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.

Table of Contents

1. Abstract 3
1.1. Requirements Language 3
2. Problem Statement and Purpose of IPv4 address efficiency . . 3
2.1. Illustrative service provider use case 4
2.2. Architecture and Communication in a PPP Dual-Stack environment 5
2.3. The advantage of the dynamic IPv4 address assigning feature 7
3. Specification 9
3.1. Definition of the participating elements and their functionalities 10
3.2. Assigning IPv4 address parameter on-demand after establishing PPP session with IPv6 connectivity 11
3.3. Releasing unused IPv4 address parameters 12
3.4. Timer Considerations 13
4. Potential for optimization 14
4.1. Avoiding unnecessary load on BRAS/NAS and AAA 14
4.2. Reducing IPv4 traffic on external interfaces 15
5. Impacts on user experience and operation 15
5.1. Impacts on user experience and Happy Eyeballs implementations 15

5.2. Operational impacts 16
6. Acknowledgements 17
7. IANA Considerations 17
8. Security Considerations 17
9. References 18
9.1. Normative Reference 18
9.2. Informative References 19
Appendix A. Workplan 19
Authors' Addresses 19

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).

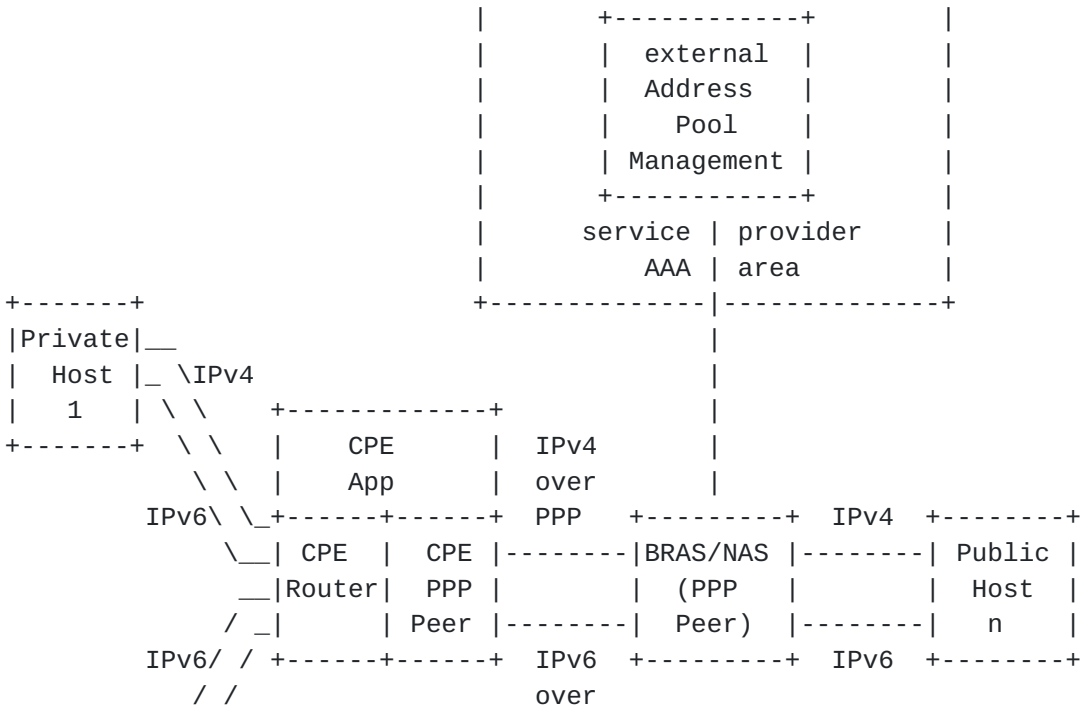
The last two points are mainly left to the responsibility of the network and service providers. The approach, sketched in this document, is based on the operational scenario that the customer starts a Dual-Stack PPP session in "IPv6-only" mode first and "adds" IPv4 later on only in the case that applications 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.



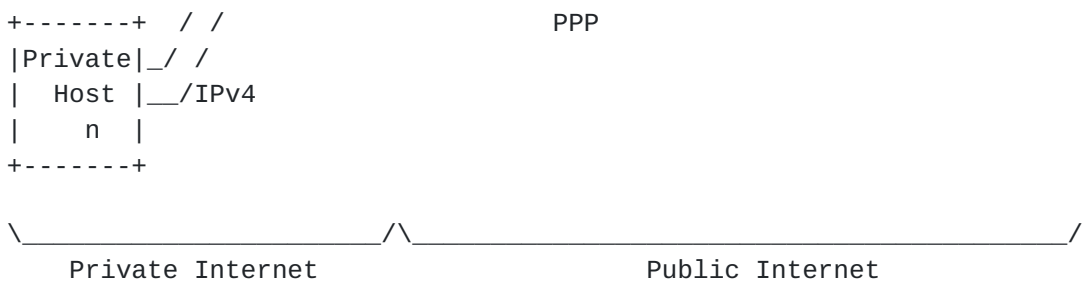


Figure 1: PPP Dual-Stack architecture

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.

o 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.

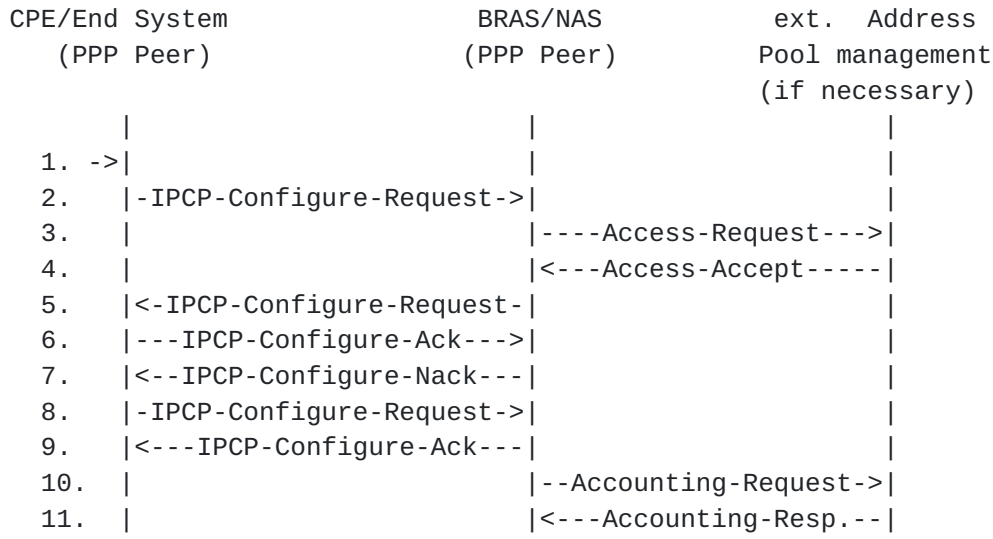


Figure 2: Message flow for assigning IPv4 address parameter

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.

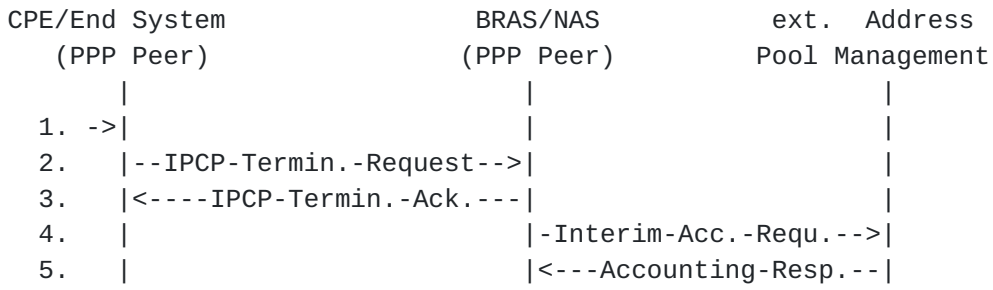


Figure 3: Message flow for releasing IPv4 address parameter

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