IPv6 Operations   E. Davies
Internet-Draft   Consultant
Expires: April 9, 2006   S. Krishnan
                                                          Ericsson
                                                         P. Savola
                                                         CSC/Funet
                                                   October 6, 2005

IPv6 Transition/Co-existence Security Considerations
draft-ietf-v6ops-security-overview-03.txt

Status of this Memo

Copyright Notice

Abstract

   The transition from a pure IPv4 network to a network where IPv4 and
   IPv6 co-exist brings a number of extra security considerations that
   need to be taken into account when deploying IPv6 and operating the
   dual-protocol network and the associated transition mechanisms. This
   document attempts to give an overview of the various issues grouped

   into three categories:
   o issues due to the IPv6 protocol itself,
   o issues due to transition mechanisms, and
   o issues due to IPv6 deployment.

Table of Contents

1. Introduction

The transition from a pure IPv4 network to a network where IPv4 and
IPv6 co-exist brings a number of extra security considerations that
need to be taken into account when deploying IPv6 and operating the
dual-protocol network with its associated transition mechanisms.
This document attempts to give an overview of the various issues
grouped into three categories:

o   issues due to the IPv6 protocol itself,

o   issues due to transition mechanisms, and

o   issues due to IPv6 deployment.

It is important to understand that we have to be concerned not about
replacing IPv4 with IPv6 (in the short term), but with adding IPv6 to
be operated in parallel with IPv4 [I-D.savola-v6ops-transarch].

This document also describes two matters which have been wrongly
identified as potential security concerns for IPv6 in the past and
explains why they are unlikely to cause problems: considerations
about probing/mapping IPv6 addresses (Appendix A), and considerations
with respect to privacy in IPv6 (Appendix B).

2. Issues Due to IPv6 Protocol

2.1. IPv6 Protocol-specific Issues

There are significant differences between the features of IPv6 and
IPv4: some of these specification changes may result in potential
security issues. Several of these issues have been discussed in
separate drafts but are summarized here to avoid normative references
which may not become RFCs. The following specification-related
problems have been identified, but this is not necessarily a complete
list:

2.1.1. Routing Headers and Hosts

All IPv6 nodes must be able to process Routing Headers [RFC2460].
This RFC can be interpreted, although it is not clearly stated, to
mean that all nodes (including hosts) must have this processing
enabled. This can result in hosts forwarding received traffic if
there are segments left in the Routing Header when it arrives at the
host.

A number of potential security issues associated with this behavior
were documented in [I-D.savola-ipv6-rh-hosts]. Some of these issues
have been resolved (a separate routing header type is now used for
Mobile IPv6 [RFC3775] and ICMP Traceback has not been standardized),

but two issues remain:

o Routing headers can be used to evade access controls based on
   destination addresses. This could be achieved by sending a packet
   ostensibly to a publicly accessible host address but with a
   routing header containing a 'forbidden' address. If the publicly
   accessible host is processing routing headers it will forward the
   packet to the destination address in the routing header which
   would have been forbidden by the packet filters if the address had
   been in the destination field when the packet was checked.

o If the packet source address in the previous case can be spoofed,
   any host could be used to mediate an anonymous reflection denial-
   of-service attack by having any publicly accessible host redirect
   the attack packets.

2.1.2. Routing Headers for Mobile IPv6 and Other Purposes

   In addition to the basic Routing Header (Type 0), which is intended
   to influence the trajectory of a packet through a network by
   specifying a sequence of router 'waypoints', Routing Header (Type 2)
   has been defined as part of the Mobile IPv6 specifications in
   [RFC3775]. The Type 2 Routing Header is intended for use by hosts to
   handle 'interface local' forwarding needed when packets are sent to
   the care-of address of a mobile node which is away from its home
   address.

   It is important that nodes treat the different types of routing
   header appropriately. It should be possible to apply separate
   filtering rules to the different types of Routing Header. By design,
   hosts must process Type 2 Routing Headers to support Mobile IPv6 but
   routers should not: to avoid the issues in Section 2.1.1 it may be
   desirable to forbid or limit the processing of Type 0 Routing Headers
   in hosts and some routers.

   Routing Headers are an extremely powerful and general capability.
   Alternative future uses of Routing Headers need to be carefully
   assessed to ensure that they do not open new avenues of attack that
   can be exploited.

2.1.3. Site-scope Multicast Addresses

   IPv6 supports multicast addresses with site scope which can
   potentially allow an attacker to identify certain important resources
   on the site if misused.

   Particular examples are the 'all routers' (FF05::2) and 'all Dynamic
   Host Configuration Protocol (DHCP) servers' (FF05::1:3) addresses
   defined in [RFC2375]: an attacker that is able to infiltrate a
   message destined for these addresses on to the site will potentially

receive in return information identifying key resources on the site.
This information can then be the target of directed attacks ranging
from simple flooding to more specific mechanisms designed to subvert
the device.

Some of these addresses have current legitimate uses within a site.
The risk can be minimized by ensuring that all firewalls and site
boundary routers are configured to drop packets with site scope
destination addresses. Also nodes should not join multicast groups
for which there is no legitimate use on the site and site routers
should be configured to drop packets directed to these unused
addresses.

2.1.4. ICMPv6 and Multicast

It is possible to launch a denial-of-service (DoS) attack using IPv6
which could be amplified by the multicast infrastructure.

Unlike ICMP for IPv4, ICMPv6 [RFC2463] allows error notification
responses to be sent when certain unprocessable packets are sent to
multicast addresses.

The cases in which responses are sent are:
o The received packet is longer than the next link MTU: 'Packet Too
   Big' responses are needed to support Path MTU Discovery for
   multicast traffic.
o The received packet contains an unrecognized option in a hop-by-
   hop or destination options extension header with the first two
   bits of the option type set to binary '10': 'Parameter Problem'
   responses are intended to inform the source that some or all of
   the recipients cannot handle the option in question.

If an attacker can craft a suitable packet sent to a multicast
destination, it may be possible to elicit multiple responses directed
at the victim (the spoofed source of the multicast packet). On the
other hand, the use of 'reverse path forwarding' checks to eliminate
loops in multicast forwarding automatically limits the range of
addresses which can be spoofed.

In practice an attack using oversize packets is unlikely to cause
much amplification unless the attacker is able to carefully tune the
packet size to exploit a network with smaller MTU in the edge than
the core. Similarly a packet with an unrecognized hop-by-hop option
would be dropped by the first router. However a packet with an
unrecognized destination option could generate multiple responses.

In addition to amplification, this kind of attack would potentially
consume large amounts of forwarding state resources in routers on

multicast-enabled networks. These attacks are discussed in more
detail in [I-D.savola-v6ops-firewalling].

2.1.5. Anycast Traffic Identification and Security

IPv6 introduces the notion of anycast addresses and services.
Originally the IPv6 standards disallowed using an anycast address as
the source address of a packet. Responses from an anycast server
would therefore supply a unicast address for the responding server.
To avoid exposing knowledge about the internal structure of the
network, it is recommended that anycast servers now take advantage of
the ability to return responses with the anycast address as the
source address if possible.

If the server needs to use a unicast address for any reason, it may
be desirable to consider using specialized addresses for anycast
servers which are not used for any other part of the network to
restrict the information exposed. Alternatively operators may wish
to restrict the use of anycast services from outside the domain, thus
requiring firewalls to filter anycast requests. For this purpose,
firewalls need to know which addresses are being used for anycast
services: these addresses are arbitrary and not distinguishable from
any other IPv6 unicast address by structure or pattern.

One particular class of anycast addresses that should be given
special attention is the set of Subnet-Router anycast addresses
defined in The IPv6 Addressing Architecture [RFC3513]. All routers
are required to support these addresses for all subnets for which
they have interfaces. For most subnets using global unicast
addresses, filtering anycast requests to these addresses can be
achieved by dropping packets with the lower 64 bits (the Interface
Identifier) set to all zeroes.

