INTERNET DRAFT
<draft-huitema-ngtrans-unmaneval-01.txt>
November 1, 2002
Expires May 1, 2002

C. Huitema
Microsoft
R. Austein
Bourgeois Dilettante
S. Satapati
Cisco Systems, Inc.
Ronald van der Pol
NLnet Labs

Evaluation of Transition Mechanisms for Unmanaged Networks

Status of this memo

Abstract

In a companion paper we defined the "unmanaged networks" scope, which typically correspond to home networks or small office networks, and the requirements for IPv6 transition mechanism in that scope. We start from this analysis and evaluate here the suitability of mechanisms defined in the NGTRANS working group.

1 Introduction

In a companion paper [UNMANREQ] we defined the "unmanaged networks" scope, which typically correspond to home networks or small office networks, and the requirements for IPv6 transition mechanism in that scope. We start from this analysis and evaluate here the suitability of mechanisms defined in the NGTRANS working group.

The requirements for unmanaged networks are expressed by analyzing four classes of application: local, client, peer to peer, and servers, and considering four cases of deployment. These are:

A) Gateway does not provide IPv6

B) ISP and gateway are dual stack
C) Gateway is IPv6 capable, dual stack, ISP is not
D) ISP is IPv6 only

This document analyses the issues involved in the transition from IPv4 to IPv6. One of the most important issues is that of naming and addressing.

During the transition phase from IPv4 to IPv6 there will be IPv4 only, dual stack or IPv6 only nodes that want to communicate to other IPv4 only, dual stack or IPv6 only nodes. When both nodes do not speak the same version of IP, some translation will be needed. The issues involved are described in the next chapters. This analysis outlines two types of requirement: connectivity requirements, i.e. how to ensure that nodes can exchange IP packets, and naming requirements, i.e. how to ensure that nodes can resolve each-other's names.

Note that draft-00 is essentially a pro-forma place holder. Many of the discussion sections are incomplete. We expect that the content will evolve significantly during and after the interim meeting of the NGTRANS/V6OPS WG.

2 Meeting case A requirements

Different connectivity requirements appear at different stages of the IPv6 deployment:

- In case A, isolated hosts located behind a NAT need to acquire some form of connectivity.
- In case B, an IPv6 capable gateway must be able to obtain IPv6 connectivity and an IPv6 prefix from an IPv6 capable ISP. The network may include IPv4 only hosts, IPv6 only hosts, and dual stack hosts. Various mechanisms are needed to let IPv4 hosts and IPv6 hosts interoperate.
- In case C, an upgraded gateway must be able to provide IPv6 connectivity to the unmanaged network independently of the local ISP. Otherwise, the application requirements are the same as in case B.
- In case D, the ISP only provide IPv6 services; an IPv4 only host must be able to interact through the gateway and the IPv6 only ISP with IPv4 only servers on the Internet.

In this section, we first evaluate how mechanisms already defined or being worked on in the IETF meet these requirements. We then consider the "remaining holes" and recommend specific developments.

2.1 Evaluation of connectivity mechanisms

The following evaluation is fractional and preliminary, and does not necessarily reflect consensus of all the authors.

2.1.1 TEREDO

TEREDO is a mechanism designed to provide IPv6 connectivity to hosts behind NATs. Hosts use servers to find out a "mapped" IPv4 address and UDP port; they build an IPv6 address that includes the IPv4 address of their preferred server, and their own mapped IPv4 address and mapped port. A mechanism of bubbles, relayed by the servers, is used for establishing contacts between Teredo nodes, or for discovering the appropriate Teredo relay serving an IPv6 peer; the actual IPv6 packets are carried in UDP packets exchanged directly between the nodes, or exchanged through the relay serving an IPv6 peer.

Teredo appears to be a good fit for providing IPv6 connectivity to hosts behind NAT, in case A of IPv6 deployment. The service is designed for minimizing the cost of deploying the server, which matches the requirement of minimizing the cost of the "supporting infrastructure" for peer-to-peer applications.

