsley

Wesley George

Joel M. Halpern

Christian Jaquenet

7. IANA Considerations

This memo includes no request to IANA.

TBD.

All drafts are required to have an IANA considerations section (see Guidelines for Writing an IANA Considerations Section in RFCs [RFC5226] for a guide). If the draft does not require IANA to do anything, the section contains an explicit statement that this is the case (as above). If there are no requirements for IANA, the section will be removed during conversion into an RFC by the RFC Editor.

8. Security Considerations

TBD.

All drafts are required to have a security considerations section. See RFC 3552 [RFC3552] for a guide.

9. References

9.1. Normative Reference

[BBF-TR-124-i2] Broadbandforum, "Functional Requirements for Broadband Residential Gateway Devices (Issue 2)", May 2010.

[BBF-TR-187] Broadbandforum, "Technical Report TR187 IPv6 over PPP Broadband Access (Issue 1)", May 2010.

[BBF-TR-242] Broadbandforum, "Draft TR242 IPv6 Transition Mechanisms for Broadband Networks", .

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC1661] Simpson, W., "The Point-to-Point Protocol (PPP)", STD 51, RFC 1661, July 1994.

[RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC2472] Haskin, D. and E. Allen, "IP Version 6 over PPP", RFC 2472, December 1998.

[RFC2661] Townsley, W., Valencia, A., Rubens, A., Pall, G., Zorn, G., and B. Palter, "Layer Two Tunneling Protocol "L2TP"", RFC 2661, August 1999.

[RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.

[RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G., and J. Arkko, "Diameter Base Protocol", RFC 3588, September 2003.

[RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, October 2005.

[RFC6204] Singh, H., Beebee, W., Donley, C., Stark, B., and O. Troan, "Basic Requirements for IPv6 Customer Edge Routers", RFC 6204, April 2011.

[RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, August 2011.

[RFC6555] Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", RFC 6555, April 2012.

9.2. Informative References

[RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, July 2003.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

Appendix A. Workplan

Authors' Addresses

Karsten Fleischhauer (editor)
Deutsche Telekom AG
Heinrich-Hertz-Strasse 3-7
64295 Darmstadt
DE

Phone: +49 6151 58 12831
Email: k.fleischhauer@telekom.de

Olaf Bonness
Deutsche Telekom AG
Winterfeldtstr. 21-27
10781 Berlin
DE

Phone: +49 30 835358826
Email: olaf.bonness@telekom.de

Internet Engineering Task Force
Internet-Draft
Updates: 1812, 1122, 4084
(if approved)
Intended status: Standards Track
Expires: December 3, 2011

W. George
Time Warner Cable
C. Donley
Cablelabs
C. Liljenstolpe
Telstra
L. Howard
Time Warner Cable

June 1, 2011

IPv6 Support Required for all IP-capable nodes
draft-george-ipv6-required-02

Abstract

Given the global lack of available IPv4 space, and limitations in IPv4 extension and transition technologies, this document deprecates the concept that an IP-capable node MAY support IPv4 _only_, and redefines an IP-capable node as one which supports either IPv6 _only_ or IPv4/IPv6 dual-stack. This document updates RFC1812, 1122 and 4084 to reflect the change in requirements.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on December 3, 2011.

Copyright Notice

Table of Contents

1. Introduction

IP version 4 (IPv4) has served to connect public and private hosts all over the world for over 30 years. However, due to the success of the Internet in finding new and innovative uses for IP networking, billions of hosts are now connected via the Internet and requiring unique addressing. This demand has led to the exhaustion of the IANA global pool of unique IPv4 addresses [IANA-exhaust], and will be followed by the exhaustion of the free pools for each Regional Internet Registry (RIR), the first of which is APNIC [APNIC-exhaust]. While transition technologies and other means to extend the lifespan of IPv4 do exist, nearly all of them come with tradeoffs that prevent them from being optimal long-term solutions when compared with deployment of IP version 6 (IPv6) as a means to allow continued growth on the Internet. See [I-D.ietf-intarea-shared-addressing-issues] and [I-D.donley-nat444-impacts] for some discussion on this topic.

IPv6 [RFC1883] was proposed in 1995 as, among other things, a solution to the limitations on globally unique addressing that IPv4's 32-bit addressing space represented, and has been under continuous refinement and deployment ever since. [RFC2460]. The exhaustion of IPv4 and the continued growth of the internet worldwide has created the driver for widespread IPv6 deployment.

However, the IPv6 deployment necessary to reduce reliance on IPv4 has been hampered by a lack of ubiquitous hardware and software support throughout the industry. Many vendors, especially in the consumer space have continued to view IPv6 support as optional. Even today they are still selling "IP capable" or "Internet Capable" devices which are not IPv6-capable, which has continued to push out the point at which the natural hardware refresh cycle will significantly increase IPv6 support in the average home or enterprise network. They are also choosing not to update existing software to enable IPv6 support on software-updatable devices, which is a problem because it is not realistic to expect that the hardware refresh cycle will single-handedly purge IPv4-only devices from the active network in a reasonable amount of time. This is a significant problem, especially in the consumer space, where the network operator often has no control over the hardware the consumer chooses to use. For the same reason that the average consumer is not making a purchasing decision based on the presence of IPv6 support in their Internet-capable devices and services, consumers are unlikely to replace their still-functional Internet-capable devices simply to add IPv6 support - they don't know or don't care about IPv6, they simply want their devices to work as advertised.

This lack of support is making the eventual IPv6 transition

considerably more difficult, and drives the need for expensive and complicated transition technologies to extend the life of IPv4-only devices as well as eventually to interwork IPv4-only and IPv6-only hosts. While IPv4 is expected to coexist on the Internet with IPv6 for many years, a transition from IPv4 as the dominant Internet Protocol towards IPv6 as the dominant Internet Protocol will need to occur. The sooner the majority of devices support IPv6, the less protracted this transition period will be.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Requirements and Recommendation

This draft updates the following documents:

Updates [RFC1812] to note that IP nodes SHOULD no longer support IPv4 only. This is to ensure that those using it as a guideline for IP implementations use the other informative references in this document as a guideline for proper IPv6 implementations.

Updates [RFC1122] to redefine generic "IP" support to include and require IPv6 for IP-capable nodes and routers.

Updates [RFC4084] to move "Version Support" from Section 4, "Additional Terminology" to Section 2, "General Terminology." This is to reflect the idea that version support is now critical to defining the types of IP service, especially with respect to Full Internet Connectivity.

From a practical perspective, the requirements proposed by this draft mean that:

New IP implementations MUST support IPv6.

Current IP implementations SHOULD support IPv6.

IPv6 support MUST be equivalent or better in quality and functionality when compared to IPv4 support in an IP implementation.