2.1.6. Address Privacy Extensions Interact with DDoS Defenses

The purpose of the privacy extensions for stateless address auto-
configuration [RFC3041][I-D.ietf-ipv6-privacy-addrs-v2] is to change
the interface identifier (and hence the global scope addresses
generated from it) from time to time. By varying the addresses used,
eavesdroppers and other information collectors find it more difficult
to identify which transactions actually relate to a specific node.

A security issue may result from this if the frequency of node
address change is sufficiently great to achieve the intended aim of
the privacy extensions: with a relatively high rate of change, the
observed behavior of the node could look very like that of a
compromised node which was the source of a distributed denial of
service (DDoS). It would thus be difficult for any future defenses

against DDoS attacks to distinguish between a high rate of change of
addresses resulting from genuine use of the privacy extensions and a
compromised node being used as the source of a DDoS with 'in-prefix'
spoofed source addresses as described in [I-D.dupont-ipv6-
rfc3041harmful].

Even if a node is well behaved, the change in the address could make
it harder for a security administrator to define a policy rule (e.g.
access control list) that takes into account a specific node.

2.1.7. Dynamic DNS: Stateless Address Auto-Configuration, Privacy Extensions and SEND

The introduction of Stateless Address Auto-Configuration (SLAAC)
[RFC2462] with IPv6 provides an additional challenge to the security
of Dynamic DNS (DDNS). With manual addressing or the use of DHCP,
the number of security associations that need to be maintained to
secure access to the DNS server is limited, assuming any necessary
updates are carried out by the DHCP server. This is true equally for
IPv4 and IPv6.

Since SLAAC does not make use of a single and potentially trusted
DHCP server, but depends on the node obtaining the address, securing
the insertion of updates into DDNS may need a security association
between each node and the DDNS server. This is discussed further in
[I-D.ietf-dnsop-ipv6-dns-issues].

Using the Privacy Extensions to SLAAC [RFC3041][I-D.ietf-ipv6-
privacy-addrs-v2] may significantly increase the rate of updates of
DDNS. Even if a node using the Privacy Extensions does not publish
its address for 'forward' lookup (as that would effectively
compromise the privacy which it is seeking), it may still need to
update the reverse DNS records so that reverse routability checks can
be carried out. If the rate of change needed to achieve real privacy
has to be increased as is mentioned in Section 2.1.6 the update rate
for DDNS may be excessive.

Similarly, the cryptographically generated addresses used by SEND
[RFC3971] are expected to be periodically regenerated in line with
recommendations for maximum key lifetimes. This regeneration could
also impose a significant extra load on DDNS.

2.1.8. Extension Headers

A number of issues relating to the specification of IPv6 Extension
headers have been identified. Several of these are discussed in
[I-D.savola-v6ops-firewalling].

2.1.8.1. **Processing Extension Headers in Middleboxes**

   In IPv4 deep packet inspection techniques are used to implement
   policing and filtering both as part of routers and in middleboxes
   such as firewalls. Fully extending these techniques to IPv6 would
   require inspection of all the extension headers in a packet. This is
   essential to ensure that policy constraints on the use of certain
   headers and options are enforced and to remove, at the earliest
   opportunity, packets containing potentially damaging unknown options.

   This requirement appears to conflict with Section 4 of the IPv6
   specification in [RFC2460] which requires that destination options
   are not processed at all until the packet reaches the appropriate
   destination (either the final destination or a routing header
   waypoint).

   Also [RFC2460] forbids processing the headers other than in the order
   in which they appear in the packet.

   A further ambiguity relates to whether an intermediate node should
   discard a packet which contains a header or destination option which
   it does not recognize. If the rules above are followed slavishly, it
   is not (or may not be) legitimate for the intermediate node to
   discard the packet because it should not be processing those headers
   or options.

   [RFC2460] therefore does not appear to take account of the behavior
   of middleboxes and other non-final destinations which may be
   inspecting the packet, and thereby potentially limits the security
   protection of these boxes.

2.1.8.2. **Processing Extension Header Chains**

   There is a further problem for middleboxes that want to examine the
   transport headers which are located at the end of the IPv6 header
   chain. In order to locate the transport header or other protocol
   data unit, the node has to parse the header chain.

   The IPv6 specification [RFC2460] does not mandate the use of the
   Type-Length-Value format with a fixed layout for the start of each
   header although it is used for the majority of headers currently
   defined. (Only the Type field is guaranteed in size and offset).

   A middlebox cannot therefore guarantee to be able to process header
   chains which may contain headers defined after the box was
   manufactured. As noted in Section 2.1.8.1, middleboxes ought not to
   have to know about all header types in use but still need to be able
   to skip over such headers to find the transport PDU start. This

either limits the security which can be applied in firewalls or makes
it difficult to deploy new extension header types.

At the time of writing, only the Fragment Header does not fully
conform to the TLV format used for other extension headers. In
practice, many firewalls reconstruct fragmented packets before
performing deep packet inspection, so this divergence is less
problematic than it might have been, and is at least partially
justified because the full header chain is not present in all
fragments.

Destination Options may also contain unknown options. However, the
options are encoded in TLV format so that intermediate nodes can skip
over them during processing, unlike the enclosing extension headers.

2.1.8.3. Unknown Headers/Destination Options and Security Policy

A strict security policy might dictate that packets containing either
unknown headers or destination options are discarded by firewalls or
other filters. This requires the firewall to process the whole
extension header chain which may be currently in conflict with the
IPv6 specification as discussed in Section 2.1.8.1.

Even if the firewall does inspect the whole header chain, it may not
be sensible to discard packets with items unrecognized by the
firewall: the intermediate node has no knowledge of which options and
headers are implemented in the destination node. Hence it is highly
desirable to make the discard policy configurable. This will avoid
firewalls dropping packets with legitimate items that they do not
recognize because their hardware or software is not aware of a new
definition.

2.1.8.4. Excessive Hop-by-Hop Options

IPv6 does not limit the number of hop by hop options which can be
present in a hop-by-hop option header. The lack of a limit can be
used to mount denial of service attacks affecting all nodes on a path
as described in [I-D.krishnan-ipv6-hopbyhop].

2.1.8.5. Overuse of Router Alert Option

The IPv6 router alert option specifies a hop-by-hop option that, if
present, signals the router to take a closer look at the packet.
This can be used for denial of service attacks. By sending a large
number of packets containing a router alert option an attacker can
deplete the processor cycles on the routers available to legitimate
traffic.

2.1.9. Fragmentation: Reassembly and Deep Packet Inspection

The current specifications of IPv6 in [RFC2460] do not mandate any
minimum packet size for the fragments of a packet before the last
one, except for the need to carry the unfragmentable part in all
fragments.

The unfragmentable part does not include the transport port numbers
so that it is possible that the first fragment does not contain
sufficient information to carry out deep packet inspection involving
the port numbers.

Also the reassembly rules for fragmented packets in [RFC2460] do not
mandate behavior which would minimize the effects of overlapping
fragments.

Depending on the implementation of packet reassembly and the
treatment of packet fragments in firewalls and other nodes which use
deep packet inspection for traffic filtering, this potentially leaves
IPv6 open to the sort of attacks described in [RFC1858] and [RFC3128]
for IPv4.

There is no reason to allow overlapping packet fragments and overlaps
could be prohibited in a future revision of the protocol
specification. Some implementations already drop all packets with
overlapped fragments.

Specifying a minimum size for packet fragments does not help in the
same way as it does for IPv4 because IPv6 extension headers can be
made to appear very long: an attacker could insert one or more
undefined destination options with long lengths and the 'ignore if
unknown' bit set. Given the guaranteed minimum MTU of IPv6 it seems
reasonable that hosts should be able to ensure that the transport
port numbers are in the first fragment in almost all cases and that
deep packet inspection should be very suspicious of first fragments
that do not contain them.