3 Meeting case B requirements

In case B, we assume that the gateway and the ISP are both dual stack. The hosts on the local network may be IPv4 only, dual stack, or IPv6 only. The main requirements are:

3.1 Prefix delegation

The gateway must be able to acquire an IPv6 prefix, delegated by the ISP. The possible mechanisms are RA proxy and explicit prefix delegation.

3.1.1 RA proxy

The implicit delegation mechanism assumes that the gateway is connected to the ISP by a point-to-point link. Examples of such point to point links are various types of configured tunnels and serial links, for example using PPP. The principle of RA proxy is simple: the gateway issues a "router solicitation" message on the serial link, receives a "router advertisement", learns a network prefix from the advertisement, and advertises the same prefix on the unmanaged network.

The implicit delegation mechanism cannot work if the provider's router advertises the same prefix to multiple gateways, as is the case if gateways are connected to routers through a shared media. In this situation, an explicit delegation mechanism is required; this type of mechanism is currently being studied.

3.1.2 Explicit prefix delegation

Discussion of [PREFIXDHCPV6].

3.2 Communication between IPv4-only and IPv6-only hosts

During the transition phase from IPv4 to IPv6 there will be IPv4-only, dual stack or IPv6-only nodes that want to communicate to other IPv4-only, dual stack or IPv6-only nodes. When both nodes do not speak the same version of IP, some translation will be needed. The table below shows the preferred situation with respect to nodes A and B wanting to communicating with each other.

A / B	IPv4 only	dual stack	IPv6 only
IPv4 only	IPv4 address	IPv4 address	t(IPv6)->IPv4 address
dual stack	IPv4 address	IPv4 or IPv6 address	IPv6 address
IPv6 only	t(IPv4)->IPv6 address	IPv6 address	IPv6 address

t(IPv6)->IPv4 means that some translation is involved which produces some IPv4 address for an IPv6 only host.

t(IPv4)->IPv6 means that some translation is involved which produces some IPv6 address for an IPv4 only host.

3.2.1 The problem with address translation

An obvious candidate for enabling communication between IPv4-only and IPv6-only hosts is "network address translation, protocol translation" [NAT-PT]. The NAT-PT mechanisms has two components, address translation and address discovery: address translation is the processing of IP packets by a NAT; address discovery is the mechanism by which an IPv4-only or IPv6-only host discovers the "translated address" at which packets for their correspondent should be addressed. The NAT-PT specification proposes to solve address discovery by using a DNS ALG, as specified in section 4 of [NAT-PT].

This section makes an important assumption: it assumes that the NAT-PT acts as a bridge between two networks, one IPv6-only and the other IPv6-only. As a result, the DNS-ALG will translate a DNS

request for a AAAA record coming from the IPv6 host into a request for an A record, and vice versa. The problem is that address translation does not know if the traffic originates from an IPv4 only/IPv6 only node or from a dual stack node. When a dual stack node A wants to communicate with an IPv4 only host B, the dual stack host A gets either the IPv4 address of B (preferred) or an IPv6 address which is some kind of translation of the IPv4 address of B. This latter situation is not wanted, because it means unnecessary translation between IPv4 and IPv6. This is shown in the table below.

+ A / B +	IPv4 only	dual stack	IPv6 only
IPv4 only	IPv4 address	IPv4 address	t(IPv6)->IPv4 address
dual stack	XXX IPv4 address or t(IPv6)->IPv6 XXX	IPv4 or IPv6 address	XXX IPv6 address or t(IPv6)->IPv4 XXX
IPv6 only	t(IPv4)->IPv6 address	IPv6 address	IPv6 address

The boxes with XXX-es are the cases where address translation could result in unwanted translation.

3.2.2 Possible solutions

There are at least two solutions to the problems described above: avoid the DNS ALG, or avoid using the results of the DNS ALG when they are not needed.

IPv6-only hosts can avoid the need for a DNS ALG if they can process IPv4 addresses locally. In this hypothesis, the IPv6-only node that wished to communicate with a correspondent will first request the AAAA records associated with the correspondent's name; if there are no such records, the IPv6-only node will request the A records of the correspondent, and construct as many IPv6 addresses by combining a specific address prefix with each IPv4 addresses. The same mechanism can be used when the IPv4 address is learned by other channels than the DNS, e.g. as a literal address in a URL.

