V6OPS B. Carpenter
Internet-Draft Univ. of Auckland
Intended status: Informational February 24, 2011
Expires: August 28, 2011

Advisory Guidelines for 6to4 Deployment
draft-carpenter-v6ops-6to4-teredo-advisory-02

Abstract

This document provides advice to network operators about deployment of the 6to4 technique for automatic tunneling of IPv6 over IPv4. It is principally addressed to Internet Service Providers, including those that do not yet support IPv6, and to Content Providers. The intention of the advice is to minimise both user dissatisfaction and help desk calls.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

A technique for automatic tunneling of IPv6 over IPv4, intended for situations where a user may wish to access IPv6-based services via a network that does not support IPv6, was defined a number of years ago. It is known as 6to4 [RFC3056], [RFC3068] and is quite widely deployed in end systems, especially desktop and laptop computers. Also, 6to4 is supported in a number of popular models of CPE routers, some of which have it enabled by default, leading to quite widespread unintentional deployment by end users.

Unfortunately, experience shows that the method has some problems in current deployments that can lead to connectivity failures. These failures either cause long retry delays or complete failures for users trying to connect to services. In many cases, the user may be quite unaware that 6to4 is in use, and when the user contacts a help desk, in all probability the help desk is unable to correctly diagnose the problem. Anecdotally, many help desks simply advise users to disable IPv6, thus defeating the whole purpose of the mechanism, which was to encourage early adoption of IPv6.

There is additional discussion of operational issues in [I-D.vandevelde-v6ops-harmful-tunnels]. The main goal of the present document is to offer advice to network operators on how to deal with this situation more constructively than by disabling 6to4. It briefly describes the principle of operation, then describes the problems observed, and finally offers specific advice on the available methods of avoiding the problems. Note that some of this advice applies to ISPs that do not yet support IPv6, since their customers and help desks are significantly affected in any case. Other advice applies to content providers.

We do not discuss here details of this situation that are mainly outside the scope of network operators:

1. Operating system preferences between IPv4 and IPv6 when both appear to be available [I-D.ietf-6man-rfc3484-revise].
2. Ensuring that application software deals gracefully with connectivity problems [I-D.wing-v6ops-happy-eyeballs-ipv6].
3. Some content providers have chosen to avoid the problem by hiding their IPv6 address except from customers of pre-qualified networks [I-D.ietf-v6ops-v6-aaaa-whitelisting-implications].

Note to readers of earlier versions: references to Teredo have been removed from this document. Sorry about the file name.

2. Principles of Operation

There are two variants of 6to4 which are referred to here as "Router 6to4" and "Anycast 6to4". To understand Anycast 6to4, it is necessary first to understand Router 6to4.

2.1. Router 6to4

Router 6to4 is the original version, documented in [RFC3056]. The model assumes that a user site operates native IPv6, but that its ISP provides no IPv6 service. The site border router acts as a 6to4 router. If its external global 32-bit IPv4 address is V4ADDR, the site automatically inherits the IPv6 prefix 2002:V4ADDR::/48. (The explanation in RFC 3056 is somewhat confusing, as it refers to the obsolete "Top Level Aggregator" terminology.) The prefix 2002: V4ADDR::/48 will be used and delegated for IPv6 service within the user site.

Consider two such site border routers, with global IPv4 addresses 192.0.2.170 and 190.0.2.187, and therefore inheriting the IPv6 prefixes 2002:c000:2aa::/48 and 2002:c000:2bb::/48 respectively. The routers can exchange IPv6 packets by encapsulating them in IPv4 using protocol number 41, and sending them to each other at their respective IPv4 addresses. In fact, any number of 6to4 routers connected to the IPv4 network can directly exchange IPv6 packets in this way.

Some 6to4 routers are also configured as "Relay routers." They behave as just described, but in addition they obtain native IPv6 connectivity with a normal IPv6 prefix. They announce an IPv6 route to 2002::/16. For example, assume that the 6to4 router at 190.0.2.187 is a relay router, whose address on the 6to4 side is 2002:c000:2bb::1. Suppose that a host with the 6to4 address 2002: c000:2aa::123 sends an IPv6 packet to a native IPv6 destination such as 2001:db8:123:456::321. Assume that the 6to4 router at 192.0.2.170 has its IPv6 default route set to 2002:c000:2bb::1, i.e. the relay. The packet will be delivered to the relay, encapsulated in IPv4. After decapsulation, the relay will forward the packet into native IPv6 for delivery. When the remote host replies, the packet (source 2001:db8:123:456::321, destination 2002:c000:2aa::123) will find a route to 2002::/16 and hence be delivered to a 6to4 relay. The process will be reversed and the packet will be encapsulated and forwarded to the 6to4 router at 192.0.2.170 for final delivery.

Note that this process does not require the same relay to be used in both directions. The outbound packet will go to whichever relay is configured as the default IPv6 router at the source router, and the return packet will go to whichever relay is announcing a route to

2002::/16 in the vicinity of the remote IPv6 host.

There are of course many further details in RFC 3056, most of which are irrelevant to current operational problems.

2.2. Anycast 6to4

Router 6to4 assumes that 6to4 routers and relays will be managed and configured cooperatively. In particular, 6to4 sites need to find a relay router willing to carry their outbound traffic, which becomes their default IPv6 router (except for 2002::/16). The objective of the anycast variant, defined in [RFC3068], is to avoid any need for such configuration. The intention was to make the solution available for small or domestic users, even those with a single host or simple home gateway rather than a border router. This is achieved quite simply, by defining 192.88.99.1 as the default IPv4 address for a 6to4 relay, and therefore 2002:c058:6301:: as the default IPv6 router address for a 6to4 site.

Since Anycast 6to4 implies a default configuration for the user site, it does not require any particular user action. It does require an IPv4 anycast route to be in place to a relay at 192.88.99.1. As with Router 6to4, there is no requirement that the return path goes through the same relay.

3. Problems Observed

It should be noted that Router 6to4 was not designed to be an unmanaged solution. Quite the contrary: RFC 3056 contains a number of operational recommendations intended to avoid routing issues. In practice, there are few if any deployments of Router 6to4 following these recommendations. Mostly, Anycast 6to4 has been deployed. In this case, the user site (either a single host or a small broaband gateway) discovers that it doesn't have native IPv6 connectivity, but that it does have a global IPv4 address and can resolve AAAA queries, and therefore assumes that it can send 6to4 packets to 192.88.99.1.

Empirically, 6to4 appears to suffer from a significant level of connection failure; see <https://labs.ripe.net/Members/emileaben/6to4-how-bad-is-it-really> and <http://www.potaroo.net/ispcol/2010-12/6to4fail.html>. In experiments conducted on a number of dual stack web servers, the TCP connection failure rate has been measured. In these experiments, the client's connection attempt to a server was considered to have failed when the server received a TCP SYN packet and sent a SYN/ACK packet in response, but received no ACK packet to complete the initial TCP 3-way handshake. The experiment conducted by Aben recorded a failure

rate of between 9% and 20% of all 6to4 connection attempts. The experiment conducted by Huston has recorded a failure rate of between 9% and 19% of all 6to4 clients. In this latter experiment it was further noted that between 65% to 80% of all 6to4 clients who failed to connect using 6to4 were able to make a successful connection using IPv4, while the remainder did not make any form of IPv4 connection attempt, successful or otherwise, using the mapped IPv4 address as a source address. No connection attempts were recorded by the server using embedded RFC1918 IPv4 addresses.

There have been several possible reasons offered for this form of 6to4 connection failure. One is the use of private IPv4 addresses embedded in the 6to4 address, making the return path for the 6to4 tunnel infeasible, and the second is the use of local filters and firewalls that drop incoming IP packets that use IP protocol 41. If the former case were prevalent it would be reasonable to expect that a significant proportion of failed 6to4 connections would use embedded IPv4 addresses that are either drawn from the private use (RFC 1918) address ranges, contrary to RFC 3056, or from addresses that are not announced in the Internet's IPv4 inter-domain routing table. Neither case was observed to any significant volume in the experiments conducted by Huston. Furthermore, the experimental conditions were varied to use a return 6to4 tunnel with either the native IPv4 source address of the dual stack server or an IPv4 source address of 192.88.99.1. No change in the 6to4 connection failure rate was observed between these two configurations; however, other operators have reported significant problems when replying from the native address, caused by stateful firewalls at the user site. Given that the server used its own 6to4 relay for the return path, the only difference in the IP packet itself between the successful IPv4 connections and the failed 6to4 connections was the IP protocol number, which was 6 (TCP) for the successful IPv4 connections and 41 (IPv6 payload) for the failed 6to4 connections. The inference from these experiments is that one likely reason for the high connection failure rate for 6to4 connections is the use of local filters close to the end-user that block incoming packets with protocol 41.

In a dual stack context this connection failure rate was effectively masked by the ability of the client system to recover from the failure and make a successful connection using IPv4. In this case the only effect on the client system was a delay in making the connection of between 7 and 20 seconds as the client's system timed out on the 6to4 connection attempts (see [I-D.wing-v6ops-happy-eyeballs-ipv6]).

This experience and further analysis shows that specific operational problems with Anycast 6to4 include:

1. Outbound Black Hole: 192.88.99.1 does not generate 'destination unreachable' but in fact packets sent to that address are dropped. This can happen due to routing or firewall configuration, or even because the relay that the packets happen to reach contains an ACL such that they are discarded.

   This class of problem arises because the user's ISP is accepting a route to 192.88.99.0/24 despite the fact that it doesn't go anywhere useful. Either the user site or its ISP is dropping outbound Protocol 41 traffic, or the upstream operator is unwilling to accept incoming 6to4 packets from the user's ISP. The latter is superficially compatible with the design of Router 6to4 (referred to as "unwilling to relay" in RFC 3056). However, the simple fact of announcing a route to 192.88.99.0/24 in IPv4, coupled with the behavior described in RFC 3068, amounts to announcing a default route for IPv6 to all 6to4 sites that receive the IPv4 route. This violates the assumptions of RFC 3056.

   The effect of this problem on users is that their IPv6 stack believes that it has 6to4 connectivity, but in fact all outgoing IPv6 packets are black-holed. The prevalence of this problem is hard to measure, since the resulting IPv6 packets can never be observed from the outside.
2. Inbound Black Hole: In this case, 6to4 packets sent to 192.88.99.1 are correctly delivered to a 6to4 relay, and reply packets are returned, but they are dropped by an inbound Protocol 41 filter. As far as the user is concerned, the effect is the same as the previous case: IPv6 is a black hole. Many enterprise neworks are believed to be set up in this way. Connection attempts due to this case can be observed by IPv6 server operators, in the form of SYN packets from addresses in 2002::/16 followed by no response to the resulting SYN/ACK. From the experiments cited above, this appears to be a significant problem in practice.
3. No Return Relay: If the Outbound Black Hole problem does not occur, i.e. the outgoing packet does reach the intended native IPv6 destination, the target system will send a reply packet, to 2002:c000:2aa::123 in our example above. Then 2002::/16 may or may not be successfully routed. If it is not routed, the packet will be dropped (hopefully with 'destination unreachable'). According to RFC 3056, an unwilling relay "MUST NOT advertise any 2002:: routing prefix into the native IPv6 domain"; therefore, conversely, if this prefix is advertised the relay must relay packets regardless of source and destination. However, in practice the problem arises that some relays reject packets that they should relay, based on their IPv6 source address.

Whether the native IPv6 destination has no route to 2002::/16, or it turns out to have a route to an unwilling relay, the effect is the same: all return IPv6 packets are black-holed. While there is no direct evidence of the prevalence of this problem, it certainly exists in practice.

4. Large RTT: In the event that none of the above three problems applies, and a two-way path does in fact exist between a 6to4 host and a native host, the round trip time may be quite large and variable since the paths to the two relays are unmanaged and may be complex. Overloaded relays might also cause highly variable RTT.
5. PMTUD Failure: A common link MTU size observed on the Internet today is 1500 bytes. However, when using 6to4 the path MTU is less than this due to the encapsulation header. Thus a 6to4 client will normally see a link MTU that is less than 1500, but a native IPv6 server will see 1500. Path MTU Discovery does not always work, and this can lead to connectivity failures. Even if a TCP SYN/ACK exchange works, TCP packets with full size payloads may simply be lost. These failures are disconcerting even to an informed user, since a standard 'ping' from the client to the server will succeed, because it generates small packets, and the successful SYN/ACK exchange can be traced. Also, the failure may occur on some paths but not others, so a user may be able to fetch web pages from one site, but only ping another.
6. Reverse DNS Failure: Typically a 6to4-addressed host will not have a reverse DNS delegation. If reverse DNS is used as a pseudo-security check, it will fail.
7. Bogus Address Failure: By design, 6to4 does not work and will not activate itself if the available V4ADDR is a private address [RFC1918]. However it will also not work if the available V4ADDR is a "bogon", i.e. a global address that is being used by the operator as a private address. A common case of this is a legacy wireless network using 1.1.1.0/24 as if it was a private address. In this case, 6to4 will assume it is connected to the global Internet, but there is certainly no working return path.

   This failure mode will also occur if an ISP is operating a Carrier Grade NAT between its customers and the Internet, and is using global public address space as if it were private space to do so.
8. Faulty 6to4 Implementations: It has been reported that some 6to4 implementations attempt to activate themselves even when the available IPv4 address is an RFC 1918 address. This is in direct contradiction to RFC 3056, and will produce exactly the same failure mode as Bogus Address Failure. It is of course outside the ISP's control.

9. Difficult Fault Diagnosis: The existence of all the above failure modes creates a problem of its own: very difficult fault diagnosis, especially if the only symptom reported by a user is slow access to web pages, caused by a long timeout before fallback to IPv4. Tracking down anycast routing problems and PMTUD failures is particularly hard.

The practical impact of the above problems, which are by no means universal as there is considerable successful use of Anycast 6to4, has been measured at a fraction of 1% loss of attempted connections to content servers (see <http://www.fud.no/ipv6/>). While this seems low, it amounts to a significant financial impact for content providers. Also, end users frustrated by the poor response times caused by fall-back to IPv4 connectivity [I-D.wing-v6ops-happy-eyeballs-ipv6] are considered likely to generate help desk calls with their attendant costs.

4. Advisory Guidelines

There are several types of operator involved, willingly or unwillingly, in the Anycast 6to4 scenario and they will all suffer if things work badly. There is a clear incentive for each of them to take appropriate action, as described below.

This document avoids formal normative language, because it is highly unlikely that the guidelines apply universally. Each operator will make its own decisions about which of the following guidelines are useful in its specific scenario.

4.1. Vendor Issues

Although this document is aimed principally at operators, there are some steps that implementers and vendors of 6to4 should take.

1. Some vendors of routers, including customer premises equipment, have not only included support for 6to4 in their products, but have enabled it by default. This is bad practice - it should always be a conscious decision by a user to enable 6to4. Many of the above problems only occur due to unintentional deployment of 6to4.
2. Similarly, host operating systems should not enable Anycast 6to4 by default; it should always be left to the user to switch it on.
3. Any 6to4 implementation that attempts to activate itself when the available IPv4 address is an RFC 1918 address is faulty and needs to be updated.
4. 6to4 implementations should adopt updated IETF recommendations on address selection [I-D.ietf-6man-rfc3484-revise].

4.2. Consumer ISPs, and enterprise networks, that do not support IPv6 in any way

To reduce the negative impact of Anycast 6to4 deployed (probably unknowingly) by users, and consequent user dissatisfaction and help desk calls, such ISPs should check in sequence:

1. Does the ISP have a route to 192.88.99.1? (This means an explicit route, or knowledge that the default upstream provider has an explicit route. A default route doesn't count!)
2. If so, is it functional and stable?
3. If so, is the ping time reasonably short?
4. If so, does the relay willingly accept 6to4 traffic from the ISP's IPv4 prefixes? (Note that this is an administrative as well as a technical question - is the relay's operator willing to accept the traffic?)

Unless the answer to all these questions is 'yes', subscribers will be no worse off, and possibly better off, if the route to 192.88.99.1 is blocked and generates an IPv4 'destination unreachable'. There is little operational experience with this, however.

Some implementations also perform some form of 6to4 relay qualification. For example, one host implementation (Windows) tests the Protocol 41 reachability by sending an ICMPv6 echo request with Hop Limit=1 to the relay, expecting a response or Hop Limit exceeded error back. Lack of any response indicates that the 6to4 relay does not work and it is turned off [Savola].

A more constructive approach for such an ISP is to seek out a transit provider who is indeed willing to offer outbound 6to4 relay service, so that the answer to each of the questions above is positive.

In any case, such ISPs should always allow protocol 41 through their network and firewalls. Not only is this a necessary condition for 6to4 to work, but it also allows users who want to use a configured IPv6 tunnel service to do so.

Some operators, particularly entreprise networks, block Protocol 41 on security grounds. Doing this on its own is bad practice. The strategic solution is to deploy native IPv6, making Protocol 41 redundant. In the short term, innovation should be encouraged by allowing Protocol 41 for certain users. Unfortunately, if this is not done, the 6to4 problem cannot be solved.

Operators should never use "bogon" address space such as the example of 1.1.1.0/24 for customers, since IPv4 exhaustion means that all such addresses are likely to be in real use in the near future. (Also see [I-D.ietf-intarea-shared-addressing-issues].) An operator

that is unable to immediately drop this practice should ensure that 192.88.99.1 generates IPv4 'destination unreachable'. It has been suggested that they could also run a dummy 6to4 relay at that address which always returns ICMPv6 'destination unreachable' as a 6to4 packet. However, these techniques are not very effective, since most current end-user 6to4 implementations will ignore them.

If an operator is providing legitimate global addresses to customers (neither RFC 1918 nor bogon addresses), and also running Carrier Grade NAT (Large Scale NAT) between this address space and the global address space of the Internet, then 6to4 cannot work properly. Such an operator should also take care to return 'destination unreachable' for 6to4 traffic. Alternatively, they could offer untranslated address space to the customers concerned.

A customer who is intentionally using 6to4 may also need to create AAAA records, and the operator should be able to support this, even if the DNS service itself runs exclusively over IPv4. However, customers should be advised to consider carefully whether their 6to4 service is sufficiently reliable for this.

Operators could, in principle, offer reverse DNS support for 6to4 users [RFC5158], although this is not straightforward for domestic customers.

Finally, enterprise operators who have complete administrative control of all end-systems may choose to disable 6to4 in those systems as an integral part of their plan to deploy IPv6.

4.2.1. 6to4 as the first step to IPv6 operation

An IPv4 operator could choose to install a well-managed 6to4 relay, connected to an IPv6-in-IPv4 tunnel to an IPv6 operator. This could serve as a small first step before the operator proceeds to native IPv6 deployment. The routing guidelines in Section 4.4 would apply.

4.3. Consumer ISPs, and enterprise networks, that do support IPv6

Once an operator does support IPv6 service, whether experimentally or in production, it is almost certain that users will get better results using this service than by continuing to use 6to4. Therefore, these operators are encouraged to advise their users to disable 6to4 and they should not create DNS records for any 6to4 addresses.

Such an operator may automatically fall into one of the following two categories (transit provider or content provider), so the guidelines in Section 4.4 or in Section 4.5 will apply instead.

Operators in this category should make sure that no routers are unintentionally or by default set up as active 6to4 relays. Unmanaged 6to4 relays will be a source of problems.

4.4. Transit ISPs and Internet Exchange Points

We assume that transit ISPs and IXPs have IPv6 connectivity. To reduce the negative impact of Anycast 6to4 on all their client networks, it is strongly recommended that they each run an Anycast 6to4 relay service. This will have the additional advantage that they will terminate the 6to4 IPv4 packets, and can then forward the decapsulated IPv6 traffic according to their own policy. Otherwise, they will blindly forward all the encapsulated IPv6 traffic to a competitor who does run a relay.

It is of critical importance that routing to this service is carefully managed:

1. The IPv4 prefix 192.88.99.0/24 must be announced only towards client IPv4 networks whose outbound 6to4 packets will be accepted.
2. The IPv6 prefix 2002::/16 must be announced towards native IPv6. The relay must accept all traffic towards 2002::/16 that reaches it, so the scope reached by this announcement should be carefully planned. It must reach all client IPv6 networks of the transit ISP or IXP. If it reaches a wider scope, the relay will be offering a free ride to non-clients.
3. The evidence is mixed, but it seems best to ensure that when the relay sends 6to4 packets back towards a 6to4 user, they should have 192.88.99.1 as their IPv4 source address (not the relay's unicast IPv4 address). This is to avoid problems if the user is behind a stateful firewall that drops inbound packets from addresses that have not been seen in outbound traffic.
4. The relay should be capable of responding correctly to ICMPv6 echo requests encapsulated in IPv4 protocol 41, typically with outer destination address 192.88.99.1 and inner destination address 2002:c058:6301::. (As noted previously, some 6to4 hosts are known to send echo requests with Hop Limit = 1, which allows them to rapidly detect the presence or absence of a relay in any case, but operators cannot rely on this behaviour.)
5. Protocol 41 must not be filtered in any IPv4 network or firewalls.
6. As a matter of general practice, which is essential for 6to4 to work well, IPv6 PMTUD must be possible, which means that ICMPv6 must not be blocked anywhere [RFC4890]. This also requires that the relay has a sufficiently high ICMP error generation threshold. For a busy relay, a typical default rate limit of 100 packets per second is too slow. On a busy relay, 1000pps or more might be needed. If ICMPv6 "Packet too Big" error messages are

rate-limited, users will experience PMTUD failure.

7. The relay must have adequate performance, and since load prediction is extremely hard, it must be possible to scale it up or, perhaps better, to replicate it as needed. Since the relay process is stateless, any reasonable method of load sharing between multiple relays will do.
8. The relay must of course be connected directly to global IPv4 space, with no NAT.

Operators in this category should make sure that no routers are unintentionally or by default set up as active 6to4 relays. Unmanaged 6to4 relays will be a source of problems.

4.5. Content providers and their ISPs

We assume that content providers and their ISPs have IPv6 connectivity, and that content servers are dual stacked. There is a need to avoid the situation where a client host, configured with Anycast 6to4, suceeds in sending an IPv6 packet to the server, but the 6to4 return path fails as described above. To avoid this, there must be a locally positioned 6to4 relay. Large content providers are advised to operate their own relays, and ISPs should do so in any case. There must be a 2002::/16 route from the content server to the relay. As noted in the previous section, the corresponding route advertisement must be carefully scoped, since any traffic that arrives for 2002::/16 must be relayed.

Such a relay may be dedicated entirely to return traffic, in which case it need not respond to the 6to4 anycast address.

Nevertheless, it seems wisest to ensure that when the relay sends 6to4 packets back towards a 6to4 user, they should have 192.88.99.1 as their IPv4 source address (not the relay's unicast IPv4 address). As noted above, this is to avoid problems if the user is behind a stateful firewall that drops UDP packets from addresses that have not been seen in outbound traffic. However, it is also necessary that 192.88.99.1 is not blocked by upstream ingress filtering - this needs to be tested.

Without careful engineering, there is nothing to make the return path as short as possible. It is highly desirable to arrange the scope of advertisements for 2002::/16 such that content providers have a short path to the relay, and the relay should have a short path to the ISP border. Care should be taken about shooting off advertisements for 2002::/16 into BGP4; they will become traffic magnets. If every ISP with content provider customers operates a relay, there will be no need for any of them to be advertised beyond each ISP's own customers.

Protocol 41 must not be filtered in the ISP's IPv4 network or firewallls. If the relays are placed outside the content provider's firewall, the latter may filter protocol 41 if desired.

The relay must have adequate performance, and since load prediction is extremely hard, it must be possible to scale it up or, perhaps better, to replicate it as needed. Since the relay process is stateless, any reasonable method of load sharing between multiple relays will do.

The relay must of course be connected directly to global IPv4 space, with no NAT.

An option for content servers is to embed the relay function directly in the content server. This is in fact trivial, since it can be achieved by enabling a local 6to4 interface on the server, and using it to route 2002::/16 for outbound packets. (This might not allow use of 192.88.99.1 as the source address.) Further details are to be found at <http://www.potaroo.net/ispcol/2010-05/v6hints.html>. However, in this case Protocol 41 must be allowed by the firewalls.

Content providers who do embed the relay function in this way could, in theory, accept inbound 6to4 traffic as well. This is highly unadvisable since, according the the rules of 6to4, they would then have to relay traffic for other IPv6 destinations too. So they should not be reachable via 192.88.99.1. Also, they should certainly not create an AAAA record for their 6to4 address - their inbound IPv6 access should be native, and advertising a 6to4 address might well lead to uRPF ingress filtering problems.

To avoid the path MTU problem described above, content servers should also set their IPv6 MTU to a safe value. From experience, 1280 bytes (the minimum allowed for IPv6) is recommended; again see <http://www.potaroo.net/ispcol/2010-05/v6hints.html>. Of course, ICMPv6 "Message too Big" must not be blocked or rate-limited anywhere [RFC4890].

To avoid the problem of missing reverse DNS delegations, content providers should not rely on these as a pseudo-security check for IPv6 clients.

Operators and content providers should make sure that no routers are unintentionally or by default set up as active 6to4 relays. Unmanaged 6to4 relays will be a source of problems.

5. Tunnels Managed by ISPs

There are various ways, such as tunnel brokers [RFC3053], 6rd [RFC5969], and the proposed 6a44 [I-D.despres-softwire-6a44], by which Internet Service Providers can provide tunneled IPv6 service to subscribers in a managed way, in which the subscriber will acquire an IPv6 prefix under a normal provider-based global IPv6 prefix. Most of the issues described for 6to4 do not arise in these scenarios. However, for tunnels used by clients behind a firewall, it is essential that IPv4 Protocol 41 is not blocked.

As a matter of general practice, IPv6 PMTUD must be possible, which means that ICMPv6 "Message too Big" must not be blocked or rate-limited anywhere [RFC4890].

6. Security Considerations

There is a general discussion of security issues for IPv6-in-IPv4 tunnels in [I-D.ietf-v6ops-tunnel-security-concerns], and [RFC3964] discusses 6to4 security. In summary, tunnels create a challenge for many common security mechanisms, simply because a potentially suspect packet is encapsulated inside a harmless outer packet. All these considerations apply to the automatic mechanisms discussed in this document. However, it should be noted that if an operator provides well managed servers and relays for 6to4, non-encapsulated IPv6 packets will pass through well defined points (the native IPv6 interfaces of those servers and relays) at which security mechanisms may be applied.

A blanket recommendation to block Protocol 41 is not compatible with mitigating the 6to4 problems described in this document.

7. IANA Considerations

This document makes no request of the IANA.

8. Acknowledgements

Useful comments and contributions were made by Emile Aben, Tore Anderson, Jack Bates, Cameron Byrne, Remi Despres, Jason Fesler, Wes George, Geoff Huston, Eric Kline, Victor Kuarsingh, Martin Levy, David Malone, Martin Millnert, Keith Moore, Pekka Savola, Mark Smith, Nathan Ward, James Woodyatt, and others.

This document was produced using the xml2rfc tool [RFC2629].

9. Change log

draft-carpenter-v6ops-6to4-teredo-advisory-02: updated after further comments, removed references to Teredo, 2011-02-24

draft-carpenter-v6ops-6to4-teredo-advisory-01: updated after WG discussion, 2011-02-10

draft-carpenter-v6ops-6to4-teredo-advisory-00: original version, 2011-02-03

10. Informative References

[I-D.despres-softwire-6a44]
Despres, R., Carpenter, B., and S. Jiang, "Native IPv6 Across NAT44 CPEs (6a44)", draft-despres-softwire-6a44-01 (work in progress), October 2010.

[I-D.ietf-6man-rfc3484-revise]
Matsumoto, A., Kato, J., and T. Fujisaki, "Update to RFC 3484 Default Address Selection for IPv6", draft-ietf-6man-rfc3484-revise-01 (work in progress), October 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-04 (work in progress), February 2011.

[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-04 (work in progress), October 2010.

[I-D.ietf-v6ops-v6-aaaa-whitelisting-implications]
Livingood, J., "IPv6 AAAA DNS Whitelisting Implications", draft-ietf-v6ops-v6-aaaa-whitelisting-implications-03 (work in progress), February 2011.

[I-D.vandevelde-v6ops-harmful-tunnels]
Velde, G., Troan, O., and T. Chown, "Non-Managed IPv6 Tunnels considered Harmful", draft-vandevelde-v6ops-harmful-tunnels-01 (work in progress), August 2010.

[I-D.wing-v6ops-happy-eyeballs-ipv6]
Wing, D. and A. Yourtchenko, "Happy Eyeballs: Trending Towards Success with Dual-Stack Hosts", draft-wing-v6ops-happy-eyeballs-ipv6-01 (work in progress), October 2010.

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

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

[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.

[RFC3068] Huitema, C., "An Anycast Prefix for 6to4 Relay Routers", RFC 3068, June 2001.

[RFC3964] Savola, P. and C. Patel, "Security Considerations for 6to4", RFC 3964, December 2004.

[RFC4890] Davies, E. and J. Mohacsi, "Recommendations for Filtering ICMPv6 Messages in Firewalls", RFC 4890, May 2007.

[RFC5158] Huston, G., "6to4 Reverse DNS Delegation Specification", RFC 5158, March 2008.

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

[Savola] Savola, P., "Observations of IPv6 Traffic on a 6to4 Relay", ACM SIGCOMM CCR 35 (1) 23-28, 2006.

Author's Address

Brian Carpenter
Department of Computer Science
University of Auckland
PB 92019
Auckland,   1142
New Zealand

Email: brian.e.carpenter@gmail.com