

Network Working Group
Internet-Draft
Intended status: Informational
Expires: July 1, 2011

J. Arkko
Ericsson
F. Baker
Cisco Systems
December 28, 2010

Guidelines for Using IPv6 Transition Mechanisms during IPv6 Deployment
draft-arkko-ipv6-transition-guidelines-14

Abstract

The Internet continues to grow beyond the capabilities of IPv4. An expansion in the address space is clearly required. With its increase in the number of available prefixes and addresses in a subnet, and improvements in address management, IPv6 is the only real option on the table. Yet, IPv6 deployment requires some effort, resources, and expertise. The availability of many different deployment models is one reason why expertise is required. This document discusses the IPv6 deployment models and migration tools, and recommends ones that have been found to work well in operational networks in many common situations.

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 July 1, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Terminology	4
3. Principles	4
3.1. Goals	5
3.2. Choosing a Deployment Model	6
4. Guidelines for IPv6 Deployment	8
4.1. Native Dual Stack	8
4.2. Crossing IPv4 Islands	10
4.3. IPv6-Only Core Network	11
4.4. IPv6-only Deployment	12
5. Conclusions	14
6. Further Reading	15
7. Security Considerations	15
8. IANA Considerations	16
9. References	16
9.1. Normative References	16
9.2. Informative References	17
Appendix A. Acknowledgments	20
Authors' Addresses	20

1. Introduction

The Internet continues to grow beyond the capabilities of IPv4. The tremendous success of the Internet has strained the IPv4 address space, which is no longer sufficient to fuel future growth. At the time of this writing, August 2010, the IANA "free pool" contains only 14 unallocated unicast IPv4 /8 prefixes. Credible estimates based on past behavior suggest that the RIRs will exhaust their remaining address space by early 2012, apart from the development of a market in IPv4 address space. An expansion in the address space is clearly required. With its increase in the number of available prefixes and addresses in a subnet, and improvements in address management, IPv6 is the only real option on the table.

John Curran, in his Internet Transition Plan [RFC5211], gives estimated dates for significant points in the transition; while the tail of the process will likely be long, it is clear that deployment is a present reality and requirement.

Accordingly, many organizations have employed or are planning to employ IPv6 in their networks. Yet, IPv6 deployment requires some effort, resources, and expertise. This is largely a natural part of maintaining and evolving a network: changing requirements are taken into account in normal planning, procurement and update cycles. Very large networks have successfully adopted IPv6 alongside IPv4, with surprisingly little effort.

However, in order to successfully make this transition, some amount of new expertise is required. Different types of experience will be required: basic understanding of IPv6 mechanisms, debugging tools, product capabilities and caveats when used with IPv6, and so on. The availability of many different IPv6 deployment models and tools is an additional reason why expertise is required. These models and tools have been developed over the years at the IETF, some for specific circumstances and others for more general use. They differ greatly in their principles of operation. Over time, views about the best ways to employ the tools have evolved. Given the number of options, network managers are understandably confused. They need guidance on recommended approaches to IPv6 deployment.

The rest of this document is organized as follows. Section 2 introduces some terminology, Section 3 discusses some of the general principles behind choosing particular deployment models and tools, Section 4 goes through the recommended deployment models for common situations, and Section 5 provides some concluding remarks about the choice between these models.

Many networks can follow one of the four scenarios described in this

document. However, variations will certainly occur in the details, and there will be questions such as the particular choice of tunneling solution for which there is no "one size fits all" answer. Network managers must each take the responsibility of choosing the best solution for their own case. This document does not attempt to provide guidance for all possible networking situations. Also, a systematic operational plan for the transition is required, but the details depend entirely on the individual network.

2. Terminology

In this document, the following terms are used.

IPv4/IPv4 NAT: refers to any IPv4-to-IPv4 network address translation algorithm, both "Basic NAT" and "Network Address/Port Translator (NAPT)", as defined by [RFC2663].

Dual Stack: refers to a technique for providing complete support for both Internet protocols -- IPv4 and IPv6 -- in hosts and routers [RFC4213].

Dual Stack Lite: also called "DS-Lite", refers to a technique that employs tunneling and IPv4/IPv4 NAT to provide IPv4 connectivity over IPv6 networks [I-D.ietf-softwire-dual-stack-lite].

IPv4-only domain: as defined in [I-D.ietf-behave-v6v4-framework], a routing domain in which applications can only use IPv4 to communicate, whether due to host limitations, application limitations, or network limitations.

IPv6-only domain: as defined in [I-D.ietf-behave-v6v4-framework], a routing domain in which applications can only use IPv6 to communicate, whether due to host limitations, application limitations, or network limitations.

NAT-PT: refers to a specific, old design of a Network Address Translator - Protocol Translator defined in [RFC2766] and deprecated due to the reasons stated in [RFC4966].

3. Principles

The primary goal is to facilitate the continued growth of the networking industry and deployment of Internet technology at relatively low capital and operational expense without destabilizing deployed services or degrading customer experience. This is at risk with IPv4 due to the address runout; economics teaches us that a

finite resource, when stressed, becomes expensive, either in the actual cost of the resource or in the complexity of the technology and processes required to manage it. It is also at risk while both IPv4 and IPv6 are deployed in parallel, as it costs more to run two technologies than one. To this end, since IPv4 clearly will not scale to meet our insatiable requirements, the primary technical goals are the global deployment of IPv6 both in the network, in its service infrastructure, and by applications, resulting in the end of the requirement to deploy two IP versions and the obsolescence of transitional mechanisms. Temporary goals in support of this focus on enabling parts of the Internet to employ IPv6 and disable IPv4 before the entire Internet has done so.

3.1. Goals

The end goal is network-wide native IPv6 deployment, resulting in the obsolescence of transitional mechanisms based on encapsulation, tunnels, or translation, and also resulting in the obsolescence of IPv4. Transition mechanisms, taken as a class, are a means to an end, to simplify the process for the network administration.

However, the goals, constraints, and opportunities for IPv6 deployment differ from one case to another. There is no single right model for IPv6 deployment, just like there is no one and only model for IPv4 network design. Some guidelines can be given, however. Common deployment models that have been found to work well are discussed in Section 4, and the small set of standardized IETF migration tools support these models. But first it may be useful to discuss some general principles that guide our thinking about what is a good deployment model.

It is important to start the deployment process in a timely manner. Most of the effort is practical -- network audit, network component choices, network management, planning, implementation -- and at the time of this writing, reasonably easily achievable. There is no particular advantage to avoiding dealing with IPv6 as part of the normal network planning cycle. The migration tools already exist, and while additional features continue to be developed it is not expected that they radically change what networks have to do. In other words, there is no point in waiting for an improved design.

There are only a few exceptional networks where co-existence with IPv4 is not a consideration at all. These networks are typically new deployments, strictly controlled by a central authority, and have no need to deal with legacy devices. For example, specialized machine-to-machine networks that communicate only to designated servers, such as Smart Grids, can easily be deployed as IPv6-only networks. Mobile telephone network operators, especially those using 3GPP, have

seriously considered IPv6-only operation, and some have deployed it. Research networks that can be separated from the IPv4 Internet to find out what happens are also a candidate. In most other networks IPv4 has to be considered. A typical requirement is that older, IPv4-only applications, systems, or services must be accommodated. Most networks that cross administrative boundaries or allow end user equipment have such requirements. Even in situations where the network consists of only new, IPv6-capable devices it is typically required that the devices can communicate with the IPv4 Internet.

It is expected that after a period of supporting both IPv4 and IPv6, IPv4 can eventually be turned off. This should happen gradually. For instance, a service provider network might stop providing IPv4 service within its own network, while still allowing its IPv6 customers to access the rest of the IPv4 Internet through overlay, proxy, or translation services. Regardless of progress in supporting IPv6, it is widely expected that some legacy applications and some networks will continue to run only over IPv4 for many years. All deployment scenarios need to deal with this situation.

3.2. Choosing a Deployment Model

The first requirement is that the model or tool actually allows communications to flow and services to appropriately be delivered to customers without perceived degradation. While this sounds too obvious to even state, it is sometimes not easy to ensure that a proposed model does not have failure modes related to supporting older devices, for instance. A network that is not serving all of its users is not fulfilling its task.

The ability to communicate is also far more important than fine-grained performance differences. In general, it is not productive to focus on optimization of a design that is designed to be temporary, such as a migration solution necessarily is. Consequently, existing tools are often preferred over new ones, even if for some specific circumstance it would be possible to construct a slightly more efficient design.

Similarly, migration tools that can be disposed after a period of co-existence are preferred over tools that require more permanent changes. Such permanent changes may incur costs even after the transition to IPv6 has been completed.

Looking back on the deployment of Internet technology, some of the factors that have been important for success include
[RFC5218, Baker.Shanghai]

- o The ability to offer a valuable service. In the case of the Internet, connectivity has been this service.
- o The ability to deploy the solution in an incremental fashion.
- o Simplicity. This has been a key factor in making it possible for all types of devices to support the Internet protocols.
- o Openly available implementations. These make it easier for researchers, start-ups and others to build on or improve existing components.
- o The ability to scale. The IPv4 Internet grew far larger than its original designers had anticipated, and scaling limits only became apparent 20-30 years later.
- o The design supports robust interoperability rather than mere correctness. This is important in order to ensure that the solution works in different circumstances and in an imperfectly controlled world.

Similar factors are also important when choosing IPv6 migration tools. Success factors should be evaluated in the context of a migration solution. For instance, incremental deployability and lack of dependencies to components that are under someone else's control are key factors.

It is also essential that any chosen designs allow the network to be maintained, serviced, diagnosed and measured. The ability of the network to operate under many different circumstances and surprising conditions is a key. Any large network that employs brittle components will incur significant support costs.

Properly executed IPv6 deployment normally involves a step-wise approach where individual functions or parts of the network are updated at different times. For instance, IPv6 connectivity has to be established and tested before DNS entries with IPv6 addresses can be provisioned. Or specific services can be moved to support IPv6 earlier than others. In general, most deployment models employ a very similar network architecture for both IPv4 and IPv6. The principle of changing only the minimum amount necessary is applied here. As a result, some features of IPv6, such as the ability to have an effectively unlimited number of hosts on a subnet, may not be available in the short term.

4. Guidelines for IPv6 Deployment

This section presents a number of common scenarios along with recommended deployment tools for them. We start from the most obvious deployment situation where native connectivity is available and both IP versions are used. Since native IPv6 connectivity is not available in all networks, our second scenario looks at ways of arranging such connectivity over the IPv4 Internet. The third scenario is more advanced and looks at a service provider network that runs only on IPv6 but which is still capable of providing both IPv6 and IPv4 services. The fourth and most advanced scenario focuses on translation, at the application or the network layer.

Note that there are many other possible deployment models and existing specifications to support such models. These other models are not necessarily frowned upon. However, they are not expected to be the mainstream deployment models, and consequently, the associated specifications are typically not IETF standards track RFCs. Network managers should not adopt these non-mainstream models lightly, however, as there is little guarantee that they work well. There are also models that are believed to be problematic. An older model to IPv6 - IPv4 translation (NAT-PT) [RFC2766] suffers from a number of drawbacks arising from, for example, its attempt to capture DNS queries on path [RFC4966]. Another example regarding the preference to employ tunneling instead of double translation will be discussed later in this document.

4.1. Native Dual Stack

The simplest deployment model is Dual Stack: one turns on IPv6 throughout one's existing IPv4 network and allows applications using the two protocols to operate as ships in the night. This model is applicable to most networks - home, enterprise, service provider, or content provider network.

The purpose of this model is to support any type of device and communication, and to make it an end-to-end choice which IP version is used between the peers. There are minimal assumptions about the capabilities and configuration of hosts in these networks. Native connectivity avoids problems associated with the configuration of tunnels and Maximum Transfer Unit (MTU) settings. As a result, these networks are robust and reliable. Accordingly, this is the recommended deployment model for most networks, and supported by IETF standards such as dual stack [RFC4213] and address selection [RFC3484]. Similarly, while there are some remaining challenges, this model is also preferred by many service providers and network managers [RFC6036] [I-D.arkko-ipv6-only-experience].

The challenges associated with this model are two-fold. First, while dual-stack allows each individual network to deploy IPv6 on their own, actual use still requires participation from all parties between the peers. For instance, the peer must be reachable over IPv6, have an IPv6 address to itself, and advertise such an address in the relevant naming service (such as the DNS). This can create a situation where IPv6 has been turned on in a network but there is little actual traffic. One direct way to affect this situation is to ensure that major destinations of traffic are prepared to receive IPv6 traffic. Current Internet traffic is highly concentrated on selected content provider networks, and making a change in even a small number of these networks can have significant effects. This was recently observed when YouTube started supporting IPv6 [networkworld.youtube]. There are scenarios where these means are insufficient. The following sections discuss deployment models that enable parts of the network deploy IPv6 faster than other parts.

The second challenge is that not all applications deal gracefully with situations where one of the alternative destination addresses works unreliably. For instance, if IPv6 connectivity is unreliable it may take a long time for some applications to switch over to IPv4. As a result, many content providers are shying away from advertising IPv6 addresses in DNS. This in turn exacerbates the first challenge. Long term, the use of modern application toolkits and APIs solves this problem. In the short term some content providers and user network managers have made a mutual agreement to resolve names to IPv6 addresses. Such agreements are similar to peering agreements and have been seen as necessary by many content providers. These "whitelisting" practices have some downsides as well, however. In particular, they create a dependency on an external party for moving traffic to IPv6. Nevertheless, there are many types of traffic in the Internet and only some of it requires such careful coordination. Popular peer-to-peer systems can automatically and reliably employ IPv6 connectivity where it is available, for instance.

Despite these challenges the native dual stack connectivity model remains the recommended approach. It is responsible for a large part of the progress on world-wide IPv6 deployment to date. The largest IPv6 networks; notably national research and education networks, Internet II, Renater, and others, employ this approach.

The original intent of dual stack was to deploy both IP versions alongside each other before IPv4 addresses were to run out. As we know, this never happened and deployment now has to take place with limited IPv4 addresses. Employing dual stack together with a traditional IPv4 address translator (IPv4/IPv4 NAT) is a very common configuration. If the address translator is acceptable for the network from a pure IPv4 perspective, this model can be recommended

from a dual stack perspective as well. The advantage of IPv6 in this model is that it allows direct addressing of specific nodes in the network, creating a contrast to the translated IPv4 service as noted in [RFC2993] and [I-D.ietf-intarea-shared-addressing-issues]. As a result, it allows the construction of IPv6-based applications that offer more functionality.

There may also be situations where a traditional IPv4 address translator is no longer sufficient. For instance, in typical residential networks, each subscriber is given one global IPv4 address, and the subscriber's IPv4/IPv4 NAT device may use this address with as many devices as it can handle. As IPv4 address space becomes more constrained and without substantial movement to IPv6, it is expected that service providers will be pressured to assign a single global IPv4 address to multiple subscribers. Indeed, in some deployments this is already the case. The dual stack model is still applicable even in these networks, but the IPv4/IPv4 Network Address Transition (NAT) functionality may need to be relocated and enhanced. On some networks it is possible to employ overlapping private address space [I-D.miles-behave-l2nat] [I-D.arkko-dual-stack-extra-lite]. Other networks may require a combination of IPv4/IPv4 NAT enhancements and tunneling. These scenarios are discussed further in Section 4.3.

4.2. Crossing IPv4 Islands

Native IPv6 connectivity is not always available, but fortunately it can be established using tunnels. Tunneling introduces some additional complexity and has a risk of MTU or other mis-configurations. However, its benefit is that it decouples addressing inside and outside the tunnel, making it easy to deploy IPv6 without having to modify routers along the path. Tunneling should be used when native connectivity can not be established, such as when crossing another administrative domain or a router that cannot be easily re-configured.

There are several types of tunneling mechanisms, including manually configured IPv6-over-IPv4 tunnels [RFC4213], 6to4 [RFC3056], automatic host-based tunnels [RFC4380], tunnel brokers [RFC3053], running IPv6 over MPLS with IPv6 Provider Edge Routers (6PE) [RFC4798], the use of Virtual Private Network (VPN) or mobility tunnels to carry both IPv4 and IPv6 [RFC4301] [RFC5454] [RFC5555] [RFC5844] and many others. More advanced solutions provide a mesh-based framework of tunnels [RFC5565].

On a managed network, there are no major challenges with tunneling beyond the possible configuration and MTU problems. Tunneling is very widely deployed both for IPv6 connectivity and other reasons,

and well understood. In general, the IETF recommends that tunneling be used if it is necessary to cross a segment of IP version X when communicating from IP version Y to Y. An alternative design would be to employ protocol translation twice. However, this design involves problems similar to those created by IPv4 address translation and is largely untried technology in any larger scale.

On an unmanaged network there have been a number of problems, however. In general, solutions aimed at early adopters (such as 6to4) have at times caused IPv6 connectivity to appear to be available on a network when in fact there is no connectivity. This in turn has lead to need by the content providers to serve IPv6 results for DNS queries only for trusted peers with known high-quality connectivity.

The IPv6 Rapid Deployment (6RD) [RFC5969] approach is a newer version of the 6to4 tunneling solution without the above drawbacks. It offers systematic IPv6 tunneling over IPv4 across an ISP, correspondence between IPv4 and IPv6 routing, and can be deployed within an ISP without the need to rely on other parties.

4.3. IPv6-Only Core Network

An emerging deployment model uses IPv6 as the dominant protocol at a service provider network, and tunnels IPv4 through this network in a manner converse to the one described in the previous section. There are several motivations for choosing this deployment model:

- o There may not be enough public or private IPv4 addresses to support network management functions in an end-to-end fashion, without segmenting the network into small parts with overlapping address space.
- o IPv4 address sharing among subscribers may involve new address translation nodes within the service provider's network. IPv6 can be used to reach these nodes. Normal IPv4 routing is insufficient for this purpose, as the same addresses would be used in several parts of the network.
- o It may be simpler for the service provider to employ a single-version network.

The recommended tool for this model is Dual Stack Lite [I-D.ietf-softwire-dual-stack-lite]. Dual Stack Lite provides both relief for IPv4 address shortage and makes forward progress on IPv6 deployment, by moving service provider networks and IPv4 traffic over IPv6. Given the IPv6 connectivity that Dual Stack Lite runs over, it becomes easy to provide IPv6 connectivity all the way to the end

users as well.

4.4. IPv6-only Deployment

Our final deployment model breaks the requirement that all parties must upgrade to IPv6 before any end-to-end communications use IPv6. This model makes sense when the following conditions are met:

- o There is a fact or requirement that there be an IPv4-only domain and an IPv6-only domain.
- o There is a requirement that hosts in the IPv4-only domain access servers or peers in the IPv6-only domain and vice versa.

This deployment model would fit well, for instance, a corporate or mobile network that offers IPv6-only networking but where users still wish to access content from the IPv4 Internet.

When we say "IPv4-only" or "IPv6-only", we mean that the applications can communicate only using IPv4 or IPv6; this might be due to lack of capabilities in the applications, host stacks, or the network; the effect is the same. The reason to switch to an IPv6-only network may be a desire to test such a configuration, or to simplify the network. It is expected that as IPv6 deployment progresses, the second reason will become more prevalent. One particular reason for considering an IPv6-only domain is the effect of overlapping private address space to applications. This is important in networks that have exhausted both public and private IPv4 address space and where arranging an IPv6-only network is easier than dealing with the overlapping address space in applications.

Note that the existence of an IPv6-only domain requires that all devices are indeed IPv6-capable. In today's mixed networking environments with legacy devices this can not always be guaranteed. But it can be arranged in networks where all devices are controlled by a central authority. For instance, newly built corporate networks can ensure that the latest device versions are in use. Some networks can also be engineered to support different services over an underlying network and as such, can support IPv6-only networking more easily. For instance, a cellular network may support IPv4-only connectivity for the installed base of existing devices and IPv6-only connectivity for incremental growth with newer IPv6 capable handsets. Similarly, a broadband ISP may support dual stack connectivity for customers that require both IPv4 and IPv6, and offer IPv6-only and NAT64 service for others. In the case of 3GPP and DOCSIS 3.0 access networks, the underlying access network architecture allows the flexibility to run different services in parallel to suit the various needs of the customer and the network operator.

It is also necessary for the network operator to have some level of understanding of what applications are used in the network, enabling him to ensure that any communication exchange is in fact predictable, capable of using IPv6, and translatable. In such a case, full interoperability can be expected. This has been demonstrated with some mobile devices, for instance. Note that the requirements on applications are similar to those in networks employing IPv4 NAT technology.

One obvious IPv6-only deployment approach applies to applications that include proxies or relays. One might position a web proxy, a mail server, or a SIP (Session Initiation Protocol) and media stream back-to-back user agent across the boundary between IPv4 and IPv6 domains, so that the application terminates IPv4 sessions on one side and IPv6 sessions on the other. Doing this preserves the end-to-end nature of communications from the gateway to the communicating peer. For obvious reasons, this solution is preferable to the implementation of Application Layer Gateways in network layer translators.

The other approach is network layer IPv4/IPv6 translation as described in IPv4/IPv6 Translation [I-D.ietf-behave-v6v4-framework] [I-D.ietf-behave-v6v4-xlate] [I-D.ietf-behave-v6v4-xlate-stateful] [RFC6052] [I-D.ietf-behave-dns64] [I-D.ietf-behave-ftp64]. IPv4/IPv6 translation at the network layer is similar in its advantages and disadvantages to IPv4/IPv4 translation. It allows a network to provide two types of services to IPv6-only hosts:

- o a relatively small set of systems may be configured with IPv4-mapped addresses, enabling stateless interoperation between IPv4-only and IPv6-only domains, each of which can use the other as peers or servers, and
- o a larger set of systems with global IPv6 addresses, which can access IPv4 servers using stateful translation but which are inaccessible as peers or servers from the IPv4-only domain.

The former service is used today in some university networks, and the latter in some corporate and mobile networks. The stateless service is naturally better suited for servers, and the stateful service for large numbers of client devices. The latter case occurs typically in a public network access setting. The two services can of course also be used together. In this scenario, network layer translation provides for straightforward services for most applications crossing the IPv4-only/IPv6-only boundary.

One challenge in this model is that as long as IPv4 addresses are still shared, issues similar to those caused by IPv4 NATs will still

appear [I-D.ietf-intarea-shared-addressing-issues]. Another challenge relates to communications involving IPv4 referrals. IPv4-literals within certain protocols and formats such as HTML, will fail when passed to IPv6-only hosts since the host does not have an IPv4 address to source the IPv4 communications or an IPv4 route. Measurements on the public internet show that literals appear in a tiny but measurable part of web pages [I-D.arkko-ipv6-only-experience], though whether this poses a practical problem is debatable. If this poses a particular problem for the types of applications in use, proxy configurations could be modified to use a proxy for the traffic in question, hosts could be modified to understand how they can map IPv4 literals to IPv6 addresses, or native dual stack could be employed instead.

5. Conclusions

The fundamental recommendation is to turn IPv6 on. Section 4 described four deployment models to do that, presented in rough order of occurrence in the world at the time of this writing. The first models are the most widely deployed today. All four models are recommended by the IETF, though again the first models should be considered first.

As noted in Section 1, variations occur in details and network managers are ultimately in charge of choosing the best solution for their own case. Benefits and challenges discussed in the previous sections should be considered when weighing deployment alternatives. The transition mechanisms that operators have deployed have been a mixed blessing; native dual stack deployments are not used to their full extent if peers have not upgraded, tunnel mechanisms that don't follow the routing of the underlying network have been problematic, and translation has its faults as well. Nevertheless, operators have successfully deployed very large networks with these models.

Some additional considerations are discussed below.

- o There is a tradeoff between ability to connect as many different types of devices as possible and the ability to move forward with deployment as independently as possible. As an example, native dual stack ensures best connectivity but requires updates in peer systems before actual traffic flows over IPv6. Conversely, IPv6-only networks are very sensitive to what kind of devices they can support, but can be deployed without any expectation of updates on peer systems.
- o Greenfield networks and networks with existing IPv4 devices and users need to be treated differently. In the latter case turning

on IPv6 in addition to IPv4 seems the rational choice. In the former case an IPv6-only model may make sense.

- o The right deployment model choices also vary as time goes by. For instance, a tunneling solution that makes sense today may become a native dual stack solution as the network and devices in the network evolve. Or an IPv6-only network becomes feasible when a sufficient fraction of client devices become IPv6-enabled.

No matter which deployment model is chosen, many of the important implications of IPv6 deployment are elsewhere within the network: IPv6 needs to be taken into account in network management systems and operations, address assignments, service agreements, firewalls and intrusion detection systems, and so on.

6. Further Reading

Various aspects of IPv6 deployment have been covered in several documents. Of particular interest may be the basic dual stack definition [RFC4213], application aspects [RFC4038], deployment in Internet Service Provider Networks [RFC4029] [RFC6036], deployment in enterprise networks [RFC4057] [RFC4852], IPv6-only deployment [I-D.arkko-ipv6-only-experience], and considerations in specific access networks such as cellular networks [RFC3314] [RFC3574] [RFC4215] [I-D.ietf-v6ops-v6-in-mobile-networks] or 802.16 networks [RFC5181].

This document provides general guidance on IPv6 deployment models that have been found suitable for most organizations. The purpose of this document is not to enumerate all special circumstances that may warrant other types of deployment models or the details of the necessary transition tools. Many of the special cases and details have been discussed in the above documents.

7. Security Considerations

While there are detailed differences between the security properties and vulnerabilities between IPv4 and IPv6, in general they provide a very similar level of security, and are subject to the same threats. With both protocols, specific security issues are more likely to be found at the practical level than in the specifications. The practical issues include, for instance, bugs or available security mechanisms on a given product. When deploying IPv6, it is important to ensure that the necessary security capabilities exist on the network components even when dealing with IPv6 traffic. For instance, firewall capabilities have often been a challenge in IPv6

deployments.

This document has no impact on the security properties of specific IPv6 transition tools. The security considerations relating to the transition tools are described in the relevant documents, for instance, [RFC4213] [I-D.ietf-behave-dns64] [I-D.ietf-softwire-dual-stack-lite] [I-D.ietf-v6ops-tunnel-security-concerns].

8. IANA Considerations

This document has no IANA implications.

9. References

9.1. Normative References

- [RFC2663] Srisuresh, P. and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations", RFC 2663, August 1999.
- [RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.
- [RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, October 2005.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.
- [RFC4380] Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", RFC 4380, February 2006.
- [RFC5454] Tsirtsis, G., Park, V., and H. Soliman, "Dual-Stack Mobile IPv4", RFC 5454, March 2009.
- [RFC5555] Soliman, H., "Mobile IPv6 Support for Dual Stack Hosts and Routers", RFC 5555, June 2009.
- [RFC5565] Wu, J., Cui, Y., Metz, C., and E. Rosen, "Softwire Mesh Framework", RFC 5565, June 2009.

9.2. Informative References

- [I-D.arkko-dual-stack-extra-lite]
Arkko, J. and L. Eggert, "Scalable Operation of Address Translators with Per-Interface Bindings", draft-arkko-dual-stack-extra-lite-03 (work in progress), October 2010.
- [I-D.arkko-ipv6-only-experience]
Arkko, J. and A. Keranen, "Experiences from an IPv6-Only Network", draft-arkko-ipv6-only-experience-00 (work in progress), July 2010.
- [I-D.arkko-townsley-coexistence]
Arkko, J. and M. Townsley, "IPv4 Run-Out and IPv4-IPv6 Co-Existence Scenarios", draft-arkko-townsley-coexistence-06 (work in progress), October 2010.
- [I-D.ietf-behave-v6v4-framework]
Baker, F., Li, X., Bao, C., and K. Yin, "Framework for IPv4/IPv6 Translation", draft-ietf-behave-v6v4-framework-10 (work in progress), August 2010.
- [RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, October 2010.
- [I-D.ietf-behave-v6v4-xlate]
Li, X., Bao, C., and F. Baker, "IP/ICMP Translation Algorithm", draft-ietf-behave-v6v4-xlate-23 (work in progress), September 2010.
- [I-D.ietf-behave-v6v4-xlate-stateful]
Bagnulo, M., Matthews, P., and I. Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", draft-ietf-behave-v6v4-xlate-stateful-12 (work in progress), July 2010.
- [I-D.ietf-behave-dns64]
Bagnulo, M., Sullivan, A., Matthews, P., and I. Beijnum, "DNS64: DNS extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", draft-ietf-behave-dns64-11 (work in progress), October 2010.
- [I-D.ietf-behave-ftp64]

Beijnum, I., "An FTP ALG for IPv6-to-IPv4 translation", draft-ietf-behave-ftp64-05 (work in progress), September 2010.

- [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-02 (work in progress), October 2010.
- [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-06 (work in progress), August 2010.
- [RFC6036] Carpenter, B. and S. Jiang, "Emerging Service Provider Scenarios for IPv6 Deployment", RFC 6036, October 2010.
- [I-D.miles-behave-l2nat]
Miles, D. and M. Townsley, "Layer2-Aware NAT", draft-miles-behave-l2nat-00 (work in progress), March 2009.
- [RFC2766] Tsirtsis, G. and P. Srisuresh, "Network Address Translation - Protocol Translation (NAT-PT)", RFC 2766, February 2000.
- [RFC2993] Hain, T., "Architectural Implications of NAT", RFC 2993, November 2000.
- [RFC3053] Durand, A., Fasano, P., Guardini, I., and D. Lento, "IPv6 Tunnel Broker", RFC 3053, January 2001.
- [RFC3056] Carpenter, B. and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds", RFC 3056, February 2001.
- [RFC3314] Wasserman, M., "Recommendations for IPv6 in Third Generation Partnership Project (3GPP) Standards", RFC 3314, September 2002.
- [RFC3574] Soinenen, J., "Transition Scenarios for 3GPP Networks", RFC 3574, August 2003.
- [RFC4029] Lind, M., Ksinant, V., Park, S., Baudot, A., and P. Savola, "Scenarios and Analysis for Introducing IPv6 into ISP Networks", RFC 4029, March 2005.

- [RFC4038] Shin, M-K., Hong, Y-G., Hagino, J., Savola, P., and E. Castro, "Application Aspects of IPv6 Transition", RFC 4038, March 2005.
- [RFC4057] Bound, J., "IPv6 Enterprise Network Scenarios", RFC 4057, June 2005.
- [RFC4215] Wiljakka, J., "Analysis on IPv6 Transition in Third Generation Partnership Project (3GPP) Networks", RFC 4215, October 2005.
- [RFC4798] De Clercq, J., Ooms, D., Prevost, S., and F. Le Faucheur, "Connecting IPv6 Islands over IPv4 MPLS Using IPv6 Provider Edge Routers (6PE)", RFC 4798, February 2007.
- [RFC4852] Bound, J., Pouffary, Y., Klynsma, S., Chown, T., and D. Green, "IPv6 Enterprise Network Analysis - IP Layer 3 Focus", RFC 4852, April 2007.
- [RFC4966] Aoun, C. and E. Davies, "Reasons to Move the Network Address Translator - Protocol Translator (NAT-PT) to Historic Status", RFC 4966, July 2007.
- [RFC5181] Shin, M-K., Han, Y-H., Kim, S-E., and D. Premec, "IPv6 Deployment Scenarios in 802.16 Networks", RFC 5181, May 2008.
- [RFC5211] Curran, J., "An Internet Transition Plan", RFC 5211, July 2008.
- [RFC5218] Thaler, D. and B. Aboba, "What Makes For a Successful Protocol?", RFC 5218, July 2008.
- [RFC5844] Wakikawa, R. and S. Gundavelli, "IPv4 Support for Proxy Mobile IPv6", RFC 5844, May 2010.
- [RFC5969] Townsley, W. and O. Troan, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification", RFC 5969, August 2010.
- [Baker.Shanghai] Baker, F., "The view from IPv6 Operations WG (and we'll talk about translation)", Presentation in the China Mobile Workshop on IPv6 Deployment in Cellular Networks, <http://ipv6ws.arkko.com/presentations/3GPP-IETF-V6OPS-Discussion.pdf>, Shanghai, China, November 2009.

[networkworld.youtube]

Marsan, C., "YouTube support of IPv6 seen in dramatic traffic spike", Network World article <http://www.networkworld.com/news/2010/020110-youtube-ipv6.html>, February 2010.

[I-D.ietf-v6ops-tunnel-security-concerns]

Krishnan, S., Thaler, D., and J. Hoagland, "Security Concerns With IP Tunneling", draft-ietf-v6ops-tunnel-security-concerns-03 (work in progress), October 2010.

[I-D.ietf-v6ops-v6-in-mobile-networks]

Koodli, R., "Mobile Networks Considerations for IPv6 Deployment", draft-ietf-v6ops-v6-in-mobile-networks-02 (work in progress), October 2010.

Appendix A. Acknowledgments

The authors would like to thank the many people who have engaged in discussions around this topic over the years. Some of the material in this document comes originally from Fred Baker's presentation in a workshop in Shanghai [Baker.Shanghai]. In addition, the authors would like to thank Mark Townsley with whom the Jari Arkko wrote an earlier document [I-D.arkko-townsley-coexistence]. Brian Carpenter submitted an in-depth review and provided significant new text. Cameron Byrne provided significant feedback on the key recommendations in this memo. The authors would also like thank Dave Thaler, Alain Durand, Randy Bush, and Dan Wing who have always provided valuable guidance in this field. Finally, the authors would like to thank Suresh Krishnan, Fredrik Garneij, Mohamed Boucadair, Remi Despres, Kurtis Lindqvist, Shawn Emery, Dan Romascanu, Tim Polk, Ralph Droms, Sean Turner, Tina Tsou, Nevil Brownlee, and Joel Jaeggli who have commented on early versions of this memo.

Authors' Addresses

Jari Arkko
Ericsson
Jorvas 02420
Finland

Email: jari.arkko@piuha.net

Fred Baker
Cisco Systems
Santa Barbara, California 93117
USA

Email: fred@cisco.com

IPv6 Operations Working Group
Internet-Draft
Intended status: Informational
Expires: April 28, 2011

S. Krishnan
Ericsson
D. Thaler
Microsoft
J. Hoagland
Symantec
October 25, 2010

Security Concerns With IP Tunneling
draft-ietf-v6ops-tunnel-security-concerns-04

Abstract

A number of security concerns with IP tunnels are documented in this memo. The intended audience of this document includes network administrators and future protocol developers. The primary intent of this document is to raise the awareness level regarding the security issues with IP tunnels as deployed today.

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 28, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

1. Introduction	3
2. Tunnels May Bypass Security	3
2.1. Network Security Bypass	3
2.2. IP Ingress and Egress Filtering Bypass	5
2.3. Source Routing After the Tunnel Client	6
3. Challenges in Inspecting and Filtering Content of Tunneled Data Packets	6
3.1. Inefficiency of Selective Network Filtering of All Tunneled Packets	7
3.2. Problems with deep packet inspection of tunneled data packets	8
4. Increased Exposure Due to Tunneling	9
4.1. NAT Holes Increase Attack Surface	9
4.2. Exposure of a NAT Hole	11
4.3. Public Tunnels Widen Holes in Restricted NATs	12
5. Tunnel Address Concerns	12
5.1. Feasibility of Guessing Tunnel Addresses	12
5.2. Profiling Targets Based on Tunnel Address	13
6. Additional Security Concerns	14
6.1. Attacks Facilitated By Changing Tunnel Server Setting	14
7. Mechanisms to secure the use of tunnels	16
8. Acknowledgments	17
9. Security Considerations	17
10. IANA Considerations	17
11. Informative References	17
Authors' Addresses	19

1. Introduction

With NAT devices becoming increasingly more prevalent, there have recently been many tunneling protocols developed that go through NAT devices or firewalls by tunneling over UDP or TCP. For example, Teredo [RFC4380], L2TPv2 [RFC2661], and L2TPv3 [RFC3931] all tunnel IP packets over UDP. Similarly, many SSL VPN solutions that tunnel IP packets over HTTP (and hence over TCP) are deployed today.

This document discusses security concerns with tunneling IP packets, and includes recommendations where relevant.

The primary intent of this document is to help improve security deployments using tunnel protocols. In addition, the document aims to provide information that can be used in any new or updated tunnel protocol specification. The intended audience of this document includes network administrators and future protocol developers.

2. Tunnels May Bypass Security

2.1. Network Security Bypass

2.1.1. Problem

Tunneled IP traffic may not receive the intended level of inspection or policy application by network-based security devices unless such devices are specifically tunnel-aware. This reduces defense in depth and may cause security gaps. This applies to all network-located devices and to any end-host based firewalls whose existing hooking mechanism(s) would not show them the IP packet stream after the tunnel client does decapsulation or before it does encapsulation.

2.1.2. Discussion

Evasion by tunneling is often a problem for network-based security devices such as network firewalls, intrusion detection and prevention systems, and router controls. To provide such functionality in the presence of tunnels, the developer of such devices must add support for parsing each new protocol. There is typically a significant lag between when the security developer recognizes that a tunnel will be used (or will be remotely usable) to a significant degree and when the parsing can be implemented in a product update, the update tested and released, and customers begin using the update. Late changes in the protocol specification or in the way it is implemented can cause additional delays. This becomes a significant security concern when a delay in applied coverage is occurring frequently. One way to cut down on this lag is for security developers to follow the progress of

new IETF protocols but this will still not account for any new proprietary protocols.

For example, for L2TP or Teredo, an unaware network security device would inspect or regulate the outer IP and the IP-based UDP layer as normal, but it would not recognize that there is an additional IP layer contained inside the UDP payload to which it needs to apply the same controls as it would to a native packet. (Of course, if the device discards the packet due to something in the IP or UDP header, such as referring to an unknown protocol, the embedded packet is no longer a concern.) In addition, if the tunnel does encryption, the network-based security device may not be able to do much, just as if IPsec end-to-end encryption were used without tunneling.

Network security controls being not applied must be a concern to those that set them up, since those controls are supposed to provide an additional layer of defense against external attackers. If network controls are being bypassed due to the use of tunneling, the strength of the defense (i.e. the number of layers of defense) is reduced. Since security administrators may have a significantly reduced level of confidence without this layer, this becomes a concern to them.

One implication of the security control bypass is that defense in depth has been reduced, perhaps down to zero unless a local firewall is in use as recommended in [RFC4380]. However, even if there are host-based security controls that recognize tunnels, security administrators may not have configured them with full security control parity, even if all controls that were maintained by the network are available on the host. Thus there may be gaps in desired coverage.

Compounding this is that, unlike what would be the case for native IP, some network administrators will not even be aware that their hosts are globally reachable, if the tunnel provides connectivity to/from the Internet; for example, they may not be expecting this for hosts behind a stateful firewall. In addition, Section 3.2 discusses how it may not be efficient to find all tunneled traffic for network devices to examine.

2.1.3. Recommendations

Security administrators who do not consider tunneling an acceptable risk should disable tunnel functionality either on the end-nodes (hosts) or on the network nodes at the perimeter of their network. However, there may be an awareness gap. Thus, due to the possible negative security consequences, tunneling functionality should be easy to disable on the host and through a central management facility

if one is provided.

To minimize security exposure due to tunnels, we recommend that a tunnel be an interface of last resort, independent of IP version. Specifically, we suggest that when both native and tunneled access to a remote host is available, that the native access be used in preference to tunneled access except when the tunnel endpoint is known to not bypass security (e.g., an IPsec tunnel to a gateway provided by the security administrator of the network). This should also promote greater efficiency and reliability.

Note that although Rule 7 of [RFC3484] section 6 will prefer native connectivity over tunnels, this rule is only a tie-breaker when a choice is not made by earlier rules; hence tunneling mechanisms that are tied to a particular range of IP address space will be decided based on the prefix precedence. For example, using the prefix policy mechanism of [RFC3484] section 2.1, Teredo might have a precedence of 5 so that native IPv4 is preferred over Teredo.

2.2. IP Ingress and Egress Filtering Bypass

2.2.1. Problem

IP addresses inside tunnels are not subject to ingress and egress filtering in the network they tunnel over, unless extraordinary measures are taken. Only the tunnel endpoints can do such filtering.

2.2.2. Discussion

Ingress filtering (sanity-checking incoming destination addresses) and egress filtering (sanity-checking outgoing source addresses) are done to mitigate attacks and to make it easier to identify the source of a packet and are considered to be a good practice. e.g. ingress filtering at the network perimeter should not allow packets with a source address that belongs to the network to enter the network from the outside the network. This function is most naturally (and in the general case, by requirement) done at network boundaries. Tunneled IP traffic bypassing this network control is a specific case of Section 2.1, but is illustrative.

2.2.3. Recommendations

Tunnel servers can apply ingress and egress controls to tunneled IP addresses passing through them to and from tunnel clients.

Tunnel clients could make an effort to conduct ingress and egress filtering.

Implementations of protocols that embed an IPv4 address in a tunneled IPv6 address directly between peers should perform filtering based on checking the correspondence.

Implementations of protocols that accept tunneled packets directly from a server, relay or protocol peer do filtering the same way as it would be done on a native link with traffic from a router.

Some protocols such as 6to4 [RFC3056], Teredo, and ISATAP [RFC5214] allow both other hosts and a router over a common tunnel. To perform host-based filtering with such protocols a host would need to know the outer IP address of each router from which it could receive traffic, so that packets from hosts beyond the router will be accepted even though the source address would not embed the router's IP address. Router addresses might be learned via Secure Neighbor Discovery (SEND) [RFC3971] or some other mechanism (e.g., [RFC5214] section 8.3.2).

2.3. Source Routing After the Tunnel Client

2.3.1. Problem

If the encapsulated IP packet specifies source routing beyond the recipient tunnel client, the host may forward the IP packet to the specified next hop. This may be unexpected and contrary to administrator wishes and may have bypassed network-based source routing controls.

2.3.2. Discussion

A detailed discussion of issues related to source routing can be found in [RFC5095] and [SECA-IP].

2.3.3. Recommendations

Tunnel clients should by default discard tunneled IP packets that specify additional routing, as recommended in [RFC5095] and [SECA-IP], though they may also allow the user to configure what source routing types are allowed. All pre-existing source routing controls should be upgraded to apply these controls to tunneled IP packets as well.

3. Challenges in Inspecting and Filtering Content of Tunneled Data Packets

3.1. Inefficiency of Selective Network Filtering of All Tunneled Packets

3.1.1. Problem

There is no mechanism to both efficiently and immediately filter all tunneled packets (other than the obviously faulty method of filtering all packets). This limits the ability to prevent tunnel use on a network.

3.1.2. Discussion

Given concerns about tunnel security or a network's lack of preparedness for tunnels, a network administrator may wish to simply block all use of tunnels that bypass security policies. He or she may wish to do so using network controls; this could be either due to not having the capability to disable tunneling on all hosts attached to the network or due to wanting an extra layer of prevention.

One simple method of doing this easily for many tunnel protocols is to block outbound packets to the UDP or TCP port used (e.g., destination UDP port is 3544 for Teredo, UDP port 1701 for L2TP, etc.). This prevents a tunnel client from establishing a new tunnel. However, existing tunnels will not necessarily be affected if the blocked port is used only for initial setup. In addition, if the blocking is applied on the outside of the client's NAT device, the NAT device will retain the port mapping for the client. In some cases, however, blocking all traffic to a given outbound port (e.g., port 80) may interfere with non-tunneled traffic so this should be used with caution.

Another simple alternative, if the tunnel server addresses are well-known, is to filter out all traffic to/from such addresses.

The other approach is to find all packets to block in the same way as would be done for inspecting all packets (Section 3.2). However; this faces the difficulties in terms of efficiency of filtering, as is discussed there.

3.1.3. Recommendations

Developers of protocols that tunnel over UDP or TCP (including HTTP) to reach the Internet should disable their protocols in networks that wish to enforce security policies on the user traffic. (Windows, for example, disables Teredo by default if it detects that it is within an enterprise network that contains a Windows domain controller.)

Administrators of such networks may wish to filter all tunneled

traffic at the boundaries of their networks. It is sufficient to filter out the tunneled connection requests (if they can be identified) to stop further tunneled traffic. The easiest mechanism for this would be to filter out outgoing traffic sent to the destination port defined by the tunneling protocol, and incoming traffic with that source port. Similarly, in certain cases, it is also possible to use the IP protocol field to identify and filter tunneled packets. e.g. 6to4 [RFC3056] is a tunneling mechanism that uses the IPv4 packets to carry encapsulated IPv6 packets, and can be identified by the IPv4 protocol type 41.

3.2. Problems with deep packet inspection of tunneled data packets

3.2.1. Problem

There is no efficient mechanism for network-based devices, which are not the tunnel endpoint, to inspect the contents of all tunneled data packets, the way they can for native packets. This makes it difficult to apply the same controls as they do to native IP.

3.2.2. Discussion

Some tunnel protocols are easy to identify, such as if all data packets are encapsulated using a well-known UDP or TCP port that is unique to the protocol.

Other protocols, however, either use dynamic ports for data traffic, or else share ports with other protocols (e.g., tunnels over HTTP).

The implication of this is that network-based devices that wish to passively inspect (and perhaps selectively apply policy to) all encapsulated traffic must inspect all TCP or UDP packets (or at least all packets not part of a session that is known not to be a tunnel). This is imperfect since a heuristic must then be applied to determine if a packet is indeed part of a tunnel. This may be too slow to make use of in practice, especially if it means that all TCP or UDP packets must be taken off of the device's "fast path".

One heuristic that can be used on packets to determine if they are tunnel-related or not is as follows. For each known tunnel protocol, attempt parsing the packet as if it were a packet of that protocol, destined to the local host (i.e., where the local host has the destination address in the inner IP header, if any). If all syntax checks pass, up to and including the inner IP header (if the tunnel doesn't use encryption), then treat the packet as if it is a tunneled packet of that protocol.

It is possible that non-tunnel packets will match as tunneled using

this heuristic, but tunneled packets (of the known types of tunnels) should not escape inspection, absent implementation bugs.

For some protocols, it may be possible to monitor setup exchanges to know to expect that data will be exchanged on certain ports later. (Note that this does not necessarily apply to Teredo, for example, since communicating with another Teredo client behind a cone NAT [RFC5389] device does not require such signaling. In such cases this control will not work. However, deprecation of the cone bit as discussed in [RFC5991] means this technique may indeed work with updated Teredo implementations.)

3.2.3. Recommendations

As illustrated above, it should be clear that inspecting the contents of tunneled data packets is highly complex and often impractical. For this reason, if a network wishes to monitor IP traffic, tunneling across, as opposed to tunneling to, the security boundary is not recommended. For example, to provide an IPv6 transition solution, the network should provide native IPv6 connectivity or a tunnel solution (e.g., ISATAP or 6over4) that encapsulates data packets between hosts and a router within the network.

4. Increased Exposure Due to Tunneling

4.1. NAT Holes Increase Attack Surface

4.1.1. Problem

If the tunnel allows inbound access from the public Internet, the opening created in a NAT device due to a tunnel client increases its Internet attack surface area. If vulnerabilities are present, this increased exposure can be used by attackers and their programs.

If the tunnel allows inbound access only from a private network (e.g., a remote network to which one has VPN'ed), the opening created in the NAT device still increases its attack surface area, although not as much as in the public Internet case.

4.1.2. Discussion

When a tunnel is active, a mapped port is maintained on the NAT device through which remote hosts can send packets and perhaps establish connections. The following sequence is intended to sketch out the processing on the tunnel client host that can be reached through this mapped port; the actual processing for a given host may be somewhat different.

1. Link-layer protocol processing
2. (Outer) IP host firewall processing
3. (Outer) IP processing by stack
4. UDP/TCP processing by stack
5. Tunnel client processing
6. (Inner) IP host firewall processing
7. (Inner) IP processing by stack
8. Various upper layer processing may follow

The inner firewall (and other security) processing may or may not be present, but if it is, some of the inner IP processing may be filtered. (For example, [RFC4380] section 7.1 recommends that an IPv6 host firewall be used on all Teredo clients.)

(By the virtue of the tunnel being active, we can infer that the inner host firewall is unlikely to do any filtering based on the outer IP.) Any of this processing may expose vulnerabilities an attacker can exploit; similarly these may expose information to an attacker. Thus, even if firewall filtering is in place (as is prudent) and filters all incoming packets, the exposed area is larger than if a native IP Internet connection were in place, due to the processing that takes place before the inner IP is reached (specifically, the UDP/TCP processing, the tunnel client processing, and additional IP processing, especially if one is IPv4 and the other is IPv6).

One possibility is that a layer 3 targeted worm makes use of a vulnerability in the exposed processing. The main benefit tunneling provides to worms is enabling L3 reachability to the end host. Even a thoroughly firewalled host could be subject to a worm that spreads with a single UDP packet if the right remote code vulnerability is present.

4.1.3. Recommendations

This problem seems inherent in tunneling being active on a host, so the solution seems to be to minimize tunneling use.

For example, it can be active only when it is really needed and only for as long as needed. So, the tunnel interface can be initially not configured and only used when it is entirely the last resort. The

interface should then be deactivated (ideally, automatically) again as soon as possible. Note however that the hole will remain in the NAT device for some amount of time after this, so some processing of incoming packets is inevitable unless the client's native IP address behind the NAT device is changed.

4.2. Exposure of a NAT Hole

4.2.1. Problem

Attackers are more likely to know about a tunnel client's NAT hole than a typical hole in the NAT device. If they know about the hole, they could try to use it.

4.2.2. Discussion

There are at least three reasons why an attacker may be more likely to learn of the tunnel client's exposed port than a typical NAT exposed port:

1. The NAT mapping for a tunnel is typically held open for a significant period of time, and kept stable. This increases the chance of it being discovered.
2. In some protocols (e.g., Teredo), the external IP address and port are contained in the client's address that is used end-to-end and possibly even advertised in a name resolution system. While the tunnel protocol itself might only distribute this address in IP headers, peers, routers, and other on-path nodes still see the client's IP address. Although this point does not apply directly to protocols (e.g., L2TP) that do not construct the inner IP address based on the outer IP address, the inner IP address is still known to peers, routers, etc. and can still be reached by attackers without knowing the external IP address or port.
3. The tunnel protocol often contains more messages that are exchanged and with more parties (e.g., due to a longer path length) than without using the tunnel, offering more chance for visibility into the port and address in use.

4.2.3. Recommendations

The recommendations from Section 4.1 seem to apply here as well: minimize tunnel use.

4.3. Public Tunnels Widen Holes in Restricted NATs

4.3.1. Problem

Tunnels that allow inbound connectivity from the Internet (e.g., Teredo, tunnel brokers, etc) essentially disable the filtering behavior of the NAT for all tunnel client ports. This eliminates NAT devices filtering for such ports and may eliminate the need for an attacker to spoof an address.

4.3.2. Discussion

NATs that implement Address-Dependent or Address and Port-Dependent Filtering [RFC4787] limit the source of incoming packets to just those that are a previous destination. This poses a problem for tunnels that intend to allow inbound connectivity from the Internet.

Various protocols (e.g., Teredo, STUN [RFC5389], etc.) provide a facility for peers, upon request, to become a previous destination. This works by sending a "bubble" packet via a server, which is passed to the client, and then sent by the client (through the NAT) to the originator.

This removes any NAT-based barrier to attackers sending packets in through the client's service port. In particular, an attacker would no longer need to either be an actual previous destination or to forge its addresses as a previous destination. When forging, the attacker would have had to learn of a previous destination and then would face more challenges in seeing any returned traffic.

4.3.3. Recommendations

If the tunnel can provide connectivity to the Internet, the tunnel client should run a host firewall on the tunnel interface. Also, minimizing public tunnel use (see Section 4.1.3) would lower the attack opportunity related to this exposure.

5. Tunnel Address Concerns

5.1. Feasibility of Guessing Tunnel Addresses

5.1.1. Problem

For some types of tunneling protocols, it may be feasible to guess IP addresses assigned to tunnels, either when looking for a specific client or when looking for an arbitrary client. This is in contrast to native IPv6 addresses in general, but is no worse than for native

IPv4 addresses today.

For example, some protocols (e.g., 6to4 and Teredo) use well-defined address ranges. As another example, using well-known public servers for Teredo or tunnel brokers also implies using a well known address range.

5.2. Profiling Targets Based on Tunnel Address

5.2.1. Problem

An attacker encountering an address associated with a particular tunneling protocol or well-known tunnel server has the opportunity to infer certain relevant pieces of information that can be used to profile the host before sending any packets. This can reduce the attacker's footprint and increase the attacker's efficiency.

5.2.2. Discussion

The tunnel address reveals some information about the nature of the client.

- o That a host has a tunnel address associated with a given protocol means that the client is running on some platform for which there exists a tunnel client implementation of that protocol. In addition, if some platforms have that protocol installed by default and where the host's default rules for using it make it susceptible to being in use, then it is more likely to be running on such a platform than on one where it is not used by default. For example, as of this writing, seeing a Teredo address suggests that the host it is on is probably running Windows.
- o Similarly, the use of an address associated with a particular tunnel server also suggests some information. Tunnel client software is often deployed, installed, and/or configured using some degree of automation. It seems likely that the majority of the time the tunnel server that results from the initial configuration will go unchanged from the initial setting. Moreover, the server that is configured for use may be associated with a particular means of installation, which often suggests the platform. For example, if the server field in a Teredo address is one of the IPv4 addressees to which `teredo.ipv6.microsoft.com` resolves, it suggests that the host is running Windows.
- o The external IPv4 address of a NAT device can of course be readily associated with a particular organization or at least an ISP, and hence putting this address in an IPv6 address reveals this information. However, this is no different than using a native IP

address, and hence is not new with tunneling.

- o It is also possible that external client port numbers may be more often associated with particular client software or the platform on which it is running. The usefulness of this for platform determination is, however, reduced by the different NAT port number assignment behaviors. In addition, the same observations would apply to use of UDP or TCP over native IP as well, and hence this is not new with tunneling.

The platform, tunnel client software, or organization information can be used by an attacker to target attacks more carefully. For example, an attacker may decide to attack an address only if it is likely to be associated with a particular platform or tunnel client software with a known vulnerability. (This is similar to the ability to guess some platforms based on the OUI in the EUI-64 portion of an IPv6 address generated from a MAC address, since some platforms are commonly used with interface cards from particular vendors.)

5.2.3. Recommendations

If installation programs randomized the server setting, that would reduce the extent to which they can be profiled. Similarly, administrators can choose to change the default setting to reduce the degree to which they can be profiled ahead of time.

Randomizing the tunnel client port in use would mitigate any profiling that can be done based on the external port, especially if multiple different tunnel clients did this. Further discussion on randomizing ports can be found at [TSV-PORT].

It is recommended that tunnel protocols minimize the propagation of knowledge about whether the NAT is a cone NAT.

6. Additional Security Concerns

6.1. Attacks Facilitated By Changing Tunnel Server Setting

6.1.1. Problem

If an attacker could either change a tunnel client's server setting or change the IP addresses to which a configured host name resolves (e.g., by intercepting DNS queries) AND the tunnel is not authenticated, it would let the attacker become a man in the middle. This would allow them to at least monitor peer communication and at worst to impersonate the remote peer.

6.1.2. Discussion

A client's server has good visibility into the client's communication with IP peers. If the server were switched to one that records this information and makes it available to third parties (e.g., advertisers, competitors, spouses, etc.) then sensitive information would be disclosed, especially if the client's host prefers the tunnel over native IP. Assuming the server provides good service, the user would not have reason to suspect the change.

Full interception of IP traffic could also be arranged (including pharming) which would allow any number of deception or monitoring attacks including phishing. We illustrate this with an example phishing attack scenario.

It is often assumed that the tunnel server is a trusted entity. It may be possible for malware or a malicious user to quietly change the client's tunnel server setting and have the user be unaware their trust has been misplaced for an indefinite period of time. However, malware or a malicious user can do much worse than this, so this is not a significant concern. Hence it is only important that an attacker on the network cannot change the client's server setting.

1. A phisher sets up a malicious tunnel server (or tampers with a legitimate one). This server, for the most part, provides correct service.
2. An attacker, by some means, switches the host's tunnel server setting, or spoofs a DNS reply, to point to the above server. If neither DNS nor the tunnel setup is secured (i.e., if the client does not authenticate the information), then the attacker's tunnel server is seen as legitimate.
3. A user on the victim host types their bank's URL into his/her browser.
4. The bank's hostname resolves to one or more IP addresses and the tunnel is selected for socket connection for whatever reason (e.g., the tunnel provides IPv6 connectivity and the bank has an IPv6 address).
5. The tunnel client uses the server for help in connecting to the bank's IP address. Some tunneling protocols use a separate channel for signaling vs data, but this still allows the server to place itself in the data path by an appropriate signal to the client. For example, in Teredo, the client sends a ping request through a server which is expected to come back through a data relay, and a malicious server can simply send it back itself to

indicate that is a data relay for the communication.

6. The rest works pretty much like any normal phishing transaction, except that the attacker acts as a tunnel server (or data relay, for protocols such as Teredo) and a host with the bank's IP address.

This pharming type attack is not unique to tunneling. Switching DNS server settings to a malicious DNS server or DNS cache poisoning in a recursive DNS resolver could have a similar effect.

6.1.3. Recommendations

In general, anti-phishing and anti-fraud provisions should help with aspects of this, as well as software that specifically monitors for tunnel server changes.

Perhaps the best way to mitigate tunnel-specific attacks is to have the client either authenticate the tunnel server, or at least the means by which the tunnel server's IP address is determined. For example, SSL VPNs use https URLs and hence authenticate the server as being the expected one. Another mechanism, when IPv6 Router Advertisements are sent over the tunnel is to use SEcure Neighbor Discovery (SEND) [RFC3971] to verify that the client trusts the server.

On the host, it should require an appropriate level of privilege in order to change the tunnel server setting (as well as other non-tunnel-specific settings such as the DNS server setting, etc.). Making it easy to see the current tunnel server setting (e.g., not requiring privilege for this) should help detection of changes.

The scope of the attack can also be reduced by limiting tunneling use in general but especially in preferring native IPv4 to tunneled IPv6; this is because it is reasonable to expect that banks and similar web sites will continue to be accessible over IPv4 for as long as a significant fraction of their customers are still IPv4-only. Please refer to Section 3 of [TUNNEL-LOOPS] for a detailed description and mitigation measures for a class of attacks based on IPv6 automatic tunnels.

7. Mechanisms to secure the use of tunnels

This document described several security issues with tunnels. This does not mean that tunnels need to be avoided at any cost. On the contrary, tunnels can be very useful if deployed, operated and used properly. The threats against IP tunnels are documented here. If

the threats can be mitigated, network administrators can efficiently and securely use tunnels in their network. Several measures can be taken in order to secure the operation of IPv6 tunnels:

- o Operating on-premise tunnel servers/relays so that the tunneled traffic does not cross border routers.
- o Setting up internal routing to steer traffic to these servers/relays
- o Setting up of firewalls [RFC2979] to allow known and controllable tunneling mechanisms and disallow unknown tunnels.

8. Acknowledgments

The authors would like to thank Remi Denis-Courmont, Fred Templin, Jordi Palet Martinez, James Woodyatt, Christian Huitema, Brian Carpenter, Nathan Ward, Kurt Zeilenga, Joel Halpern, Erik Kline, Alfred Hoenes and Fernando Gont for reviewing earlier versions of the document and providing comments to make this document better.

9. Security Considerations

This entire document discusses security concerns with tunnels.

10. IANA Considerations

This document does not require any IANA action.

11. Informative References

- [RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.
- [RFC2529] Carpenter, B. and C. Jung, "Transmission of IPv6 over IPv4 Domains without Explicit Tunnels", RFC 2529, March 1999.
- [RFC2661] Townsley, W., Valencia, A., Rubens, A., Pall, G., Zorn, G., and B. Palter, "Layer Two Tunneling Protocol "L2TP"", RFC 2661, August 1999.
- [RFC2979] Freed, N., "Behavior of and Requirements for Internet Firewalls", RFC 2979, October 2000.

- [RFC3056] Carpenter, B. and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds", RFC 3056, February 2001.
- [RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.
- [RFC3931] Lau, J., Townsley, M., and I. Goyret, "Layer Two Tunneling Protocol - Version 3 (L2TPv3)", RFC 3931, March 2005.
- [RFC3971] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.
- [RFC4380] Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", RFC 4380, February 2006.
- [RFC4787] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.
- [RFC5095] Abley, J., Savola, P., and G. Neville-Neil, "Deprecation of Type 0 Routing Headers in IPv6", RFC 5095, December 2007.
- [RFC5214] Templin, F., Gleeson, T., and D. Thaler, "Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)", RFC 5214, March 2008.
- [RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008.
- [RFC5991] Thaler, D., Krishnan, S., and J. Hoagland, "Teredo Security Updates", RFC 5991, September 2010.
- [SECA-IP] Gont, F., "Security Assessment of the Internet Protocol version 4", draft-ietf-opsec-ip-security-03 (work in progress), April 2010.
- [TSV-PORT] Larsen, M. and F. Gont, "Transport Protocol Port Randomization Recommendations", draft-ietf-tsvwg-port-randomization-09 (work in progress), August 2010.
- [TUNNEL-LOOPS] Nakibly, G. and F. Templin, "Routing Loop Attack using IPv6 Automatic Tunnels: Problem Statement and Proposed

Mitigations", draft-ietf-v6ops-tunnel-loops-00 (work in progress), September 2010.

Authors' Addresses

Suresh Krishnan
Ericsson
8400 Decarie Blvd.
Town of Mount Royal, QC
Canada

Phone: +1 514 345 7900 x42871
Email: suresh.krishnan@ericsson.com

Dave Thaler
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
USA

Phone: +1 425 703 8835
Email: dthaler@microsoft.com

James Hoagland
Symantec Corporation
350 Ellis St.
Mountain View, CA 94043
US

Email: Jim_Hoagland@symantec.com
URI: <http://symantec.com/>

Individual Submission
Internet-Draft
Intended status: Informational
Expires: August 14, 2011

J. Korhonen, Ed.
Nokia Siemens Networks
J. Soininen
Renesas Mobile
B. Patil
T. Savolainen
G. Bajko
Nokia
K. Iisakkila
Renesas Mobile
February 10, 2011

IPv6 in 3GPP Evolved Packet System
draft-korhonen-v6ops-3gpp-eps-06

Abstract

Internet connectivity and use of data services in 3GPP based mobile networks has increased rapidly as a result of smart phones, broadband service via HSPA and HSPA+ networks, competitive service offerings by operators and a large number of applications. Operators who have deployed networks based on 3GPP architectures are facing IPv4 address shortages. With the impending exhaustion of available IPv4 addresses from the registries there is an increased emphasis for operators to migrate to IPv6. This document describes the support for IPv6 in 3GPP network architectures.

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 August 14, 2011.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the

document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	4
2. 3GPP Terminology and Concepts	5
2.1. Terminology	5
2.2. The concept of APN	8
3. IP over 3GPP GPRS	9
3.1. Introduction to 3GPP GPRS	9
3.2. PDP Context	10
4. IP over 3GPP EPS	11
4.1. Introduction to 3GPP EPS	11
4.2. PDN Connection	12
4.3. EPS bearer model	13
5. Address Management	13
5.1. IPv4 Address Configuration	14
5.2. IPv6 Address Configuration	14
5.3. Prefix Delegation	15
6. 3GPP Dual-Stack Approach to IPv6	15
6.1. 3GPP Networks Prior to Release-8	15
6.2. 3GPP Release-8 and -9 Networks	16
6.3. PDN Connection Establishment Process	17
6.4. Mobility of 3GPP IPv4v6 Type of Bearers	20
7. Dual-Stack Approach to IPv6 Transition in 3GPP Networks	20
8. Deployment issues	21
8.1. Overlapping IPv4 Addresses	21
8.2. IPv6 for transport	22
8.3. Operational Aspects of Running Dual-Stack Networks	23
8.4. Operational Aspects of Running a Network with IPv6 Only Bearers	23
8.5. Restricting Outbound IPv6 Roaming	24
8.6. Inter-rat Handovers and IP Versions	25
8.7. Provisioning of IPv6 Subscribers and Various Combinations During Initial Network Attachment	26
9. IANA Considerations	27
10. Security Considerations	27
11. Summary and Conclusion	27
12. Acknowledgements	28
13. Informative References	28
Authors' Addresses	30

1. Introduction

IPv6 has been specified in the 3rd Generation Partnership Project (3GPP) standards since the early architectures developed for R99 General Packet Radio Service (GPRS). However, the support for IPv6 in commercially deployed networks by the end of 2010 is nearly non-existent. There are many factors that can be attributed to the lack of IPv6 deployment in 3GPP networks. The most relevant one is essentially the same as the reason for IPv6 not being deployed by other networks as well, i.e. the lack of business and commercial incentives for deployment. 3GPP network architectures have also evolved since 1999 (since R99). The most recent version of the 3GPP architecture, the Evolved Packet System (EPS), which is commonly referred to as SAE, LTE or Release-8, is a packet centric architecture. The number of subscribers and devices that are using the 3GPP networks for Internet connectivity and data services has also increased significantly. With the subscriber growth numbers projected to increase even further and the IPv4 addresses depletion problem looming in the near term, 3GPP operators and vendors have started the process of identifying the scenarios and solutions needed to transition to IPv6.

This document describes the establishment of IP connectivity in 3GPP network architectures, specifically in the context of IP bearers for 3GPP GPRS and for 3GPP EPS. It provides an overview of how IPv6 is supported as per the current set of 3GPP specifications. Some of the issues and concerns with respect to deployment and shortage of private IPv4 addresses within a single network domain are also discussed.

The IETF has specified a set of tools and mechanisms that can be utilized for transitioning to IPv6. In addition to operating dual-stack networks during the transition from IPv4 to IPv6 phase, the two alternative categories for the transition are encapsulation and translation. Most of the mechanisms available in the toolbox can be categorized into either translation or encapsulation approaches. The IETF continues to specify additional solutions for enabling the transition based on the deployment scenarios and operator/ISP requirements. There is no single approach for transition to IPv6 that can meet the needs for all deployments and models. The 3GPP scenarios for transition, described in [3GPP.23.975], can be addressed using transition mechanisms that are already available in the toolbox. The objective of transition to IPv6 in 3GPP networks is to ensure that:

1. Legacy devices and hosts which have an IPv4 only stack will continue to be provided with IP connectivity to the Internet and services,

2. Devices which are dual-stack can access the Internet either via IPv6 or IPv4. The choice of using IPv6 or IPv4 depends on the capability of:
 - A. the application on the host,
 - B. the support for IPv4 and IPv6 bearers by the network and/or,
 - C. the capability of the server(s) and other end points.

3GPP networks are capable of providing a host with IPv4 and IPv6 connectivity today, albeit in many cases with upgrades to network elements such as the SGSN and GGSN.

2. 3GPP Terminology and Concepts

2.1. Terminology

Access Point Name

Access Point Name (APN) is a fully qualified domain name and resolves to a specific gateway in an operators network. The APNs are piggybacked on the administration of the DNS namespace.

Packet Data Protocol Context

A Packet Data Protocol (PDP) Context is the equivalent of a virtual connection between the host and a gateway.

General Packet Radio Service

General Packet Radio Service (GPRS) is a packet oriented mobile data service available to users of the 2G and 3G cellular communication systems Global System for Mobile communications (GSM), and specified by 3GPP.

Packet Data Network

Packet Data Network (PDN) is a packet based network that either belongs to the operator or is an external network such as Internet and corporate intranet. The user eventually accesses services in one or more PDNs. The operator's packet domain network are separated from packet data networks either by GGSNs or PDN Gateways (PDN-GW).

Gateway GPRS Support Node

Gateway GPRS Support Node (GGSN) is a gateway function in GPRS, which provides connectivity to Internet or other PDNs. The host attaches to a GGSN identified by an APN assigned to it by an operator. The GGSN also serves as the topological anchor for addresses/prefixes assigned to the mobile host.

Packet Data Network Gateway

Packet Data Network Gateway (PDN-GW) is a gateway function in Evolved Packet System (EPS), which provides connectivity to Internet or other PDNs. The host attaches to a PDN-GW identified by an APN assigned to it by an operator. The PDN-GW also serves as the topological anchor for addresses/prefixes assigned to the mobile host.

Serving Gateway

Serving Gateway (SGW) is a gateway function in EPS, which terminates the interface towards E-UTRAN. The SGW is the Mobility Anchor point for layer-2 mobility (inter-eNodeB handovers). For each User Equipment connected with the EPS, at any given point of time, there is only one SGW. The SGW is essentially the user plane part of the GPRS' SGSN forwarding packets between a PDN-GW.

Serving Gateway Support Node

Serving Gateway Support Node (SGSN) is a network element that is located between the radio access network (RAN) and the gateway (GGSN). A per mobile host point to point (p2p) tunnel between the GGSN and SGSN transports the packets between the mobile host and the gateway.

GPRS tunnelling protocol

GPRS Tunnelling Protocol (GTP) [3GPP.29.060] [3GPP.29.274] is a tunnelling protocol defined by 3GPP. It is a network based mobility protocol and similar to Proxy Mobile IPv6 (PMIPv6) [RFC5213]. However, GTP also provides functionality beyond mobility such as inband signaling related to Quality of Service (QoS) and charging among others.

Evolved Packet System

Evolved Packet System (EPS) is an evolution of the 3GPP GPRS system characterized by higher-data-rate, lower-latency, packet-optimized system that supports multiple Radio Access Technologies

(RAT). The EPS comprises the Evolved Packet Core (EPC) together with the evolved radio access network (E-UTRA and E-UTRAN).

Mobility Management Entity

Mobility Management Entity (MME) is a network element that is responsible for control plane functionalities, including authentication, authorization, bearer management, layer-2 mobility, etc. The MME is essentially the control plane part of the GPRS' SGSN and not located on the user plane data path, i.e. user plane traffic bypasses the MME.

UMTS Terrestrial Radio Access Network

UMTS Terrestrial Radio Access Network (UTRAN) is communications network, commonly referred to as 3G, and consists of NodeBs (3G base station) and Radio Network Controllers (RNC) which make up the UMTS radio access network. The UTRAN allows connectivity between the mobile host/device and the core network. UTRAN comprises of WCDMA, HSPA and HSPA+ radio technologies.

Wideband Code Division Multiple Access

The Wideband Code Division Multiple Access (WCDMA) is the radio interface used in UMTS networks.

High Speed Packet Access

The High Speed Packet Access (HSPA) and the Evolved High Speed Packet Access (HSPA+) are enhanced versions of the WCDMA and UTRAN, thus providing more data throughput and lower latencies.

Evolved UTRAN

Evolved UTRAN (E-UTRAN) is communications network, sometimes referred to as 4G, and consists of eNodeBs (4G base station) which make up the E-UTRAN radio access network. The E-UTRAN allows connectivity between the mobile host/device and the core network.

eNodeB

The eNodeB is a base station entity that supports the Long Term Evolution (LTE) air interface.

GSM EDGE Radio Access Network

GSM EDGE Radio Access Network (GERAN) is communications network, commonly referred to as 2G or 2.5G, and consists of base stations

and Base Station Controllers (BSC) which make up the GSM EDGE radio access network. The GERAN allows connectivity between the mobile host/device and the core network.

UE, MS, MN and Mobile

The terms UE (User Equipment), MS (Mobile Station), MN (Mobile Node) and, mobile refer to the devices which are hosts with ability to obtain Internet connectivity via a 3GPP network. The terms UE, MS, MN and devices are used interchangeably within this document.

PCC

The Policy and Charging Control (PCC) framework is used for QoS policy and charging control. It is optional for 3GPP EPS but needed if dynamic policy and charging control by means of PCC rules based on user and services are desired.

HLR

The Home Location Register (HLR) is a pre-Release-5 database (the reality regarding releases is different, though) for a given subscriber. It is the entity containing the subscription-related information to support the network entities actually handling calls/sessions.

HSS

The Home Subscriber Server (HSS) is a database for a given subscriber and got introduced in 3GPP Release-5. It is the entity containing the subscription-related information to support the network entities actually handling calls/sessions.

2.2. The concept of APN

The Access Point Name (APN) essentially refers to a gateway in the 3GPP network. The 'complete' APN is expressed in a form of a Fully Qualified Domain Name (FQDN) and also piggybacked on the administration of the DNS namespace, thus effectively allowing the discovery of gateways using the DNS. Mobile hosts/devices can choose to attach to a specific gateway in the packet core. The gateway provides connectivity to the Packet Data Network (PDN) such as the Internet. An operator may also include gateways which do not provide Internet connectivity, rather a connectivity to closed network providing a set of operator's own services. A mobile host/device can be attached to one or more gateways simultaneously. The gateway in a 3GPP network is the GGSN or PDN-GW. Figure 1 below illustrates the

APN-based network connectivity concept.

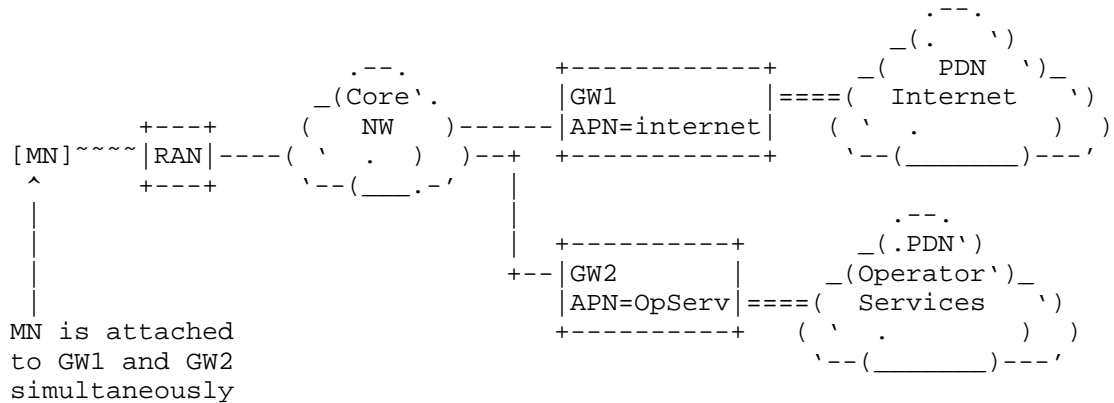


Figure 1: Mobile host/device attached to multiple APNs simultaneously

3. IP over 3GPP GPRS

3.1. Introduction to 3GPP GPRS

A simplified 2G/3G GPRS architecture is illustrated in Figure 2. This architecture basically covers the GPRS core network since R99 to Release-7, and radio access technologies such as GSM (2G), EDGE (2G, often referred as 2.5G), WCDMA (3G) and HSPA(+) (3G, often referred as 3.5G). The architecture shares obvious similarities with the Evolved Packet System (EPS) as will be seen in Section 4. Based on Gn/Gp interfaces, the GPRS core network functionality is logically implemented on two network nodes, the SGSN and the GGSN.

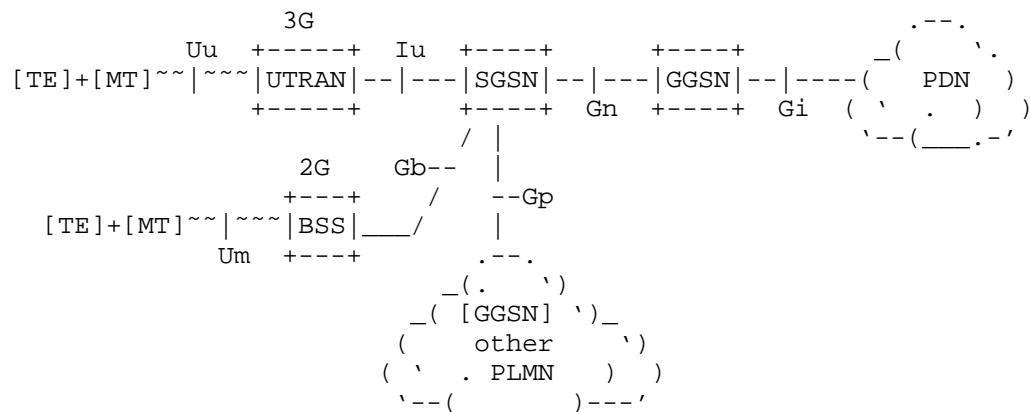


Figure 2: Overview of the 2G/3G GPRS Logical Architecture

- Gn/Gp: These interfaces provide a network based mobility service for a mobile host and are used between a SGSN and a GGSN. The Gn interface is used when GGSN and SGSN are located inside one operator (i.e. PLMN). The Gp-interface is used if the GGSN and the SGSN are located in different operator domains (i.e. 'other' PLMN). GTP protocol is defined for the Gn/Gp interfaces (both GTP-C for the control plane and GTP-U for the user plane).
- Gb: Is the Base Station System (BSS) to SGSN interface, which is used to carry information concerning packet data transmission and layer-2 mobility management. The Gb-interface is based on either on Frame Relay or IP.
- Iu: Is the Radio Network System (RNS) to SGSN interface, which is used to carry information concerning packet data transmission and layer-2 mobility management. The user plane part of the Iu-interface (actually the Iu-PS) is based on GTP-U. The control plane part of the Iu-interface is based on Radio Access Network Application Protocol (RANAP).
- Gi: It is the interface between the GGSN and a PDN. The PDN may be an operator external public or private packet data network or an intra-operator packet data network.
- Uu/Um: Are either 2G or 3G radio interfaces between a mobile terminal and a respective radio access network.

The SGSN is responsible for the delivery of data packets from and to the mobile hosts within its geographical service area when a direct tunnel option is not used. If the direct tunnel is used, then the user plane goes directly between the RNS and the GGSN. The control plane traffic always goes through the SGSN. For each mobile host connected with the GPRS, at any given point of time, there is only one SGSN.

3.2. PDP Context

A PDP context is an association between a mobile host represented by one IPv4 address and/or one /64 IPv6 prefix and a PDN represented by an APN. Each PDN can be accessed via a gateway (typically a GGSN or PDN-GW). On the device/mobile host a PDP context is equivalent to a network interface. A host may hence be attached to one or more gateways via separate connections, i.e. PDP contexts. Each primary PDP context has its own IPv4 address and/or one /64 IPv6 prefix assigned to it by the PDN and anchored in the corresponding gateway.

Applications on the host use the appropriate network interface (PDP context) for connectivity to a specific PDN. Figure 3 represents a high level view of what a PDP context implies in 3GPP networks.

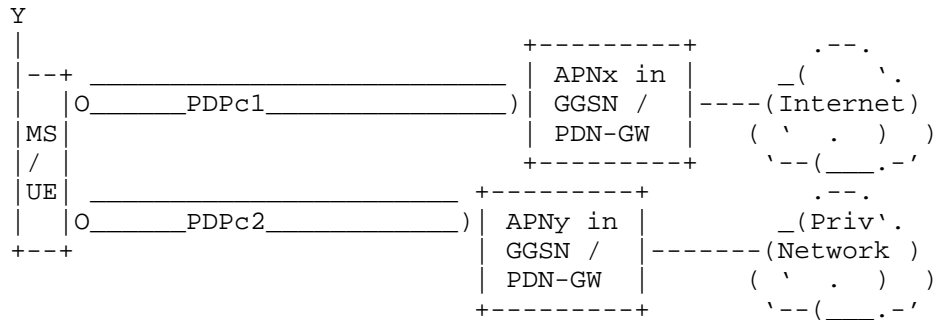


Figure 3: PDP contexts between the MS/UE and gateway

In the above figure there are two PDP contexts at the MS/UE (UE=User Equipment in 3GPP parlance). The 'PDPc1' PDP context that is connected to APNx provided Internet connectivity and the 'PDPc2' PDP context provides connectivity to a private IP network via APNy (as an example this network may include operator specific services such as MMS (Multi media service)). An application on the host such as a web browser would use the PDP context that provides Internet connectivity for accessing services on the Internet. An application such as MMS would use APNy in the figure above because the service is provided through the private network.

4. IP over 3GPP EPS

4.1. Introduction to 3GPP EPS

In its most basic form, the EPS architecture consists of only two nodes on the user plane, a base station and a core network Gateway (GW). The basic EPS architecture is illustrated in Figure 4. The Mobility Management Entity (MME) node performs control-plane functionality and is separated from the node(s) that performs bearer-plane functionality (GW), with a well-defined open interface between them (S11). The optional interface S5 can be used to split the Gateway (GW) into two separate nodes, the Serving Gateway (SGW) and the PDN-GW. This allows independent scaling and growth of traffic throughput and control signal processing. The functional split of gateways also allows for operators to choose optimized topological locations of nodes within the network and enables various deployment models including the sharing of radio networks between different operators.

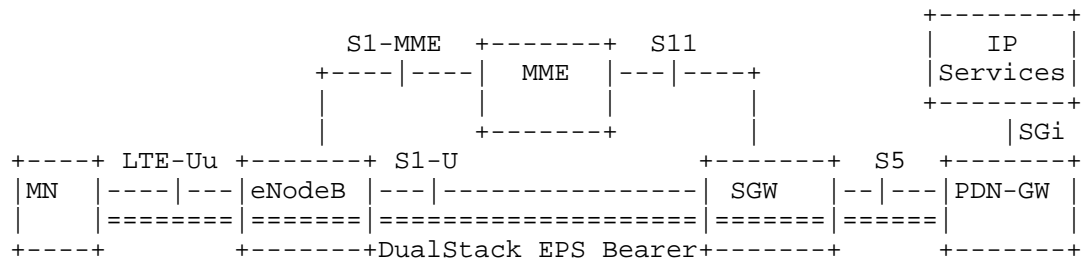


Figure 4: EPS Architecture for 3GPP Access

- 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 (GTP user plane).
- S1-MME:** Reference point for the control plane protocol between eNodeB and MME.
- 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.

The eNodeB is a base station entity that supports the Long Term Evolution (LTE) air interface and includes functions for radio resource control, user plane ciphering, and other lower layer functions. MME is responsible for control plane functionalities, including authentication, authorization, bearer management, layer-2 mobility, etc.

The SGW is the Mobility Anchor point for layer-2 mobility. For each MN connected with the EPS, at any given point of time, there is only one SGW.

4.2. PDN Connection

A PDN connection is an association between a mobile host represented by one IPv4 address and/or one /64 IPv6 prefix, and a PDN represented by an APN. The PDN connection is the EPC equivalent of the GPRS PDP context. Each PDN can be accessed via a gateway (a PDN-GW). PDN is responsible for the IP address/prefix allocation to the mobile host. On the device/mobile host a PDN connection is equivalent to a network interface. A host may hence be attached to one or more gateways via

separate connections, i.e. PDN connections. Each PDN connection has its own IP address/prefix assigned to it by the PDN and anchored in the corresponding gateway. Applications on the host use the appropriate network interface (PDN connection) for connectivity.

4.3. EPS bearer model

The logical concept of a bearer has been defined to be an aggregate of one or more IP flows related to one or more services. An EPS bearer exists between the Mobile Node (MN i.e. a mobile host) and the PDN-GW and is used to provide the same level of packet forwarding treatment to the aggregated IP flows constituting the bearer. Services with IP flows requiring a different packet forwarding treatment would therefore require more than one EPS bearer. The mobile host performs the binding of the uplink IP flows to the bearer while the PDN-GW performs this function for the downlink packets.

In order to provide low latency for always on connectivity, a default bearer will be provided at the time of startup and an IPv4 address and/or IPv6 prefix gets assigned to the mobile host (this is different from GPRS, where mobile hosts are not automatically assigned with an IP address or prefix). This default bearer will be allowed to carry all traffic which is not associated with a dedicated bearer. Dedicated bearers are used to carry traffic for IP flows that have been identified to require a specific packet forwarding treatment. They may be established at the time of startup; for example, in the case of services that require always-on connectivity and better QoS than that provided by the default bearer. The default bearer and the dedicated bearer(s) associated to it share the same IP address(es)/prefix.

An EPS bearer is referred to as a GBR bearer if dedicated network resources related to a Guaranteed Bit Rate (GBR) value that is associated with the EPS bearer are permanently allocated (e.g. by an admission control function in the eNodeB) at bearer establishment/modification. Otherwise, an EPS bearer is referred to as a non-GBR bearer. The default bearer is always non-GBR, with the resources for the IP flows not guaranteed at eNodeB, and with no admission control. However, the dedicated bearer can be either GBR or non-GBR. A GBR bearer has a Guaranteed Bit Rate (GBR) and Maximum Bit Rate (MBR) while more than one non-GBR bearer belonging to the same UE shares an Aggregate Maximum Bit Rate (AMBR). Non-GBR bearers can suffer packet loss under congestion while GBR bearers are immune to such losses.

5. Address Management

5.1. IPv4 Address Configuration

Mobile host's IPv4 address configuration is always performed during PDP context/EPS bearer setup procedures (on layer-2). DHCPv4-based [RFC2131] address configuration is supported by the 3GPP specifications, but is not used in wide scale. The mobile host must always support layer-2 based address configuration, since DHCPv4 is optional for both mobile hosts and networks.

5.2. IPv6 Address Configuration

IPv6 Stateless Address Autoconfiguration (SLAAC) as specified in [RFC4862] is the only supported address configuration mechanism. Stateful DHCPv6-based address configuration is not supported by 3GPP specifications [RFC3315]. On the other hand, Stateless DHCPv6-service to obtain other configuration information is supported [RFC3736]. This implies that the M-bit must always be set to zero and the O-bit may be set to one in the Router Advertisement (RA) sent to the UE.

3GPP network allocates each default bearer a unique /64 prefix, and uses layer-2 signaling to suggest user equipment an Interface Identifier that is guaranteed not to conflict with gateway's Interface Identifier. The UE may configure link local address using this Interface Identifier, but is allowed to use also other Interface Identifiers and as many globally scoped addresses as it needs. There is no restriction, for example, of using Privacy Extension for SLAAC [RFC4941] or other similar types of mechanisms.

In the 3GPP link model the /64 prefix assigned to the UE is always off-link (i.e. the L-bit in the Prefix Information Option (PIO) in the RA must be set to zero). If the advertised prefix is used for SLAAC then the A-bit in the PIO must be set to one. The details of the 3GPP link-model and address configuration is described in Section 11.2.1.3.2a of [3GPP.29.061]. More specifically, the GGSN/PDN-GW guarantees that the /64 prefix is unique for the mobile host. Therefore, there is no need to perform any Duplicate Address Detection (DAD) on addresses the mobile host creates (i.e., the 'DupAddrDetectTransmits' variable in the mobile host should be zero). The GGSN/PDN-GW is not allowed to generate any globally unique IPv6 addresses for itself using the /64 prefix assigned to the mobile host in the RA.

The current 3GPP architecture limits number of prefixes in each bearer to a single /64 prefix. If the mobile host finds more than one prefix in the RA, it only considers the first one and silently discard the others [3GPP.29.061]. Therefore, multi-homing within a single bearer is not possible. Renumbering without closing layer-2

connection is also not possible. The lifetime of /64 prefix is bound to lifetime of layer-2 connection even if the advertised prefix lifetime would be longer than the layer-2 connection lifetime.

5.3. Prefix Delegation

IPv6 prefix delegation is a part of Release-10 and is not covered by any earlier release. However, the /64 prefix allocated for each default bearer (and to the user equipment) may be shared to local area network by user equipment implementing Neighbor Discovery proxy (ND proxy) [RFC4389] functionality.

Release-10 prefix delegation uses the DHCPv6-based prefix delegation [RFC3633]. The model defined for Release-10 requires aggregatable prefixes, which means the /64 prefix allocated for the default bearer (and to the user equipment) must be part of the shorter delegated prefix. DHCPv6 prefix delegation has an explicit limitation described in Section 12.1 of [RFC3633] that a prefix delegated to a requesting router cannot be used by the delegating router (i.e., the PDN-GW in this case). This implies the shorter 'delegated prefix' cannot be given to the requesting router (i.e. the user equipment) as such but has to be delivered by the delegating router (i.e. the PDN-GW) in such a way the /64 prefix allocated to the default bearer is not part of the 'delegated prefix'. IETF is working on a solution for DHCPv6-based prefix delegation to exclude a specific prefix from the 'delegated prefix' [I-D.ietf-dhc-pd-exclude].

6. 3GPP Dual-Stack Approach to IPv6

6.1. 3GPP Networks Prior to Release-8

3GPP standards prior to Release-8 provide IPv6 access for cellular devices with PDP contexts of type IPv6 [3GPP.23.060]. For dual-stack access, a PDP context of type IPv6 is established in parallel to the PDP context of type IPv4, as shown in Figure 5 and Figure 6. For IPv4-only service, connections are created over the PDP context of type IPv4 and for IPv6-only service connections are created over the PDP context of type IPv6. The two PDP contexts of different type may use the same APN (and the gateway), however, this aspect is not explicitly defined in standards. Therefore, cellular device and gateway implementations from different vendors may have varying support for this functionality.

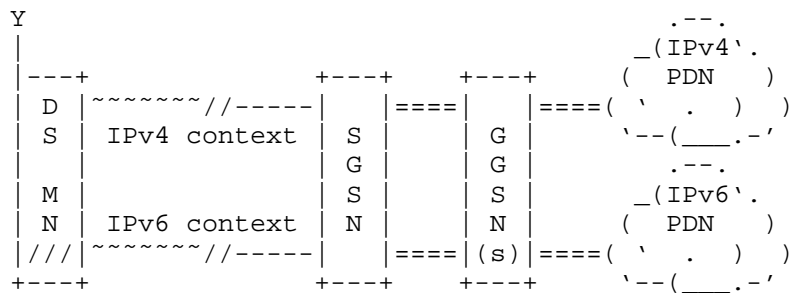


Figure 5: A dual-stack mobile host connecting to both IPv4 and IPv6 Internet using parallel IPv4-only and IPv6-only PDP contexts

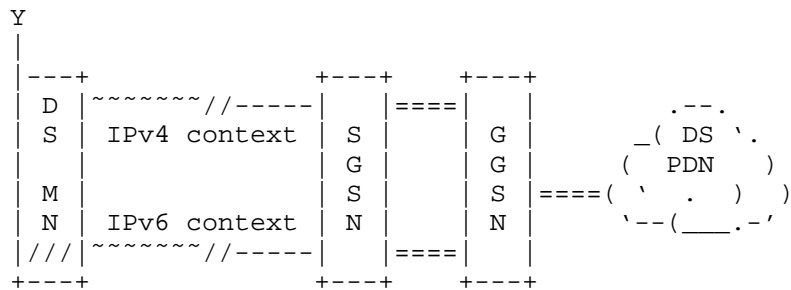


Figure 6: A dual-stack mobile host connecting to dual-stack Internet using parallel IPv4-only and IPv6-only PDP contexts

The approach of having parallel IPv4 and IPv6 type of PDP contexts open is not optimal, because two PDP contexts require double the signaling and consume more network resources than a single PDP context. In the figure above the IPv4 and IPv6 PDP contexts are attached to the same GGSN. While this is possible, the DS MS may be attached to different GGSNs in the scenario where one GGSN supports IPv4 PDN connectivity while another GGSN provides IPv6 PDN connectivity.

6.2. 3GPP Release-8 and -9 Networks

Since 3GPP Release-8, the powerful concept of a dual-stack type of PDN connection and EPS bearer have been introduced [3GPP.23.401]. This enables parallel use of both IPv4 and IPv6 on a single bearer (IPv4v6), as illustrated in Figure 7, and makes dual stack simpler than in earlier 3GPP releases. As of Release-9, GPRS network nodes also support dual-stack type (IPv4v6) PDP contexts.

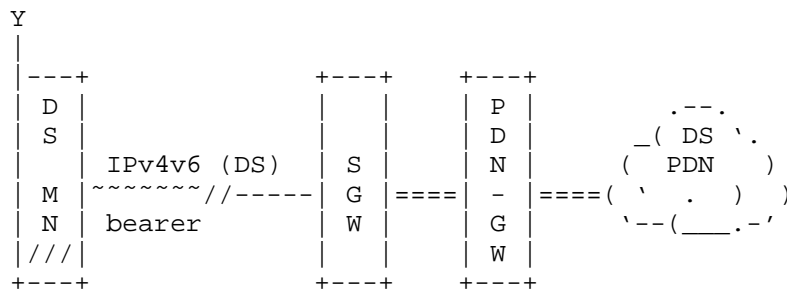


Figure 7: A dual-stack mobile host connecting to dual-stack Internet using a single IPv4v6 type PDN connection

The following is a description of the various PDP contexts/PDN bearer types that are specified by 3GPP:

1. For 2G/3G access to GPRS core (SGSN/GGSN) pre-Release-9 there are two IP PDP Types, IPv4 and IPv6. Two PDP contexts are needed to get dual stack connectivity.
2. For 2G/3G access to GPRS core (SGSN/GGSN) from Release-9 there are three IP PDP Types, IPv4, IPv6 and IPv4v6. Minimum one PDP context is needed to get dual stack connectivity.
3. For 2G/3G access to EPC core (PDN-GW via S4 Release-8 SGSN) from Release-8 there are three IP PDP Types, IPv4, IPv6 and IPv4v6 which gets mapped to PDN Connection type. Minimum one PDP Context is needed to get dual stack connectivity.
4. For LTE (E-UTRAN) access to EPC core from Release-8 there are three IP PDN Types, IPv4, IPv6 and IPv4v6. Minimum one PDN Connection is needed to get dual stack connectivity.

6.3. PDN Connection Establishment Process

The PDN connection establishment process is specified in detail in 3GPP specifications. Figure 8 illustrates the high level process and signaling involved in the establishment of a PDN connection.

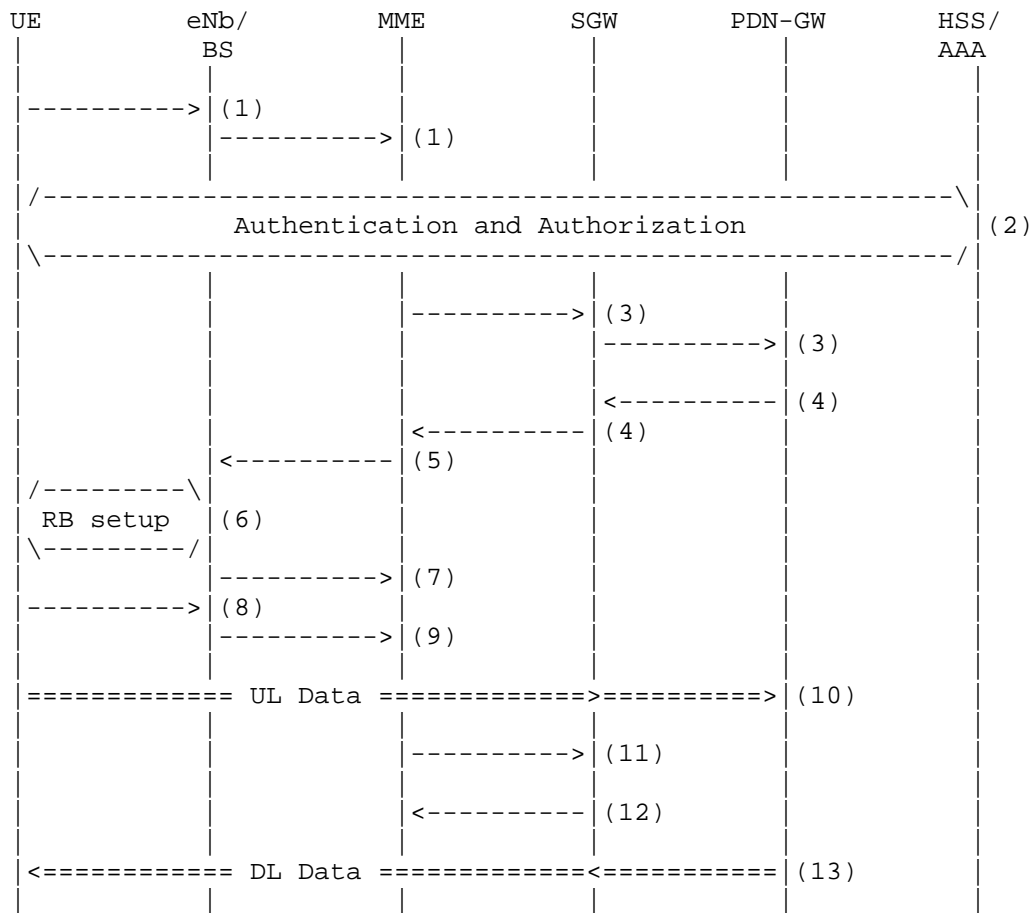


Figure 8: Simplified PDN connection setup procedure in Release-8

1. The UE (i.e the MS) requires a data connection and hence decides to establish a PDN connection with a PDN-GW. The UE sends an "Attach Request" (layer-2) to the BS. The BS forwards this attach request to the MME.
2. Authentication of the UE with the AAA server/HSS follows. If the UE is authorized for establishing a data connection, the following steps continue
3. The MME sends a "Create Session Request" message to the Serving-GW. The SGW forwards the create session request to the PDN-GW. The SGW knows the address of the PDN-GW to forward the create session request to as a result of this information having been obtained by the MME during the authentication/authorization

phase.

The UE IPv4 address and/or IPv6 prefix get assigned during this step. If a subscribed IPv4 address and/or IPv6 prefix is statically allocated for the UE for this APN, then the MME already passes the address information to the SGW and eventually to the PDN-GW in the "Create Session Request" message. Otherwise, the PDN-GW manages the address assignment to the UE (there is another variation to this where IPv4 address allocation is delayed until the UE initiates a DHCPv4 exchange but this is not discussed here).

4. The PDN-GW creates a PDN connection for the UE and sends "Create Session Response" message to the SGW from which the session request message was received from. The SGW forwards the response to the corresponding MME which originated the request.
5. The MME sends the "Attach Accept/Initial Context Setup request" message to the eNodeB/BS.
6. The radio bearer between the UE and the eNb is reconfigured based on the parameters received from the MME
7. The eNb sends "Initial Context Response" message to the MME.
8. The UE sends a "Direct Transfer" message to the eNodeB which includes the Attach complete signal.
9. The eNodeB forwards the Attach complete message to the MME.
10. The UE can now start sending uplink packets to the PDN GW.
11. The MME sends a "Modify Bearer Request" message to the SGW.
12. The SGW responds with a "Modify Bearer Response" message. At this time the downlink connection is also ready
13. The UE can now start receiving downlink packets

The type of PDN connection established between the UE and the PDN-GW can be any of the types described in the previous section. The DS PDN connection, i.e the one which supports both IPv4 and IPv6 packets is the default one that will be established if no specific PDN connection type is specified by the UE in Release-8 networks.

6.4. Mobility of 3GPP IPv4v6 Type of Bearers

3GPP discussed at length various approaches to support mobility between Release-8 and pre-Release-8 networks for the new dual-stack type of bearers.

The chosen approach for mobility is as follows, in short: if a mobile is known to be at risk for doing handovers between Release-8 and pre-Release-8 networks, only single stack bearers are used. Essentially meaning:

1. If a network knows a mobile may do handovers between Release-8 and pre-Release-8 networks (segment), network will only provide single stack bearers, even if the mobile host requests dual-stack bearers. This can happen e.g. if an operator is using pre-Release-8 SGSNs in some parts of the network. The single stack bearers of Release-8 are easy to map one-to-one to pre-Release-8 bearers.
2. If a network knows a mobile will not be able to do handover to pre-Release-8 network (segment), it will provide mobile with dual-stack bearers on request. This can happen e.g. if an operator has upgraded their SGSNs to support dual-stack bearers, or if an operator is running LTE-only network.

When a network operator and their roaming partners have upgraded their networks to Release-8, it is possible to use the new IPv4v6 dual-stack type of bearers. A Release-8 mobile device always requests for a dual-stack bearer, but accepts what is assigned by the network.

7. Dual-Stack Approach to IPv6 Transition in 3GPP Networks

3GPP networks can natively transport IPv4 and IPv6 packets between the mobile station/UE and the gateway (GGSN or PDN-GW) as a result of establishing either a dual-stack PDP context or parallel IPv4 and IPv6 PDP contexts.

Current deployments of 3GPP networks primarily support IPv4 only. These networks can be upgraded to also support IPv6 PDP contexts. By doing so devices and applications that are IPv6 capable can start utilizing the IPv6 connectivity. This will also ensure that legacy devices and applications continue to work with no impact. As newer devices start using IPv6 connectivity, the demand for actively used IPv4 connections is expected to slowly decrease, helping operators with a transition to IPv6. With a dual-stack approach, there is always the potential to fallback to IPv4. A device which may be

roaming in a network wherein IPv6 is not supported by the visited network could fall back to using IPv4 PDP contexts and hence the end user would at least get some connectivity. Unfortunately, dual-stack approach as such does not lower the number of used IPv4 addresses. Every dual-stack bearer still needs to be given an IPv4 address, private or public. This is a major concern with dual-stack bearers concerning IPv6 transition. However, if the majority of active IP communication has moved over to IPv6, then in case of NAT44 [RFC1918] IPv4 connections the number of active IPv4 connections can still be expected to gradually decrease and thus giving some level of relief regarding NAT44 function scalability.

As the networks evolve to support Release-8 EPS architecture and the dual-stack PDP contexts, newer devices will be able to leverage such capability and have a single bearer which supports both IPv4 and IPv6. Since IPv4 and IPv6 packets are carried as payload within GTP between the MS and the gateway (GGSN/PDN-GW) the transport network capability in terms of whether it supports IPv4 or IPv6 on the interfaces between the eNodeB and SGW or, SGW and PDN-GW is immaterial.

8. Deployment issues

8.1. Overlapping IPv4 Addresses

Given the shortage of globally routable public IPv4 addresses, operators tend to assign private IPv4 addresses [RFC1918] to hosts when they establish an IPv4 only PDP context or an IPv4v6 type PDN context. About 16 million hosts can be assigned a private IPv4 address that is unique within a domain. However, in case of many operators the number of subscribers is greater than 16 million. The issue can be dealt with by assigning overlapping RFC 1918 IPv4 addresses to hosts. As a result the IPv4 address assigned to a host within the context of a single operator realm would no longer be unique. This has the obvious and known issues of NATed IP connection in the Internet. Direct host to host connectivity becomes complicated, unless the hosts are within the same private address range pool and/or anchored to the same gateway, referrals using IP addresses will have issues and so forth. These are generic issues and not only a concern of the EPS. However, 3GPP as such does not have any mandatory language concerning NAT44 functionality in EPC. Obvious deployment choices apply also to EPC:

1. Very large network deployments are partitioned, for example, based on geographical areas. This partitioning allows overlapping IPv4 address ranges to be assigned to hosts that are in different areas. Each area has its own pool of gateways

that are dedicated for a certain overlapping IPv4 address range (referred here later as a zone). Standard NAT44 functionality enables the communication between hosts that are assigned the same IPv4 address but belong to different zones, yet are part of the same operator domain.

2. A mobile host/device attaches to a gateway as part of the attach process. The number of hosts that a gateway supports is in the order of 1 to 10 million. Hence all the hosts assigned to a single gateway can be assigned private IPv4 addresses. Operators with large subscriber bases have multiple gateways and hence the same [RFC1918] IPv4 address space can be reused across gateways. The IPv4 address assigned to a host is unique within the scope of a single gateway.
3. New services requiring direct connectivity between hosts should be build on IPv6. Possible existing IPv4-only services and applications requiring direct connectivity can be ported to IPv6.

8.2. IPv6 for transport

The various reference points of the 3GPP architecture such as S1-U, S5 and S8 are based on either GTP or PMIPv6. The underlying transport for these reference points can be IPv4 or IPv6. GTP has been able to operate over IPv6 transport (optionally) since R99 and PMIPv6 has supported IPv6 transport starting from its introduction in Release-8. The user plane traffic between the mobile host and the gateway can use either IPv4 or IPv6. These packets are essentially treated as payload by GTP/PMIPv6 and transported accordingly with no real attention paid to the information (at least from a routing perspective) contained in the IPv4 or IPv6 headers. The transport links between the eNodeB and the SGW, and the link between the SGW and PDN-GW can be migrated to IPv6 without any direct implications to the architecture.

Currently, the inter-operator (for 3GPP technology) roaming networks are all IPv4 only (see Inter-PLMN Backbone Guidelines [GSMA.IR.34]). Eventually these roaming networks will also get migrated to IPv6, if there is a business reason for that. The migration period can be prolonged considerably because the 3GPP protocols always tunnel user plane traffic in the core network and as described earlier the transport network IP version is not in any way tied to user plane IP version. Furthermore, the design of the inter-operator roaming networks is such that the user plane and transport network IP addressing is completely separated from each other. The inter-operator roaming network itself is also completely separated from the Internet. Only those core network nodes that must be connected to the inter-operator roaming networks are actually visible there, and

be able to send and receive (tunneled) traffic within the inter-operator roaming networks. Obviously, in order the roaming to work properly, the operators have to agree on supported protocol versions so that the visited network does not, for example, unnecessarily drop user plane IPv6 traffic.

8.3. Operational Aspects of Running Dual-Stack Networks

Operating dual-stack networks does imply cost and complexity to a certain extent. However these factors are mitigated by the assurance that legacy devices and services are unaffected and there is always a fallback to IPv4 in case of issues with the IPv6 deployment or network elements. The model also enables operators to develop operational experience and expertise in an incremental manner.

Running dual-stack networks requires the management of multiple IP address spaces. Tracking of hosts needs to be expanded since it can be identified by either an IPv4 address or IPv6 prefix. Network elements will also need to be dual-stack capable in order to support the dual-stack deployment model.

Deployment and migration cases described in Section 6.1 for providing dual-stack like capability may mean doubled resource usage in operator's network. This is a major concern against providing dual-stack like connectivity using techniques discussed in Section 6.1. Also handovers between networks with different capabilities in terms of networks being dual-stack like service capable or not, may turn out hard to comprehend for users and for application/services to cope with. These facts may add other than just technical concerns for operators when planning to roll out dual-stack service offerings.

8.4. Operational Aspects of Running a Network with IPv6 Only Bearers

It is possible to allocate IPv6 only type bearers to mobile hosts in 3GPP networks. IPv6 only bearer type has been part of the 3GPP specification since the beginning. In 3GPP Release-8 (and later) it was defined that a dual-stack mobile host (or when the radio equipment has no knowledge of the host IP stack capabilities) must first attempt to establish a dual-stack bearer and then possibly fall back to single IP version bearer. A Release-8 (or later) mobile host with IPv6 only stack can directly attempt to establish an IPv6 only bearer. The IPv6 only behavior is up to a subscription provisioning or a PDN-GW configuration, and the fallback scenarios do not necessarily cause additional signaling.

Although the bullets below introduce IPv6 to IPv4 address translation and specifically discuss NAT64 technology [I-D.ietf-behave-v6v4-framework], the current 3GPP Release-8

architecture does not describe the use of address translation or NAT64. It is up to a specific deployment whether address translation is part of the network or not. Some operational aspects to consider for running a network with IPv6 only bearers:

- o The mobile hosts must have an IPv6 capable stack and a radio interface capable of establishing an IPv6 PDP context or PDN connection.
- o The GGSN/PDN-GW must be IPv6 capable in order to support IPv6 bearers. Furthermore, the SGSN/MME must allow the creation of PDP Type or PDN Type of IPv6.
- o Many of the common applications are IP version agnostic and hence would work using an IPv6 bearer. However, applications that are IPv4 specific would not work.
- o Inter-operator roaming is another aspect which causes issues, at least during the ramp up phase of the IPv6 deployment. If the visited network to which outbound roamers attach to does not support PDP/PDN Type IPv6, then there needs to be a fallback option. The fallback option in this specific case is mostly up to the mobile host to implement. Several cases are discussed in the following sections.
- o If and when a mobile host using IPv6 only bearer needs to access to IPv4 Internet/network, a translation of some type from IPv6 to IPv4 has to be deployed in the network. NAT64 (and DNS64) is one solution that can be used for this purpose and works for a certain set of protocols (read TCP and UDP, and when applications actually use DNS for resolving name to IP addresses).

8.5. Restricting Outbound IPv6 Roaming

Roaming was briefly touched upon in Sections 8.2 and 8.4. While there is interest in offering roaming service for IPv6 enabled mobile hosts and subscriptions, not all visited networks are prepared for IPv6 outbound roamers. There are basically two issues. First, the visited network (S4-)SGSN does not support the IPv6 PDP Context or IPv4v6 PDP Context types. These should mostly concern pre-Release-8 networks but there is no definitive rule as the deployed feature sets vary depending on implementations and licenses. Second, the visited network might not be commercially ready for IPv6 outbound roamers, while everything might work technically at the user plane level. This would lead to "revenue leakage" especially from the visited operator point of view (note that the use of visited network GGSN/PDN-GW does not really exist in real deployments today). Therefore, it might be in the interest of operators to prohibit roaming

selectively within specific visited networks.

Unfortunately, it is not mandatory to implement/deploy 3GPP standards based solution to selectively prohibit IPv6 roaming without also prohibiting other packet services (such as IPv4 roaming). However, there are few possibilities how this can be done in real deployments. The examples given below are either optional and/or vendor specific features to the 3GPP EPC:

- o Using Policy and Charging Control (PCC) [3GPP.23.203] functionality and its rules to fail, for example, the bearer authorization when a desired criteria is met. In this case that would be PDN/PDP Type IPv6/IPv4v6 and a specific visited network. The rules can be provisioned either in the home network or locally in the visited network.
- o Some Home Location Register (HLR) and Home Subscriber Server (HSS) subscriber databases allow prohibiting roaming in a specific (visited) network for a specified PDN/PDP Type.

The obvious problems are that these solutions are not mandatory, are not unified across networks, and therefore also lack well-specified fall back mechanism from the mobile host point of view.

8.6. Inter-rat Handovers and IP Versions

It is obvious that when operators start to incrementally deploy EPS (and E-UTRAN) along with the existing UTRAN/GERAN, handovers between different radio technologies (inter-rat handovers) become inevitable. In case of inter-rat handovers 3GPP supports the following IP addressing scenarios:

- o E-UTRAN IPv4v6 bearer has to map one to one to UTRAN/GERAN IPv4v6 bearer.
- o E-UTRAN IPv6 bearer has to map one to one to UTRAN/GERAN IPv6 bearer.
- o E-UTRAN IPv4 bearer has to map one to one to UTRAN/GERAN IPv4 bearer.

Other types of configurations are considered network planning mistakes. What the above rules essentially imply is that the network migration has to be planned and subscriptions provisioned based on the lowest common nominator, if inter-rat handovers are desired. For example, if some part of the UTRAN network cannot serve anything but IPv4 bearers, then the E-UTRAN is also forced to provide only IPv4 bearers. Various combinations of subscriber provisioning regarding

IP versions are discussed further in Section 8.7.

8.7. Provisioning of IPv6 Subscribers and Various Combinations During Initial Network Attachment

Subscribers' provisioned PDP/PDN Types have multiple configurations. The supported PDP/PDN Type is provisioned per each APN for every subscriber. The following PDN Types are possible in the HSS for a Release-8 subscription [3GPP.23.401]:

- o IPv4v6 PDN Type (note that IPv4v6 PDP Type does not exist in HLR).
- o IPv6 only PDN Type
- o IPv4 only PDN Type.
- o IPv4_or_IPv6 PDN Type (note that IPv4_or_IPv6 PDP Type does not exist in HLR).

A Release-8 dual-stack mobile host must always attempt to establish a PDP/PDN Type IPv4v6 bearer. The same also applies when the modem part of the mobile host does not have exact knowledge whether the host operating system IP stack is a dual-stack capable or not. A mobile host that is IPv6 only capable must attempt to establish a PDP/PDN Type IPv6 bearer. Last, a mobile host that is IPv4 only capable must attempt to establish a PDN/PDP Type IPv4 bearer.

In a case the PDP/PDN Type requested by a mobile host does not match what has been provisioned for the subscriber in the HSS (or HLR), the mobile host possibly falls back to a different PDP/PDN Type. The network (i.e. the MME or the SGSN) is able to inform the mobile host during the network attachment signaling why it did not get the requested PDP/PDN Type. These response/cause codes are documented in [3GPP.24.008][3GPP.24.301]. Possible fall back cases include (as documented in [3GPP.23.401]):

- o Requested & provisioned PDP/PDN Types match -> requested.
- o Requested IPv4v6 & provisioned IPv6 -> IPv6 and a mobile host receives indication that IPv6-only bearer is allowed.
- o Requested IPv4v6 & provisioned IPv4 -> IPv4 and the mobile host receives indication that IPv4-only bearer is allowed.
- o Requested IPv4v6 & provisioned IPv4_or_IPv6 -> IPv4 or IPv6 is selected by the MME based on an unspecified criteria. The mobile host may then attempt to establish, based on the mobile host implementation, a parallel bearer of a different PDP/PDN Type.

- o Other combinations cause the bearer establishment to fail.

In addition to PDP/PDN Types provisioned in the HSS, it is also possible for a PDN-GW (and a MME) to affect the final selected PDP/PDN Type:

- o Requested IPv4v6 & configured IPv4 or IPv6 in the PDN-GW -> IPv4 or IPv6. If the MME operator had included the "Dual Address Bearer Flag" into the bearer establishment signaling, then the mobile host receives an indication that IPv6-only or IPv4-only bearer is allowed.
- o Requested IPv4v6 & configured IPv4 or IPv6 in the PDN-GW -> IPv4 or IPv6. If the MME operator had not included the "Dual Address Bearer Flag" into the bearer establishment signaling, then the mobile host may attempt to establish, based on the mobile host implementation, a parallel bearer of different PDP/PDN Type.

If for some reason a SGSN does not understand the requested PDP Type, then the PDP Type is handled as IPv4. If for some reason a MME does not understand the requested PDN Type, then the PDN Type is handled as IPv6.

9. IANA Considerations

This document has no requests to IANA.

10. Security Considerations

This document does not introduce any security related concerns.

11. Summary and Conclusion

The 3GPP network architecture and specifications enable the establishment of IPv4 and IPv6 connections through the use of appropriate PDP context types. The current generation of deployed networks can support dual-stack connectivity if the packet core network elements such as the SGSN and GGSN have the capability. With Release-8, 3GPP has specified a more optimal PDP context type which enables the transport of IPv4 and IPv6 packets within a single PDP context between the mobile station and the gateway.

As devices and applications are upgraded to support IPv6 they can start leveraging the IPv6 connectivity provided by the networks while maintaining the fall back to IPv4 capability. Enabling IPv6

connectivity in the 3GPP networks by itself will provide some degree of relief to the IPv4 address space as many of the applications and services can start to work over IPv6. However without comprehensive testing of different applications and solutions that exist today and are widely used, for their ability to operate over IPv6 PDN connections, an IPv6 only access would cause disruptions.

12. Acknowledgements

The authors thank Shabnam Sultana, Sri Gundavelli, Hui Deng, and Zhenqiang Li, Mikael Abrahamsson, James Woodyatt and Cameron Byrne for their reviews and comments on this document.

13. Informative References

- [3GPP.23.060]
3GPP, "General Packet Radio Service (GPRS); Service description; Stage 2", 3GPP TS 23.060 8.8.0, March 2010.
- [3GPP.23.203]
3GPP, "Policy and charging control architecture (PCC)", 3GPP TS 23.203 8.11.0, September 2010.
- [3GPP.23.401]
3GPP, "General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access", 3GPP TS 23.401 10.2.1, January 2011.
- [3GPP.23.975]
3GPP, "IPv6 Migration Guidelines", 3GPP TR 23.975 1.1.1, June 2010.
- [3GPP.24.008]
3GPP, "Mobile radio interface Layer 3 specification", 3GPP TS 24.008 8.12.0, December 2010.
- [3GPP.24.301]
3GPP, "Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS)", 3GPP TS 24.301 8.8.0, December 2010.
- [3GPP.29.060]
3GPP, "General Packet Radio Service (GPRS); GPRS Tunnelling Protocol (GTP) across the Gn and Gp interface", 3GPP TS 29.274 8.8.0, April 2010.
- [3GPP.29.061]

3GPP, "Interworking between the Public Land Mobile Network (PLMN) supporting packet based services and Packet Data Networks (PDN)", 3GPP TS 29.061 8.5.0, April 2010.

[3GPP.29.274]

3GPP, "3GPP Evolved Packet System (EPS); Evolved General Packet Radio Service (GPRS) Tunnelling Protocol for Control plane (GTPv2-C)", 3GPP TS 29.060 8.11.0, December 2010.

[GSMA.IR.34]

GSMA, "Inter-PLMN Backbone Guidelines", GSMA PRD IR.34.4.9, March 2010.

[I-D.ietf-behave-v6v4-framework]

Baker, F., Li, X., Bao, C., and K. Yin, "Framework for IPv4/IPv6 Translation", draft-ietf-behave-v6v4-framework-10 (work in progress), August 2010.

[I-D.ietf-dhc-pd-exclude]

Korhonen, J., Savolainen, T., Krishnan, S., and O. Troan, "Prefix Exclude Option for DHCPv6-based Prefix Delegation", draft-ietf-dhc-pd-exclude-01 (work in progress), January 2011.

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

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, March 1997.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC3633] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, December 2003.

[RFC3736] Droms, R., "Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6", RFC 3736, April 2004.

[RFC4389] Thaler, D., Talwar, M., and C. Patel, "Neighbor Discovery Proxies (ND Proxy)", RFC 4389, April 2006.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless

Address Autoconfiguration", RFC 4862, September 2007.

[RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, September 2007.

[RFC5213] Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., and B. Patil, "Proxy Mobile IPv6", RFC 5213, August 2008.

Authors' Addresses

Jouni Korhonen (editor)
Nokia Siemens Networks
Linnoitustie 6
FI-02600 Espoo
FINLAND

Email: jouni.nospam@gmail.com

Jonne Soininen
Renesas Mobile

Email: jonne.soininen@renesasmobile.com

Basavaraj Patil
Nokia
6021 Connection drive
Irving, TX 75039
USA

Email: basavaraj.patil@nokia.com

Teemu Savolainen
Nokia
Hermiankatu 12 D
FI-33720 Tampere
FINLAND

Email: teemu.savolainen@nokia.com

Gabor Bajko
Nokia
323 Fairchild drive 6
Mountain view, CA 94043
USA

Email: gabor.bajko@nokia.com

Kaisu Iisakkila
Renesas Mobile

Email: kaisu.iisakkila@renesasmobile.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 24, 2016

N. Matsuhira
Fujitsu Limited
July 23, 2015

Stateless Automatic IPv4 over IPv6 Encapsulation / Decapsulation
Technology: Specification
draft-matsuhira-sa46t-spec-11

Abstract

This document specifies Stateless Automatic IPv4 over IPv6 Encapsulation / Decapsulation Technology (SA46T) base specification. SA46T makes backbone network to IPv6 only. And also, SA46T can stack many IPv4 networks, i.e. the networks using same IPv4 (private) addresses, without interdependence.

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 January 24, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents

(<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Architecture of SA46T	3
3. Basic Network Configuration	5
4. Basic Function of SA46T	6
4.1. IPv4 over IPv6 Encapsulation / Decapsulation	6
4.2. SA46T address architecture	7
4.3. Route Advertisement	8
5. SA46T address format	9
5.1. IPv6 Global Unicast Address as SA46T address	9
5.2. Global SA46T address format	10
6. Stacking IPv4 Networks	10
7. Redundancy of SA46T	12
8. Configuration of SA46T and address allocation	12
9. Example of SA46T Operation	16
9.1. Basic SA46T Operation	16
9.2. SA46T Operation with plane ID	18
10. Characteristic	21
11. IANA Considerations	22
12. Security Considerations	22
13. Acknowledgements	22
14. References	23
14.1. Normative References	23
14.2. References	23
Appendix A. Test implementation of SA46T	24
Appendix B. SA46T experiments	24
B.1. WIDE camp at Sept 2010	24
B.2. NICT JGN2Plus Testbed at Feb 2011	24
B.3. Some corporate network	25
B.4. Interop 2011 Tokyo at Jun 2011	25
Author's Address	25

1. Introduction

This document provides Stateless Automatic IPv4 over IPv6 Encapsulation / Decapsulation Technology (SA46T) base specification.

The basic strategy for IPv6 deployment is dual stack. Viewing this strategy from operational side, operation cost of dual stack is higher than single stack operation. Viewing from future, IPv6 only operation is more reasonable rather than IPv4 only operation. Therefore IPv6 only operation is desired.

SA46T makes backbone network to IPv6 only. And also, SA46T can stack many IPv4 networks, i.e. the networks using same IPv4 (private) address, without interdependence.

2. Architecture of SA46T

IP address contain two information, one is locator information, and another is identifier information. This is basic architecture of internet protocol, and also the Internet, and no difference between IPv4 and IPv6.

Locator is a information related "Where", and identifier is a information related "Who". That mean, IP address's semantics is "Where's Who" meaning. Host is identified whole IP address information, that is "Where's Who", however route to the host is identified just locator information in IP address, that is "Where". See Figure 1.

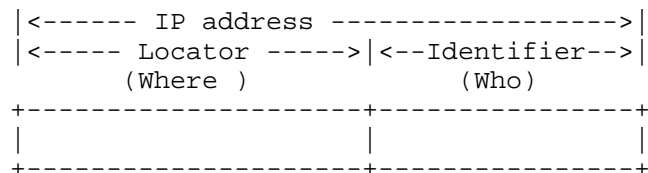


Figure 1

In IPv4 address space, some host has IPv4 address, which consist n bits length identifier and $32 - n$ bits locator. In Where's Who representation, $32 - n$ bits "Where" and n bits "Who".

Keeping such "Where's Who" relation, IPv4 address can be represent as IPv6 address by expanding "Where" information from $32 - n$ bits to $128 - n$ bits. Expanding "Where" information, IPv4 address can be mapped

to IPv6 address. Figure 2 shows such expanding.

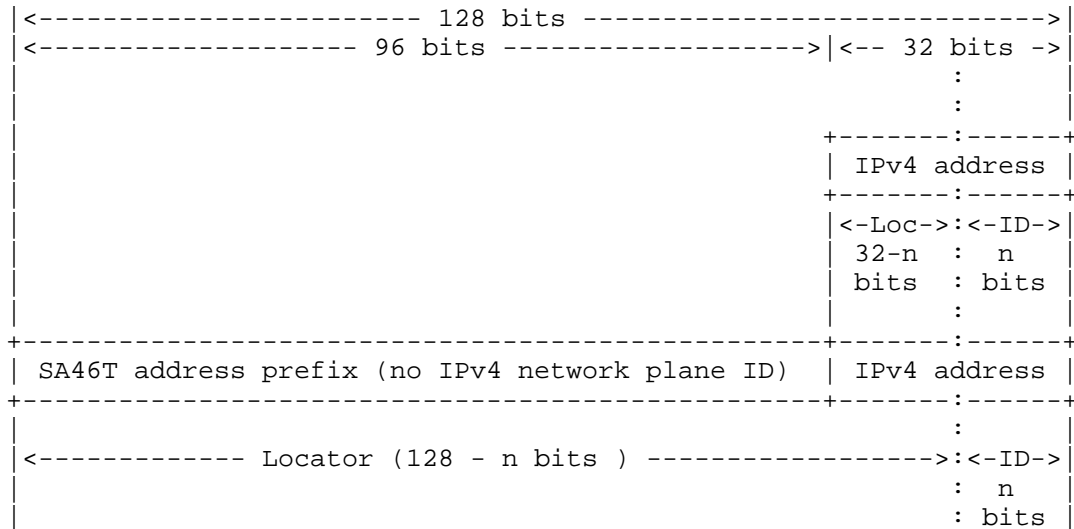


Figure 2

IPv4 address space contain private address, that is non globally unique IP address. If some identifier which distinguish private address can introduce in IPv6 address space, we can treat IPv4 private address as different address in IPv6 address space. This document define such identifier as "IPv4 network plane ID". "IPv6 network plane ID" can provide VPN (Virtual Private Network) like service.

That is SA46T address. In SA46T address, "Where" information's bit length is 128 - n bits, and "Who" information's bit length is n bits. Figure 3 shows summary of IPv4 address and SA46T address relation.

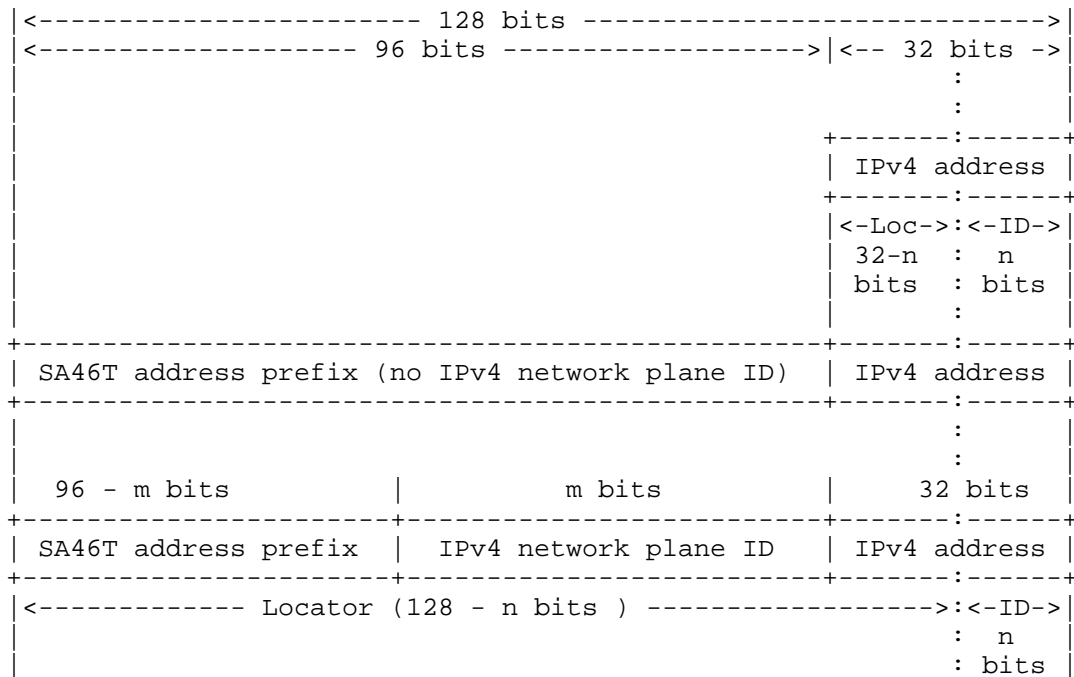


Figure 3

3. Basic Network Configuration

Figure 4 shows network configuration with SA46T. The network consists of three parts. Backbone network, stub network, and SA46T.

Backbone network is operated with IPv6 only. Stub network has three cases. IPv4 only, Dual Stack (both IPv4 and IPv6), and IPv6 only.

SA46T connects backbone network and stub network in case IPv4 still works in that stub network. If stub network is IPv6 only, SA46T is not needed.

Campus network, corporate network, and ISP network are the example for such network.

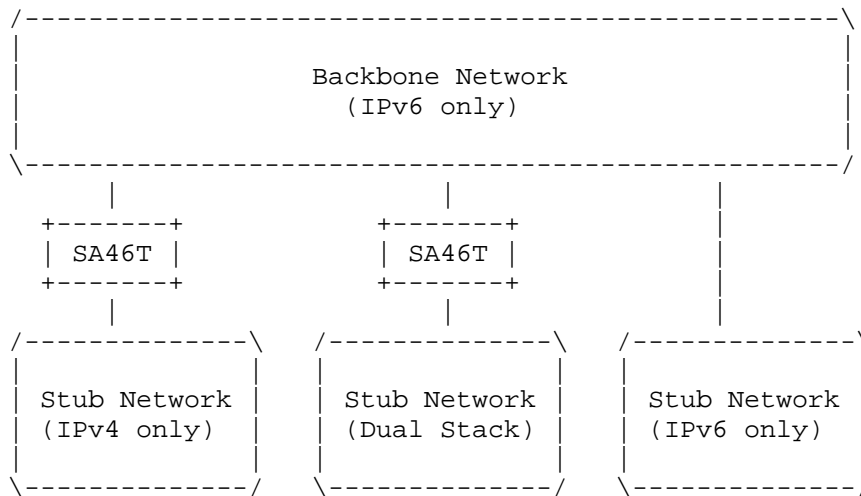


Figure 4

4. Basic Function of SA46T

SA46T has mainly two function. One is IPv4 over IPv6 Encapsulation / Decapsulation, and another is advertise route for stub network.

4.1. IPv4 over IPv6 Encapsulation / Decapsulation

SA46T encapsulates IPv4 packet to IPv6 from stub network to backbone network, and decapsulates IPv6 packet to IPv4 from backbone network to stub network. Figure 5 shows such movement.

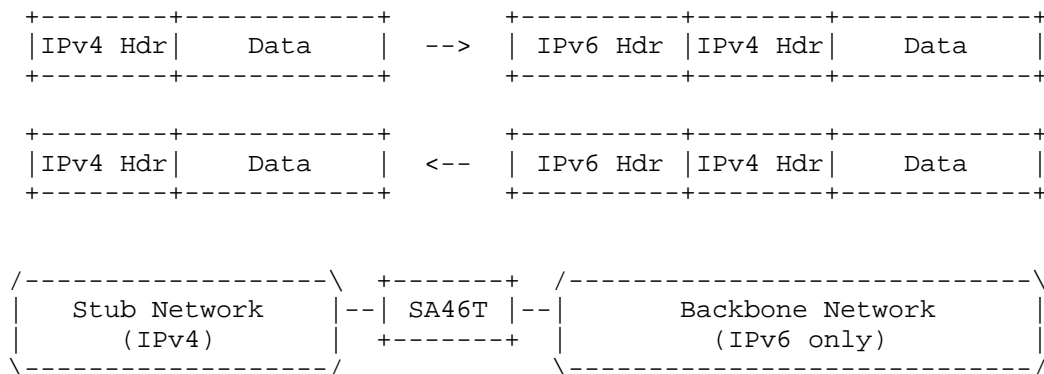


Figure 5

SA46T MUST support tunnel MTU discovery [RFC1853]. When encapsulated IPv6 Packet size exceed path MTU and inner IPv4 packet have the Don't Fragment bit is set, SA46T MUST return ICMP Destination unreachable message with Type3 Code4, fragmentation needed and DS set [RFC0792].

In case IPv6, SA46T just relays IPv6 packet.

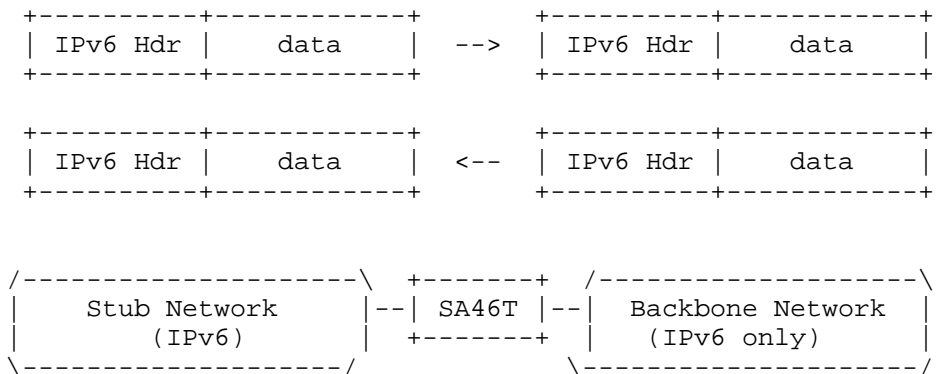


Figure 6

By IPv4 over IPv6 function, SA46T make backbone network to IPv6 only.

4.2. SA46T address architecture

SA46T address is a IPv6 address used in outer IPv6 header which encapsulate IPv4 packet by SA46T.

Figure 7 shows SA46T address architecture

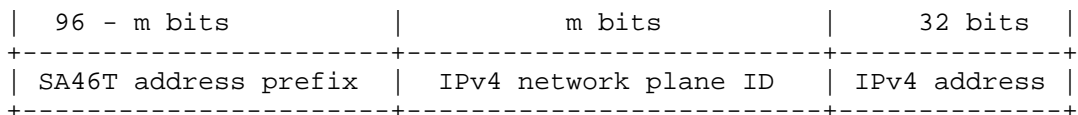


Figure 7

SA46T address consists of three parts as follows.

SA46T address prefix

SA46T address prefix indicates this packet is encapsulated by SA46T and MUST be encapsulated by SA46T. This value is preconfigured to all SA46T in the networks.

IPv4 network plane ID

IPv4 network plane ID is an identifier of IPv4 network stack over IPv6 backbone network. This value is preconfigured depend on the SA46T belong which IPv4 network plane. For more detail see Section 6.

IPv4 address

IPv4 address in inner IPv4 packet.

SA46T address is resolved copying IPv4 address in inner IPv4 packet, and preconfigured values, SA46T prefix and IPv4 network plane ID.

Table 1 shows SA46T IPv4 network plane ID length (m) and number of plane.

m	# of plane
16	65536
32	4294967296
64	18446744073709551616

Table 1

4.3. Route Advertisement

SA46T converts stub network's IPv4 route to SA46T IPv6 route and advertises to backbone network. And reverse direction, SA46T converts SA46T IPv6 route to IPv4 route, that advertises other IPv4 stub networks.

If IPv4 stub network's prefix length is n, the prefix length of SA46T IPv6 route which converts from that IPv4 prefix is $128 - 32 + n$. Table 2 shows detail value.

IPv4 prefix length	SA46T IPv6 prefix length
/8	/104
/16	/112
/24	/120

Table 2

The IPv4 route for stub network is map to SA46T IPv6 route one to one, so number of route of IPv4 is same as number of route of SA46T IPv6 route. Total number of route is same as when backbone network operate dual stack, without SA46T.

In stub network, usual dynamic routing protocol for IPv4 and IPv6 can be used such as RIPv2 [RFC2453], RIPv6 [RFC2080], OSPFv2 [RFC2328], OSPFv3 [RFC2740] and IS-IS [RFC1195][RFC5308]. Similarly, in backbone network, usual dynamic routing protocol for IPv6 can be used such as RIPv6 [RFC2080], OSPFv3 [RFC2740] and IS-IS [RFC5308] .

If want using default route, default SA46T advertise the route [SA46T address prefix/(96 - m)] as default route. If want using different default route by IPv4 network plane ID, default SA46T in IPv4 network plane #1 advertise the route [SA46T address prefix + IPv4 network plane ID #1 / 96] as default route. Figure 15 in Section 9 show the example using default route.

5. SA46T address format

SA46T can be used closely in the backbone network, so SA46T address does not be advertised outside of the backbone network, and IPv6 packet which contains SA46T address does not be forwarded outside of the backbone network.

So, SA46T address format and SA46T address prefix can be decided each backbone network. But for your information, one example is shown as follows. That is based on IPv6 Global Unicast Address.

Of course, SA46T can be used in the Internet, or between the ASs. This case is discussed shortly in Section 5.2.

5.1. IPv6 Global Unicast Address as SA46T address

This example is based on IPv6 Global Unicast Address Format [RFC3587].

Figure 8 shows IPv6 Global Unicast Address Format.

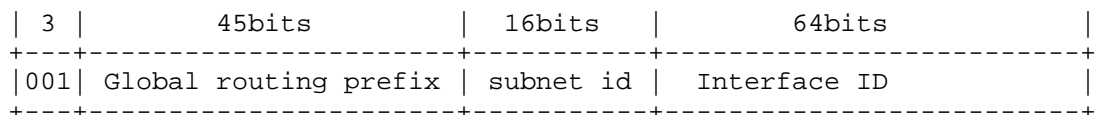


Figure 8

Figure 9 shows SA46T address format using part of IPv6 Global Unicast Address.

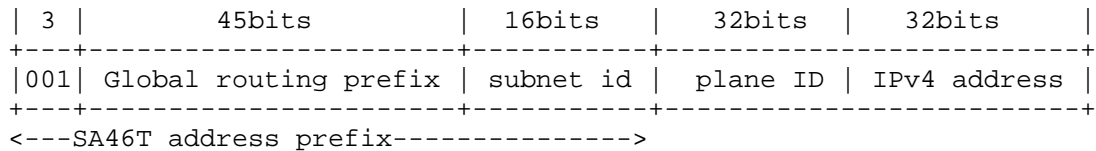


Figure 9

Where:

Global routing prefix

global routing prefix

subnet id

indication for SA46T prefix. Example is 0x5A46.

plane id

IPv4 network plane ID. The value 0 should be for the global IPv4 Internet.

IPv4 address

IPv4 address of inner IPv4 packet

5.2. Global SA46T address format

SA46T can be used in The Internet, or between AS. This is achieved by recognizing SA46T address format as common address. Such address should be Global SA46T address.

Global SA46T address format and prefix requires IANA assignment of IPv6 address prefix. Global SA46T address is proposed in [I-D.draft-matsuhira-sa46t-gaddr].

6. Stacking IPv4 Networks

SA46T can provide VPN like service to stub networks by using different IPv4 network plane ID value. Table 3 shows example of IPv4 network plane ID and its usage.

If backbone network operator provide IPv4 privates network service to Organization A, backbone network operator sets IPv4 network plane ID value =1 to the SA46T which connects stub network of organization A. If there are five stub network of organization A, backbone network operator sets same IPv4 network plane ID = 1, to five SA46Ts which connect stub network of organization A. If there are one hundred stub network of organization B, backbone network operator sets same IPv4 network plane ID = 2, to one hundred SA46Ts which connect stub network of organization B. If a new stub network in organization B join, backbone network operator configures same IPv4 network plane ID = 2, to the new stub network only, which connect stub network of organization B, and no configuration is needed to one hundred SA46Ts which are already connected.

Such configuration, that means same stub network group to same IPv4 network plane ID value, is simple and easy to understand, so, it is expected that possibility of misconfiguration is very low. And also, number of configuration is minimum, that mean, number of configuration is same as number of stub networks, and add new stub network, configure to new one only.

Describe above, SA46T can provide VPN like service, for example, Intranet or extranet. And, after IPv4 global address running out, some service provider may want to reuse IPv4 private address. SA46T can provide such IPv4 private address networks over single IPv6 backbone network. By SA46T, some service providers may reuse IPv4 private address.

IPv4 network plane ID value	usage
0	IPv4 Internet (Global)
1	IPv4 Private network for Organization A (Intranet)
2	IPv4 Private network for Organization B (Intranet)
3	IPv4 Private network for Group A (Extranet)
4	IPv4 Private network for Group B (Extranet)
5	Net10 reuse network for consumer group A (Private address access)
6	Net10 reuse network for consumer group B (Private address access)
7	Net10 reuse network for consumer group C (Private address access)
....

Table 3

7. Redundancy of SA46T

SA46T brings no limit for redundancy. Figure 10 shows such example in case two connection between backbone network and stub network. Number of link between backbone network and stub network is not limited, and different type of link can be used, for example, for wire and wireless.

Configuration of SA46Ts, which connect same stub network, is same. That mean same SA46T prefix and same IPv4 network plane ID value.

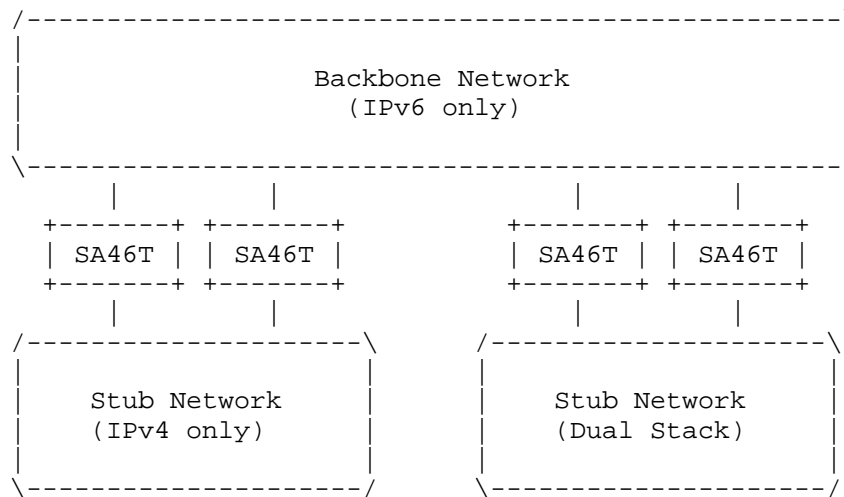


Figure 10

8. Configuration of SA46T and address allocation

Configuration of SA46T require just three information, SA46T address prefix, IPv4 Network plane ID, and prefix length of SA46T route. These information could explain just only one line, "<SA46T address prefix><IPv4 network plane ID>/ prefix length of SA46T route".

When there are N numbers SA46Ts in a certain backbone network, configure one line per SA46T to the N numbers SA46Ts are needed. Total line is just N. If adding new SA46T to the backbone network, configure one line to the new SA46T only is needed, and addition or change does not needed to existing N numbers SA46Ts. Now new 1 line

and total numbers of line is $N+1$.

Static configured tunnel require $N(N-1)$ configurations. So, SA46T needs less configuration than static configured tunnel, especially when value of N is large number.

SA46T require few configuration, so when numbers of SA46T is small, manual configuration may be enough. However, when large number of SA46T needed in big network, configuration via server may useful. For automatic configuration of SA46T, IPv4 address allocation in stub network should consider, both static address allocation and automatic address allocation. In the latter case, using DHCP should be reasonable.

Figure 11 shows example of configuration database for SA46T. As identifier of SA46T, MAC address is used, however, other information may be used.

When stub network connected SA46T is configured with dynamic address, allocate IPv4 address in allocatable IPv4 address block to the stub network side interface of SA46T at startup phase. That is default router address in the stub network. When SA46T receive DHCP request from a host in stub network, DHCP server allocate IP address from allocatable IPv4 address block, and notify IP address of DNS server and IP address of default router.

When stub network connected SA46T is configured with static address, a value of allocatable IPv4 address block should be 0.0.0.0/0 and a value of DNS Server should be 0.0.0.0..

Identifier of SA46T (e.g. MAC addr)	SA46T address prefix + IPv4 network plane ID + prefix length	Allocatable IPv4 address block	DNS Server (IPv4)
Identifier of SA46T (e.g. MAC addr)	SA46T address prefix + IPv4 network plane ID + prefix length	Allocatable IPv4 address block	DNS Server (IPv4)
Identifier of SA46T (e.g. MAC addr)	SA46T address prefix + IPv4 network plane ID + prefix length	Allocatable IPv4 address block	DNS Server (IPv4)
~ :	~ :	~ :	~ :
Identifier of SA46T (e.g. MAC addr)	SA46T address prefix + IPv4 network plane ID + prefix length	Allocatable IPv4 address block	DNS Server (IPv4)

Figure 11

Figure 12 shows timeline diagram of message exchange between SA46T and host in stub network and SA46T configuration server when stub network is configured with dynamic address. Protocol between SA46T and SA46T configuration server including SA46T server discovery may be defined in future.

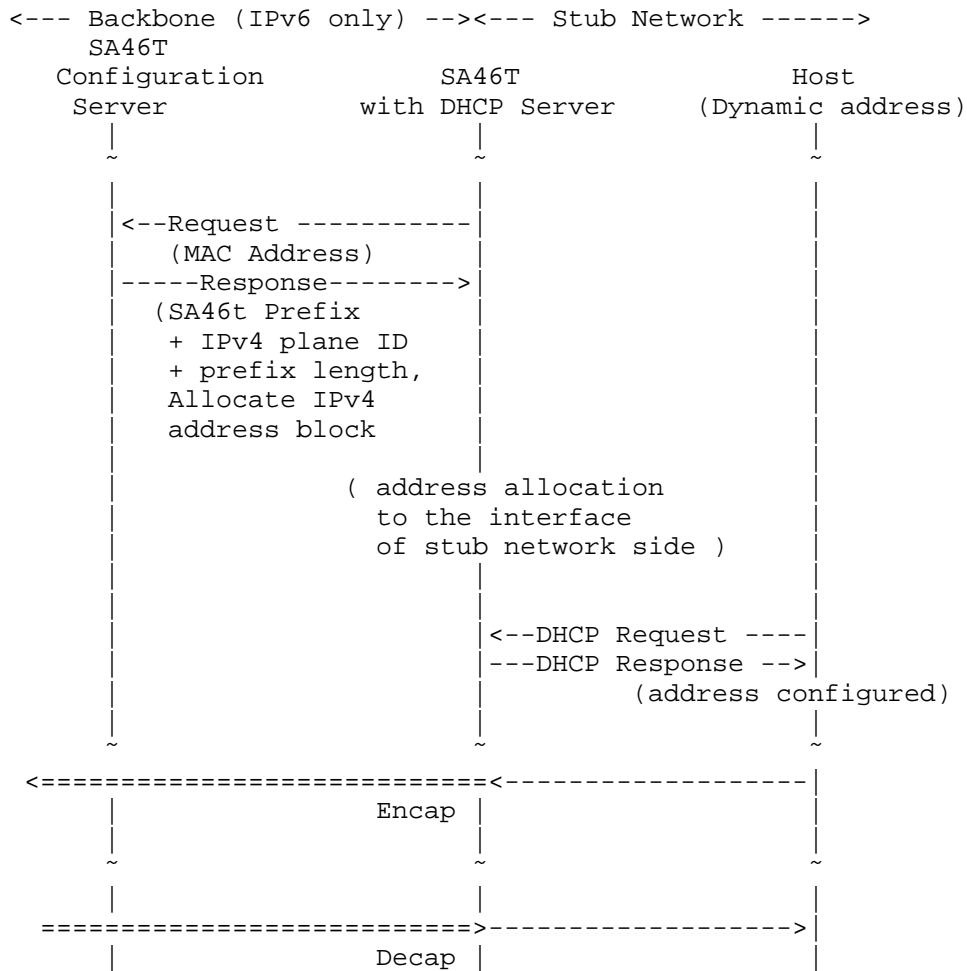


Figure 12

Figure 13 shows timeline diagram of message exchange between SA46T and host in stub network and SA46T configuration server when stub network is configured with static address. Such static address configuration may be used mainly at server zone, so such stub network may be well managed, so SA46T may also configured manually.

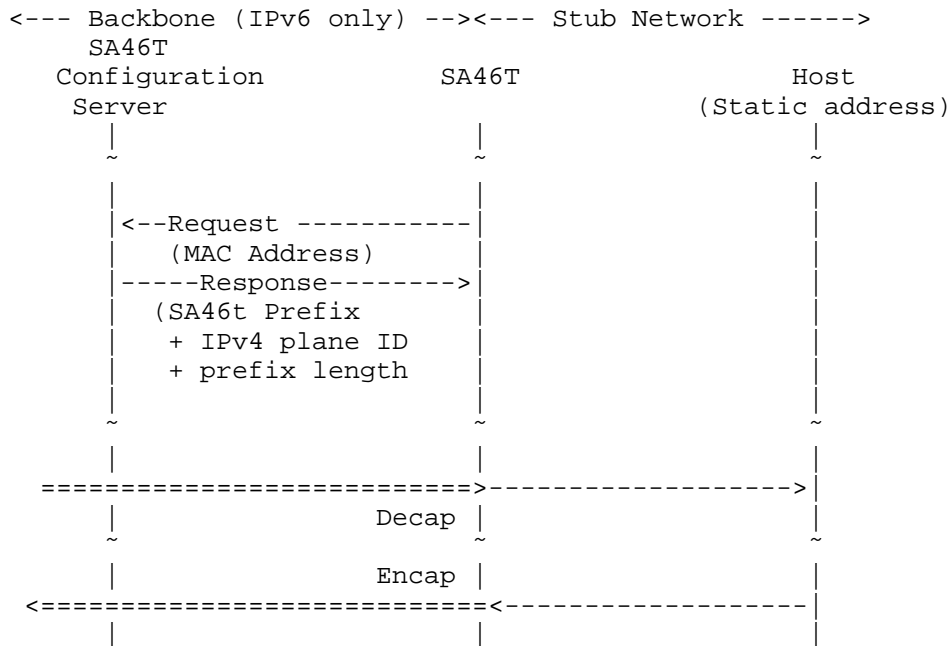


Figure 13

9. Example of SA46T Operation

9.1. Basic SA46T Operation

Figure 14 shows SA46T operation which does not use IPv4 network plane ID. In this example, two stub network is connected to backbone network via SA46T. One stub network is 10.1.1.0/24 sub network, and the other is 10.1.2.0/24 sub network.

When SA46T receives IPv4 route advertisement, then SA46T convert this IPv4 route to IPv6 route by address resolution to SA46T address, and advertise this IPv6 route to backbone network. When SA46T receives IPv6 route advertisements, then SA46T converts this IPv6 route to IPv4 route if this IPv6 route is match SA46T address (same prefix with SA46T), and advertise this IPv4 route to stub network.

In this example. IPv4 route, 10.1.1.0/24 is converted to IPv6 route, <SA46Tprefix>:10.1.1.0/120, and IPv4 route, 10.1.2.0/24 is converted to IPv6 route, <SA46Tprefix>:10.1.2.0/120 at SA46T from stub network to backbone network. And, from backbone network to stub network, IPv6 route, <SA46Tprefix>:10.1.1.0/120 is converted to IPv4 route,

10.1.1.0/24, and IPv6 route, <SA46Tprefix>:10.1.2.0/120 is converted to IPv4 route, 10.1.2.0/24.

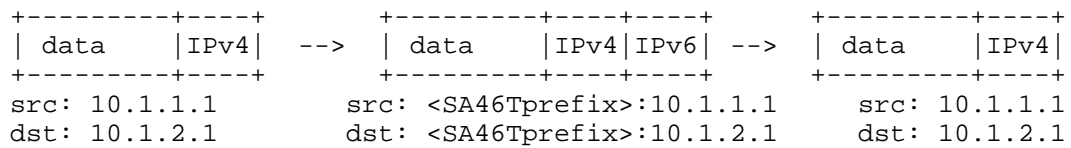
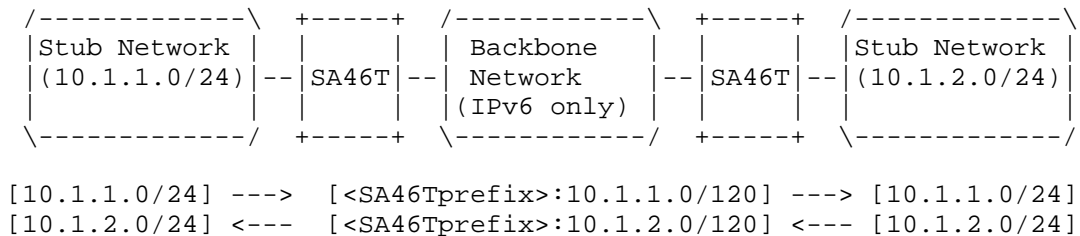


Figure 14

Figure 15 shows the example using default route. Default route is useful in case most packets are routed same path. Typically, access network is one of the example. Although using default route, communication between stub networks can be done. Communication between host 10.1.1.1 and host 10.1.2.1 can be done inside in access network, and does not pass over default SA46T.

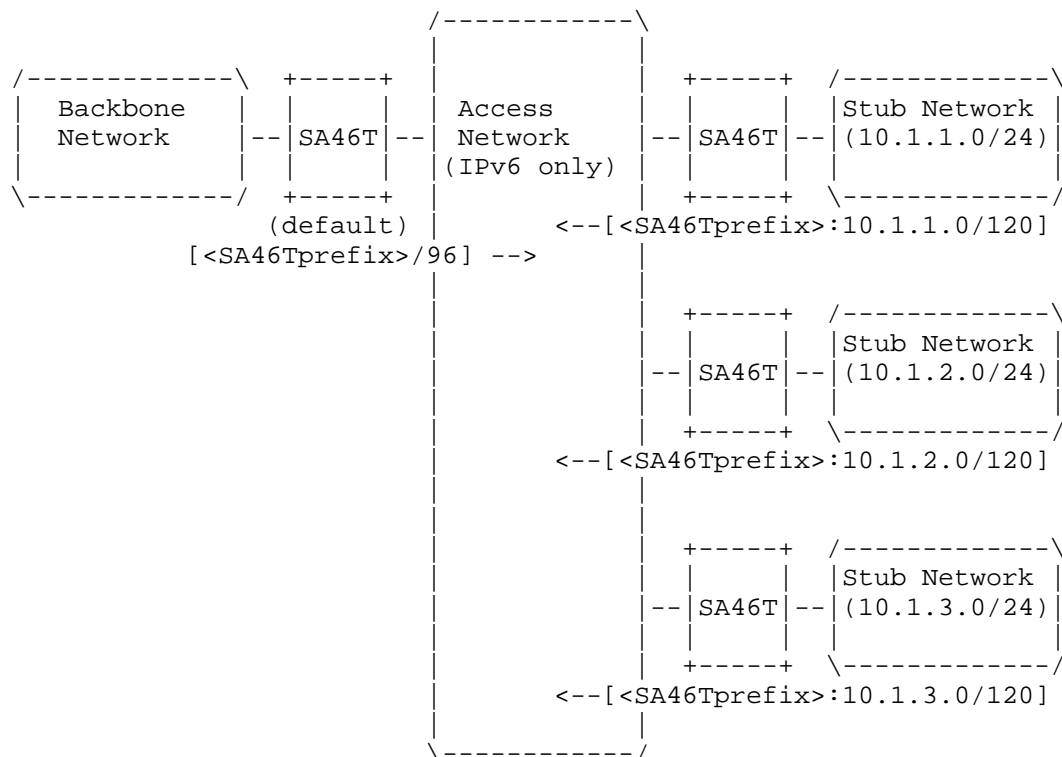


Figure 15

9.2. SA46T Operation with plane ID

Figure 16 shows SA46T operation which uses IPv4 network plane ID. In this example, there are two planes, and two stub network in each plane is connected to backbone network via SA46T. In each plane, one stub network is 10.1.1.0/24 sub network, and the other is 10.1.2.0/24 sub network, that means same IPv4 address is used in different plane.

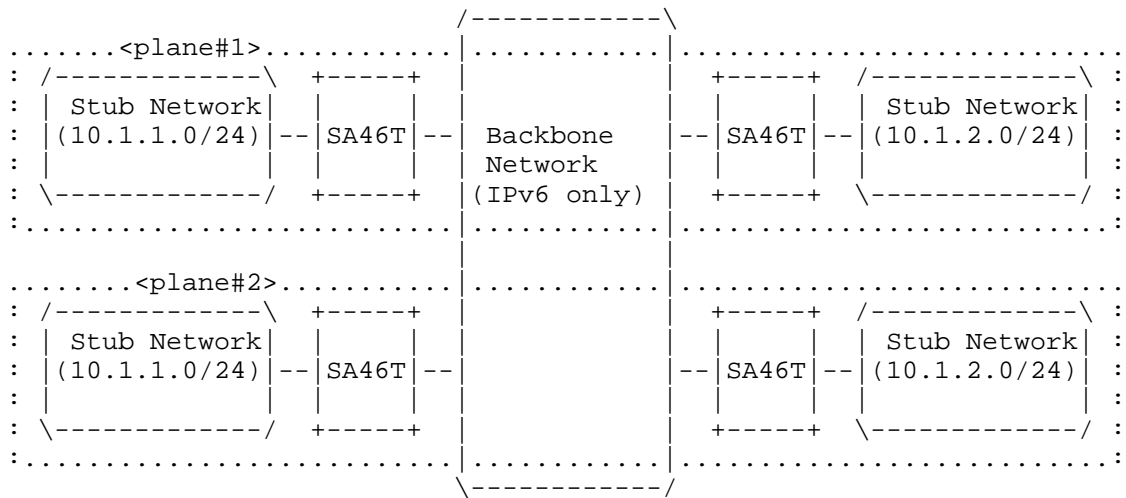
When SA46T receives IPv4 route advertisements, then SA46T converts this IPv4 route to IPv6 route by address resolution to SA46T address, and advertise this IPv6 route to backbone network. When SA46T receives IPv6 route advertisements, then SA46T converts this IPv6 route to IPv4 route if this IPv6 route is match SA46T address (same prefix with SA46T), and advertises this IPv4 route to stub network.

In this example in plane #1. IPv4 route, 10.1.1.0/24 is converted to IPv6 route, <SA46Tprefix>#1>:10.1.1.0/120, and IPv4 route, 10.1.2.0/24 is converted to IPv6 route, <SA46Tprefix>#1>:10.1.2.0/

120 at SA46T from stub network to backbone network. And, from backbone network to stub network, IPv6 route, <SA46Tprefix><#1>:10.1.1.0/120 is converted to IPv4 route, 10.1.1.0/24, and IPv6 route, <SA46Tprefix><#1>:10.1.2.0/120 is converted to IPv4 route, 10.1.2.0/24.

And also, In this example in plane #2. IPv4 route, 10.1.1.0/24 is converted to IPv6 route, <SA46Tprefix><#2>:10.1.1.0/120, and IPv4 route, 10.1.2.0/24 is converted to IPv6 route, <SA46Tprefix><#2>:10.1.2.0/120 at SA46T from stub network to backbone network. And, from backbone network to stub network, IPv6 route, <SA46Tprefix><#2>:10.1.1.0/120 is converted to IPv4 route, 10.1.1.0/24, and IPv6 route, <SA46Tprefix><#2>:10.1.2.0/120 is converted to IPv4 route, 10.1.2.0/24.

In IPv6 space, address <SA46Tprefix><#1>:10.1.1.1 and address <SA46Tprefix><#2>:10.1.1.1 are different address, route <SA46Tprefix><#1>:10.1.1.0/120 and route <SA46Tprefix><#2>:10.1.1.0/120 are different route, although in IPv4 space, address 10.1.1.1 in plane #1 and 10.1.1.1 in plane#2 are same address, route 10.1.1.0/24 in plane#1 and route 10.1.1.0/24 in plane#2 are same route.



<<plane #1>>

```

[10.1.1.0/24] ---> [<SA46Tprefix><#1>:10.1.1.0/120] ---> [10.1.1.0/24]
[10.1.2.0/24] <--- [<SA46Tprefix><#1>:10.1.2.0/120] <--- [10.1.2.0/24]

```

```

+-----+-----+ +-----+-----+-----+ +-----+-----+
| data   | IPv4 | --> | data   | IPv4 | IPv6 | --> | data   | IPv4 |
+-----+-----+ +-----+-----+-----+ +-----+-----+
src: 10.1.1.1      src: <SA46Tprefix><#1>:10.1.1.1      src: 10.1.1.1
dst: 10.1.2.1      dst: <SA46Tprefix><#1>:10.1.2.1      dst: 10.1.2.1

```

<<plane#2>>

```

[10.1.1.0/24] ---> [<SA46Tprefix><#2>:10.1.1.0/120] ---> [10.1.1.0/24]
[10.1.2.0/24] <--- [<SA46Tprefix><#2>:10.1.2.0/120] <--- [10.1.2.0/24]

```

```

+-----+-----+ +-----+-----+-----+ +-----+-----+
| data   | IPv4 | --> | data   | IPv4 | IPv6 | --> | data   | IPv4 |
+-----+-----+ +-----+-----+-----+ +-----+-----+
src: 10.1.1.1      src: <SA46Tprefix><#2>:10.1.1.1      src: 10.1.1.1
dst: 10.1.2.1      dst: <SA46Tprefix><#2>:10.1.2.1      dst: 10.1.2.1

```

Figure 16

Figure 17 shows the example using default route with IPv4 network plane. In this case, default SA46T may configure different by each IPv4 network plane.

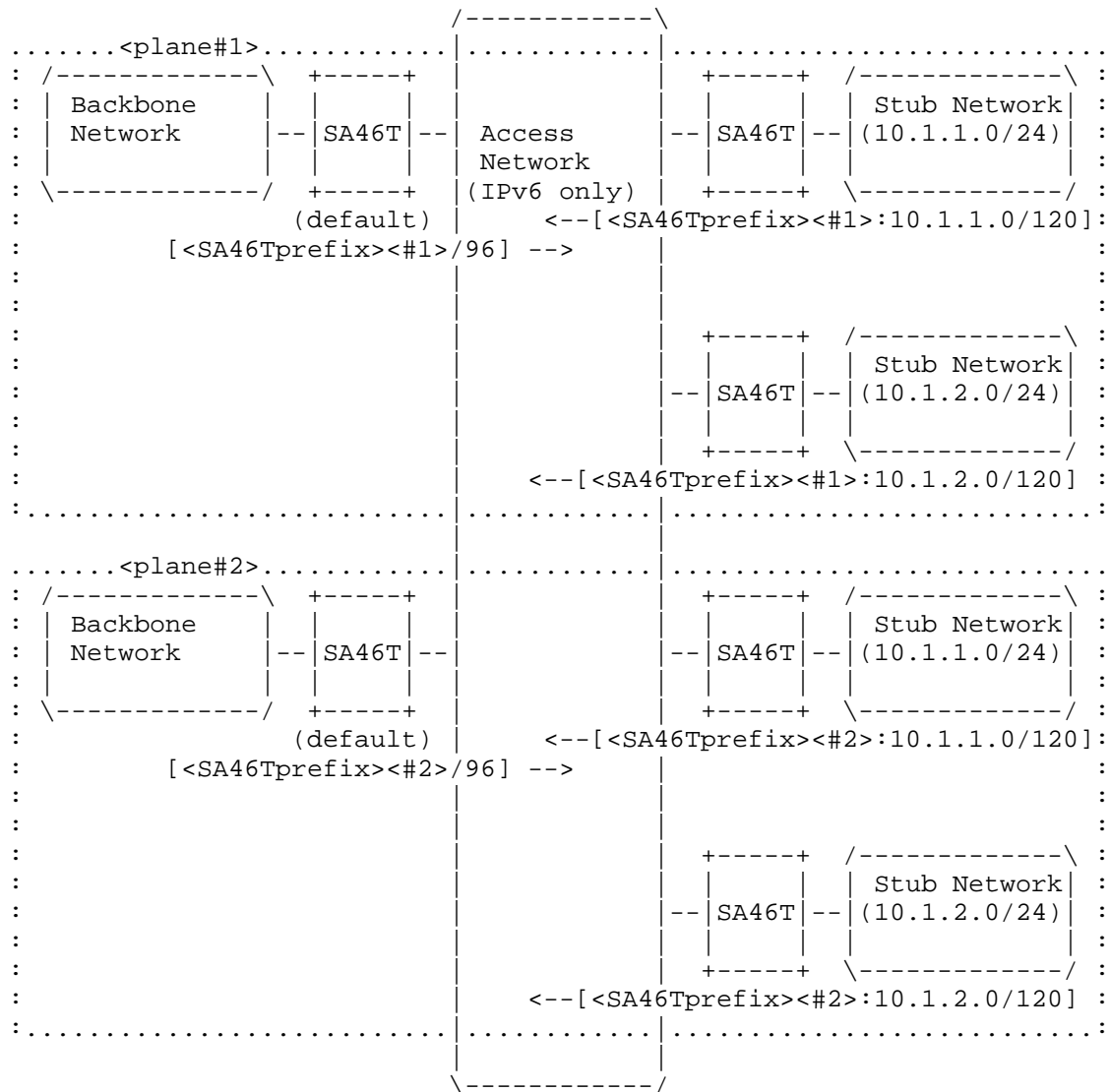


Figure 17

10. Characteristic

SA46T has following useful characteristics.

- o Reduce backbone network operation cost with IPv6 single stack (at least less than Dual Stack)
- o Can allocate IPv4 address to stub networks, which used in backbone network before installing SA46T
- o Less configuration
- o No need for special protocol
- o No dependent Layer 2 network
- o Can Stack IPv4 Private networks
- o Easy stop IPv4 operation in stub network for future (just remove SA46T)
- o Provide redundancy

11. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

12. Security Considerations

SA46T use automatic Encapsulation / Decapsulation technologies. Security consideration related tunneling technologies are discussed in RFC2893[RFC2893], RFC2267[RFC2267], etc.

13. Acknowledgements

This document is based on Naoki Matsuhira's original ideas and an individual effort of the author.

Review and encouragement have been provided by many peoples. Particular Akira Kato at WIDE Project / Keio University and Masanobu Katoh at Fujitsu in initial stage. And many discussions and assists are provided from Toshiya Asaba, Osamu Nakamura, Yoshiki Ishida, Ichiro Mizukoshi, Noriyuki Shigechika, Miya Kohno, Yoshinobu Matsuzaki, Akira Nakagawa. And comments and discussions are provided in IETF meeting from Fred Baker, Brian Carpenter, Randy Bush, Dave Thaler and Alain Duland. If there is a comment not refrected, it is

surely because of my English language capability, and the author still want reflect it include missing.

The author would like to thank all above people, and others discussed with in WIDE project meeting and inside Fujitsu.

Originally, SA46T is an abbreviation for "Stateless Automatic IPv4 over IPv6 Tunneling". Now, SA46T is an abbreviation for "Stateless Automatic IPv4 over IPv6 Encapsulation / Decapsulation Technology". This change was made in response to the indication from the softwire WG chair at 4th softwire interim meeting in September 2011.

14. References

14.1. Normative References

- [I-D.draft-matsuhira-sa46t-gaddr]
Matsuhira, N., "Stateless Automatic IPv4 over IPv6 Encapsulation / Decapsulation Technology: Global SA46T Address Format", January 2014.
- [RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, September 1981.
- [RFC1853] Simpson, W., "IP in IP Tunneling", RFC 1853, October 1995.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3587] Hinden, R., Deering, S., and E. Nordmark, "IPv6 Global Unicast Address Format", RFC 3587, August 2003.

14.2. References

- [RFC1195] Callon, R., "Use of OSI IS-IS for routing in TCP/IP and dual environments", RFC 1195, December 1990.
- [RFC2080] Malkin, G. and R. Minnear, "RIPng for IPv6", RFC 2080, January 1997.
- [RFC2267] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", RFC 2267, January 1998.
- [RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998.
- [RFC2453] Malkin, G., "RIP Version 2", STD 56, RFC 2453,

November 1998.

- [RFC2740] Coltun, R., Ferguson, D., and J. Moy, "OSPF for IPv6", RFC 2740, December 1999.
- [RFC2893] Gilligan, R. and E. Nordmark, "Transition Mechanisms for IPv6 Hosts and Routers", RFC 2893, August 2000.
- [RFC5308] Hopps, C., "Routing IPv6 with IS-IS", RFC 5308, October 2008.

Appendix A. Test implementation of SA46T

Test implementation of SA46T is developed for evaluation the SA46T technology. This implementation is developed as module in kernel space of CentOS. The amount of development is about 300 step with C language.

Appendix B. SA46T experiments

B.1. WIDE camp at Sept 2010

SA46T implementation is tested at WIDE camp in 4.5 days at September 2010. Attendees of WIDE camp served SA46T service via Wireless LAN. SA46T provide both IPv4 and IPv6. IPv4 packets are encapsulated and decapsulated in camp net, that mean this test is in LAN environments. This time single IPv4 plane was used.

About 200 peoples joins this experiments and 275 clients are used, include Windows, MacOS, Linux, FreeBSD, iPhone and iPod Touch, etc. IPv4 address is allocated via DHCP. There are no change in clients, servers, and network equipment, just add SA46T. Total, four SA46T boxes were used in this experiments.

SA46T work fine and very stable.

B.2. NICT JGN2Plus Testbed at Feb 2011

SA46T implementation is tested at NICT JGN2Plus testbed at February 2011. This test is held at WAN environments. SA46T is setted up at Sapporo, Osaka, Okayama and Okinawa in Japan and Thai, and carry HDTV Live Stream and 3D HDTV Live stream. Experimental period is about an one month. Total, five SA46T boxes were used in this experiments.

In JGN2Plus, OSPFv3 was used, and BGP4+ is used for peering with Thai.

This time, single IPv4 plane was used too.

SA46T work fine and very stable, too.

B.3. Some corporate network

SA46Ts are installed some corporate network. This installation is done with secrets basically, that mean, nobody know SA46T was installed, and if there are some trouble, someone claim or report the problem.

After few month trial, there was no problem.

B.4. Interop 2011 Tokyo at Jun 2011

SA46T is demonstrated at Interop 2011 Tokyo at June 2011.

At this time, three planes were used. Plane #0 is used for Internet access, using IPv4 Global address. Visitor can have a experiments with SA46T from the cables which connected to SA46T in access corner. Plane #1 is used for closed network, such like between Data Center network and enterprise network. In this plane, private addresses are used. Plane #2 is used for video streaming. In this plane, same private addresses which used in Plane#1 are used by intention. And this plane in Interop ShowNet and NICT and Thai were connected.

Total, nine SA46T boxes are used in this demonstration.

About 128,000 peoples visit in this event, and see many demonstration include SA46T.

SA46T work fine and very stable, too.

Author's Address

Naoki Matsuhira
Fujitsu Limited
1-1, Kamikodanaka 4-chome, Nakahara-ku
Kawasaki, 211-8588
Japan

Phone: +81-44-754-3466
Fax:
Email: matsuhira@jp.fujitsu.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 13, 2012

B. Sarikaya
F. Xia
Huawei USA
T. Lemon
Nominum
February 10, 2012

DHCPv6 Prefix Delegation in Long Term Evolution (LTE) Networks
draft-sarikaya-v6ops-prefix-delegation-11.txt

Abstract

As interest on IPv6 deployment is increasing in cellular networks several migration issues are being raised and IPv6 prefix management is the one addressed in this document. Based on the idea that DHCPv6 servers can manage prefixes, we address prefix management issues such as the access router offloading delegation and release tasks of the prefixes to a DHCPv6 server using DHCPv6 Prefix Delegation. The access router first requests a prefix for an incoming mobile node from the DHCPv6 server. The access router may next do stateless or stateful address allocation to the mobile node, e.g. with a Router Advertisement or using DHCP. We also describe prefix management using Authentication Authorization and Accounting servers.

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 August 13, 2012.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal

Provisions Relating to IETF Documents
(<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Terminology and Acronyms	4
3. Prefix Delegation Using DHCPv6	5
3.1. Prefix Request Procedure for Stateless Address Configuration	5
3.2. Prefix Request Procedure for Stateful Address Configuration	7
3.3. MN as Requesting Router in Prefix Delegation	8
3.4. Prefix Release Procedure	8
3.5. Miscellaneous Considerations	9
3.5.1. How to Generate IAID	9
3.5.2. Policy to Delegate Prefixes	10
4. Prefix Delegation Using RADIUS and Diameter	10
5. Security Considerations	11
6. IANA Considerations	11
7. Acknowledgements	11
8. Informative References	12
Authors' Addresses	13

1. Introduction

Figure 1 illustrates the key elements of a typical cellular access network. In a Long Term Evolution (LTE) network, access router is the packet data network (PDN) gateway [ThreeGPP23401].

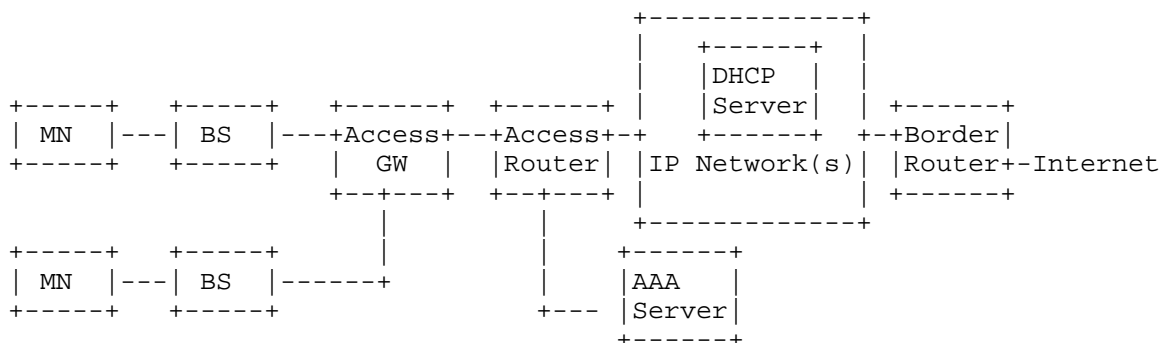


Figure 1: Key elements of a typical cellular network

Mobile node (MN) attaches to a base station (BS) through LTE air interface. A BS manages connectivity of UEs and extends connections to an Access Gateway (GW), e.g. the serving gateway (S-GW) in an LTE network. The access gateway and the Access Router (AR) are connected with an IP network. The access router is the first hop router of MNs and it is in charge of address/prefix management.

Access router is connected to an IP network which is owned by the operator which is connected to the public Internet via a Border Router. The network contains servers for subscriber management including Quality of Service, billing and accounting as well as Dynamic Host Configuration Protocol (DHCP) server [RFC6342].

As to IPv6 addressing, because mobile network links are point-to-point (p2p) Per-MN interface prefix model is used [RFC3314], [RFC3316]. In Per-MN interface prefix model, prefix management is an issue.

When an MN attaches an AR, the AR requests one or more prefixes for the MN. When the MN detaches the AR, the prefixes should be released. When the MN becomes idle, the AR should hold the prefixes allocated.

This document describes how to use DHCPv6 Prefix Delegation (PD) in mobile networks such as networks based on standards developed by the 3rd Generation Partnership Project (3GPP) and it could easily be adopted to Worldwide Interoperability for Microwave Access (WiMAX)

Forum as well. In view of migration to IPv6, the number of mobile nodes connected to the network at a given time may become very high. Traditional techniques such as prefix pools are not scalable. In such cases DHCPv6 PD becomes the viable approach to take.

The techniques described in this document have not been approved either by the IETF or by 3GPP, except what is described below in Section 3.3. This document is not a standard or best current practice. This document is published only as a possibility for consideration by operators.

This document is useful when address space needs to be managed by DHCPv6-PD. There are obviously other means of managing address space, including having the AR track internally what address space is used by what mobile.

2. Terminology and Acronyms

3GPP 3rd Generation Partnership Project

AAA Authentication Authorization and Accounting

AR Access Router

BS Base Station

DHCP Dynamic Host Control Protocol

E-UTRAN Evolved Universal Terrestrial Radio Access Network

GPRS General Packet Radio Service

LTE Long Term Evolution

MN Mobile node

PDN Packet data network

PD Prefix Delegation

p2p Point-to-point

Serving Gateway S-GW

WiMAX Worldwide Interoperability for Microwave Access

3. Prefix Delegation Using DHCPv6

Access router refers to the cellular network entity that has DHCP Client. According to [ThreeGPP23401] DHCP Client is located in PDN Gateway. So AR is the PDN Gateway in LTE architecture.

3.1. Prefix Request Procedure for Stateless Address Configuration

There are two function modules in the AR, DHCP Client and DHCP Relay. DHCP messages should be relayed if the AR and a DHCP server are not connected directly, otherwise DHCP relay function in the AR is not necessary. Figure 2 illustrates the scenario that the AR and the DHCP Server aren't connected directly:

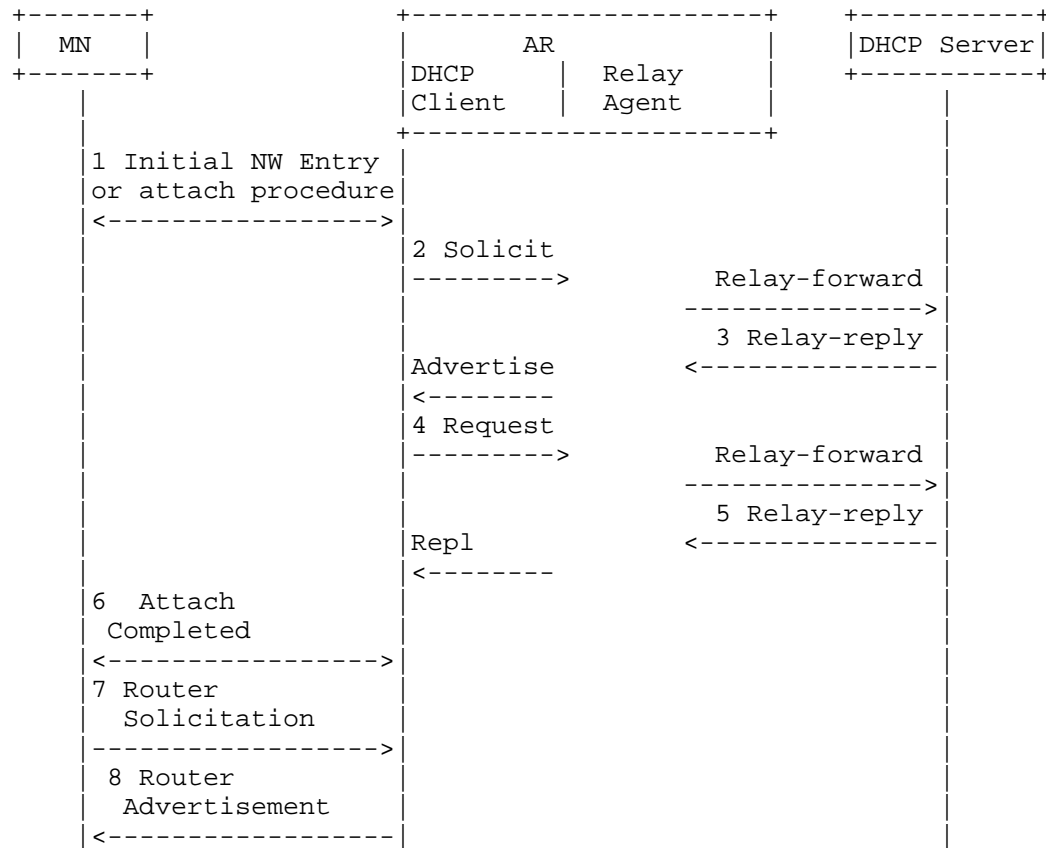


Figure 2: Prefix request

1. An MN (UE=User Equipment in 3GPP) performs initial network entry and authentication procedures, a.k.a. attach procedure.
2. On successful completion of Step 1, the AR initiates DHCP Solicit procedure to request prefixes for the MN. The DHCP Client in AR creates and transmits a Solicit message as described in sections 17.1.1, "Creation of Solicit Messages" and 17.1.2, "Transmission of Solicit Messages" of [RFC3315]. The DHCP Client in AR that supports DHCPv6 Prefix Delegation [RFC3633] creates an Identity Association for Prefix Delegation (IA_PD) and assigns it an Identity Association Identifier (IAID). The client must include the IA_PD option in the Solicit message. DHCP Client as Requesting Router must set prefix-length field to a value less than, e.g. 48 or equal to 64 to request a /64 prefix. Next, the Relay Agent in AR sends Relay-Forward message to the DHCP Server encapsulating Solicit message.
3. The DHCP server sends an Advertise message to the AR in the same way as described in section 17.2.2, "Creation and transmission of Advertise messages" of [RFC3315]. Advertise message with IA_PD shows that the DHCP server is capable of delegating prefixes. This message is received encapsulated in Relay-Reply message by the Relay Agent in AR and sent as Advertise message to the DHCP Client in AR.
4. The AR (DHCP Client and Relay Agent) uses the same message exchanges as described in section 18, "DHCP Client-Initiated Configuration Exchange" of [RFC3315] and [RFC3633] to obtain or update prefixes from the DHCP server. The AR (DHCP Client and Relay Agent) and the DHCP server use the IA_PD Prefix option to exchange information about prefixes in much the same way as IA Address options are used for assigned addresses. This is accomplished by the AR sending a DHCP Request message and the DHCP server sending a DHCP Reply message.
5. AR stores the prefix information it received in the Reply message.
6. A connection between MN and AR is established and the link becomes active. This step completes the PDP Context Activation Procedure in UMTS and PDN connection establishment in LTE networks.
7. The MN may send a Router Solicitation message to solicit the AR to send a Router Advertisement message.
8. The AR advertises the prefixes received in IA_PD option to MN with router advertisement (RA) once the PDP Context/PDN connection is established or in response to Router Solicitation message sent from the MN.

4-way exchange between AR as requesting router (RR) and DHCP server as delegating router (DR) in Figure 2 may be reduced into a two message exchange using the Rapid Commit option [RFC3315]. DHCP Client in AR acting as RR includes a Rapid Commit option in the

Solicit message. DR then sends a Reply message containing one or more prefixes.

3.2. Prefix Request Procedure for Stateful Address Configuration

Stateful address configuration requires a different architecture than shown in Figure 2. There are two function modules in the AR, DHCP Server and DHCP Client.

After the initial attach is completed, a connection to the AR is established for the MN. DHCP Client function at the AR as requesting router and DHCP server as delegating router follow Steps 2 through 5 of the procedure shown in Figure 2 to get the new prefix for this interface of MN from IA_PD Option exchange defined in [RFC3633].

DHCPv6 client at the MN sends DHCP Request to AR. DHCP Server function at the AR must use the IA_PD option received in DHCP PD exchange to assign an address to MN. IA_PD option must contain the prefix. AR sends DHCP Reply message to MN containing IA address option (IAADDR). Figure 3 shows the message sequence.

MN configures its interface with the address assigned by DHCP server in DHCP Reply message.

In Figure 3 AR may be the home gateway of a fixed network to which MN gets connected during MN's handover.

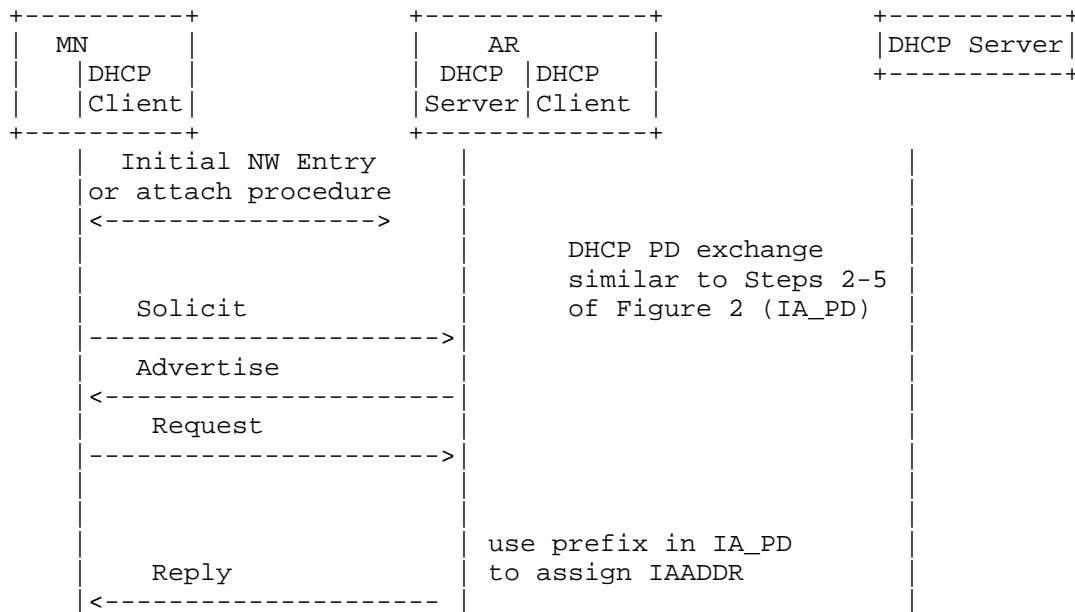


Figure 3: Stateful Address Configuration Following PD

3.3. MN as Requesting Router in Prefix Delegation

AR may use DHCPv6 prefix delegation exchange to get a delegated prefix shorter than /64 by setting prefix-length field to a value less than 64, e.g. 56 to get a /56 prefix. Each newly attaching MN first goes through the steps in Figure 2 in which AR requests a shorter prefix to establish a default connection with the MN.

MN may next request additional prefixes (/64 or shorter) from the AR using DHCPv6 prefix delegation where MN is the requesting router and AR is the delegating router [RFC6459], Section 5.3.1.2.6 in [ThreeGPP23401]. In this case the call flow is similar to Figure 3. Solicit message must include the IA_PD option with prefix-length field set to 64. MN may request more than one /64 prefixes. AR as delegating router must delegate these prefixes excluding the prefix assigned to the default connection.

3.4. Prefix Release Procedure

Prefixes can be released in two ways, prefix aging or DHCP release procedure. In the former way, a prefix should not be used by an MN when the prefix ages, and the DHCP Server can delegate it to another MN. A prefix lifetime is delivered from the DHCPv6 server to the MN

through DHCP IA_PD Prefix option [RFC3633] and RA Prefix Information option [RFC4861]. Figure 4 illustrates how the AR releases prefixes to a DHCP Server which isn't connected directly:

1. An MN detachment signaling, such as switch-off or handover, triggers prefix release procedure.
2. The AR initiates a Release message to give back the prefixes to the DHCP server.
3. The server responds with a Reply message, and then the prefixes can be reused by other MNs.

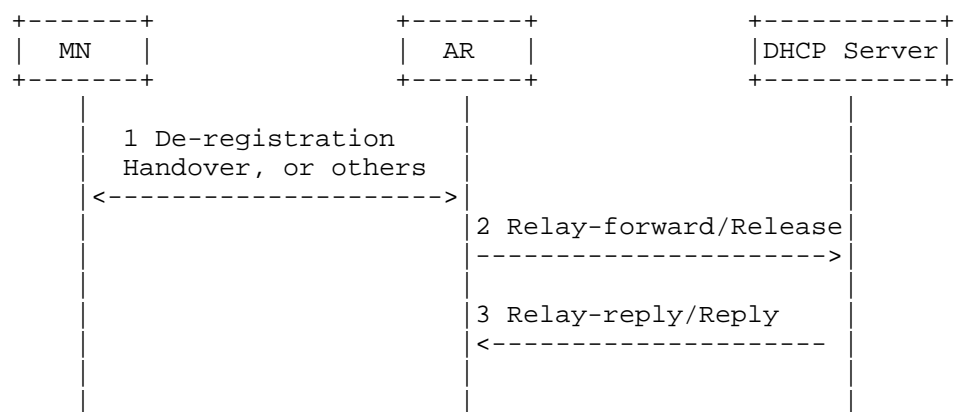


Figure 4: Prefix Release

3.5. Miscellaneous Considerations

3.5.1. How to Generate IAID

IAID is 4 bytes in length and should be unique in an AR scope. Prefix table should be maintained. Prefix table contains IAID, MAC address and the prefix(es) assigned to MN. In LTE networks, International Mobile station Equipment Identity (IMEI) uniquely identifies MN's interface and thus corresponds to the MAC address. MAC address of the interface should be stored in the prefix table and this field is used as the key for searching the table.

IAID should be set to Start_IAID, an integer of 4 octets. The following IAID generation algorithm is used:

1. Set this IAID value in IA_PD Prefix Option. Request prefix for this MN as in Section 3.1 or Section 3.2.
2. Store IAID, MAC address and the prefix(es) received in the next entry of the prefix table.

3. Increment IAID.

Prefix table entry for an MN that hands over to another AR must be removed. IAID value is released to be reused.

3.5.2. Policy to Delegate Prefixes

In point-to-point links, if /64 prefixes of all the MNs connected to one or more ARs are broadcast dynamically upstream as the route information this causes high routing protocol traffic (IGP, OSPF, etc.) due to Per-MN interface prefixes. There are two solutions this problem. One is to use static configuration, which would be preferable in many cases. No routing protocols are needed, because each AR has a known piece of address space. If the DHCP servers know this space, too, then they will assign from that space to a particular AR.

The other method is to use route aggregation. For example, each AR can be assigned a /48 or /32 prefix (aggregate prefix, aka service provider common prefix) while each interface of MN can be assigned a /64 prefix. The /64 prefix is an extension of /48 one, for example, an AR's /48 prefix is 2001:DB8:0::/48, an interface of MN is assigned 2001:DB8:0:2::/64 prefix. The border router (BR) in Figure 1 may be manually configured to broadcast only individual AR's /48 or /32 prefix information to Internet.

4. Prefix Delegation Using RADIUS and Diameter

In the initial network entry procedure Figure 2, AR as Remote Authentication Dial In User Service (RADIUS) client sends Access-Request message with MN information to RADIUS server. If the MN passes the authentication, the RADIUS server may send Access-Accept message with prefix information to the AR using Framed-IPv6-Prefix attribute. AAA server also provides routing information to be configured for MN on the AR using Framed-IPv6-Route attribute. Using such a process AR can handle initial prefix assignments to MNs but managing lifetime of the prefixes is totally left to the AR. Framed-IPv6-Prefix is not designed to support delegation of IPv6 prefixes. For this Delegated-IPv6-Prefix attribute can be used which is discussed next.

[RFC4818] defines a RADIUS attribute Delegated-IPv6-Prefix that carries an IPv6 prefix to be delegated. This attribute is usable within either RADIUS or Diameter. [RFC4818] recommends the delegating router to use AAA server to receive the prefixes to be delegated using Delegated-IPv6-Prefix attribute/AVP.

DHCP server as the delegating router in Figure 2 may send an Access-Request packet containing Delegated-IPv6-Prefix attribute to the RADIUS server to request prefixes. In the Access-Request message, the delegating router may provide a hint that it would prefer a prefix, for example, a /48 prefix. As the RADIUS server is not required to honor the hint, the server may delegate longer prefix, e.g. /56 or /64 in an Access-Accept message containing Delegated-IPv6-Prefix attribute [RFC4818]. The attribute can appear multiple times when RADIUS server delegates multiple prefixes to the delegating router. The delegating router sends the prefixes to the requesting router using IA_PD Option and AR as RR uses them for MN's as described in Section 3.

When Diameter is used, DHCP server as the delegating router in Figure 2 sends AA-Request message. AA-Request message may contain Delegated-IPv6-Prefix AVP. Diameter server replies with AA-Answer message. AA-Answer message may contain Delegated-IPv6-Prefix AVP. The AVP can appear multiple times when Diameter server assigns multiple prefixes to MN. The Delegated-IPv6-Prefix AVP may appear in an AA-Request packet as a hint by the AR to the Diameter server that it would prefer a prefix, for example, a /48 prefix. Diameter server may delegate in an AA-Answer message with a /64 prefix which is an extension of the /48 prefix. As in the case of RADIUS, the delegating router sends the prefixes to the requesting router using IA_PD Option and AR as RR uses them for MN's as described in Section 3.

5. Security Considerations

This draft introduces no additional messages. Comparing to [RFC3633], [RFC2865] and [RFC3588] there is no additional threats to be introduced. DHCPv6, RADIUS and Diameter security procedures apply.

6. IANA Considerations

None.

7. Acknowledgements

We are grateful to Suresh Krishnan, Hemant Singh, Qiang Zhao, Ole Troan, Qin Wu, Jouni Korhonen, Cameron Byrne, Brian Carpenter, Jari Arkko and Jason Lin who provided in depth reviews of this document that have led to several improvements.

8. Informative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.
- [RFC3314] Wasserman, M., "Recommendations for IPv6 in Third Generation Partnership Project (3GPP) Standards", RFC 3314, September 2002.
- [RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.
- [RFC3316] Arkko, J., Kuijpers, G., Soliman, H., Loughney, J., and J. Wiljakka, "Internet Protocol Version 6 (IPv6) for Some Second and Third Generation Cellular Hosts", RFC 3316, April 2003.
- [RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G., and J. Arkko, "Diameter Base Protocol", RFC 3588, September 2003.
- [RFC3633] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, December 2003.
- [RFC4818] Salowey, J. and R. Droms, "RADIUS Delegated-IPv6-Prefix Attribute", RFC 4818, April 2007.
- [RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.
- [RFC6342] Koodli, R., "Mobile Networks Considerations for IPv6 Deployment", RFC 6342, August 2011.
- [RFC6459] Korhonen, J., Soininen, J., Patil, B., Savolainen, T., Bajko, G., and K. Iisakkila, "IPv6 in 3rd Generation Partnership Project (3GPP) Evolved Packet System (EPS)", RFC 6459, January 2012.
- [ThreeGPP23401] "3GPP TS 23.401 V11.0.0, General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access (Release 11).",

2011.

Authors' Addresses

Behcet Sarikaya
Huawei USA
5340 Legacy Dr.
Plano, TX 75074

Email: sarikaya@ieee.org

Frank Xia
Huawei USA
1700 Alma Dr. Suite 500
Plano, TX 75075

Phone: +1 972-509-5599
Email: xiayangsong@huawei.com

Ted Lemon
Nominum
2000 Seaport Blvd
Redwood City, CA 94063

Phone:
Email: mellon@nominum.com

Homenet
Internet-Draft
Intended status: Informational
Expires: May 3, 2012

E. Vyncke
A. Yourtchenko
M. Townsley
Cisco Systems
October 31, 2011

Advanced Security for IPv6 CPE
draft-vyncke-advanced-ipv6-security-03.txt

Abstract

This document describes how an IPv6 residential Customer Premise Equipment (CPE) can leverage modern security techniques to have strong security, while retaining as much of the end-to-end reachability of IPv6 as possible.

It is a re-submission in the framework of the HOMENET working group. The reputation part of this document should leverage the work done in the REPUTE working group of the Application are.

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 May 3, 2012.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Threats	3
3. Overview	4
3.1. Rules for Security Policy	5
3.2. Security Analysis	6
4. IANA Considerations	7
5. Security Considerations	7
6. Acknowledgements	8
7. References	8
7.1. Normative References	8
7.2. Informative References	8
Authors' Addresses	8

1. Introduction

Internet access in residential IPv4 deployments generally consist of a single IPv4 address provided by the service provider for each home. Residential CPE then translates the single address into multiple private addresses allowing more than one device in the home, but at the cost of losing end-to-end reachability. IPv6 allows all devices to have a unique, global, IP address, restoring end-to-end reachability directly between any device. Such reachability is very powerful for ubiquitous global connectivity, and is often heralded as one of the significant advantages to IPv6 over IPv4. Despite this, concern about exposure to inbound packets from the IPv6 Internet (which would otherwise be dropped by the address translation function if they had been sent from the IPv4 Internet) remain. This document describes firewall functionality for an IPv6 CPE which departs from the "simple security" model described in [RFC6092]. The intention is to provide an example of a security model which allows most traffic, including incoming unsolicited packets and connections, to traverse the CPE unless the CPE identifies the traffic as potentially harmful based on a set of signatures (and other correlation data and heuristics) that are kept up to date on a regular basis. The computational resources necessary to support some, not all, functionalities of this model are likely more intensive than those described in [RFC6092], but are easily within the realm of what is commonly available in 2011 on medium to high-end network based firewall systems for small and medium businesses, or host-based commercial firewalls that run on laptop and desktop PCs. This set of techniques is also known as Universal Threat Mitigation (UTM).

2. Threats

For a typical residential network connected to the Internet over a broadband connection, the threats can be classified into:

- o denial of service by packet flooding: overwhelming either the access bandwidth or the bandwidth of a slower link in the residential network (like a slow home automation network) or the CPU power of a slow IPv6 host (like networked thermostat or any other sensor type nodes)
- o denial of service by service requests: like sending print jobs from the Internet to an ink jet printer until the ink cartridge is empty or like filing some file server with junk data
- o unauthorized use of services: like accessing a webcam or a file server which are open to anonymous access within the residential network but should not be accessed freely and anonymously from

outside of the home network

- o exploiting a vulnerability in the host in order to get access to data or to execute some arbitrary code in the attacked host. Exploitation can be further divided in two classes:
 1. day-0 attack when this attack has never been seen before (hence nothing can really detect it) and
 2. day+n attack where this attack is known and can be detected by the use of an attack signature
- o trojanized host (belonging to a Botnet) can communicate via a covert channel to its master and launch attacks to Internet targets.

3. Overview

The basic goal is to provide an adaptive security policy which aims to block known harmful traffic and allow the rest, restoring as much of end-to-end communication as possible. In addition, new protocols may evolve and be deployed over time; only if they become a threat vector does the CPE firewall receive a signature update (including dynamic correlation data) to classify and block them. This is in direct contrast to [RFC6092], which requires built-in knowledge of a number of protocols, or requires Internet communication to be limited to a handful of protocols that the CPE understands how to process.

- o Intrusion Prevention System (IPS) is a signature-based technology which inspects a pre-defined set of protocols at all layers (from layer-3 to layer-7) and uses a vast set of heuristics to detect attacks within one or several flow. Upon detection, the flow is terminated and an event is logged for further optional auditing. As exploits are added every day, the signature database must be updated daily and is usually quite large (more than 100 MB). This requires both large local storage (large flash or even a hard disk) and a subscription to an update service.
- o Reputation database is a centralized database which gives a reputation score to any IPv6 address (or prefix). The score varies from untrusted to trusted. Untrusted IPv6 addresses are typically addresses of a well-known attacker or from a Botnet member or from an ISP with a poor track of security... Protocols exist to dynamically request a reputation (based on DNS or HTTP). This usually requires a subscription. Note: in IPv6 the reputation database concept is still in its infancy, for example, little experience exists on the scope of the reputation: a host

/128, a LAN prefix /64 or a delegated prefix size of /56 or /48...

- o Local correlation uses another set of heuristics (like TCP distribution of Initial Sequence Number or used TCP ports or protocol handshake banners) to assert the variety of local hosts (namely operating system (OS) version and set of application) and raise or decrease the importance of a specific attack signature. For example, if the OS of host A is OS-A, then there is no point to inspect traffic to or from host A for attacks which are only relevant to OS-B.
- o Global correlation leverage all IPS distributed on the Internet to build the reputation database as well as changing the relevance of an IPS signature (for example, a propagating worm will trigger a lot of identical signatures on several IPS, this should raise the relevance of a specific signature up to the point of blocking all inbound/outbound connections on a specific layer-4 port).

The above techniques are common in the large network where budget is enough to buy firewalls, IPS and subscribe to signature or reputation source. The authors of this document believes that competition and Moore's law will make the set of those techniques (commonly referred to as 'Universal Threat Mitigation') affordable for consumer space.

3.1. Rules for Security Policy

These are an example set of rules to be applied. Each would normally be configurable, either by the user directly or on behalf of the user by a subscription service. The default preferred state hasn't been listed, though it is expected that all rules would be on by default.

If we named all hosts on the residential side of the CPE as 'inside' and all hosts on the Internet as 'outside', then the behavior of the CPE is described by a small set of rules:

1. Rule RejectBogon: apply unicast reverse path forwarding (RPF) checks (anti-spoofing) for all inbound and outbound traffic (implicitly blocking link-local and ULA in the same shot)
2. Rule BlockBadReputation: block all inbound and outbound packets whose outside IPv6 address has a bad reputation score
3. Rule AllowReturn: inspect all outbound traffic and allow the return traffic matching the states (5-tuple + TCP sequence number or any layer-4 state), apply IPS on the outbound (to block Botnet) and inbound (to block malicious/cracked servers which could inject malware) with IPS. If the protocol is not supported/recognized by the IPS, accept it anyway.

4. Rule AllowToPublicDnsHost: allow all inbound traffic to any inside address which is listed in the public DNS with a AAAA record (this requires that the CPE/RG can do a zone transfer, i.e., that the CPE/RG appears like a secondary name server), all inbound traffic is also inspected with IPS. If the protocol is not supported/recognized by the IPS, accept it anyway.
5. Rule ProtectLocalOnly: block all inbound traffic to any inside address as long as the inside address has never sent a packet to the outside. The intent is to protect local-only devices like thermostat or printers. Most (if not all) hosts expecting inbound connections have to send a couple of outbound packets to the outside (registration, DNS request, ...). This is the usual IPv4 firewall behavior augmented with IPS and reputation
6. Rule CryptoIntercept: at the exception of IPsec, all inbound connections that are encrypted (notably TLS [RFC5246]) must be intercepted (this is terminated by the CPE that will present its own self-signed certificate to the remote party which should have installed the CPE self-signed certificate in a secure way in its trust anchors store) in order to allow for further inspection. The decrypted flow is then passed again through those rules and encrypted again before being forwarded to the local host. This is actually a Man-in-the-Middle attack done for a good reason: protect the naive residential user. Of course, documentation and GUI MUST be provided to educate the user and help him/her to understand how to do it in a secure way. Note: this technique is also used nowadays by large enterprise web proxies with the self-signed certificate being securely distributed to all clients.
7. Rule ParanoidOpeness: allow all unsolicited inbound connections rate limited to protect against port and address scanning attacks or overloading devices or slow links within the home. The connection MUST be inspected by the IPS engine. If the connection is anonymous or using a default password (like connecting to a webcam as a guest), then the flow SHOULD be dropped. If the IPS detects an attack, then the flow MUST be closed. If the protocol is not recognized as supported by the IPS, the flow MAY be allowed.

3.2. Security Analysis

This proposal of 'paranoid openness' stops the following attacks:

- o unauthorized use of services/denial of service: because all anonymous access to inside servers are blocked.

- o Denial of services on low bandwidth or low CPU inside hosts IFF those hosts never access the Internet
- o Exploiting of a day+1 attack, those attacks are blocked with the IPS signature and address reputation database

The CryptoIntercept part can also be leveraged as a small Certification Authority (CA) that could generate RSA key pairs and X.509 certificates at the CPE/RG owner's request. Those key pairs and certificates can then be given to trusted devices or users (like the owner's laptop so that he/she could easily and safely connect from the outside).

This proposal cannot help with the following attacks:

- o flooding the access link to the Internet, this is exactly the same as with the old layers-3/4 firewall approach as only the ISP can effectively stop the flooding of the CE-PE link;
- o weak password on inside services, of course the IPS component will detect multiple failed attempts (dictionary attack) and report the offender to the Global Correlation system;
- o exploiting of day-0 attack: until now, these day-0 attacks are caused either by rapidly propagating worms (then the global correlation of unusual traffic pattern will raise an alert and block the traffic after a couple of hundred's of successful attacks) or by targeted attacks against high-profile targets (like Government or banks or ;..) which should be protected by conventional less open security policies;
- o exploiting a vulnerability in a rare or new protocol (not yet supported by the IPS), this case will probably never occur on a wide scale in a residential use of Internet.

4. IANA Considerations

There are no extra IANA consideration for this document.

5. Security Considerations

All security considerations have been done in the Security Analysis Section 3.2.

It is also advisable that the inbound rate limiter system could be added to the [RFC6092] as it is light and does not depend on a

centralized policy server.

6. Acknowledgements

Many thanks to Ole Troan, Stuart Cheshire, Dave Oran and Eliot Lear for the review of the -00 version and to Ron Bonica, Sam Hartmans, Lee Howard, Greg Lebovitz, Jordi Palet, Tina Tsou and others for their comments during and after the first presentation at the Hiroshima IETF meeting in November 2009.

A previous IETF work has similar ideas
[I-D.palet-v6ops-ipv6security].

7. References

7.1. Normative References

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

7.2. Informative References

- [I-D.palet-v6ops-ipv6security]
Palet, J., Vives, A., Martinez, G., and A. Gomez, "IPv6 distributed security requirements",
draft-palet-v6ops-ipv6security-02 (work in progress),
February 2005.
- [RFC2993] Hain, T., "Architectural Implications of NAT", RFC 2993,
November 2000.
- [RFC6092] Woodyatt, J., "Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service", RFC 6092,
January 2011.

Authors' Addresses

Eric Vyncke
Cisco Systems
De Kleetlaan 6a
Diegem 1831
Belgium

Phone: +32 2 778 4677
Email: evyncke@cisco.com

Andrew Yourtchenko
Cisco Systems
De Kleetlaan 6a
Diegem 1831
Belgium

Phone: +32 2 704 5494
Email: ayourtch@cisco.com

Mark Townsley
Cisco Systems
11, Rue Camille Desmoulins
Issy Les Moulineaux 92782
France

Phone: +33 15 804 3483
Email: townsley@cisco.com

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: April 28, 2011

H. Singh
W. Beebee
Cisco Systems, Inc.
C. Donley
CableLabs
B. Stark
AT&T
O. Troan, Ed.
Cisco Systems, Inc.
October 25, 2010

Advanced Requirements for IPv6 Customer Edge Routers
draft-wbeebee-v6ops-ipv6-cpe-router-bis-04

Abstract

This document continues the work undertaken by the IPv6 CE Router Phase I work in the IETF v6ops Working Group. Advanced requirements or Phase II work is covered in this document.

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 28, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Requirements Language	3
2. Terminology	3
3. Conceptual Configuration Variables	4
4. Architecture	4
5. Advanced Features and Feature Requirements	6
5.1. DNS	6
5.2. Multicast Behavior	6
5.3. ND Proxy	7
5.4. Prefix Delegation on LAN interface(s) (More details are TBD)	8
5.5. Routed network behavior(General Cases TBD)	8
5.6. Transition Technologies Support	9
5.6.1. Dual-Stack(DS)-Lite	9
5.6.2. 6rd	10
5.6.3. Transition Technologies Coexistence	10
5.7. Quality Of Service	11
5.8. Unicast Data Forwarding	11
5.9. ZeroConf	11
6. Security Considerations	11
7. Acknowledgements	11
8. Contributors	12
9. IANA Considerations	12
10. References	12
10.1. Normative References	12
10.2. Informative References	15
Authors' Addresses	15

1. Introduction

This document defines Advanced IPv6 features for a residential or small office router referred to as an IPv6 CE router. Typically these routers also support IPv4. The IPv6 End-user Network Architecture for such a router is described in [I-D.ietf-v6ops-ipv6-cpe-router]. This version of the document includes the requirements for Advanced features.

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. Terminology

End-user Network	one or more links attached to the IPv6 CE router that connect IPv6 hosts.
IPv6 Customer Edge router	a node intended for home or small office use which forwards IPv6 packets not explicitly addressed to itself. The IPv6 CE router connects the end-user network to a service provider network.
IPv6 host	any device implementing an IPv6 stack receiving IPv6 connectivity through the IPv6 CE router
LAN interface	an IPv6 CE router's attachment to a link in the end-user network. Examples are Ethernet (simple or bridged), 802.11 wireless or other LAN technologies. An IPv6 CE router may have one or more network layer LAN Interfaces.
Service Provider	an entity that provides access to the Internet. In this document, a Service Provider specifically offers Internet access using IPv6, and may also offer IPv4 Internet access. The Service Provider can provide such access over a variety of different transport methods such as DSL, cable, wireless, and others.

WAN interface an IPv6 CE router's attachment to a link used to provide connectivity to the Service Provider network; example link technologies include Ethernets (simple or bridged), PPP links, Frame Relay, or ATM networks as well as Internet-layer (or higher-layer) "tunnels", such as tunnels over IPv4 or IPv6 itself.

3. Conceptual Configuration Variables

The CE Router maintains such a list of conceptual optional configuration variables.

1. Enable an IGP on the LAN.

4. Architecture

This document extends the architecture described in [I-D.ietf-v6ops-ipv6-cpe-router] to cover a strictly larger set of operational scenarios. In particular, QoS, multicast, DNS, routed network in the home, transition technologies, and conceptual configuration variables. This document also extends the model described in [I-D.ietf-v6ops-ipv6-cpe-router] to a two router topology where the two routers are connected back-to-back (the LAN of one router is connected to the WAN of the other router). This topology is depicted below:

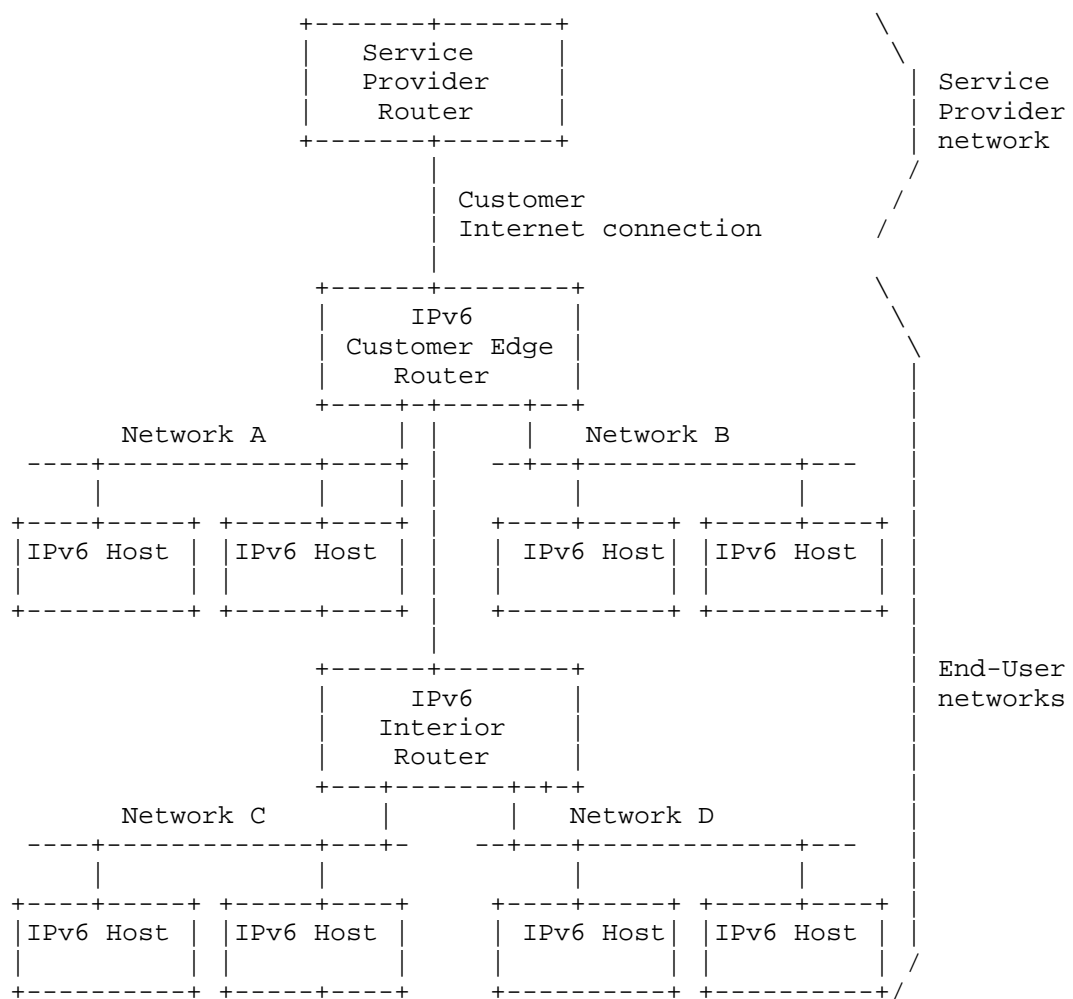


Figure 1.

For DNS, the operational expectation is that the end-user would be able to access home hosts from the home using DNS names instead of more cumbersome IPv6 addresses. Note that this is distinct from the requirement to access home hosts from outside the home.

End-users are expected to be able to receive multicast video in the home without requiring the CE router to include the cost of supporting full multicast routing protocols.

5. Advanced Features and Feature Requirements

The IPv6 CE router will need to support connectivity to one or more access network architectures. This document describes an IPv6 CE router that is not specific to any particular architecture or Service Provider, and supports all commonly used architectures.

5.1. DNS

D-1: For local DNS queries for configuration, the CE Router may include a DNS server to handle local queries. Non-local queries can be forwarded unchanged to a DNS server specified in the DNS server DHCPv6 option. The CE Router may also include DNS64 functionality which is specified in [I-D.bagnulo-behave-dns64].

D-2: The local DNS server MAY also handle renumbering from the Service Provider provided prefix for local names used exclusively inside the home (the local AAAA and PTR records are updated). This capability provides connectivity using local DNS names in the home after a Service Provider renumbering. A CE Router MAY add local DNS entries based on dynamic requests from the LAN segment(s). The protocol to carry such requests from hosts to the CE Router is yet to be described.

5.2. Multicast Behavior

This section is only applicable to a CE Router with at least one LAN interface. A host in the home is expected to receive multicast video. Note the CE Router resides at edge of the home and the Service Provider, and the CE Router has at least one WAN connection for multiple LAN connections. In such a multiple LAN to a WAN topology at the CE Router edge, it is not necessary to run a multicast routing protocol and thus MLD Proxy as specified in [RFC4605] can be used. The CE Router discovers the hosts via a MLDv2 Router implementation on a LAN interface. A WAN interface of the CE Router interacts with the Service Provider router by sending MLD Reports and replying to MLD queries for multicast Group memberships for hosts in the home.

The CE router SHOULD implement MLD Proxy as specified in [RFC4605]. For the routed topology shown in Figure 1, each router implements a MLD Proxy. If the CE router implements MLD Proxy, the requirements on the CE Router for MLD Proxy are listed below.

WAN requirements, MLD Proxy:

WMLD-1: Consistent with [RFC4605], the CE router MUST NOT implement the router portion of MLDv2 for the WAN interface.

LAN requirements, MLD Proxy:

LMMLD-1: The CPE Router MUST follow the model described for MLD Proxy in [RFC4605] to implement multicast.

LMMLD-2: Consistent with [RFC4605], the LAN interfaces on the CPE router MUST NOT implement an MLDv2 Multicast Listener.

LAN requirements:

LM-1: If the CE Router has bridging configured between the LAN interfaces, then the LAN interfaces MUST support snooping of MLD [RFC3810] messages.

5.3. ND Proxy

LAN requirements:

LNDP-1: If the CE Router has only one /64 prefix to be used across multiple LAN interfaces and the CE Router supports any two LAN interfaces that cannot bridge data between them because the two interfaces have disparate MAC layers, then the CE Router MUST support Proxying Neighbor Advertisements as specified in Section 7.2.8 of [RFC4861]. If any two LAN interfaces support bridging between the interfaces, then Proxying Neighbor Advertisements is not necessary between the two interfaces. Legacy 3GPP networks have the following requirements:

1. No DHCPv6 prefix is delegated to the CE Router.
2. Only one /64 is available on the WAN link.
3. The link types between the WAN interface and LAN interface(s) are disparate and, therefore, can't be bridged.
4. No NAT66 is to be used.
5. Each LAN interface needs global connectivity.
6. Uses SLAAC to configure LAN interface addresses.

For these legacy 3GPP networks, the CPE Router MUST support ND Proxy between the WAN and LAN interface(s). If a CE

Router will never be deployed in an environment with these characteristics, then ND Proxy is not necessary.

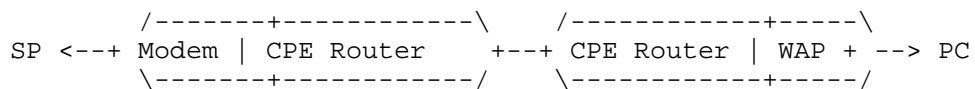
5.4. Prefix Delegation on LAN interface(s) (More details are TBD)

This section is only applicable to a CE Router with at least one LAN interface. The LAN interface(s) are delegated prefixes subnetted from the delegated prefix acquired by the WAN interface and the ULA prefix. After the CE router has assigned prefixes for all of its internally defined needs (its interfaces and any other purposes defined in its internal logic), any leftover prefixes are available for delegation. Any automated prefix delegation mechanism is TBD.

5.5. Routed network behavior(General Cases TBD)

CPE Router Behavior in a routed network:

R-1: One example of the CPE Router use in the home is shown below. The home has a broadband modem combined with a CPE Router, all in one device. The LAN interface of the device is connected to another standalone CPE Router that supports a wireless access point. To support such a network, this document recommends using prefix delegation of the prefix obtained either via IA_PD from WAN interface or a ULA from the LAN interface. The network interface of the downstream router may obtain an IA_PD via stateful DHCPv6. If the CPE router supports the routed network through automatic prefix delegation, the CPE router MUST support a DHCPv6 server or DHCPv6 relay agent. Further, if an IA_PD is used, the Service Provider or user MUST allocate an IA_PD or ULA prefix short enough to be delegated and subsequently used for SLAAC. Therefore, a prefix length shorter than /64 is needed. The CPE Router MAY support and IGP in the home network.



WAP = Wireless Access Point

Figure 2.

5.6. Transition Technologies Support

5.6.1. Dual-Stack(DS)-Lite

Even as users migrate from IPv4 to IPv6 addressing, a significant percentage of Internet resources and content will remain accessible only through IPv4. Also, many end-user devices will only support IPv4. As a consequence, Service Providers require mechanisms to allow customers to continue to access content and resources using IPv4 even after the last IPv4 allocations have been fully depleted. One technology that can be used for IPv4 address extension is DS-Lite.

DS-Lite enables a Service Provider to share IPv4 addresses among multiple customers by combining two well-known technologies: IP in IP (IPv4-in-IPv6) tunneling and Carrier Grade NAT. More specifically, Dual-Stack-Lite encapsulates IPv4 traffic inside an IPv6 tunnel at the IPv6 CE Router and sends it to a Service Provider Address Family Translation Router (AFTR). Configuration of the IPv6 CE Router to support IPv4 LAN traffic is outside the scope of this document.

The IPv6 CE Router SHOULD implement DS-Lite functionality as specified in [I-D.ietf-softwire-dual-stack-lite].

WAN requirements:

- DLW-1: To facilitate IPv4 extension over an IPv6 network, if the CE Router supports DS-Lite functionality, the CE Router WAN interface MUST implement a B4 Interface as specified in [I-D.ietf-softwire-dual-stack-lite].
- DLW-2: If the IPv6 CE Router implements DS-Lite functionality, the CE Router MUST support using a DS-Lite DHCPv6 option [I-D.ietf-softwire-ds-lite-tunnel-option] to configure the DS-Lite tunnel. The IPv6 CE Router MAY use other mechanisms to configure DS-Lite parameters. Such mechanisms are outside the scope of this document.
- DLW-3: IPv6 CE Router MUST NOT perform IPv4 Network Address Translation (NAT) on IPv4 traffic encapsulated using DS-Lite.
- DLW-4: If the IPv6 CE Router is configured with a non-RFC1918 IPv4 address on its WAN interface, the IPv6 CE Router MUST disable the DS-Lite B4 element.

DLW-5: If DS-Lite is operational on the IPv6 CE Router, multicast data MUST NOT be sent on any DS-Lite tunnel.

5.6.2. 6rd

The IPv6 CE Router can be used to offer IPv6 service to a LAN, even when the WAN access network only supports IPv4. One technology that supports IPv6 service over an IPv4 network is IPv6 Rapid Deployment (6rd). 6rd encapsulates IPv6 traffic from the end user LAN inside IPv4 at the IPv6 CE Router and sends it to a Service Provider Border Relay (BR). The IPv6 CE Router calculates a 6rd delegated IPv6 prefix during 6rd configuration, and sub-delegates the 6rd delegated prefix to devices in the LAN.

The IPv6 CE Router SHOULD implement 6rd functionality as specified in [RFC5969].

6rd requirements:

6RD-1: If the IPv6 CE Router implements 6rd functionality, the CE Router WAN interface MUST support at least one 6rd Virtual Interface and 6rd CE functionality as specified in [RFC5969].

6RD-2: If the IPv6 CE Router implements 6rd CE functionality, it MUST support using the 6rd DHCPv4 Option (212) for 6rd configuration. The IPv6 CE Router MAY use other mechanisms to configure 6rd parameters. Such mechanisms are outside the scope of this document.

6RD-3: If 6rd is operational on the IPv6 CE Router, multicast data MUST NOT be sent on any 6rd tunnel.

5.6.3. Transition Technologies Coexistence

Run the following four in parallel to provision CPE router connectivity to the Service Provider:

1. Initiate IPv4 address acquisition.
2. Initiate IPv6 address acquisition as specified by [I-D.ietf-v6ops-ipv6-cpe-router].
3. If 6rd is provisioned, initiate 6rd.
4. If DS-Lite is provisioned, initiate DS-Lite.

The default route for IPv6 through the native physical interface should have preference over the 6rd tunnel interface. The default

route for IPv4 through the native physical interface should have preference over the DS-Lite tunnel interface.

5.7. Quality Of Service

Q-1: The CPE router MAY support differentiated services [RFC2474].

5.8. Unicast Data Forwarding

The null route introduced by the WPD-6 requirement in [I-D.ietf-v6ops-ipv6-cpe-router] has lower precedence than other routes except for the default route.

5.9. ZeroConf

The CE Router MAY support manual configuration via the web using a URL string like `http://router.local` as per multicast DNS (mDNS). Zero-configuration is vendor-dependent.

6. Security Considerations

None.

7. Acknowledgements

Thanks to the following people (in alphabetical order) for their guidance and feedback:

Mikael Abrahamsson, Merete Asak, Scott Beuker, Mohamed Boucadair, Rex Bullinger, Brian Carpenter, Remi Denis-Courmont, Gert Doering, Alain Durand, Katsunori Fukuoka, Tony Hain, Thomas Herbst, Kevin Johns, Stephen Kramer, Victor Kuarsingh, Francois-Xavier Le Bail, David Miles, Shin Miyakawa, Jean-Francois Mule, Michael Newbery, Carlos Pignataro, John Pomeroy, Antonio Querubin, Teemu Savolainen, Matt Schmitt, Hiroki Sato, Mark Townsley, Bernie Volz, James Woodyatt, Dan Wing and Cor Zwart

This draft is based in part on CableLabs' eRouter specification. The authors wish to acknowledge the additional contributors from the eRouter team:

Ben Bekele, Amol Bhagwat, Ralph Brown, Eduardo Cardona, Margo Dolas, Toerless Eckert, Doc Evans, Roger Fish, Michelle Kuska, Diego Mazzola, John McQueen, Harsh Parandekar, Michael Patrick, Saifur Rahman, Lakshmi Raman, Ryan Ross, Ron da Silva, Madhu Sudan, Dan Torbet and Greg White.

8. Contributors

The following people have participated as co-authors or provided substantial contributions to this document: Ralph Droms, Kirk Erichsen, Fred Baker, Jason Weil, Lee Howard, Jean-Francois Tremblay, Yiu Lee, John Jason Brzozowski and Heather Kirksey.

9. IANA Considerations

This memo includes no request to IANA.

10. References

10.1. Normative References

- [I-D.bagnulo-behave-dns64]
Bagnulo, M., Sullivan, A., Matthews, P., Beijnum, I., and M. Endo, "DNS64: DNS extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", draft-bagnulo-behave-dns64-02 (work in progress), March 2009.
- [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-05 (work in progress), September 2010.
- [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-06 (work in progress), August 2010.
- [I-D.ietf-v6ops-ipv6-cpe-router]
Singh, H., Beebee, W., Donley, C., Stark, B., and O. Troan, "Basic Requirements for IPv6 Customer Edge Routers", draft-ietf-v6ops-ipv6-cpe-router-07 (work in progress), August 2010.
- [I-D.vyncke-advanced-ipv6-security]
Vyncke, E. and M. Townsley, "Advanced Security for IPv6 CPE", draft-vyncke-advanced-ipv6-security-01 (work in progress), March 2010.
- [RFC1122] Braden, R., "Requirements for Internet Hosts -

Communication Layers", STD 3, RFC 1122, October 1989.

- [RFC2080] Malkin, G. and R. Minnear, "RIPng for IPv6", RFC 2080, January 1997.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2464] Crawford, M., "Transmission of IPv6 Packets over Ethernet Networks", RFC 2464, December 1998.
- [RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.
- [RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, May 2000.
- [RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.
- [RFC3633] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, December 2003.
- [RFC3646] Droms, R., "DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3646, December 2003.
- [RFC3736] Droms, R., "Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6", RFC 3736, April 2004.
- [RFC3810] Vida, R. and L. Costa, "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, June 2004.
- [RFC4075] Kalusivalingam, V., "Simple Network Time Protocol (SNTP) Configuration Option for DHCPv6", RFC 4075, May 2005.
- [RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, October 2005.
- [RFC4242] Venaas, S., Chown, T., and B. Volz, "Information Refresh Time Option for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 4242, November 2005.

- [RFC4294] Loughney, J., "IPv6 Node Requirements", RFC 4294, April 2006.
- [RFC4443] Conta, A., Deering, S., and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, March 2006.
- [RFC4541] Christensen, M., Kimball, K., and F. Solensky, "Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches", RFC 4541, May 2006.
- [RFC4605] Fenner, B., He, H., Haberman, B., and H. Sandick, "Internet Group Management Protocol (IGMP) / Multicast Listener Discovery (MLD)-Based Multicast Forwarding ("IGMP/MLD Proxying")", RFC 4605, August 2006.
- [RFC4632] Fuller, V. and T. Li, "Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan", BCP 122, RFC 4632, August 2006.
- [RFC4779] Asadullah, S., Ahmed, A., Popoviciu, C., Savola, P., and J. Palet, "ISP IPv6 Deployment Scenarios in Broadband Access Networks", RFC 4779, January 2007.
- [RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.
- [RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.
- [RFC4864] Van de Velde, G., Hain, T., Droms, R., Carpenter, B., and E. Klein, "Local Network Protection for IPv6", RFC 4864, May 2007.
- [RFC5072] S.Varada, Haskins, D., and E. Allen, "IP Version 6 over PPP", RFC 5072, September 2007.
- [RFC5571] Storer, B., Pignataro, C., Dos Santos, M., Stevant, B., Toutain, L., and J. Tremblay, "Softwire Hub and Spoke Deployment Framework with Layer Two Tunneling Protocol Version 2 (L2TPv2)", RFC 5571, June 2009.
- [RFC5969] Townsley, W. and O. Troan, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification", RFC 5969, August 2010.

10.2. Informative References

[I-D.ietf-behave-v6v4-framework]

Baker, F., Li, X., Bao, C., and K. Yin, "Framework for IPv4/IPv6 Translation", draft-ietf-behave-v6v4-framework-10 (work in progress), August 2010.

[UPnP-IGD]

UPnP Forum, "Universal Plug and Play (UPnP) Internet Gateway Device (IGD)", November 2001, <<http://www.upnp.org/standardizeddcps/igd.asp>>.

Authors' Addresses

Hemant Singh
Cisco Systems, Inc.
1414 Massachusetts Ave.
Boxborough, MA 01719
USA

Phone: +1 978 936 1622
Email: shemant@cisco.com
URI: <http://www.cisco.com/>

Wes Beebee
Cisco Systems, Inc.
1414 Massachusetts Ave.
Boxborough, MA 01719
USA

Phone: +1 978 936 2030
Email: wbeebee@cisco.com
URI: <http://www.cisco.com/>

Chris Donley
CableLabs
858 Coal Creek Circle
Louisville, CO 80027
USA

Email: c.donley@cablelabs.com

Barbara Stark
AT&T
725 W Peachtree St
Atlanta, GA 30308
USA

Email: barbara.stark@att.com

Ole Troan (editor)
Cisco Systems, Inc.
Veversmauet 8
N-5017 BERGEN,
Norway

Email: ot@cisco.com