Dual stack hosts can avoid using unnecessary translation if they can

recognize translated addresses, and then use address selection rules to give priority to non-translated addresses. Translated addresses can be either IPv6 addresses obtained by combining an IPv4 address with a specific address prefix, or IPv4 addresses allocated by the NAT-PT as a representation of an IPv6 address. The dual stack host can recognize translated IPv6 addresses if the know the value of the specific address prefix used by the NAT-PT; they can recognize the translated IPv4 addresses if the know the address range from which the NAT-PT picks them.

The 1st mechanism will allow IPv6 only hosts to direct requests for IPv4 hosts to the NAT, possibly using a "bump in the API" mechanism. If we assume that this mechanism exists, then there is never a need for a NAT to intercept requests for AAAA records: the IPv6 only node will issue requests for A records if need be.

The second mechanism allows a NAT to map IPv6 addresses to IPv4 while reducing the risk of confusion: with proper address selection rules, a dual stack host will never use these mapped addresses instead of the regular IPv6 addresses.

Both solutions requires that the IPv6 host be aware of the specific IPv6 address prefix used by the NAT-PT, and thus require either that this 96 bit prefix be set to a conventional well known value, or that the value chosen by the NAT-PT be provisioned in each IPv6 client. Dual stack host will also need to recognize the IPv4 address range used by the NAT-PT.

3.3 Resolution of local names to IPv6 addresses

To be developed. Compare dynamic DNS update and multicast name resolution. Study solution for reverse lookup. Consider the case of privacy addresses.

4 Meeting case C requirements

Case C is very similar to case B, the difference being that the ISP is not dual stack. The gateway must thus use some form of tunneling mechanism to obtain IPv6 connectivity, and an address prefix.

4.1 Tunneling mechanisms

4.1.1 6to4

The [6TO4] technology allows routers to derive a global scope IPv6 prefix from a global IPv4 address. This technology is a very good fit for the second phase of the transition, as it can be programmed in the "upgraded gateway", and can provide value to the gateway users without requiring explicit support from the ISP. This technology has however a clear limitation: it requires that the

gateway obtains at least one global IPv4 address from the local ISP.

Another potential limitation of the technology is the reliance on publicly accessible "6to4 relay routers" that accept packets from 6to4 routers and relay them to the "regular" IPv6 Internet. These relays all listen to the same IPv4 anycast address [RFC3056], which enables gateways to start operating as 6to4 routers without requiring any explicit configuration. As the deployment of IPv6 progresses, a growing fraction of the traffic originating from 6to4 routers will have to be carried through these relays, potentially leading to severe congestion of the relays.

There are three possible ways to alleviate this congestion. First, one can hope that many actors will deploy 6to4 relay routers, in order to facilitate the deployment of IPv6; congestion would be alleviated by the provision of a large number of gateways. Second, one could develop some "route optimization" process, so that the traffic would flow through a "shortcut path" rather than through the 6to4 relays; the relays would then avoid congestion by carrying only a small fraction of the traffic. Third, if neither the first nor the second solution materialize, some gateways may enter into contractual agreements with relay service providers; in this case, the 6to4 technology would become merely a variant of the configured tunnel technologies.

4.1.2 Tunnel broker

Configured tunnels require a contractual agreement with an IPv6 provider, which comes in addition to the existing agreement with the IPv4 provider; different technologies have different domains of application:

- Many tunnel technologies use a global IPv4 address to identify the "client end" of the tunnel, thus inheriting the same "global IPv4 address" requirement as 6TO4;

- A variant of the [TEREDO] technology could be used to establish tunnels over UDP when the client cannot use a global IPv4 address; this variant is however not standardized.

- Practical deployment of tunnel technologies requires the introduction of accounting/billing functions; the existing tunnel broker specification, [TUNNELS], does not describe how these functions should be implemented.