2.1.10. Fragmentation Related DoS Attacks

Packet reassembly in IPv6 hosts also opens up the possibility of
various fragment-related security attacks. Some of these are
analogous to attacks identified for IPv4. Of particular concern is a
DoS attack based on sending large numbers of small fragments without
a terminating last fragment which would potentially overload the
reconstruction buffers and consume large amounts of CPU resources.

Mandating the size of packet fragments could reduce the impact of
this kind of attack by limiting the rate at which fragments could

arrive and limiting the number of fragments which need to be
processed.

[2.1.11](). **Link-Local Addresses and Securing Neighbor Discovery**

All IPv6 nodes are required to configure a link-local address on each
interface. This address is used to communicate with other nodes
directly connected to the link accessed via the interface, especially
during the neighbor discovery and auto-configuration processes.
Link-local addresses are fundamental to the operation of the Neighbor
Discovery Protocol (NDP) [RFC2461] and SLAAC [RFC2462]. NDP also
provides the functionality of associating link layer and IP addresses
provided by the Address Resolution Protocol (ARP) in IPv4 networks.

The standard version of NDP is subject to a number of security
threats related to ARP spoofing attacks on IPv4. These threats have
been documented in [RFC3756] and mechanisms to combat them specified
in SEcure Neighbor Discovery (SEND) [RFC3971]. SEND is an optional
mechanism which is particularly applicable to wireless and other
environments where it is difficult to physically secure the link.

Because the link-local address can, by default, be acquired without
external intervention or control, it allows an attacker to commence
communication on the link without needing to acquire information
about the address prefixes in use or communicate with any authorities
on the link. This feature gives a malicious node the opportunity to
mount an attack on any other node which is attached to this link;
this vulnerability exists in addition to possible direct attacks on
NDP. Link-local addresses may also facilitate the unauthorized use
of the link bandwidth ('bandwidth theft') to communicate with another
unauthorized node on the same link.

Link-local addresses allocated from the prefix 169.254.0.0/16 are
available in IPv4 as well and procedures for using them are described
in [I-D.ietf-zeroconf-ipv4-linklocal] but the security issues were
not as pronounced as for IPv6 for the following reasons:
o link-local addresses are not mandatory in IPv4 and are primarily
   intended for isolated or ad hoc networks that cannot acquire a
   routable IPv4 address by other means,
o IPv4 link-local addresses are not universally supported across
   operating systems, and
o the IPv4 link-local address should be removed when a non-link-
   local address is configured on the interface and will generally
   not be allocated unless other means of acquiring an address are
   not available.

These vulnerabilities can be mitigated in several ways. A general
solution will require

   o authenticating the link layer connectivity, for example by using
      IEEE 802.1x functionality, port-based MAC address security
      (locking), or physical security, and

   o using SEcure Neighbor Discovery (SEND) to create a
      cryptographically generated link-local address as described in
      [RFC3971] which is tied to the authenticated link layer address.
   This solution would be particularly appropriate in wireless LAN
   deployments where it is difficult to physically secure the
   infrastructure

   In wired environments, where the physical infrastructure is
   reasonably secure, it may be sufficient to ignore communication
   requests originating from a link-local address for other than local
   network management purposes. This requires that nodes should only
   accept packets with link-local addresses for a limited set of
   protocols including NDP, MLD and other functions of ICMPv6.

2.1.12. Securing Router Advertisements

   As part of the Neighbor Discovery process, routers on a link
   advertise their capabilities in Router Advertisement messages. The
   version of NDP defined in [RFC2461] does not protect the integrity of
   these messages or validate the assertions made in the messages with
   the result that any node which connects to the link can maliciously
   claim to offer routing services which it will not fulfill, and
   advertise inappropriate prefixes and parameters. These threats have
   been documented in [RFC3756].

   A malicious node may also be able to carry out a DoS attack by
   deprecating an established valid prefix (by advertising it with a
   zero lifetime). Similar DoS attacks are possible if the optional
   Router Selection mechanism is implemented as described in the
   security considerations of [I-D.ietf-ipv6-router-selection].

   SEND [RFC3971] can be used to provide verification that routers are
   authorized to provide the services they advertise through a
   certificate-based mechanism. This capability of SEND is also
   particularly appropriate for wireless environments where clients are
   reliant on the assertions of the routers rather than a physically
   secured connection.

2.1.13. Host to Router Load Sharing

   If a host deploys the optional Host to Router Load Sharing mechanism
   [I-D.ietf-ipv6-host-load-sharing] a malicious application could try
   to subvert the load sharing algorithm to direct a large volume of
   traffic to a subset of the routers.

However, as described in [I-D.ietf-ipv6-host-load-sharing], this is
not considered a significant threat as

1. load-sharing routers should be provisioned to handle the load in
    any case (e.g., if one of them goes down),

2. a privileged user can launch such an attack using raw sockets in
    any case, and

3. a user or application could just attack the routers directly or
    in other, simpler ways.

2.1.14. Mobile IPv6

Mobile IPv6 offers significantly enhanced security compared with
Mobile IPv4 especially when using optimized routing and care-of
addresses. Return routability checks are used to provide relatively
robust assurance that the different addresses which a mobile node
uses as it moves through the network do indeed all refer to the same
node. The threats and solutions are described in [RFC3775] and a
more extensive discussion of the security aspects of the design can
be found in [I-D.ietf-mip6-ro-sec].

2.1.14.1. Obsolete Home Address Option in Mobile IPv6

The Home Address option specified in early drafts of Mobile IPv6
would have allowed a trivial source spoofing attack: hosts were
required to substitute the source address of incoming packets with
the address in the option, thereby potentially evading checks on the
packet source address. This is discussed at greater length in
[I-D.savola-ipv6-rh-ha-security]. The version of Mobile IPv6 as
standardized in [RFC3775] has removed this issue by ensuring that the
Home Address destination option is only processed if there is a
corresponding binding cache entry and securing Binding Update
messages.

A number of pre-standard implementations of Mobile IPv6 were
available which implemented this obsolete and insecure option: care
should be taken to avoid running such obsolete systems.

2.2. IPv4-mapped IPv6 Addresses

Overloaded functionality is always a double-edged sword: it may yield
some deployment benefits, but often also incurs the price which comes
with ambiguity.

One example of such is IPv4-mapped IPv6 addresses: a representation
of an IPv4 address as an IPv6 address inside an operating system.
Since the original specification, the use of IPv4-mapped addresses
has been extended to a transition mechanism, Stateless IP/ICMP
Translation algorithm (SIIT) [RFC2765], where they are potentially

used in the addresses of packets on the wire.

Therefore, it becomes difficult to unambiguously discern whether an
IPv4 mapped address is really an IPv4 address represented in the IPv6
address format *or* an IPv6 address received from the wire (which may
be subject to address forgery, etc.).

In addition, special cases like these, while giving deployment
benefits in some areas, require a considerable amount of code
complexity (e.g. in the implementations of bind() system calls and
reverse DNS lookups) which is probably undesirable. Some of these
issues are discussed in [I-D.cmetz-v6ops-v4mapped-api-harmful] and
[I-D.itojun-v6ops-v4mapped-harmful].

In practice, although the packet translation mechanisms of SIIT are
specified for use in the Network Address Translator - Protocol
Translator (NAT-PT) [RFC2765], NAT-PT uses a mechanism different from
IPv4-mapped IPv6 addresses for communicating embedded IPv4 addresses
in IPv6 addresses. Also SIIT is not recommended for use as a
standalone transition mechanism. Given the issues that have been
identified, it seems appropriate that mapped addresses should not be
used on the wire. However, changing application behavior by
deprecating the use of mapped addresses in the operating system
interface would have significant impact on application porting
methods [RFC4038] and needs further study.

2.3. Increased End-to-End Transparency

One of the major design aims of IPv6 has been to maintain the
original IP architectural concept of end-to-end transparency.
Transparency can help foster technological innovation in areas such
as peer-to-peer communication but maintaining the security of the
network at the same time requires some modifications in the network
architecture. Ultimately, it is also likely to need changes in the
security model as compared with the norms for IPv4 networks.

2.3.1. IPv6 Networks without NATs

The necessity of introducing Network Address Translators (NATs) into
IPv4 networks, resulting from a shortage of IPv4 addresses, has
removed the end-to-end transparency of most IPv4 connections: the use
of IPv6 would restore this transparency. However, the use of NATs,
and the associated private addressing schemes, has become
inappropriately linked to the provision of security in enterprise
networks. The restored end-to-end transparency of IPv6 networks can
therefore be seen as a threat by poorly informed enterprise network
managers. Some seem to want to limit the end-to-end capabilities of
IPv6, for example by deploying private, local addressing and

translators, even when it is not necessary because of the abundance
of IPv6 addresses.

Recommendations for designing an IPv6 network to meet the perceived
security and connectivity requirements implicit in the current usage
of IPv4 NATs whilst maintaining the advantages of IPv6 end-to-end
transparency are described in IPv6 Network Architecture Protection
[I-D.ietf-v6ops-nap].

2.3.2. Enterprise Network Security Model for IPv6

The favored model for enterprise network security in IPv4 stresses
the use of a security perimeter policed by autonomous firewalls and
incorporating the NATs. Both perimeter firewalls and NATs introduce
asymmetry and reduce the transparency of communications through these
perimeters. The symmetric bidirectionality and transparency which
are extolled as virtues of IPv6 may seem to be at odds with this
model. Consequently network managers may even see them as
undesirable attributes, in conflict with their need to control
threats to and attacks on the networks they administer.

It is worth noting that IPv6 does not *require* end-to-end
connectivity. It merely provides end-to-end addressability; the
connectivity can still be controlled using firewalls (or other
mechanisms), and it is indeed wise to do so.

A number of matters indicate that IPv6 networks should migrate
towards an improved security model, which will increase the overall
security of the network but facilitate end-to-end communication:
o Increased usage of end-to-end security especially at the network
   layer. IPv6 mandates the provision of IPsec capability in all
   nodes and increasing usage of end-to-end security is a challenge
   to current autonomous firewalls that are unable to perform deep
   packet inspection on encrypted packets. It is also incompatible
   with NATs because they modify the packets, even when packets are
   only authenticated rather than encrypted.
o Acknowledgement that over-reliance on the perimeter model is
   potentially dangerous. An attacker who can penetrate today's
   perimeters will have free rein within the perimeter, in many
   cases. Also a successful attack will generally allow the attacker
   to capture information or resources and make use of them.
o Development of mechanisms such as 'Trusted Computing' which will
   increase the level of trust which network managers are able to
   place on hosts.
o Development of centralized security policy repositories and secure
   distribution mechanisms which, in conjunction with trusted hosts,
   will allow network managers to place more reliance on security
   mechanisms at the end points. The mechanisms are likely to

include end-node firewalling and intrusion detection systems as
well as secure protocols that allow end points to influence the
behavior of perimeter security devices.
o Review of the role of perimeter devices with increased emphasis on
intrusion detection, network resource protection and coordination
to thwart distributed denial of service attacks.

Several of the technologies required to support an enhanced security
model are still under development, including secure protocols to
allow end points to control firewalls: the complete security model
utilizing these technologies is now emerging but still requires some
development.

In the meantime, initial deployments will need to make use of similar
firewalling and intrusion detection techniques to IPv4 which may
limit end-to-end transparency temporarily, but should be prepared to
use the new security model as it develops and avoid the use of NATs
by the use of the architectural techniques described in [I-D.ietf-
v6ops-nap]. In particular, using NAT-PT [RFC2766] as a general
purpose transition mechanism should be avoided as it is likely to
limit the exploitation of end-to-end security and other IPv6
capabilities in future as explained in [I-D.ietf-v6ops-natpt-to-
exprmntl].

3. Issues Due to Transition Mechanisms

3.1. IPv6 Transition/Co-existence Mechanism-specific Issues

The more complicated the IPv6 transition/co-existence becomes, the
greater the danger that security issues will be introduced either
o in the mechanisms themselves,
o in the interaction between mechanisms, or
o by introducing unsecured paths through multiple mechanisms.
These issues may or may not be readily apparent. Hence it would be
desirable to keep the mechanisms simple, as few in number as possible
and built from as small pieces as possible to simplify analysis.

One case where such security issues have been analyzed in detail is
the 6to4 tunneling mechanism [RFC3964].

As tunneling has been proposed as a model for several more cases than
are currently being used, its security properties should be analyzed
in more detail. There are some generic dangers to tunneling:

o it may be easier to avoid ingress filtering checks

o it is possible to attack the tunnel interface: several IPv6
   security mechanisms depend on checking that Hop Limit equals 255
   on receipt and that link-local addresses are used. Sending such
   packets to the tunnel interface is much easier than gaining access
   to a physical segment and sending them there.
o automatic tunneling mechanisms are typically particularly
   dangerous as there is no pre-configured association between end
   points. Accordingly, at the receiving end of the tunnel packets
   have to be accepted and decapsulated from any source.
   Consequently, special care should be taken when specifying
   automatic tunneling techniques.

3.2. Automatic Tunneling and Relays

Two mechanisms have been (or are being) specified which use automatic
tunneling and are intended for use outside a single domain. These
mechanisms encapsulate the IPv6 packet directly in an IPv4 packet in
the case of 6to4 [RFC3056] or in an IPv4 UDP packet in the case of
Teredo [I-D.huitema-v6ops-teredo]. In each case packets can be sent
and received by any similarly equipped nodes in the IPv4 Internet.

As mentioned in Section 3.1, a major vulnerability in such approaches
is that receiving nodes must allow decapsulation of traffic sourced
from anywhere in the Internet. This kind of decapsulation function
must be extremely well secured because of the wide range of potential
sources.

An even more difficult problem is how these mechanisms are able to
establish communication with native IPv6 nodes or between the
automatic tunneling mechanisms: such connectivity requires the use of
some kind of "relay". These relays could be deployed in various
locations such as:
o all native IPv6 nodes,
o native IPv6 sites,
o in IPv6-enabled ISPs, or
o just somewhere in the Internet.

Given that a relay needs to trust all the sources (e.g., in the 6to4
case, all 6to4 routers) which are sending it traffic, there are
issues in achieving this trust and at the same time scaling the relay
system to avoid overloading a small number of relays.

As authentication of such a relay service is very difficult to
achieve, and particularly so in some of the possible deployment
models, relays provide a potential vehicle for address spoofing,
(reflected) Denial-of-Service attacks, and other threats.

Threats related to 6to4 and measures to combat them are discussed in

[RFC3964]. [I-D.huitema-v6ops-teredo] incorporates extensive
discussion of the threats to Teredo and measures to combat them.

3.3. Tunneling IPv6 Through IPv4 Networks May Break IPv4 Network Security Assumptions

NATs and firewalls have been deployed extensively in the IPv4
Internet, as discussed in Section 2.3. Operators who deploy them
typically have some security/operational requirements in mind (e.g. a
desire to block inbound connection attempts), which may or may not be
misguided.

The addition of tunneling can change the security model which such
deployments are seeking to enforce. IPv6-over-IPv4 tunneling using
protocol 41 is typically either explicitly allowed, or disallowed
implicitly. Tunneling IPv6 over IPv4 encapsulated in UDP constitutes
a more difficult problem as UDP must usually be allowed to pass
through NATs and firewalls. Consequently, using UDP implies the
ability to punch holes in NAT's and firewalls although, depending on
the implementation, this ability may be limited or only achieved in a
stateful manner. In practice, the mechanisms have been explicitly
designed to traverse both NATs and firewalls in a similar fashion.

One possible view is that use of tunneling is especially questionable
in home/SOHO environments where the level of expertise in network
administration is typically not very high; in these environments the
hosts may not be as tightly managed as in others (e.g., network
services might be enabled unnecessarily), leading to possible
security break-ins or other vulnerabilities.

Holes can be punched both intentionally and unintentionally. In
cases where the administrator or user makes an explicit decision to
create the hole, this is less of a problem, although (for example)
some enterprises might want to block IPv6 tunneling explicitly if
employees were able to create such holes without reference to
administrators. On the other hand, if a hole is punched
transparently, it is likely that a proportion of users will not
understand the consequences: this will very probably result in a
serious threat sooner or later.

When deploying tunneling solutions, especially tunneling solutions
which are automatic and/or can be enabled easily by users who do not
understand the consequences, care should be taken not to compromise
the security assumptions held by the users.

For example, NAT traversal should not be performed by default unless
there is a firewall producing a similar by-default security policy to
that provided by IPv4 NAT. IPv6-in-IPv4 (protocol 41) tunneling is

less of a problem, as it is easier to block if necessary; however, if
the host is protected in IPv4, the IPv6 side should be protected as
well.

As has been shown in Appendix A, it is relatively easy to determine
the IPv6 address corresponding to an IPv4 address in tunneling
deployments. It is therefore vital NOT to rely on "security by
obscurity" i.e., assuming that nobody is able to guess or determine
the IPv6 address of the host especially when using automatic
tunneling transition mechanisms.

The network architecture must provide separate IPv4 and IPv6
firewalls with tunneled IPv6 traffic arriving encapsulated in IPv4
packets routed through the IPv4 firewall before being decapsulated,
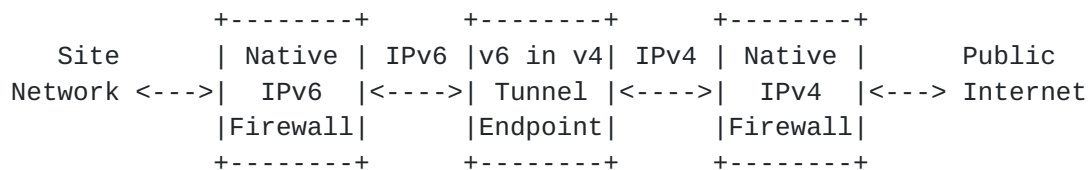and then through the IPv6 firewall as shown in Figure 1.

```
                +--------+       +--------+       +--------+
     Site       | Native | IPv6 |v6 in v4| IPv4 | Native |       Public
   Network <--->|  IPv6  |<---->| Tunnel |<---->|  IPv4  |<---> Internet
                |Firewall|       |Endpoint|       |Firewall|
                +--------+       +--------+       +--------+
```

Figure 1: Tunneled Traffic and Firewalls

4. Issues Due to IPv6 Deployment

4.1. Avoiding the Trap of Insecure IPv6 Service Piloting

Because IPv6 is a new service for many networks, network managers
will often opt to make a pilot deployment in a part of the network to
gain experience and understand the problems as well as the benefits
that may result from a full production quality IPv6 service.

Unless IPv6 service piloting is done in a manner which is as secure
as possible there is a risk that security in the pilot which does not
match up to what is achievable with current IPv4 production service
can adversely impact the overall assessment of the IPv6 pilot
deployment. This may result in a decision to delay or even avoid
deploying an IPv6 production service. For example, hosts and routers
might not be protected by IPv6 firewalls, even if the corresponding
IPv4 service is fully protected by firewalls as described in
[I-D.ietf-v6ops-v6onbydefault]. This is particularly critical where
IPv6 capabilities are turned on by default in new equipment or new
releases of operating systems: network managers may not be fully
aware of the security exposure that this creates.

In some cases a perceived lack of availability of IPv6 firewalls and

other security capabilities, such as intrusion detection systems may
have lead network managers to resist any kind of IPv6 service
deployment. These problems may be partly due to the relatively slow
development and deployment of IPv6-capable security equipment, but
the major problems appear to have been a lack of information, and
more importantly a lack of documented operational experience on which
managers can draw. In actual fact, as of the time of writing (2005)
there are a significant number of alternative IPv6 packet filters and
firewalls already in existence, which could be used for provide
sufficient access controls.

However, there are a small number of areas that where the available
equipment and capabilities may still be a barrier to secure
deployment:
o 'Personal firewalls' intended for use on hosts are not yet widely
   available.
o Enterprise firewalls are at an early stage of development and may
   not provide the full range of capabilities needed to implement the
   necessary IPv6 filtering rules. network managers often expect the
   same devices that support and are used for IPv4 today to also
   become IPv6-capable -- even though this is not really required and
   the equipment may not have the requisite hardware capabilities to
   support fast packet filtering for IPv6. Suggestions for the
   appropriate deployment of firewalls are given in Section 3.3 -- as
   will be seen from this section it is usually desirable that the
   firewalls are in separate boxes and there is no necessity for them
   to be same model of equipment.
o A lesser factor may be that some design decisions in the IPv6
   protocol make it more difficult for firewalls to be implemented
   and work in all cases, and to be fully future proof (e.g. when new
   extension headers are used) as discussed in Section 2.1.8: it is
   significantly more difficult for intermediate nodes to process the
   IPv6 header chains than IPv4 packets.
o Adequate Intrusion Detection Systems (IDS) are more difficult to
   construct for IPv6. IDSs are now beginning to become available
   but the pattern-based mechanisms used for IPv4 may not be the most
   appropriate for long-term development of these systems as end-to-
   end encryption becomes more prevalent. Future systems may be more
   reliant on traffic flow pattern recognition.
o Implementations of high availability capabilities supporting IPv6
   are also in short supply. In particular, development of the IPv6
   version of the Virtual Router Redundancy Protocol (VRRP)
   [I-D.ietf-vrrp-ipv6-spec] has lagged the development of the main
   IPv6 protocol although alternatives may be available for some
   environments.

In all these areas developments are ongoing and they should not be
considered a long-term bar to the deployment of IPv6 either as a

pilot or production service. The necessary tools are now available
to make a secure IPv6 deployment and with careful selection of
components and design of the network architecture a successful pilot
or production IPv6 service can be deployed. Recommendations for
secure deployment and appropriate management of IPv6 networks can be
found in the documentation archives of the European Union 6net
project [SIXNET] and in the Deployment Guide published by the IPv6
Promotion Council of Japan [JpIPv6DC].

4.2. DNS Server Problems

Some DNS server implementations have flaws that severely affect DNS
queries for IPv6 addresses as discussed in [RFC4074]. These flaws
can be used for DoS attacks affecting both IPv4 and IPv6 by inducing
caching DNS servers to believe that a domain is broken and causing
the server to block access to all requests for the domain for a
precautionary period.

4.3. Addressing Schemes and Securing Routers

Whilst in general terms brute force scanning of IPv6 subnets is
essentially impossible due to the enormously larger address space of
IPv6 and the 64 bit interface identifiers (see Appendix A), this will
be obviated if administrators do not take advantage of the large
space to use unguessable interface identifiers.

Because the unmemorability of complete IPv6 addresses there is a
temptation for administrators to use small integers as interface
identifiers when manually configuring them, as might happen on point-
to-point links or when provisioning complete addresses from a DHCPv6
server. Such allocations make it easy for an attacker to find active
nodes that they can then port scan.

To make use of the larger address space properly, administrators
should be very careful when entering IPv6 addresses in their
configurations (e.g. Access Control List), since numerical IPv6
addresses are more prone to human error than IPv4 due to their length
and unmemorability.

It is also essential to ensure that the management interfaces of
routers are well secured as the router will usually contain a
significant cache of neighbor addresses in its neighbor cache.

4.4. Consequences of Multiple Addresses in IPv6

One positive consequence of IPv6 is that nodes which do not require
global access can communicate locally just by the use of a link-local
address (if very local access is sufficient) or across the site by

using a Unique Local Address (ULA). In either case it is easy to
ensure that access outside the assigned domain of activity can be
controlled by simple filters (which may be the default for link-
locals). However, the security hazards of using link-local addresses
for non-management purposes as documented in Section 2.1.11 should be
borne in mind.

On the other hand, the possibility that a node or interface can have
multiple global scope addresses makes access control filtering both
on ingress and egress more complex and requires higher maintenance
levels. Vendors and network administrators need to be aware that
multiple addresses are the norm rather than the exception in IPv6:
when building and selecting tools for security and management a
highly desirable feature is the ability to be able to 'tokenize'
access control lists and configurations in general to cater for
multiple addresses and/or address prefixes.

The addresses could be from the same network prefix (for example,
privacy mechanisms [RFC3041][I-D.ietf-ipv6-privacy-addrs-v2] will
periodically create new addresses taken from the same prefix and two
or more of these may be active at the same time), or from different
prefixes (for example, when a network is multihomed or is
implementing anycast services). In either case, it is possible that
a single host could be using several different addresses with
different prefixes. It would be desirable that the security
administrator should be able to identify that the same host is behind
all these addresses.

Some network administrators may find the mutability of addresses when
privacy mechanisms are used in their network to be undesirable
because of the current difficulties in maintaining access control
lists and knowing the origin of traffic. In general, disabling the
use of privacy addresses is only possible if the full stateful DHCPv6
mechanism [RFC3315] is used to allocate IPv6 addresses and DHCPv6
requests for privacy addresses are not honored.

4.5. Deploying ICMPv6

In IPv4 it is commonly accepted that some filtering of ICMP packets
by firewalls is essential to maintain security. Because of the
extended use that is made of ICMPv6 [RFC2461] with a multitude of
functions, the simple set of dropping rules that are usually applied
in IPv4 need to be significantly developed for IPv6. The blanket
dropping of all ICMP messages that is used in some very strict
environments is simply not possible for IPv6.

In an IPv6 firewall, policy needs to allow some messages through the
firewall but also has to permit certain messages to and from the

firewall, especially those with link-local sources on links to which
the firewall is attached. These messages must be permitted to ensure
that Neighbor Discovery [RFC2462], Multicast Listener Discovery
[RFC2710], [RFC3810] and Stateless Address Configuration [RFC2463]
work as expected.

Recommendations for filtering ICMPv6 messages can be found in
[I-D.davies-v6ops-icmpv6-filtering-bcp].

4.5.1. Problems Resulting from ICMPv6 Transparency

As described in Section 4.5, certain ICMPv6 error packets need to be
passed through a firewall in both directions. This means that some
ICMPv6 error packets can be exchanged between inside and outside
without any filtering.

Using this feature, malicious users can communicate between the
inside and outside of a firewall bypassing the administrator's
inspection (proxy, firewall etc.). For example it might be possible
to carry out a covert conversation through the payload of ICMPv6
error messages or tunnel inappropriate encapsulated IP packets in
ICMPv6 error messages. This problem can be alleviated by filtering
ICMPv6 errors using a stateful packet inspection mechanism to ensure
that the packet carried as a payload is associated with legitimate
traffic to or from the protected network.

4.6. IPsec Transport Mode

IPsec provides security to end-to-end communications at the network
layer (layer 3). The security features available include access
control, connectionless integrity, data origin authentication,
protection against replay attacks, confidentiality, and limited
traffic flow confidentiality (see [RFC2401] section 2.1). IPv6
mandates the implementation of IPsec in all conforming nodes, making
the usage of IPsec to secure end-to-end communication possible in a
way which is generally not available to IPv4.

To secure IPv6 end-to-end communications, IPsec transport mode would
generally be the solution of choice. However, use of these IPsec
security features can result in novel problems for network
administrators and decrease the effectiveness of perimeter firewalls
because of the increased prevalence of encrypted packets on which the
firewalls cannot perform deep packet inspection and filtering.

One example of such problems is the lack of security solutions in the
middlebox, including effective content-filtering, ability to provide
DoS prevention based on the expected TCP protocol behavior, and
intrusion detection. Future solutions to this problem are discussed

in Section 2.3.2. Another example is an IPsec-based DoS (e.g.,
sending malformed ESP/AH packets) which can be especially detrimental
to software-based IPsec implementations.

4.7. Reduced Functionality Devices

With the deployment of IPv6 we can expect the attachment of a very
large number of new IPv6-enabled devices with scarce resources and
low computing capacity. The resource limitations are generally
because of a market requirement for cost reduction. Although the
IPv6 Node Requirements [I-D.ietf-ipv6-node-requirements] specifies
some mandatory security capabilities for every conformant node, these
do not include functions required for a node to be able to protect
itself. Accordingly, some such devices may not be able even to
perform the minimum set of functions required to protect themselves
(e.g. 'personal' firewall, automatic firmware update, enough CPU
power to endure DoS attacks). This means a different security scheme
may be necessary for such reduced functionality devices.

4.8. Operational Factors when Enabling IPv6 in the Network

There are a number of reasons which make it essential to take
particular care when enabling IPv6 in the network equipment:

Initially, IPv6-enabled router software may be less mature than
current IPv4-only implementations and there is less experience with
configuring IPv6 routing, which can result in disruptions to the IPv6
routing environment and (IPv6) network outages.

IPv6 processing may not happen at (near) line speed (or at a
comparable performance level to IPv4 in the same equipment). A high
level of IPv6 traffic (even legitimate, e.g. Network News Transport
Protocol, NNTP) could easily overload IPv6 processing especially when
it is software-based without the hardware support typical in high-end
routers. This may potentially have deleterious knock-on effects on
IPv4 processing, affecting availability of both services.
Accordingly, if people don't feel confident enough in the IPv6
capabilities of their equipment, they will be reluctant to enable it
in their "production" networks.

Sometimes essential features may be missing from early releases of
vendors' software; an example is provision of software enabling IPv6
telnet/SSH access (e.g., to the configuration application of a
router), but without the ability to turn it off or limit access to
it!

Sometimes the default IPv6 configuration is insecure. For example,
in one vendor's implementation, if you have restricted IPv4 telnet to

   only a few hosts in the configuration, you need to be aware that IPv6
   telnet will be automatically enabled, that the configuration commands
   used previously do not block IPv6 telnet, IPv6 telnet is open to the
   world by default, and that you have to use a separate command to also
   lock down the IPv6 telnet access.

   Many operator networks have to run interior routing protocols for
   both IPv4 and IPv6. It is possible to run the both in one routing
   protocol, or have two separate routing protocols; either approach has
   its tradeoffs [RFC4029]. If multiple routing protocols are used, one
   should note that this causes double the amount of processing when
   links flap or recalculation is otherwise needed -- which might more
   easily overload the router's CPU, causing slightly slower convergence
   time.

4.9. Ingress Filtering Issues Due to Privacy Addresses

   [RFC3041][I-D.ietf-ipv6-privacy-addrs-v2] describes a method for
   creating temporary addresses on IPv6 nodes to address privacy issues
   created by the use of a constant identifier. In a network, which
   implements such a mechanism, with a large number of nodes, new
   temporary addresses may be created at a fairly high rate. This might
   make it hard for ingress filtering mechanisms to distinguish between
   legitimately changing temporary addresses and spoofed source
   addresses, which are "in-prefix" (They use a topologically correct
   prefix and non-existent interface ID). This can be addressed by
   using finer grained access control mechanisms on the network egress
   point.

4.10. Security Issues Due to ND Proxies

   In order to span a single subnet over multiple physical links, a new
   capability is being introduced in IPv6 to proxy Neighbor Discovery
   messages. This node will be called an NDProxy (see [I-D.ietf-ipv6-
   ndproxy]. NDProxies are susceptible to the same security issues as
   the ones faced by hosts using unsecured Neighbor Discovery or ARP.
   These proxies may process unsecured messages, and update the neighbor
   cache as a result of such processing, thus allowing a malicious node
   to divert or hijack traffic. This may undermine the advantages of
   using SEND [RFC3971].

   To resolve the security issues introduced by NDProxies, SEND needs to
   be extended to be NDProxy aware.

5. IANA Considerations

   This memo does not contain any actions for IANA.

[6](#). **Security Considerations**

   This memo attempts to give an overview of security considerations of
   the different aspects of IPv6, particularly as they relate to the
   transition to a network in which IPv4- and IPv6-based communications
   need to coexist.

[7](#). **Acknowledgements**

   Alain Durand, Alain Baudot, Luc Beloeil, Tim Chown, Andras Kis-Szabo,
   Alvaro Vives, Janos Mohacsi and Mark Smith provided feedback to
   improve this memo. Satoshi Kondo, Shinsuke Suzuki and Alvaro Vives
   provided additional inputs in cooperation with the Deployment Working
   Group of the Japanese IPv6 Promotion Council and the Euro6IX IST co-
   funded project, together with inputs from Jordi Palet, Brian
   Carpenter, and Peter Bieringer. Michael Wittsend and Michael Cole
   discussed issues relating to probing/mapping and privacy.

[8](#). **References**

[8.1](#). **Normative References**

   [I-D.huitema-v6ops-teredo]
              Huitema, C., "Teredo: Tunneling IPv6 over UDP through
              NATs", [draft-huitema-v6ops-teredo-05](#) (work in progress),
              April 2005.

   [I-D.ietf-ipv6-privacy-addrs-v2]
              Narten, T., "Privacy Extensions for Stateless Address
              Autoconfiguration in IPv6",
              [draft-ietf-ipv6-privacy-addrs-v2-04](#) (work in progress),
              May 2005.

   [I-D.ietf-v6ops-natpt-to-exprmntl]
              Aoun, C. and E. Davies, "Reasons to Move NAT-PT to
              Experimental", [draft-ietf-v6ops-natpt-to-exprmntl-01](#) (work
              in progress), July 2005.

   [I-D.ietf-vrrp-ipv6-spec]
              Hinden, R., "Virtual Router Redundancy Protocol for IPv6",
              [draft-ietf-vrrp-ipv6-spec-07](#) (work in progress),
              October 2004.

   [RFC2375] Hinden, R. and S. Deering, "IPv6 Multicast Address
              Assignments", [RFC 2375](#), July 1998.

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

[RFC2461]   Narten, T., Nordmark, E., and W. Simpson, "Neighbor
            Discovery for IP Version 6 (IPv6)", RFC 2461,
            December 1998.

[RFC2462]   Thomson, S. and T. Narten, "IPv6 Stateless Address
            Autoconfiguration", RFC 2462, December 1998.

[RFC2463]   Conta, A. and S. Deering, "Internet Control Message
            Protocol (ICMPv6) for the Internet Protocol Version 6
            (IPv6) Specification", RFC 2463, December 1998.

[RFC2710]   Deering, S., Fenner, W., and B. Haberman, "Multicast
            Listener Discovery (MLD) for IPv6", RFC 2710,
            October 1999.

[RFC3041]   Narten, T. and R. Draves, "Privacy Extensions for
            Stateless Address Autoconfiguration in IPv6", RFC 3041,
            January 2001.

[RFC3056]   Carpenter, B. and K. Moore, "Connection of IPv6 Domains
            via IPv4 Clouds", RFC 3056, February 2001.

[RFC3513]   Hinden, R. and S. Deering, "Internet Protocol Version 6
            (IPv6) Addressing Architecture", RFC 3513, April 2003.

[RFC3775]   Johnson, D., Perkins, C., and J. Arkko, "Mobility Support
            in IPv6", RFC 3775, June 2004.

[RFC3810]   Vida, R. and L. Costa, "Multicast Listener Discovery
            Version 2 (MLDv2) for IPv6", RFC 3810, June 2004.

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

8.2. Informative References

[FNAT]   Bellovin, S., "Technique for Counting NATted Hosts", Proc.
            Second Internet Measurement Workshop , November 2002,
            <http://www.research.att.com/~smb/papers/fnat.pdf>.

[I-D.chown-v6ops-port-scanning-implications]
            Chown, T., "IPv6 Implications for TCP/UDP Port Scanning",
            draft-chown-v6ops-port-scanning-implications-01 (work in
            progress), July 2004.

   [I-D.cmetz-v6ops-v4mapped-api-harmful]
            Metz, C. and J. Hagino, "IPv4-Mapped Address API
            Considered Harmful",
            draft-cmetz-v6ops-v4mapped-api-harmful-01 (work in
            progress), October 2003.

   [I-D.davies-v6ops-icmpv6-filtering-bcp]
            Davies, E. and J. Mohacsi, "Best Current Practice for
            Filtering ICMPv6 Messages in Firewalls",
            draft-davies-v6ops-icmpv6-filtering-bcp-00 (work in
            progress), July 2005.

   [I-D.dupont-ipv6-rfc3041harmful]
            Dupont, F. and P. Savola, "RFC 3041 Considered Harmful",
            draft-dupont-ipv6-rfc3041harmful-05 (work in progress),
            June 2004.

   [I-D.ietf-dnsop-ipv6-dns-issues]
            Durand, A., "Operational Considerations and Issues with
            IPv6 DNS", draft-ietf-dnsop-ipv6-dns-issues-11 (work in
            progress), July 2005.

   [I-D.ietf-ipv6-host-load-sharing]
            Hinden, R. and D. Thaler, "IPv6 Host to Router Load
            Sharing", draft-ietf-ipv6-host-load-sharing-04 (work in
            progress), June 2005.

   [I-D.ietf-ipv6-ndproxy]
            Thaler, D., "Neighbor Discovery Proxies (ND Proxy)",
            draft-ietf-ipv6-ndproxy-03 (work in progress), July 2005.

   [I-D.ietf-ipv6-node-requirements]
            Loughney, J., "IPv6 Node Requirements",
            draft-ietf-ipv6-node-requirements-11 (work in progress),
            August 2004.

   [I-D.ietf-ipv6-router-selection]
            Draves, R. and D. Thaler, "Default Router Preferences and
            More-Specific Routes", draft-ietf-ipv6-router-selection-07
            (work in progress), January 2005.

   [I-D.ietf-mip6-ro-sec]
            Nikander, P., "Mobile IP version 6 Route Optimization
            Security Design Background", draft-ietf-mip6-ro-sec-03
            (work in progress), May 2005.

   [I-D.ietf-v6ops-nap]
            Velde, G., "IPv6 Network Architecture Protection",

draft-ietf-v6ops-nap-01 (work in progress), June 2005.

[I-D.ietf-v6ops-v6onbydefault]
          Roy, S., Durand, A., and J. Paugh, "Issues with Dual Stack
          IPv6 on by Default", draft-ietf-v6ops-v6onbydefault-03
          (work in progress), July 2004.

[I-D.ietf-zeroconf-ipv4-linklocal]
          Aboba, B., "Dynamic Configuration of Link-Local IPv4
          Addresses", draft-ietf-zeroconf-ipv4-linklocal-17 (work in
          progress), July 2004.

[I-D.itojun-v6ops-v4mapped-harmful]
          Metz, C. and J. Hagino, "IPv4-Mapped Addresses on the Wire
          Considered Harmful",
          draft-itojun-v6ops-v4mapped-harmful-02 (work in progress),
          October 2003.

[I-D.krishnan-ipv6-hopbyhop]
          Krishnan, S., "Arrangement of Hop-by-Hop options",
          draft-krishnan-ipv6-hopbyhop-00 (work in progress),
          June 2004.

[I-D.savola-ipv6-rh-ha-security]
          Savola, P., "Security of IPv6 Routing Header and Home
          Address Options", draft-savola-ipv6-rh-ha-security-02
          (work in progress), March 2002.

[I-D.savola-ipv6-rh-hosts]
          Savola, P., "Note about Routing Header Processing on IPv6
          Hosts", draft-savola-ipv6-rh-hosts-00 (work in progress),
          February 2002.

[I-D.savola-v6ops-firewalling]
          Savola, P., "Firewalling Considerations for IPv6",
          draft-savola-v6ops-firewalling-02 (work in progress),
          October 2003.

[I-D.savola-v6ops-transarch]
          Savola, P., "A View on IPv6 Transition Architecture",
          draft-savola-v6ops-transarch-03 (work in progress),
          January 2004.

[I-D.schild-v6ops-guide-v4mapping]
          Schild, C., "Guide to Mapping IPv4 to IPv6 Subnets",
          draft-schild-v6ops-guide-v4mapping-00 (work in progress),
          January 2004.

   [JpIPv6DC]
              Deployment WG, "IPv6 Deployment Guideline (2005 Edition)",
              IPv6 Promotion Council (Japan) Deployment Working Group,
              2005, <http://www.v6pc.jp/>.

   [RFC1858] Ziemba, G., Reed, D., and P. Traina, "Security
              Considerations for IP Fragment Filtering", RFC 1858,
              October 1995.

   [RFC2401] Kent, S. and R. Atkinson, "Security Architecture for the
              Internet Protocol", RFC 2401, November 1998.

   [RFC2765] Nordmark, E., "Stateless IP/ICMP Translation Algorithm
              (SIIT)", RFC 2765, February 2000.

   [RFC2766] Tsirtsis, G. and P. Srisuresh, "Network Address
              Translation - Protocol Translation (NAT-PT)", RFC 2766,
              February 2000.

   [RFC3128] Miller, I., "Protection Against a Variant of the Tiny
              Fragment Attack (RFC 1858)", RFC 3128, June 2001.

   [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.

   [RFC3756] Nikander, P., Kempf, J., and E. Nordmark, "IPv6 Neighbor
              Discovery (ND) Trust Models and Threats", RFC 3756,
              May 2004.

   [RFC3971] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure
              Neighbor Discovery (SEND)", RFC 3971, March 2005.

   [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.

   [RFC4074] Morishita, Y. and T. Jinmei, "Common Misbehavior Against
              DNS Queries for IPv6 Addresses", RFC 4074, May 2005.

   [SIXNET]   6Net, "Large Scale International IPv6 Pilot Network", EU
              Information Society Technologies Project , 2005,
              <http://www.6net.org/>.

Appendix A. **IPv6 Probing/Mapping Considerations**

   One school of thought wants the IPv6 numbering topology (either at
   network or node level) [I-D.schild-v6ops-guide-v4mapping] to match
   IPv4 as exactly as possible, whereas others see IPv6 as giving more
   flexibility to the address plans, not wanting to constrain the design
   of IPv6 addressing. Mirroring the address plans may also be seen as
   a security threat because an IPv6 deployment may have different
   security properties from IPv4.

   Given the relatively immature state of IPv6 network security, if an
   attacker knows the IPv4 address of the node and believes it to be
   dual-stacked with IPv4 and IPv6, he might want to try to probe the
   corresponding IPv6 address, based on the assumption that the security
   defenses might be lower. This might be the case particularly for
   nodes which are behind a NAT in IPv4, but globally addressable in
   IPv6. Naturally, this is not a concern if similar and adequate
   security policies are in place.

   On the other hand, brute-force scanning or probing of addresses is
   computationally infeasible due to the large search space of interface
   identifiers on most IPv6 subnets (somewhat less than 64 bits wide,
   depending on how identifiers are chosen), always provided that
   identifiers are chosen at random out of the available space, as
   discussed in [I-D.chown-v6ops-port-scanning-implications].

   For example, automatic tunneling mechanisms typically use
   deterministic methods for generating IPv6 addresses, so probing/
   port-scanning an IPv6 node is simplified. The IPv4 address is
   embedded at least in 6to4, Teredo and ISATAP addresses.
   Additionally, it is possible (in the case of 6to4 in particular) to
   learn the address behind the prefix; for example, Microsoft 6to4
   implementation uses the address 2002:V4ADDR::V4ADDR while older Linux
   and FreeBSD implementations default to 2002:V4ADDR::1. This could
   also be used as one way to identify an implementation and hence
   target any specific weaknesses.

   One proposal has been to randomize the addresses or subnet identifier
   in the address of the 6to4 router. This does not really help, as the
   6to4 router (whether a host or a router) will return an ICMPv6 Hop
   Limit Exceeded message, revealing the IP address. Hosts behind the
   6to4 router can use methods such as RFC 3041 addresses to conceal
   themselves, provided that they are not meant to be reachable by
   sessions started from elsewhere: they would still require a globally
   accessible static address if they wish to receive communications
   initiated elsewhere.

   To conclude, it seems that when an automatic tunneling mechanism is

being used, given an IPv4 address, the corresponding IPv6 address
could possibly be guessed with relative ease. This has significant
implications if the IPv6 security policy is less adequate than that
for IPv4.

Appendix B. IPv6 Privacy Considerations

The generation of IPv6 addresses of IPv6 addresses from MAC addresses
potentially allows the behavior of users to be tracked in a way which
may infringe their privacy. [RFC3041] specifies mechanisms which can
be used to reduce the risk of infringement. It has also been claimed
that IPv6 harms the privacy of the user, either by exposing the MAC
address, or by exposing the number of nodes connected to a site.

Additional discussion of privacy issues can be found in the IPv6
Network Architecture Protection document [I-D.ietf-v6ops-nap].

B.1. Exposing MAC Addresses

Using stateless address autoconfiguration results in the MAC address
being incorporated in an EUI64 that exposes the model of network
card. The concern has been that a user might not want to expose the
details of the system to outsiders, e.g., fearing a resulting
burglary if a thief identifies expensive equipment from the vendor
identifier embedded in MAC addresses.

In most cases, this seems completely unfounded. First, such an
address must be learned somehow -- this is a non-trivial process; the
addresses are visible e.g., in web site access logs, but the chances
that a random web site owner is collecting this kind of information
(or whether it would be of any use) are quite slim. Being able to
eavesdrop the traffic to learn such addresses (e.g., by the
compromise of DSL or Cable modem physical media) seems also quite
far-fetched. Further, using RFC 3041 addresses for such purposes is
straightforward if worried about the risk. Second, the burglar would
have to be able to map the IP address to the physical location;
typically this would only be possible with information from the
private customer database of the ISP and, for large sites, the
administrative records of the site.

B.2. Exposing Multiple Devices

Another concern that has been aired involves the user wanting to
conceal the presence of a large number of computers or other devices
connected to a network; NAT can "hide" all this equipment behind a
single address, but is not perfect either [FNAT].

One practical reason why some administrators may find this desirable
is being able to thwart certain ISPs' business models. These models
require payment based on the number of connected computers, rather
than the connectivity as a whole.

Similar feasibility issues as described above apply. To a degree,
the number of machines present could be obscured by the sufficiently
frequent re-use of RFC 3041 addresses -- that is, if during a short
period, dozens of generated addresses seem to be in use, it's
difficult to estimate whether they are generated by just one host or
multiple hosts.

B.3. Exposing the Site by a Stable Prefix

When an ISP provides IPv6 connectivity to its customers, it delegates
a fixed global routing prefix (usually a /48) to them.

Due to this fixed allocation, it is easier to correlate the global
routing prefix to a network site. In case of consumer users, this
correlation leads to a privacy issue, since a site is often
equivalent to an individual or a family in such a case. That is,
some users might be concerned about being able to be tracked based on
their /48 allocation if it is static [I-D.dupont-ipv6-
rfc3041harmful]. On the other hand many users may find having a
static allocation desirable as it allows them to offer services
hosted in their network more easily.

This problem remains unsolved even when a user changes his/her
interface ID or subnet ID, because malicious users can still discover
this binding. This problem can be solved by untraceable IPv6
addresses as described in [I-D.ietf-v6ops-nap].

Authors' Addresses

    Elwyn B. Davies
    Consultant
    Soham, Cambs
    UK

    Phone: +44 7889 488 335
    Email: elwynd@dial.pipex.com

    Suresh Krishnan
    Ericsson
    8400 Decarie Blvd.
    Town of Mount Royal, QC H4P 2N2
    Canada

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

    Pekka Savola
    CSC/Funet

    Email: psavola@funet.fi