Helpful informative references can be found in [RFC4294], soon to be updated by [I-D.ietf-6man-node-req-bis] and in [RFC6204]

Current and new IP Networking implementations SHOULD support IPv4 and IPv6 coexistence (dual-stack), but MUST NOT require IPv4 for proper and complete function.

It is expected that many existing devices and implementations will not be able to support IPv6 for one or more valid technical reasons, but for maximum flexibility and compatibility, a best effort SHOULD be made to update existing hardware and software to enable IPv6 support.

3. Acknowledgements

Thanks to the following people for their reviews and comments: Marla Azinger, Brian Carpenter, Victor Kuarsingh, Jari Arkko, Scott Brim, Margaret Wasserman, Joe Touch

4. IANA Considerations

This memo includes no request to IANA.

5. Security Considerations

There are no direct security considerations generated by this document, but existing documented security considerations for implementing IPv6 will apply.

6. References

6.1. Normative References

[RFC1122] Braden, R., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, October 1989.

[RFC1812] Baker, F., "Requirements for IP Version 4 Routers", RFC 1812, June 1995.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4084] Klensin, J., "Terminology for Describing Internet Connectivity", BCP 104, RFC 4084, May 2005.

6.2. Informative References

[APNIC-exhaust]
APNIC, "APNIC Press Release", 2011, <http://www.apnic.net/__data/assets/pdf_file/0018/33246/Key-Turning-Point-in-Asia-Pacific-IPv4-Exhaustion_English.pdf >.

[I-D.donley-nat444-impacts]
Donley, C., Howard, L., Kuarsingh, V., Chandrasekaran, A., and V. Ganti, "Assessing the Impact of NAT444 on Network Applications", draft-donley-nat444-impacts-01 (work in progress), October 2010.

[I-D.ietf-6man-node-req-bis]
Jankiewicz, E., Loughney, J., and T. Narten, "IPv6 Node Requirements", draft-ietf-6man-node-req-bis-11 (work in progress), May 2011.

[I-D.ietf-intarea-shared-addressing-issues]
Ford, M., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Issues with IP Address Sharing", draft-ietf-intarea-shared-addressing-issues-05 (work in progress), March 2011.

[IANA-exhaust]
IANA, "IANA address allocation", 2011, <http://www.iana.org/assignments/ipv4-address-space/ipv4-address-space.xml>.

[RFC1883] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 1883, December 1995.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC4294] Loughney, J., "IPv6 Node Requirements", RFC 4294, April 2006.

[RFC6204] Singh, H., Beebee, W., Donley, C., Stark, B., and O. Troan, "Basic Requirements for IPv6 Customer Edge Routers", RFC 6204, April 2011.

Authors' Addresses

Wesley George
Time Warner Cable
13820 Sunrise Valley Drive
Herndon, VA 20171
US

Phone: +1 703-561-2540
Email: wesley.george@twcable.com

Chris Donley
Cablelabs
858 Coal Creek Circle
Louisville, CO 80027
US

Phone: +1-303-661-9100
Email: C.Donley@cablelabs.com

Christopher Liljenstolpe
Telstra
Level 32/242 Exhibition Street
Melbourne, VIC 3000
AU

Phone: +61-3-8647-6389
Email: cdl@asgaard.org

Lee Howard
Time Warner Cable
13820 Sunrise Valley Drive
Herndon, VA 20171
US

Phone: +1-703-345-3513
Email: lee.howard@twcable.com

Internet Area WG
Internet Draft
Updates: 791,1122,2003
Intended status: Proposed Standard
Expires: May 2013

J. Touch
USC/ISI

November 27, 2012

Updated Specification of the IPv4 ID Field
draft-ietf-intarea-ipv4-id-update-07.txt

Status of this Memo

Abstract

The IPv4 Identification (ID) field enables fragmentation and reassembly, and as currently specified is required to be unique within the maximum lifetime for all datagrams with a given source/destination/protocol tuple. If enforced, this uniqueness requirement would limit all connections to 6.4 Mbps. Because individual connections commonly exceed this speed, it is clear that existing systems violate the current specification. This document updates the specification of the IPv4 ID field in RFC791, RFC1122, and RFC2003 to more closely reflect current practice and to more closely match IPv6 so that the field's value is defined only when a datagram is actually fragmented. It also discusses the impact of these changes on how datagrams are used.

Table of Contents

1. Introduction

In IPv4, the Identification (ID) field is a 16-bit value that is unique for every datagram for a given source address, destination address, and protocol, such that it does not repeat within the maximum datagram lifetime (MDL) [RFC791][RFC1122]. As currently specified, all datagrams between a source and destination of a given protocol must have unique IPv4 ID values over a period of this MDL, which is typically interpreted as two minutes, and is related to the recommended reassembly timeout [RFC1122]. This uniqueness is currently specified as for all datagrams, regardless of fragmentation settings.

Uniqueness of the IPv4 ID is commonly violated by high speed devices; if strictly enforced, it would limit the speed of a single protocol between two IP endpoints to 6.4 Mbps for typical MTUs of 1500 bytes [RFC4963]. It is common for a single connection to operate far in excess of these rates, which strongly indicates that the uniqueness of the IPv4 ID as specified is already moot. Further, some sources have been generating non-varying IPv4 IDs for many years (e.g., cellphones), which resulted in support for such in ROHC [RFC5225].

This document updates the specification of the IPv4 ID field to more closely reflect current practice, and to include considerations taken into account during the specification of the similar field in IPv6.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [RFC2119].

In this document, the characters ">>" proceeding an indented line(s) indicates a requirement using the key words listed above. This convention aids reviewers in quickly identifying or finding this document's explicit requirements.

3. The IPv4 ID Field

IP supports datagram fragmentation, where large datagrams are split into smaller components to traverse links with limited maximum transmission units (MTUs). Fragments are indicated in different ways in IPv4 and IPv6:

- In IPv4, fragments are indicated using four fields of the basic header: Identification (ID), Fragment Offset, a "Don't Fragment" flag (DF), and a "More Fragments" flag (MF) [RFC791]

- In IPv6, fragments are indicated in an extension header that includes an ID, Fragment Offset, and M (more fragments) flag similar to their counterparts in IPv4 [RFC2460]

IPv4 and IPv6 fragmentation differs in a few important ways. IPv6 fragmentation occurs only at the source, so a DF bit is not needed to prevent downstream devices from initiating fragmentation (i.e., IPv6 always acts as if DF=1). The IPv6 fragment header is present only when a datagram has been fragmented, or when the source has received a "packet too big" ICMPv6 error message indicating that the path cannot support the required minimum 1280-byte IPv6 MTU and is thus subject to translation [RFC2460][RFC4443]. The latter case is relevant only for IPv6 datagrams sent to IPv4 destinations to support subsequent fragmentation after translation to IPv4.

With the exception of these two cases, the ID field is not present for non-fragmented datagrams, and thus is meaningful only for datagrams that are already fragmented or datagrams intended to be fragmented as part of IPv4 translation. Finally, the IPv6 ID field is 32 bits, and required unique per source/destination address pair for IPv6, whereas for IPv4 it is only 16 bits and required unique per source/destination/protocol triple.

This document focuses on the IPv4 ID field issues, because in IPv6 the field is larger and present only in fragments.

3.1. Uses of the IPv4 ID Field

The IPv4 ID field was originally intended for fragmentation and reassembly [RFC791]. Within a given source address, destination address, and protocol, fragments of an original datagram are matched

based on their IPv4 ID. This requires that IDs are unique within the address/protocol triple when fragmentation is possible (e.g., DF=0) or when it has already occurred (e.g., frag_offset>0 or MF=1).

Other uses have been envisioned for the IPv4 ID field. The field has been proposed as a way to detect and remove duplicate datagrams, e.g., at congested routers (noted in Sec. 3.2.1.5 of [RFC1122]) or in network accelerators. It has similarly been proposed for use at end hosts to reduce the impact of duplication on higher-layer protocols (e.g., additional processing in TCP, or the need for application-layer duplicate suppression in UDP). This is also discussed further in Section 5.1.

The IPv4 ID field is used in some diagnostic tools to correlate datagrams measured at various locations along a network path. This is already insufficient in IPv6 because unfragmented datagrams lack an ID, so these tools are already being updated to avoid such reliance on the ID field. This is also discussed further in Section 5.1.

The ID clearly needs to be unique (within MDL, within the src/dst/protocol tuple) to support fragmentation and reassembly, but not all datagrams are fragmented or allow fragmentation. This document deprecates non-fragmentation uses, allowing the ID to be repeated (within MDL, within the src/dst/protocol tuple) in those cases.

3.2. Background on IPv4 ID Reassembly Issues

The following is a summary of issues with IPv4 fragment reassembly in high speed environments raised previously [RFC4963]. Readers are encouraged to consult RFC 4963 for a more detailed discussion of these issues.

With the maximum IPv4 datagram size of 64KB, a 16-bit ID field that does not repeat within 120 seconds means that the aggregate of all TCP connections of a given protocol between two IP endpoints is limited to roughly 286 Mbps; at a more typical MTU of 1500 bytes, this speed drops to 6.4 Mbps [RFC791][RFC1122][RFC4963]. This limit currently applies for all IPv4 datagrams within a single protocol (i.e., the IPv4 protocol field) between two IP addresses, regardless of whether fragmentation is enabled or inhibited, and whether a datagram is fragmented or not.

IPv6, even at typical MTUs, is capable of 18.7 Tbps with fragmentation between two IP endpoints as an aggregate across all protocols, due to the larger 32-bit ID field (and the fact that the IPv6 next-header field, the equivalent of the IPv4 protocol field, is

not considered in differentiating fragments). When fragmentation is not used the field is absent, and in that case IPv6 speeds are not limited by the ID field uniqueness.

Note also that 120 seconds is only an estimate on the MDL. It is related to the reassembly timeout as a lower bound and the TCP Maximum Segment Lifetime as an upper bound (both as noted in [RFC1122]). Network delays are incurred in other ways, e.g., satellite links, which can add seconds of delay even though the TTL is not decremented by a corresponding amount. There is thus no enforcement mechanism to ensure that datagrams older than 120 seconds are discarded.

Wireless Internet devices are frequently connected at speeds over 54 Mbps, and wired links of 1 Gbps have been the default for several years. Although many end-to-end transport paths are congestion limited, these devices easily achieve 100+ Mbps application-layer throughput over LANs (e.g., disk-to-disk file transfer rates), and numerous throughput demonstrations with COTS systems over wide-area paths exhibit these speeds for over a decade. This strongly suggests that IPv4 ID uniqueness has been moot for a long time.

4. Updates to the IPv4 ID Specification

This document updates the specification of the IPv4 ID field in three distinct ways, as discussed in subsequent subsections:

- Use the IPv4 ID field only for fragmentation
- Avoiding a performance impact when the IPv4 ID field is used
- Encourage safe operation when the IPv4 ID field is used

There are two kinds of datagrams used in the following discussion, named as follows:

- Atomic datagrams are datagrams not yet fragmented and for which further fragmentation has been inhibited.
- Non-atomic datagrams are datagrams that either already have been fragmented or for which fragmentation remains possible.

This same definition can be expressed in pseudo code as using common logical operators (equals is ==, logical ’and’ is &&, logical ’or’ is ||, greater than is >, and parenthesis function typically) as:

- Atomic datagrams: (DF==1)&&(MF==0)&&(frag_offset==0)

o Non-atomic datagrams: (DF==0)||(MF==1)||(frag_offset>0)

The test for non-atomic datagrams is the logical negative of the test for atomic datagrams, thus all possibilities are considered.

4.1. IPv4 ID Used Only for Fragmentation

Although RFC1122 suggests the IPv4 ID field has other uses, including datagram de-duplication, such uses are already not interoperable with known implementations of sources that do not vary their ID. This document thus defines this field’s value only for fragmentation and reassembly:

>> IPv4 ID field MUST NOT be used for purposes other than fragmentation and reassembly.

Datagram de-duplication is accomplished using hash-based duplicate detection for cases where the ID field is absent (IPv6 unfragmented datagrams), which can also be applied to IPv4 atomic datagrams without utilizing the ID field [RFC6621].

In atomic datagrams, the IPv4 ID field has no meaning, and thus can be set to an arbitrary value, i.e., the requirement for non-repeating IDs within the address/protocol triple is no longer required for atomic datagrams:

>> Originating sources MAY set the IPv4 ID field of atomic datagrams to any value.

Second, all network nodes, whether at intermediate routers, destination hosts, or other devices (e.g., NATs and other address sharing mechanisms, firewalls, tunnel egresses), cannot rely on the field:

>> All devices that examine IPv4 headers MUST ignore the IPv4 ID field of atomic datagrams.

The IPv4 ID field is thus meaningful only for non-atomic datagrams - datagrams that have either already been fragmented, or those for which fragmentation remains permitted. Atomic datagrams are detected by their DF, MF, and fragmentation offset fields as explained in Section 4, because such a test is completely backward compatible; this document thus does not reserve any IPv4 ID values, including 0, as distinguished.

Deprecating the use of the IPv4 ID field for non-reassembly uses should have little - if any - impact. IPv4 IDs are already frequently

repeated, e.g., over even moderately fast connections and from some sources that do not vary the ID at all, and no adverse impact has been observed. Duplicate suppression was suggested [RFC1122] and has been implemented in some protocol accelerators, but no impacts of IPv4 ID reuse have been noted to date. Routers are not required to issue ICMPs on any particular timescale, and so IPv4 ID repetition should not have been used for validation and has not been observed, and again repetition already occurs and would have been noticed [RFC1812]. ICMP relaying at tunnel ingresses is specified to use soft state rather than a datagram cache, and should have been noted if the latter for similar reasons [RFC2003]. These and other legacy issues are discussed further in Section 5.1.

4.2. Encourage Safe IPv4 ID Use

This document makes further changes to the specification of the IPv4 ID field and its use to encourage its safe use as corollary requirements changes as follows.

RFC 1122 discusses that if TCP retransmits a segment it may be possible to reuse the IPv4 ID (see Section 6.2). This can make it difficult for a source to avoid IPv4 ID repetition for received fragments. RFC 1122 concludes that this behavior "is not useful"; this document formalizes that conclusion as follows:

>> The IPv4 ID of non-atomic datagrams MUST NOT be reused when sending a copy of an earlier non-atomic datagram.

RFC 1122 also suggests that fragments can overlap [RFC1122]. Such overlap can occur if successive retransmissions are fragmented in different ways but with the same reassembly IPv4 ID. This overlap is noted as the result of reusing IPv4 IDs when retransmitting datagrams, which this document deprecates. However, it is also the result of in-network datagram duplication, which can still occur. As a result this document does not change the need to support overlapping fragments.

4.3. IPv4 ID Requirements That Persist

This document does not relax the IPv4 ID field uniqueness requirements of [RFC791] for non-atomic datagrams, i.e.:

>> Sources emitting non-atomic datagrams MUST NOT repeat IPv4 ID values within one MDL for a given source address/destination address/protocol triple.

Such sources include originating hosts, tunnel ingresses, and NATs (including other address sharing mechanisms) (see Section 5.3).

This document does not relax the requirement that all network devices honor the DF bit, i.e.:

>> IPv4 datagrams whose DF=1 MUST NOT be fragmented.

>> IPv4 datagram transit devices MUST NOT clear the DF bit.

In specific, DF=1 prevents fragmenting atomic datagrams. DF=1 also prevents further fragmenting received fragments. In-network fragmentation is permitted only when DF=0; this document does not change that requirement.

5. Impact of Proposed Changes

This section discusses the impact of the proposed changes on legacy devices, datagram generation in updated devices, middleboxes, and header compression.

5.1. Impact on Legacy Internet Devices

Legacy uses of the IPv4 ID field consist of fragment generation, fragment reassembly, duplicate datagram detection, and "other" uses.

Current devices already generate ID values that are reused within the source address, destination address, protocol, and ID tuple in less than the current estimated Internet MDL of two minutes. They assume that the MDL over their end-to-end path is much lower.

Existing devices have been known to generate non-varying IDs for atomic datagrams for nearly a decade, notably some cell phones. Such constant ID values are the reason for their support as an optimization of ROHC [RFC5225]. This is discussed further in Section 5.4. Generation of IPv4 datagrams with constant (zero) IDs is also described as part of the IP/ICMP translation standard [RFC6145].

Many current devices support fragmentation that ignores the IPv4 Don’t Fragment (DF) bit. Such devices already transit traffic from sources that reuse the ID. If fragments of different datagrams reusing the same ID (within the source/destination/protocol tuple) arrive at the destination interleaved, fragmentation would fail and traffic would be dropped. Either such interleaving is uncommon, or traffic from such devices is not widely traversing these DF-ignoring devices, because significant occurrence of reassembly errors has not been reported. DF-ignoring devices do not comply with existing

standards, and it is not feasible to update the standards to allow them as compliant.

The ID field has been envisioned for use in duplicate detection, as discussed in Section 4.1 [RFC1122]. Although this document now allows IPv4 ID reuse for atomic datagrams, such reuse is already common (as noted above). Protocol accelerators are known to implement IPv4 duplicate detection, but such devices are also known to violate other Internet standards to achieve higher end-to-end performance. These devices would already exhibit erroneous drops for this current traffic, and this has not been reported.

There are other potential uses of the ID field, such as for diagnostic purposes. Such uses already need to accommodate atomic datagrams with reused ID fields. There are no reports of such uses having problems with current datagrams that reuse IDs. These and any other uses of the ID field are encouraged to apply IPv6-compatible methods for IPv4 as well.

Thus, as a result of previous requirements, this document recommends that IPv4 duplicate detection and diagnostic mechanisms apply IPv6-compatible methods, i.e., that do not rely on the ID field (e.g., as suggested in [RFC6621]). This is a consequence of using the ID field only for reassembly, as well as the known hazard of existing devices already reusing the ID field.

5.2. Impact on Datagram Generation

The following is a summary of the recommendations that are the result of the previous changes to the IPv4 ID field specification.

Because atomic datagrams can use arbitrary IPv4 ID values, the ID field no longer imposes a performance impact in those cases. However, the performance impact remains for non-atomic datagrams. As a result:

>> Sources of non-atomic IPv4 datagrams MUST rate-limit their output to comply with the ID uniqueness requirements. Such sources include, in particular, DNS over UDP [RFC2671].

Because there is no strict definition of the MDL, reassembly hazards exist regardless of the IPv4 ID reuse interval or the reassembly timeout. As a result:

>> Higher layer protocols SHOULD verify the integrity of IPv4 datagrams, e.g., using a checksum or hash that can detect reassembly errors (the UDP checksum is weak in this regard, but better than nothing).

Additional integrity checks can be employed using tunnels, as supported by SEAL, IPsec, or SCTP [RFC4301][RFC4960][RFC5320]. Such checks can avoid the reassembly hazards that can occur when using UDP and TCP checksums [RFC4963], or when using partial checksums as in UDP-Lite [RFC3828]. Because such integrity checks can avoid the impact of reassembly errors:

>> Sources of non-atomic IPv4 datagrams using strong integrity checks MAY reuse the ID within MDL values smaller than is typical.

Note, however, that such frequent reuse can still result in corrupted reassembly and poor throughput, although it would not propagate reassembly errors to higher layer protocols.

5.3. Impact on Middleboxes

Middleboxes include rewriting devices that include network address translators (NATs), address/port translators (NAPTs), and other address sharing mechanisms (ASMs). They also include devices that inspect and filter datagrams that are not routers, such as accelerators and firewalls.

The changes proposed in this document may not be implemented by middleboxes, however these changes are more likely to make current middlebox behavior compliant than to affect the service provided by those devices.

5.3.1. Rewriting Middleboxes

NATs and NAPTs rewrite IP fields, and tunnel ingresses (using IPv4 encapsulation) copy and modify some IPv4 fields, so all are considered sources, as do any devices that rewrite any portion of the source address, destination address, protocol, and ID tuple for any datagrams [RFC3022]. This is also true for other ASMs, including 4rd, IVI, and others in the "A+P" (address plus port) family [Bo11] [De11] [RFC6219]. It is equally true for any other datagram rewriting mechanism. As a result, they are subject to all the requirements of any source, as has been noted.

NATs/ASMs/rewriters present a particularly challenging situation for fragmentation. Because they overwrite portions of the reassembly tuple in both directions, they can destroy tuple uniqueness and result in a reassembly hazard. Whenever IPv4 source address, destination address, or protocol fields are modified, a NAT/ASM/rewriter needs to ensure that the ID field is generated appropriately, rather than simply copied from the incoming datagram. In specific:

>> Address sharing or rewriting devices MUST ensure that the IPv4 ID field of datagrams whose address or protocol are translated comply with these requirements as if the datagram were sourced by that device.

This compliance means that the IPv4 ID field of non-atomic datagrams translated at a NAT/ASM/rewriter needs to obey the uniqueness requirements of any IPv4 datagram source. Unfortunately, fragments already violate that requirement, as they repeat an IPv4 ID within the MDL for a given source address, destination address, and protocol triple.

Such problems with transmitting fragments through NATs/ASMs/rewriters are already known; translation is based on the transport port number, which is present in only the first fragment anyway [RFC3022]. This document underscores the point that not only is reassembly (and possibly subsequent fragmentation) required for translation, it can be used to avoid issues with IPv4 ID uniqueness.

Note that NATs/ASMs already need to exercise special care when emitting datagrams on their public side, because merging datagrams from many sources onto a single outgoing source address can result in IPv4 ID collisions. This situation precedes this document, and is not affected by it. It is exacerbated in large-scale, so-called "carrier grade" NATs [Pe11].

Tunnel ingresses act as sources for the outermost header, but tunnels act as routers for the inner headers (i.e., the datagram as arriving at the tunnel ingress). Ingresses can always fragment as originating sources of the outer header, because they control the uniqueness of that IPv4 ID field and the value of DF on the outer header independent of those values on the inner (arriving datagram) header.

5.3.2. Filtering Middleboxes

Middleboxes also include devices that filter datagrams, including network accelerators and firewalls. Some such devices reportedly feature datagram de-duplication that relies on IP ID uniqueness to identify duplicates, which has been discussed in Section 5.1.

5.4. Impact on Header Compression

Header compression algorithms already accommodate various ways in which the IPv4 ID changes between sequential datagrams [RFC1144] [RFC2508] [RFC3545] [RFC5225]. Such algorithms currently assume that the IPv4 ID is preserved end-to-end. Some algorithms already allow

assuming the ID does not change (e.g., ROHC [RFC5225]), where others include non-changing IDs via zero deltas (e.g., ECRTP [RFC3545]).

When compression assumes a changing ID as a default, having a non-changing ID can make compression less efficient. Such non-changing IDs have been described in various RFCs (e.g., footnote 21 of [RFC1144] and cRTP [RFC2508]). When compression can assume a non-changing IPv4 ID - as with ROHC and ECRTP - efficiency can be increased.

5.5. Impact of Network Reordering and Loss

Tolerance to network reordering and loss is a key feature of the Internet architecture. Although most current IP networks avoid gratuitous such events, both reordering and loss can and do occur. Datagrams are already intended to be reordered or lost, and recovery from those errors (where supported) already occurs at the transport or higher protocol layers.

Reordering is typically associated with routing transients or where multiple alternate paths exist. Loss is typically associated with path congestion or link failure (partial or complete). The impact of such events is different for atomic and non-atomic datagrams, and is discussed below. In summary, the recommendations of this document make the Internet more robust to reordering and loss by emphasizing the requirements of ID uniqueness for non-atomic datagrams and by more clearly indicating the impact of these requirements on both endpoints and datagram transit devices.

5.5.1. Atomic Datagrams Experiencing Reordering or Loss

Reusing ID values does not affect atomic datagrams when the DF bit is correctly respected, because order restoration does not depend on the datagram header. TCP uses a transport header sequence number; in some other protocols, sequence is indicated and restored at the application layer.

When DF=1 is ignored, reordering or loss can cause fragments of different datagrams to be interleaved and thus incorrectly reassembled and thus discarded. Reuse of ID values in atomic packets, as permitted by this document, can result in higher datagram loss in such cases. Such cases already can exist because there are known devices that use a constant ID for atomic packets (some cellphones), and there are known devices that ignore DF=1, but high levels of corresponding loss have not been reported. The lack of such reports indicates either a lack of reordering or loss in such cases, or a tolerance to the resulting losses. If such issues are reported, it

would be more productive to address non-compliant devices (that ignore DF=1), because it is impractical to define Internet specifications to tolerate devices that ignore those specifications. This is why this document emphasizes the need to honor DF=1, as well as that datagram transit devices need to retain the DF bit as received (i.e., rather than clear it).

5.5.2. Non-atomic Datagrams Experiencing Reordering or Loss

Non-atomic datagrams rely on the uniqueness of the ID value to tolerate reordering of fragments, notably where fragments of different datagrams are interleaved as a result of such reordering. Fragment loss can result in reassembly of fragments from different origin datagrams, which is why ID reuse in non-atomic datagrams is based on datagram (fragment) maximum lifetime, not just expected reordering interleaving.

This document does not change the requirements for uniqueness of IDs in non-atomic datagrams, and thus does not affect their tolerance to such reordering or loss. This document emphasizes the need for ID uniqueness for all datagram sources including rewriting middleboxes, the need to rate-limit sources to ensure ID uniqueness, the need to not reuse the ID for retransmitted datagrams, and the need to use higher-layer integrity checks to prevent reassembly errors - all of which result in a higher tolerance to reordering or loss events.

6. Updates to Existing Standards

The following sections address the specific changes to existing protocols indicated by this document.

6.1. Updates to RFC 791

RFC 791 states that:

> The originating protocol module of an internet datagram sets the identification field to a value that must be unique for that source-destination pair and protocol for the time the datagram will be active in the internet system.

And later that:

> Thus, the sender must choose the Identifier to be unique for this source, destination pair and protocol for the time the datagram (or any fragment of it) could be alive in the internet.

> It seems then that a sending protocol module needs to keep a table of Identifiers, one entry for each destination it has communicated with in the last maximum datagram lifetime for the internet.
>
> However, since the Identifier field allows 65,536 different values, some host may be able to simply use unique identifiers independent of destination.
>
> It is appropriate for some higher level protocols to choose the identifier. For example, TCP protocol modules may retransmit an identical TCP segment, and the probability for correct reception would be enhanced if the retransmission carried the same identifier as the original transmission since fragments of either datagram could be used to construct a correct TCP segment.

This document changes RFC 791 as follows:

- IPv4 ID uniqueness applies to only non-atomic datagrams.
- Retransmitted non-atomic IPv4 datagrams are no longer permitted to reuse the ID value.

6.2. Updates to RFC 1122

RFC 1122 states that:

> 3.2.1.5 Identification: RFC-791 Section 3.2
>
> When sending an identical copy of an earlier datagram, a host MAY optionally retain the same Identification field in the copy.
>
> DISCUSSION:
>
> Some Internet protocol experts have maintained that when a host sends an identical copy of an earlier datagram, the new copy should contain the same Identification value as the original. There are two suggested advantages: (1) if the datagrams are fragmented and some of the fragments are lost, the receiver may be able to reconstruct a complete datagram from fragments of the original and the copies; (2) a congested gateway might use the IP Identification field (and Fragment Offset) to discard duplicate datagrams from the queue.

This document changes RFC 1122 as follows:

- The IPv4 ID field is no longer permitted to be used for duplicate detection. This applies to both atomic and non-atomic datagrams.

- Retransmitted non-atomic IPv4 datagrams are no longer permitted to reuse the ID value.

6.3. Updates to RFC 2003

This document updates how IPv4-in-IPv4 tunnels create IPv4 ID values for the IPv4 outer header [RFC2003], but only in the same way as for any other IPv4 datagram source. In specific, RFC 2003 states the following, where ref. [10] is RFC 791:

> Identification, Flags, Fragment Offset
>
> These three fields are set as specified in [10]...

This document changes RFC 2003 as follows:

- The IPv4 ID field is set as permitted by RFCXXXX.

7. Security Considerations

When the IPv4 ID is ignored on receipt (e.g., for atomic datagrams), its value becomes unconstrained; that field then can more easily be used as a covert channel. For some atomic datagrams it is now possible, and may be desirable, to rewrite the IPv4 ID field to avoid its use as such a channel. Rewriting would be prohibited for datagrams protected by IPsec Authentication Header (AH), although we do not recommend use of AH to achieve this result [RFC4302].

The IPv4 ID also now adds much less to the entropy of the header of a datagram. Such entropy might be used as input to cryptographic algorithms or pseudorandom generators, although IDs have never been assured sufficient entropy for such purposes. The IPv4 ID had previously been unique (for a given source/address pair, and protocol field) within one MDL, although this requirement was not enforced and clearly is typically ignored. The IPv4 ID of atomic datagrams is not required unique, and so contributes no entropy to the header.

The deprecation of the IPv4 ID field’s uniqueness for atomic datagrams can defeat the ability to count devices behind a NAT/ASM/rewriter [Be02]. This is not intended as a security feature, however.

8. IANA Considerations

There are no IANA considerations in this document.

The RFC Editor should remove this section prior to publication

9. References

9.1. Normative References

[RFC791] Postel, J., "Internet Protocol", RFC 791 / STD 5, September 1981.

[RFC1122] Braden, R., Ed., "Requirements for Internet Hosts - Communication Layers", RFC 1122 / STD 3, October 1989.

[RFC1812] Baker, F. (Ed.), "Requirements for IP Version 4 Routers", RFC 1812 / STD 4, Jun. 1995.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119 / BCP 14, March 1997.

[RFC2003] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996.

9.2. Informative References

[Be02] Bellovin, S., "A Technique for Counting NATted Hosts", Internet Measurement Conference, Proceedings of the 2nd ACM SIGCOMM Workshop on Internet Measurement, Nov. 2002.

[Bo11] Boucadair, M., J. Touch, P. Levis, R. Penno, "Analysis of Solution Candidates to Reveal a Host Identifier in Shared Address Deployments", (work in progress), draft-boucadair-intarea-nat-reveal-analysis, Sept. 2011.

[De11] Despres, R. (Ed.), S. Matsushima, T. Murakami, O. Troan, "IPv4 Residual Deployment across IPv6-Service networks (4rd)", (work in progress), draft-despres-intarea-4rd, Mar. 2011.

[Pe11] Perreault, S., (Ed.), I. Yamagata, S. Miyakawa, A. Nakagawa, H. Ashida, "Common requirements of IP address sharing schemes", (work in progress), draft-ietf-behave-lsn-requirements, Mar. 2011.

[RFC1144] Jacobson, V., "Compressing TCP/IP Headers", RFC 1144, Feb. 1990.

[RFC2460] Deering, S., R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, Dec. 1998.

[RFC2508] Casner, S., V. Jacobson. "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links", RFC 2508, Feb. 1999.

[RFC2671] Vixie,P., "Extension Mechanisms for DNS (EDNS0)", RFC 2671, Aug. 1999.

[RFC3022] Srisuresh, P. and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, Jan. 2001.

[RFC3545] Koren, T., S. Casner, J. Geevarghese, B. Thompson, P. Ruddy, "Enhanced Compressed RTP (CRTP) for Links with High Delay, Packet Loss and Reordering", RFC 3545, Jul. 2003.

[RFC3828] Larzon, L-A., M. Degermark, S. Pink, L-E. Jonsson, Ed., G. Fairhurst, Ed., "The Lightweight User Datagram Protocol (UDP-Lite)", RFC 3828, Jul. 2004.

[RFC4301] Kent, S., K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, Dec. 2005.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, Dec. 2005.

[RFC4443] Conta, A., S. Deering, M. Gupta (Ed.), "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, March. 2006.

[RFC4960] Stewart, R. (Ed.), "Stream Control Transmission Protocol", RFC 4960, Sep. 2007.

[RFC4963] Heffner, J., M. Mathis, B. Chandler, "IPv4 Reassembly Errors at High Data Rates," RFC 4963, Jul. 2007.

[RFC5225] Pelletier, G., K. Sandlund, "RObust Header Compression Version 2 (ROHCv2): Profiles for RTP, UDP, IP, ESP and UDP-Lite", RFC 5225, Apr. 2008.

[RFC5320] Templin, F., Ed., "The Subnetwork Encapsulation and Adaptation Layer (SEAL)", RFC 5320, Feb. 2010.

[RFC6145] Li, X., C. Bao, F. Baker, "IP/ICMP Translation Algorithm," RFC 6145, Apr. 2011.

[RFC6219] Li, X., C. Bao, M. Chen, H. Zhang, J. Wu, "The China Education and Research Network (CERNET) IVI Translation Design and Deployment for the IPv4/IPv6 Coexistence and Transition", RFC 6219, May 2011.

[RFC6621] Macker, J. (Ed.), "Simplified Multicast Forwarding," RFC 6621, May 2012.

10. Acknowledgments

This document was inspired by of numerous discussions among the authors, Jari Arkko, Lars Eggert, Dino Farinacci, and Fred Templin, as well as members participating in the Internet Area Working Group. Detailed feedback was provided by Gorry Fairhurst, Brian Haberman, Ted Hardie, Mike Heard, Erik Nordmark, Carlos Pignataro, and Dan Wing. This document originated as an Independent Stream draft co-authored by Matt Mathis, PSC, and his contributions are greatly appreciated.

This document was prepared using 2-Word-v2.0.template.dot.

Author’s Address

Joe Touch
USC/ISI
4676 Admiralty Way
Marina del Rey, CA 90292-6695
U.S.A.

Phone: +1 (310) 448-9151
Email: touch@isi.edu

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: September 30, 2011

S. Matsushima
Softbank Telecom
Y. Yamazaki
Softbank Mobile
C. Sun
M. Yamanishi
J. Jiao
Softbank BB

March 29, 2011

Use case and consideration experiences of IPv4 to IPv6 transition draft-matsushima-v6ops-transition-experience-02

Abstract

Service Providers will apply their use case when conducting IPv6 transition and determine helpful solutions with the assistance of the IPv6 transition guideline document. More than one solution is possible, and decisions must be made from not only the technical point of view, but also from the economic point of view. This document describes the conclusions reached by one operator based on their considerations and their plans for IPv6 transition so as to assist others who may have similar circumstances.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 30, 2011.

Copyright Notice

Table of Contents

1. Introduction

IPv4 to IPv6 transition solutions are becoming more converged. Given the variety of operators involved, various use-case scenarios exist and efforts are underway to clarify them. Since the first group addressing IPv6 transition are technically inclined, the economic analyses needed for creating business plans are often delayed. One key factor impacting the business plan is architecture. The solution will be considered and then adopted so as to implement the most efficient architecture for each operator. In other words, the Service Provider who wants to ensure long-term viability must place greater emphasis on the economic impact of IPv6 transition. The author expect that IETF has great interest in this approach given its engineering and standardization work. Moreover, sharing the considerations described in this document would be helpful to operators who are in similar circumstances.

2. Transition overview and current status

Various transition use-cases have been published.

[I-D.huang-v6ops-v4v6tran-bb-usecase]
[I-D.lee-v6ops-tran-cable-usecase]
[I-D.tsou-v6ops-mobile-transition-guide] [I-D.sunq-v6ops-ivi-sp]

IPv6 transition guideline document
[I-D.arkko-ipv6-transition-guidelines] presents four deployment models. As our ultimate goal is the IPv6 only network, our strategy to achieving it is: (1) provide IPv6 connectivity to the existing IPv4-Only network, (2) build new IPv6-Only network, (3) migrate our customers from the IPv4-Only network to the IPv6-Only network. Along with the guideline, we had studied the "Crossing IPv4 Islands" model in the guideline to realize (1), while performing (2) in parallel. Subsequently, we started studying the "IPv6-Only Core Network" model to achieve (3). Research into a deployment model for our mobile network is now in progress.

3. Experience of IPv4-only Network and Assessment Approach

Our starting point is ensuring that the IPv4-Only network can provide IPv6 connectivity. Since our final goal is to build a IPv6-only network and migrate all customers to the network, we will not have to accommodate new customers beyond current capacity in the existing IPv4-only network. This means two things for the IPv4-only network; one, "minimized additional resources will provided to keep the network" and two, "there is less need to conserve IPv4 addresses in

the network". As the guideline document pointed out, many "IPv6 over IPv4 tunneling" solutions have already been developed. Our criterion for adopting the best solution involves not only technical pros/cons, but also the cost efficiency of providing IPv6 connectivity to all customers in the IPv4-only networks.

When the total capital and operational expense of the system is represented as "Q", and the number of customers that can be served by the system as "T", the metric of cost efficiency, "S", is given by the following simple formula:

S=Q/T

We gathered the S values of all candidate products and solutions, and decided to adopt the solution that had the lowest S value. Ignoring the price difference between the products, the stateful solutions have S values that are significantly different from those of the stateless solutions. In stateful solutions, T is the total number of system capable sessions divided by the number of sessions per customer. In stateless solutions, on the other hand, T is the total amount of system bandwidth capacity divided by the bandwidth consumption per customer.

From our experience, S(A) < S(B), that is, S(A) is always more efficient than S(B) (note S(A) is stateless, S(B) is stateful). We consequently adopt 6rd [RFC5969] for IPv4-only network. As the guideline document points out, it is not productive to implement an optimal IPv6 transition system as a temporary solution with goal of rich functionality. Many service providers hope that by allocating more resource they can increase network performance, bandwidth capacity, and the coverage of their network. In other words, we, as a service provider, want to minimize the resources allocated to such temporary solutions.

4. Considerations for IPv6-Only network

Our considerations suggest that a stateless solution should be adopted for the IPv6-only network to minimize overall resource allocation and to allocate resources to the more productive areas. In one of IPv6-only network deployment scenario, routing and addressing lie outside our control except for our own prefix, which is assigned to the customers who connect to the network. It seems like relation of operators among wholesale and retail. In that network, it is difficult to avoid assigning well known and other operator owned IPv4 prefixes if the stateless solution uses the 32bit IPv4 address to IPv6 address mapping technique. The solution must meet the requirements of: (1) The routing path for IPv4 should match

the optimized IPv6 routing path, (2) It should be capable to share one IPv4 address among customers since the number of IPv4 addresses is insufficient, (3) It must be stateless. We will adopt the solution that satisfies these three requirements. According to [I-D.sun-intarea-4rd-applicability], there are significant characteristics in particular these three requirements are satisfied. It is noted that since some customers require a service which no address sharing, a non-address sharing solution is also needed, but this does not need to be the same as the address sharing solution.

The guideline document describes that Dual-Stack-lite [I-D.ietf-softwire-dual-stack-lite] is recommended only as a transition solution on the way to the IPv6-only network. Compared to other deployment scenarios such as crossing IPv4 island and IPv6-only deployment, there are several candidate solutions for each deployment model but only one solution for the scenario. It is noted that the solutions not mentioned in the guideline are discussed in [I-D.dec-stateless-4v6], which adopt 4rd [I-D.despres-intarea-4rd] and dIVI [I-D.xli-behave-divi].

5. Considerations for Mobile network

We believe that the requirements explained in the previous section should be applied to the mobile network as well. [TR23.975], has clarified the IPv6-only deployment model in the guideline as a IPv6 transition scenario. As [I-D.arkko-ipv6-only-experience] pointed out, the operators' policy of service quality assurance may require the solution of avoiding the IPv4 referral issue [I-D.ietf-behave-v4v6-bih]

It is interesting that stateless address mapping techniques exist for both encapsulation/decapsulation and translation in the case of IPv4 crossing IPv6-only network model. This means that, the requirements listed in previous section could be achieved for the mobile network.

6. Conclusions

One of most significant areas that remain to be investigated is the physical resources of our network. We also need to minimize the investments needed to secure the IP transition (i.e. the temporary solutions) because we believe that the ultimate goal of the transition must be the long-term viability of the Internet and also the provision of our services. To ensure that, our considerations yielded the conclusion that the stateless solution should be specified for all deployment models in the guideline document. It is recommended that IETF standardize on stateless solutions for not only

the IPv4-only network, but also both the IPv6-only network and Ipv6-only deployment models in the guideline.

7. Security considerations

A stateless solution without the appropriate implementation and operation techniques would be vulnerable to denial of service attacks, routing loops, spoofing, and other such malicious acts. To eliminate these security vulnerabilities, a stateless solution, like 6rd, which is capable of validating consistency of IPv6 source address with IPv4 source address, can be used to avoid these vulnerabilities, based on its address mapping rule. If a stateless solution supports IPv4 address sharing, it must take into account the issues described in [I-D.ietf-intarea-shared-addressing-issues]. If an operator is concerned about the unnecessary bandwidth consumption created by unwanted packets from the outside, one recommended solution is to implement appropriate firewall protection for not only v4v6 transition solution, but also both native IPv4 and IPv6 networks.

8. Acknowledgements

The authors would like to thank the guideline document of IPv6 transition [I-D.arkko-ipv6-transition-guidelines], which guides us through the transition way, and has motivated the authors to write this document. We also would like to thank Miwa Fujii for her helpful suggestions and supports to share our experience with many people.

9. References

9.1. Normative References

[I-D.arkko-ipv6-transition-guidelines]
Arkko, J. and F. Baker, "Guidelines for Using IPv6 Transition Mechanisms during IPv6 Deployment", draft-arkko-ipv6-transition-guidelines-14 (work in progress), December 2010.

9.2. Informative References

[I-D.arkko-ipv6-only-experience]
Arkko, J. and A. Keranen, "Experiences from an IPv6-Only Network", draft-arkko-ipv6-only-experience-02 (work in progress), October 2010.

[I-D.dec-stateless-4v6]
Dec, W., "Stateless 4Via6 Address Sharing", draft-dec-stateless-4v6-01 (work in progress), March 2011.

[I-D.despres-intarea-4rd]
Despres, R., Matsushima, S., Murakami, T., and O. Troan, "IPv4 Residual Deployment across IPv6-Service networks (4rd) ISP-NAT's made optional", draft-despres-intarea-4rd-01 (work in progress), March 2011.

[I-D.huang-v6ops-v4v6tran-bb-usecase]
Huang, C., Li, X., and L. Hu, "Use Case For IPv6 Transition For a Large-Scale Broadband network", draft-huang-v6ops-v4v6tran-bb-usecase-01 (work in progress), October 2010.

[I-D.ietf-behave-v4v6-bih]
Huang, B., Deng, H., and T. Savolainen, "Dual Stack Hosts Using "Bump-in-the-Host" (BIH)", draft-ietf-behave-v4v6-bih-03 (work in progress), March 2011.

[I-D.ietf-intarea-shared-addressing-issues]
Ford, M., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Issues with IP Address Sharing", draft-ietf-intarea-shared-addressing-issues-05 (work in progress), March 2011.

[I-D.ietf-softwire-dual-stack-lite]
Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", draft-ietf-softwire-dual-stack-lite-07 (work in progress), March 2011.

[I-D.lee-v6ops-tran-cable-usecase]
Lee, Y. and V. Kuarsingh, "IPv6 Transition Cable Access Network Use Cases", draft-lee-v6ops-tran-cable-usecase-00 (work in progress), October 2010.

[I-D.sun-intarea-4rd-applicability]
Sun, C., Matsushima, S., and J. Jiao, "4rd Applicability Statement", draft-sun-intarea-4rd-applicability-01 (work in progress), March 2011.

[I-D.sunq-v6ops-ivi-sp]
Sun, Q., Xie, C., Li, X., Bao, C., and M. Feng, "Considerations for Stateless Translation (IVI/dIVI) in

Large SP Network", draft-sunq-v6ops-ivi-sp-02 (work in progress), March 2011.

[I-D.tsou-v6ops-mobile-transition-guide]
ZOU), T. and T. Taylor, "IPv6 Transition Guide For A Large Mobile Operator", draft-tsou-v6ops-mobile-transition-guide-00 (work in progress), October 2010.

[I-D.xli-behave-divi]
Li, X., Bao, C., and H. Zhang, "Address-sharing stateless double IVI", draft-xli-behave-divi-01 (work in progress), October 2009.

[I-D.ymbk-aplusp]
Bush, R., "The A+P Approach to the IPv4 Address Shortage", draft-ymbk-aplusp-09 (work in progress), February 2011.

[RFC5565] Wu, J., Cui, Y., Metz, C., and E. Rosen, "Softwire Mesh Framework", RFC 5565, June 2009.

[RFC5969] Townsley, W. and O. Troan, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification", RFC 5969, August 2010.

[TR23.975]
"3GPP, IPv6 migration guidelines", <http://www.3gpp.org/ftp/specs/html-info/23975.htm>.

Authors' Addresses

Satoru Matsushima
Softbank Telecom
Tokyo Shiodome Building
1-9-1,Higashi-Shibashi,Minato-Ku
Tokyo 105-7322
JAPAN

Email: satoru.matsushima@tm.softbank.co.jp

Yuji Yamazaki
Softbank Mobile
Tokyo Shiodome Building
1-9-1,Higashi-Shibashi,Minato-Ku
Tokyo 105-7322
JAPAN

Email: yuyamaza@bb.softbank.co.jp

Chunfa Sun
Softbank BB
Tokyo Shiodome Building
1-9-1,Higashi-Shibashi,Minato-Ku
Tokyo 105-7322
JAPAN

Email: c-sun@bb.softbank.co.jp

Masato Yamanishi
Softbank BB
611 Wilshire Blvd., Suite 400
Los Angeles, CA
USA

Email: myamanis@bb.softbank.co.jp

Jie Jiao
Softbank BB
Tokyo Shiodome Building
1-9-1,Higashi-Shibashi,Minato-Ku
Tokyo 105-7322
JAPAN

Email: j-jiao@bb.softbank.co.jp