The practical conclusion is that "upgraded gateways" will probably support the 6TO4 technology, and will have an optional configuration option for "configured tunnels". Configured tunnels are in practice an intermediate solution between the "automatic configuration" provided by 6to4, and the "ISP support" that characterize case B.

5 Meeting the case D requirements

To be developed.

6 Provisional recommendations

This draft is still a draft, but we can already list a set of recommendations for the V6OPS working group:

- To meet case A requirements, we need to develop and standardize the Teredo technology.

- To meet case B prefix delegation requirements, we need a standardized IPv6 prefix delegation mechanism

- To meet case B connectivity requirements, we need to revisit the NAT-PT specification, in order to clarify the use of the DNS-ALG. We also need to either reserve specific IPv4 and IPv6 address prefixes for use by NAT-PT, or define a way to provision IPv6 hosts with the IPv4 and IPv6 prefixes used by the local NAT-PT.

- To meet case C connectivity requirement, we need to continue standardization of the 6to4 mechanism.

7 IANA Considerations

This memo does not include any request to IANA.

8 Copyright

The following copyright notice is copied from RFC 2026 [Bradner, 1996], Section 10.4, and describes the applicable copyright for this document.

revoked by the Internet Society or its successors or assignees.

This document and the information contained herein is provided on an

"AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

9 Intellectual Property

The following notice is copied from RFC 2026 [Bradner, 1996], Section 10.4, and describes the position of the IETF concerning intellectual property claims made against this document.

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use other technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

10 Acknowledgements

This fractional and preliminary memo has benefited from comments of Ronald van der Pol, Margaret Wasserman and Tony Hain.

11 References

[UNMANREQ] Unmanaged Networks Transition Scope. Work in progress.

[IPV6] S. Deering, R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[NEIGHBOR] T. Narten, E. Nordmark, W. Simpson, "Neighbor Discovery for IP Version 6 (IPv6)", RFC 2461, December 1998.

[STATELESS] T. Narten, S. Thomson, "IPv6 Stateless Address

Autoconfiguration", RFC 2462, December 1998.

[6TO4] B. Carpenter, K. Moore, "Connection of IPv6 Domains via IPv4 Clouds", RFC 3056, February 2001.

[6TO4ANYCAST] C. Huitema, "An Anycast Prefix for 6to4 Relay Routers", RFC 3068, June 2001.

[TEREDO] Teredo: Tunneling IPv6 over UDP through NATs. Work in progress.

[TUNNELS] A. Durand, P. Fasano, I. Guardini. IPv6 Tunnel Broker. RFC 3053, January 2001

[PREFIXDHCPV6] IPv6 Prefix Options for DHCPv6. Work in progress.

[SIIT] E. Nordmark, Stateless IP/ICMP Translation Algorithm (SIIT). RFC 2765, February 2000.

[NAT-PT] G. Tsirtsis, P. Srisuresh, Network Address Translation - Protocol Translation (NAT-PT). RFC 2766, February 2000.

[DNS-ALG-ISSUES] Issues with NAT-PT DNS ALG in RFC2766. Work in progress.

[HALLIN-DNS-ALG] NAT-PT DNS ALG solutions. Work in progress.

[TRT] J. Hagino, K. Yamamoto. An IPv6-to-IPv4 Transport Relay Translator, RFC 3142, June 2001.

[DSTM] Dual Stack Transition Mechanism (DSTM). Work in progress.

[DNSDHCPV6] DNS Configuration options for DHCPv6. Work in progress.

[DNSANYCAST] Well known site local unicast addresses for DNS resolver. Work in progress.

[MDNS] Multicast DNS. Work in progress.

[NODEINFO] IPv6 Node Information Queries. Work in progress.

12 Authors' Addresses

Christian Huitema
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399
Email: huitema@microsoft.com

Rob Austein
Email: sra@hactrn.net

Suresh Satapati
Cisco Systems, Inc.
San Jose, CA 95134

USA
EMail: satapati@cisco.com

Ronald van der Pol
Email: Ronald.vanderPol@rvdp.org

Table of Contents: