Internet Engineering Task Force (IETF)
Request for Comments: 8504
BCP: 220
Obsoletes: 6434
Category: Best Current Practice
ISSN: 2070-1721

T. Chown
Jisc
J. Loughney
Intel
T. Winters
UNH-IOL

January 2019

IPv6 Node Requirements

Abstract

This document defines requirements for IPv6 nodes. It is expected that IPv6 will be deployed in a wide range of devices and situations. Specifying the requirements for IPv6 nodes allows IPv6 to function well and interoperate in a large number of situations and deployments.

This document obsoletes RFC 6434, and in turn RFC 4294.

Status of This Memo

This memo documents an Internet Best Current Practice.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on BCPs is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8504.

Table of Contents

1. Introduction

This document defines common functionality required by both IPv6 hosts and routers. Many IPv6 nodes will implement optional or additional features, but this document collects and summarizes requirements from other published Standards Track documents in one place.

This document tries to avoid discussion of protocol details and references RFCs for this purpose. This document is intended to be an applicability statement and to provide guidance as to which IPv6 specifications should be implemented in the general case and which specifications may be of interest to specific deployment scenarios. This document does not update any individual protocol document RFCs.

Although this document points to different specifications, it should be noted that in many cases, the granularity of a particular requirement will be smaller than a single specification, as many specifications define multiple, independent pieces, some of which may not be mandatory. In addition, most specifications define both client and server behavior in the same specification, while many implementations will be focused on only one of those roles.

This document defines a minimal level of requirement needed for a device to provide useful Internet service and considers a broad range of device types and deployment scenarios. Because of the wide range of deployment scenarios, the minimal requirements specified in this document may not be sufficient for all deployment scenarios. It is perfectly reasonable (and indeed expected) for other profiles to define additional or stricter requirements appropriate for specific usage and deployment environments. As an example, this document does not mandate that all clients support DHCP, but some deployment scenarios may deem it appropriate to make such a requirement. As another example, NIST has defined profiles for specialized requirements for IPv6 in target environments (see [USGv6]).

As it is not always possible for an implementer to know the exact usage of IPv6 in a node, an overriding requirement for IPv6 nodes is that they should adhere to Jon Postel's Robustness Principle: "Be conservative in what you do, be liberal in what you accept from others" [RFC793].

1.1. Scope of This Document

IPv6 covers many specifications. It is intended that IPv6 will be deployed in many different situations and environments. Therefore, it is important to develop requirements for IPv6 nodes to ensure interoperability.

1.2. Description of IPv6 Nodes

From "Internet Protocol, Version 6 (IPv6) Specification" [RFC8200], we have the following definitions:

IPv6 node - a device that implements IPv6.
IPv6 router - a node that forwards IPv6 packets not explicitly addressed to itself.
IPv6 host - any IPv6 node that is not a router.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Abbreviations Used in This Document

AH	Authentication Header
DAD	Duplicate Address Detection
ESP	Encapsulating Security Payload
ICMP	Internet Control Message Protocol
IKE	Internet Key Exchange
MIB	Management Information Base
MLD	Multicast Listener Discovery
MTU	Maximum Transmission Unit
NA	Neighbor Advertisement
NBMA	Non-Broadcast Multi-Access
ND	Neighbor Discovery
NS	Neighbor Solicitation
NUD	Neighbor Unreachability Detection
PPP	Point-to-Point Protocol

4. Sub-IP Layer

An IPv6 node MUST include support for one or more IPv6 link-layer specifications. Which link-layer specifications an implementation should include will depend upon what link layers are supported by the hardware available on the system. It is possible for a conformant IPv6 node to support IPv6 on some of its interfaces and not on others.

As IPv6 is run over new Layer 2 technologies, it is expected that new specifications will be issued. We list here some of the Layer 2 technologies for which an IPv6 specification has been developed. It is provided for informational purposes only and may not be complete.

- Transmission of IPv6 Packets over Ethernet Networks [RFC2464]

- Transmission of IPv6 Packets over Frame Relay Networks Specification [RFC2590]

- Transmission of IPv6 Packets over IEEE 1394 Networks [RFC3146]

- Transmission of IPv6, IPv4, and Address Resolution Protocol (ARP) Packets over Fibre Channel [RFC4338]

- Transmission of IPv6 Packets over IEEE 802.15.4 Networks [RFC4944]

- Transmission of IPv6 via the IPv6 Convergence Sublayer over IEEE 802.16 Networks [RFC5121]

- IP version 6 over PPP [RFC5072]

In addition to traditional physical link layers, it is also possible to tunnel IPv6 over other protocols. Examples include:

- Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs) [RFC4380]

- Basic Transition Mechanisms for IPv6 Hosts and Routers (see Section 3 of [RFC4213])

5. IP Layer

5.1. Internet Protocol Version 6 - RFC 8200

The Internet Protocol version 6 is specified in [RFC8200]. This specification MUST be supported.

The node MUST follow the packet transmission rules in RFC 8200.

All conformant IPv6 implementations MUST be capable of sending and receiving IPv6 packets; forwarding functionality MAY be supported. Nodes MUST always be able to send, receive, and process Fragment headers.

IPv6 nodes MUST not create overlapping fragments. Also, when reassembling an IPv6 datagram, if one or more of its constituent fragments is determined to be an overlapping fragment, the entire datagram (and any constituent fragments) MUST be silently discarded. See [RFC5722] for more information.

As recommended in [RFC8021], nodes MUST NOT generate atomic fragments, i.e., where the fragment is a whole datagram. As per [RFC6946], if a receiving node reassembling a datagram encounters an atomic fragment, it should be processed as a fully reassembled packet, and any other fragments that match this packet should be processed independently.

To mitigate a variety of potential attacks, nodes SHOULD avoid using predictable Fragment Identification values in Fragment headers, as discussed in [RFC7739].

All nodes SHOULD support the setting and use of the IPv6 Flow Label field as defined in the IPv6 Flow Label specification [RFC6437]. Forwarding nodes such as routers and load distributors MUST NOT depend only on Flow Label values being uniformly distributed. It is RECOMMENDED that source hosts support the flow label by setting the Flow Label field for all packets of a given flow to the same value chosen from an approximation to a discrete uniform distribution.

5.2. Support for IPv6 Extension Headers

RFC 8200 specifies extension headers and the processing for these headers.

Extension headers (except for the Hop-by-Hop Options header) are not processed, inserted, or deleted by any node along a packet's delivery path, until the packet reaches the node (or each of the set of nodes, in the case of multicast) identified in the Destination Address field of the IPv6 header.

Any unrecognized extension headers or options MUST be processed as described in RFC 8200. Note that where Section 4 of RFC 8200 refers to the action to be taken when a Next Header value in the current header is not recognized by a node, that action applies whether the value is an unrecognized extension header or an unrecognized upper-layer protocol (ULP).

An IPv6 node MUST be able to process these extension headers. An exception is Routing Header type 0 (RH0), which was deprecated by [RFC5095] due to security concerns and which MUST be treated as an unrecognized routing type.

Further, [RFC7045] adds specific requirements for the processing of extension headers, in particular that any forwarding node along an IPv6 packet's path, which forwards the packet for any reason, SHOULD do so regardless of any extension headers that are present.

As per RFC 8200, when a node fragments an IPv6 datagram, it MUST include the entire IPv6 Header Chain in the first fragment. The Per-Fragment headers MUST consist of the IPv6 header plus any extension headers that MUST be processed by nodes en route to the destination, that is, all headers up to and including the Routing header if present, else the Hop-by-Hop Options header if present, else no extension headers. On reassembly, if the first fragment does not include all headers through an upper-layer header, then that fragment SHOULD be discarded and an ICMP Parameter Problem, Code 3, message SHOULD be sent to the source of the fragment, with the Pointer field set to zero. See [RFC7112] for a discussion of why oversized IPv6 Extension Header Chains are avoided.

Defining new IPv6 extension headers is not recommended, unless there are no existing IPv6 extension headers that can be used by specifying a new option for that IPv6 extension header. A proposal to specify a new IPv6 extension header MUST include a detailed technical explanation of why an existing IPv6 extension header can not be used for the desired new function, and in such cases, it needs to follow the format described in Section 8 of RFC 8200. For further background reading on this topic, see [RFC6564].

5.3. Protecting a Node from Excessive Extension Header Options

As per RFC 8200, end hosts are expected to process all extension headers, destination options, and hop-by-hop options in a packet. Given that the only limit on the number and size of extension headers is the MTU, the processing of received packets could be considerable. It is also conceivable that a long chain of extension headers might be used as a form of denial-of-service attack. Accordingly, a host may place limits on the number and sizes of extension headers and options it is willing to process.

A host MAY limit the number of consecutive PAD1 options in destination options or hop-by-hop options to 7. In this case, if there are more than 7 consecutive PAD1 options present, the packet MAY be silently discarded. The rationale is that if padding of 8 or more bytes is required, then the PADN option SHOULD be used.

A host MAY limit the number of bytes in a PADN option to be less than 8. In such a case, if a PADN option is present that has a length greater than 7, the packet SHOULD be silently discarded. The rationale for this guideline is that the purpose of padding is for alignment and 8 bytes is the maximum alignment used in IPv6.

A host MAY disallow unknown options in destination options or hop-by-hop options. This SHOULD be configurable where the default is to accept unknown options and process them per [RFC8200]. If a packet with unknown options is received and the host is configured to disallow them, then the packet SHOULD be silently discarded.

A host MAY impose a limit on the maximum number of non-padding options allowed in the destination options and hop-by-hop extension headers. If this feature is supported, the maximum number SHOULD be configurable, and the default value SHOULD be set to 8. The limits for destination options and hop-by-hop options may be separately configurable. If a packet is received and the number of destination or hop-by-hop options exceeds the limit, then the packet SHOULD be silently discarded.

A host MAY impose a limit on the maximum length of Destination Options or Hop-by-Hop Options extension headers. This value SHOULD be configurable, and the default is to accept options of any length. If a packet is received and the length of the Destination or Hop-by-Hop Options extension header exceeds the length limit, then the packet SHOULD be silently discarded.

5.4. Neighbor Discovery for IPv6 - RFC 4861

Neighbor Discovery is defined in [RFC4861]; the definition was updated by [RFC5942]. Neighbor Discovery MUST be supported with the noted exceptions below. RFC 4861 states:

> Unless specified otherwise (in a document that covers operating IP over a particular link type) this document applies to all link types. However, because ND uses link-layer multicast for some of its services, it is possible that on some link types (e.g., Non-Broadcast Multi-Access (NBMA) links), alternative protocols or mechanisms to implement those services will be specified (in the appropriate document covering the operation of IP over a particular link type). The services described in this document that are not directly dependent on multicast, such as Redirects, Next-hop determination, Neighbor Unreachability Detection, etc., are expected to be provided as specified in this document. The details of how one uses ND on NBMA links are addressed in [RFC2491].

Some detailed analysis of Neighbor Discovery follows:

Router Discovery is how hosts locate routers that reside on an attached link. Hosts MUST support Router Discovery functionality.

Prefix Discovery is how hosts discover the set of address prefixes that define which destinations are on-link for an attached link. Hosts MUST support Prefix Discovery.

Hosts MUST also implement Neighbor Unreachability Detection (NUD) for all paths between hosts and neighboring nodes. NUD is not required for paths between routers. However, all nodes MUST respond to unicast Neighbor Solicitation (NS) messages.

[RFC7048] discusses NUD, in particular cases where it behaves too impatiently. It states that if a node transmits more than a certain number of packets, then it SHOULD use the exponential backoff of the retransmit timer, up to a certain threshold point.

Hosts MUST support the sending of Router Solicitations and the receiving of Router Advertisements (RAs). The ability to understand individual RA options is dependent on supporting the functionality making use of the particular option.

[RFC7559] discusses packet loss resiliency for Router Solicitations and requires that nodes MUST use a specific exponential backoff algorithm for retransmission of Router Solicitations.

All nodes MUST support the sending and receiving of Neighbor Solicitation (NS) and Neighbor Advertisement (NA) messages. NS and NA messages are required for Duplicate Address Detection (DAD).

Hosts SHOULD support the processing of Redirect functionality. Routers MUST support the sending of Redirects, though not necessarily for every individual packet (e.g., due to rate limiting). Redirects are only useful on networks supporting hosts. In core networks dominated by routers, Redirects are typically disabled. The sending of Redirects SHOULD be disabled by default on routers intended to be deployed on core networks. They MAY be enabled by default on routers intended to support hosts on edge networks.

As specified in [RFC6980], nodes MUST NOT employ IPv6 fragmentation for sending any of the following Neighbor Discovery and SEcure Neighbor Discovery messages: Neighbor Solicitation, Neighbor Advertisement, Router Solicitation, Router Advertisement, Redirect, or Certification Path Solicitation. Nodes MUST silently ignore any of these messages on receipt if fragmented. See RFC 6980 for details and motivation.

"IPv6 Host-to-Router Load Sharing" [RFC4311] includes additional recommendations on how to select from a set of available routers. [RFC4311] SHOULD be supported.

5.5. SEcure Neighbor Discovery (SEND) - RFC 3971

SEND [RFC3971] and Cryptographically Generated Addresses (CGAs) [RFC3972] provide a way to secure the message exchanges of Neighbor Discovery. SEND has the potential to address certain classes of spoofing attacks, but it does not provide specific protection for threats from off-link attackers.

There have been relatively few implementations of SEND in common operating systems and platforms since its publication in 2005; thus, deployment experience remains very limited to date.

At this time, support for SEND is considered optional. Due to the complexity in deploying SEND and its heavyweight provisioning, its deployment is only likely to be considered where nodes are operating in a particularly strict security environment.

5.6. IPv6 Router Advertisement Flags Option - RFC 5175

Router Advertisements include an 8-bit field of single-bit Router Advertisement flags. The Router Advertisement Flags Option extends the number of available flag bits by 48 bits. At the time of this writing, 6 of the original 8 single-bit flags have been assigned, while 2 remain available for future assignment. No flags have been defined that make use of the new option; thus, strictly speaking, there is no requirement to implement the option today. However, implementations that are able to pass unrecognized options to a higher-level entity that may be able to understand them (e.g., a user-level process using a "raw socket" facility) MAY take steps to handle the option in anticipation of a future usage.

5.7. Path MTU Discovery and Packet Size

5.7.1. Path MTU Discovery - RFC 8201

"Path MTU Discovery for IP version 6" [RFC8201] SHOULD be supported. From [RFC8200]:

> It is strongly recommended that IPv6 nodes implement Path MTU Discovery [RFC8201], in order to discover and take advantage of path MTUs greater than 1280 octets. However, a minimal IPv6 implementation (e.g., in a boot ROM) may simply restrict itself to sending packets no larger than 1280 octets, and omit implementation of Path MTU Discovery.

The rules in [RFC8200] and [RFC5722] MUST be followed for packet fragmentation and reassembly.

As described in RFC 8201, nodes implementing Path MTU Discovery and sending packets larger than the IPv6 minimum link MTU are susceptible to problematic connectivity if ICMPv6 messages are blocked or not transmitted. For example, this will result in connections that complete the TCP three-way handshake correctly but then hang when data is transferred. This state is referred to as a black-hole connection [RFC2923]. Path MTU Discovery relies on ICMPv6 Packet Too Big (PTB) to determine the MTU of the path (and thus these MUST NOT be filtered, as per the recommendation in [RFC4890]).

An alternative to Path MTU Discovery defined in RFC 8201 can be found in [RFC4821], which defines a method for Packetization Layer Path MTU Discovery (PLPMTUD) designed for use over paths where delivery of ICMPv6 messages to a host is not assured.

5.7.2. Minimum MTU Considerations

While an IPv6 link MTU can be set to 1280 bytes, it is recommended that for IPv6 UDP in particular, which includes DNS operation, the sender use a large MTU if they can, in order to avoid gratuitous fragmentation-caused packet drops.

5.8. ICMP for the Internet Protocol Version 6 (IPv6) - RFC 4443

ICMPv6 [RFC4443] MUST be supported. "Extended ICMP to Support Multi-Part Messages" [RFC4884] MAY be supported.

5.9. Default Router Preferences and More-Specific Routes - RFC 4191

"Default Router Preferences and More-Specific Routes" [RFC4191] provides support for nodes attached to multiple (different) networks, each providing routers that advertise themselves as default routers via Router Advertisements. In some scenarios, one router may provide connectivity to destinations that the other router does not, and choosing the "wrong" default router can result in reachability failures. In order to resolve this scenario, IPv6 nodes MUST implement [RFC4191] and SHOULD implement the Type C host role defined in RFC 4191.

5.10. First-Hop Router Selection - RFC 8028

In multihomed scenarios, where a host has more than one prefix, each allocated by an upstream network that is assumed to implement BCP 38 ingress filtering, the host may have multiple routers to choose from.

Hosts that may be deployed in such multihomed environments SHOULD follow the guidance given in [RFC8028].

5.11. Multicast Listener Discovery (MLD) for IPv6 - RFC 3810

Nodes that need to join multicast groups MUST support MLDv2 [RFC3810]. MLD is needed by any node that is expected to receive and process multicast traffic; in particular, MLDv2 is required for support for source-specific multicast (SSM) as per [RFC4607].

Previous versions of this specification only required MLDv1 [RFC2710] to be implemented on all nodes. Since participation of any MLDv1-only nodes on a link require that all other nodes on the link then operate in version 1 compatibility mode, the requirement to support MLDv2 on all nodes was upgraded to a MUST. Further, SSM is now the preferred multicast distribution method, rather than Any-Source Multicast (ASM).

Note that Neighbor Discovery (as used on most link types -- see Section 5.4) depends on multicast and requires that nodes join Solicited Node multicast addresses.

5.12. Explicit Congestion Notification (ECN) - RFC 3168

An ECN-aware router sets a mark in the IP header in order to signal impending congestion, rather than dropping a packet. The receiver of the packet echoes the congestion indication to the sender, which can then reduce its transmission rate as if it detected a dropped packet.

Nodes SHOULD support ECN [RFC3168] by implementing an interface for the upper layer to access and by setting the ECN bits in the IP header. The benefits of using ECN are documented in [RFC8087].

6. Addressing and Address Configuration

6.1. IP Version 6 Addressing Architecture - RFC 4291

The IPv6 Addressing Architecture [RFC4291] MUST be supported.

The current IPv6 Address Architecture is based on a 64-bit boundary for subnet prefixes. The reasoning behind this decision is documented in [RFC7421].

Implementations MUST also support the multicast flag updates documented in [RFC7371].

6.2. Host Address Availability Recommendations

Hosts may be configured with addresses through a variety of methods, including Stateless Address Autoconfiguration (SLAAC), DHCPv6, or manual configuration.

[RFC7934] recommends that networks provide general-purpose end hosts with multiple global IPv6 addresses when they attach, and it describes the benefits of and the options for doing so. Routers SHOULD support [RFC7934] for assigning multiple addresses to a host. A host SHOULD support assigning multiple addresses as described in [RFC7934].

Nodes SHOULD support the capability to be assigned a prefix per host as documented in [RFC8273]. Such an approach can offer improved host isolation and enhanced subscriber management on shared network segments.

6.3. IPv6 Stateless Address Autoconfiguration - RFC 4862

Hosts MUST support IPv6 Stateless Address Autoconfiguration. It is RECOMMENDED, as described in [RFC8064], that unless there is a specific requirement for Media Access Control (MAC) addresses to be embedded in an Interface Identifier (IID), nodes follow the procedure in [RFC7217] to generate SLAAC-based addresses, rather than use [RFC4862]. Addresses generated using the method described in [RFC7217] will be the same whenever a given device (re)appears on the same subnet (with a specific IPv6 prefix), but the IID will vary on each subnet visited.

Nodes that are routers MUST be able to generate link-local addresses as described in [RFC4862].

From RFC 4862:

> The autoconfiguration process specified in this document applies only to hosts and not routers. Since host autoconfiguration uses information advertised by routers, routers will need to be configured by some other means. However, it is expected that routers will generate link-local addresses using the mechanism described in this document. In addition, routers are expected to successfully pass the Duplicate Address Detection procedure described in this document on all addresses prior to assigning them to an interface.

All nodes MUST implement Duplicate Address Detection. Quoting from Section 5.4 of RFC 4862:

> Duplicate Address Detection MUST be performed on all unicast addresses prior to assigning them to an interface, regardless of whether they are obtained through stateless autoconfiguration, DHCPv6, or manual configuration, with the following exceptions [noted therein].

"Optimistic Duplicate Address Detection (DAD) for IPv6" [RFC4429] specifies a mechanism to reduce delays associated with generating addresses via Stateless Address Autoconfiguration [RFC4862]. RFC 4429 was developed in conjunction with Mobile IPv6 in order to reduce the time needed to acquire and configure addresses as devices quickly move from one network to another, and it is desirable to minimize transition delays. For general purpose devices, RFC 4429 remains optional at this time.

[RFC7527] discusses enhanced DAD and describes an algorithm to automate the detection of looped-back IPv6 ND messages used by DAD. Nodes SHOULD implement this behavior where such detection is beneficial.

6.4. Privacy Extensions for Address Configuration in IPv6 - RFC 4941

A node using Stateless Address Autoconfiguration [RFC4862] to form a globally unique IPv6 address that uses its MAC address to generate the IID will see that the IID remains the same on any visited network, even though the network prefix part changes. Thus, it is possible for a third-party device to track the activities of the node they communicate with, as that node moves around the network. Privacy Extensions for Stateless Address Autoconfiguration [RFC4941] address this concern by allowing nodes to configure an additional temporary address where the IID is effectively randomly generated. Privacy addresses are then used as source addresses for new communications initiated by the node.

General issues regarding privacy issues for IPv6 addressing are discussed in [RFC7721].

RFC 4941 SHOULD be supported. In some scenarios, such as dedicated servers in a data center, it provides limited or no benefit, or it may complicate network management. Thus, devices implementing this specification MUST provide a way for the end user to explicitly enable or disable the use of such temporary addresses.

Note that RFC 4941 can be used independently of traditional SLAAC or independently of SLAAC that is based on RFC 7217.

Implementers of RFC 4941 should be aware that certain addresses are reserved and should not be chosen for use as temporary addresses. Consult "Reserved IPv6 Interface Identifiers" [RFC5453] for more details.

6.5. Stateful Address Autoconfiguration (DHCPv6) - RFC 3315

DHCPv6 [RFC3315] can be used to obtain and configure addresses. In general, a network may provide for the configuration of addresses through SLAAC, DHCPv6, or both. There will be a wide range of IPv6 deployment models and differences in address assignment requirements, some of which may require DHCPv6 for stateful address assignment. Consequently, all hosts SHOULD implement address configuration via DHCPv6.

In the absence of observed Router Advertisement messages, IPv6 nodes MAY initiate DHCP to obtain IPv6 addresses and other configuration information, as described in Section 5.5.2 of [RFC4862].

Where devices are likely to be carried by users and attached to multiple visited networks, DHCPv6 client anonymity profiles SHOULD be supported as described in [RFC7844] to minimize the disclosure of identifying information. Section 5 of RFC 7844 describes operational considerations on the use of such anonymity profiles.

6.6. Default Address Selection for IPv6 - RFC 6724

IPv6 nodes will invariably have multiple addresses configured simultaneously and thus will need to choose which addresses to use for which communications. The rules specified in the Default Address Selection for IPv6 document [RFC6724] MUST be implemented. [RFC8028] updates Rule 5.5 from [RFC6724]; implementations SHOULD implement this rule.

7. DNS

DNS is described in [RFC1034], [RFC1035], [RFC3363], and [RFC3596]. Not all nodes will need to resolve names; those that will never need to resolve DNS names do not need to implement resolver functionality. However, the ability to resolve names is a basic infrastructure capability on which applications rely, and most nodes will need to provide support. All nodes SHOULD implement stub-resolver [RFC1034] functionality, as in Section 5.3.1 of [RFC1034], with support for:

- AAAA type Resource Records [RFC3596];

- reverse addressing in ip6.arpa using PTR records [RFC3596]; and

- Extension Mechanisms for DNS (EDNS(0)) [RFC6891] to allow for DNS packet sizes larger than 512 octets.

Those nodes are RECOMMENDED to support DNS security extensions [RFC4033] [RFC4034] [RFC4035].

A6 Resource Records [RFC2874] are classified as Historic per [RFC6563]. These were defined with Experimental status in [RFC3363].

8. Configuring Non-address Information

8.1. DHCP for Other Configuration Information

DHCP [RFC3315] specifies a mechanism for IPv6 nodes to obtain address configuration information (see Section 6.5) and to obtain additional (non-address) configuration. If a host implementation supports applications or other protocols that require configuration that is only available via DHCP, hosts SHOULD implement DHCP. For specialized devices on which no such configuration need is present, DHCP may not be necessary.

An IPv6 node can use the subset of DHCP (described in [RFC3736]) to obtain other configuration information.

If an IPv6 node implements DHCP, it MUST implement the DNS options [RFC3646] as most deployments will expect that these options are available.

8.2. Router Advertisements and Default Gateway

There is no defined DHCPv6 Gateway option.

Nodes using the Dynamic Host Configuration Protocol for IPv6 (DHCPv6) are thus expected to determine their default router information and on-link prefix information from received Router Advertisements.

8.3. IPv6 Router Advertisement Options for DNS Configuration - RFC 8106

Router Advertisement options have historically been limited to those that are critical to basic IPv6 functionality. Originally, DNS configuration was not included as an RA option, and DHCP was the recommended way to obtain DNS configuration information. Over time, the thinking surrounding such an option has evolved. It is now generally recognized that few nodes can function adequately without having access to a working DNS resolver; thus, a Standards Track document has been published to provide this capability [RFC8106].

Implementations MUST include support for the DNS RA option [RFC8106].

8.4. DHCP Options versus Router Advertisement Options for Host Configuration

In IPv6, there are two main protocol mechanisms for propagating configuration information to hosts: RAs and DHCP. RA options have been restricted to those deemed essential for basic network functioning and for which all nodes are configured with exactly the same information. Examples include the Prefix Information Options, the MTU option, etc. On the other hand, DHCP has generally been preferred for configuration of more general parameters and for parameters that may be client specific. Generally speaking, however, there has been a desire to define only one mechanism for configuring a given option, rather than defining multiple (different) ways of configuring the same information.

One issue with having multiple ways to configure the same information is that interoperability suffers if a host chooses one mechanism but the network operator chooses a different mechanism. For "closed" environments, where the network operator has significant influence over what devices connect to the network and thus what configuration mechanisms they support, the operator may be able to ensure that a particular mechanism is supported by all connected hosts. In more open environments, however, where arbitrary devices may connect (e.g., a Wi-Fi hotspot), problems can arise. To maximize interoperability in such environments, hosts would need to implement multiple configuration mechanisms to ensure interoperability.

9. Service Discovery Protocols

Multicast DNS (mDNS) and DNS-based Service Discovery (DNS-SD) are described in [RFC6762] and [RFC6763], respectively. These protocols, often collectively referred to as the 'Bonjour' protocols after their naming by Apple, provide the means for devices to discover services within a local link and, in the absence of a unicast DNS service, to exchange naming information.

Where devices are to be deployed in networks where service discovery would be beneficial, e.g., for users seeking to discover printers or display devices, mDNS and DNS-SD SHOULD be supported.

10. IPv4 Support and Transition

IPv6 nodes MAY support IPv4.

10.1. Transition Mechanisms

10.1.1. Basic Transition Mechanisms for IPv6 Hosts and Routers - RFC 4213

If an IPv6 node implements dual stack and tunneling, then [RFC4213] MUST be supported.

11. Application Support

11.1. Textual Representation of IPv6 Addresses - RFC 5952

Software that allows users and operators to input IPv6 addresses in text form SHOULD support "A Recommendation for IPv6 Address Text Representation" [RFC5952].

11.2. Application Programming Interfaces (APIs)

There are a number of IPv6-related APIs. This document does not mandate the use of any, because the choice of API does not directly relate to on-the-wire behavior of protocols. Implementers, however, would be advised to consider providing a common API or reviewing existing APIs for the type of functionality they provide to applications.

"Basic Socket Interface Extensions for IPv6" [RFC3493] provides IPv6 functionality used by typical applications. Implementers should note that RFC 3493 has been picked up and further standardized by the Portable Operating System Interface (POSIX) [POSIX].

"Advanced Sockets Application Program Interface (API) for IPv6" [RFC3542] provides access to advanced IPv6 features needed by diagnostic and other more specialized applications.

"IPv6 Socket API for Source Address Selection" [RFC5014] provides facilities that allow an application to override the default Source Address Selection rules of [RFC6724].

"Socket Interface Extensions for Multicast Source Filters" [RFC3678] provides support for expressing source filters on multicast group memberships.

"Extension to Sockets API for Mobile IPv6" [RFC4584] provides application support for accessing and enabling Mobile IPv6 [RFC6275] features.

12. Mobility

Mobile IPv6 [RFC6275] and associated specifications [RFC3776] [RFC4877] allow a node to change its point of attachment within the Internet, while maintaining (and using) a permanent address. All communication using the permanent address continues to proceed as expected even as the node moves around. The definition of Mobile IP includes requirements for the following types of nodes:

- mobile nodes

- correspondent nodes with support for route optimization

- home agents

- all IPv6 routers

At the present time, Mobile IP has seen only limited implementation and no significant deployment, partly because it originally assumed an IPv6-only environment rather than a mixed IPv4/IPv6 Internet. Additional work has been done to support mobility in mixed-mode IPv4 and IPv6 networks [RFC5555].

More usage and deployment experience is needed with mobility before any specific approach can be recommended for broad implementation in all hosts and routers. Consequently, Mobility Support in IPv6 [RFC6275], Mobile IPv6 Support for Dual Stack Hosts and Routers [RFC5555], and associated standards (such as Mobile IPv6 with IKEv2 and IPsec [RFC4877]) are considered a MAY at this time.

IPv6 for 3GPP [RFC7066] lists a snapshot of required IPv6 functionalities at the time the document was published that would need to be implemented, going above and beyond the recommendations in this document. Additionally, a 3GPP IPv6 Host MAY implement [RFC7278] to deliver IPv6 prefixes on the LAN link.

13. Security

This section describes the security specification for IPv6 nodes.

Achieving security in practice is a complex undertaking. Operational procedures, protocols, key distribution mechanisms, certificate management approaches, etc., are all components that impact the level of security actually achieved in practice. More importantly, deficiencies or a poor fit in any one individual component can significantly reduce the overall effectiveness of a particular security approach.

IPsec can provide either end-to-end security between nodes or channel security (for example, via a site-to-site IPsec VPN), making it possible to provide secure communication for all (or a subset of) communication flows at the IP layer between pairs of Internet nodes. IPsec has two standard operating modes: Tunnel-mode and Transport-mode. In Tunnel-mode, IPsec provides network-layer security and protects an entire IP packet by encapsulating the original IP packet and then prepending a new IP header. In Transport-mode, IPsec provides security for the transport layer (and above) by encapsulating only the transport-layer (and above) portion of the IP packet (i.e., without adding a second IP header).

Although IPsec can be used with manual keying in some cases, such usage has limited applicability and is not recommended.

A range of security technologies and approaches proliferate today (e.g., IPsec, Transport Layer Security (TLS), Secure SHell (SSH), TLS VPNS, etc.). No single approach has emerged as an ideal technology for all needs and environments. Moreover, IPsec is not viewed as the ideal security technology in all cases and is unlikely to displace the others.

Previously, IPv6 mandated implementation of IPsec and recommended the key-management approach of IKE. RFC 6434 updated that recommendation by making support of the IPsec architecture [RFC4301] a SHOULD for all IPv6 nodes, and this document retains that recommendation. Note that the IPsec Architecture requires the implementation of both manual and automatic key management (e.g., Section 4.5 of RFC 4301). Currently, the recommended automated key-management protocol to implement is IKEv2 [RFC7296].

This document recognizes that there exists a range of device types and environments where approaches to security other than IPsec can be justified. For example, special-purpose devices may support only a very limited number or type of applications, and an application-specific security approach may be sufficient for limited management or configuration capabilities. Alternatively, some devices may run on extremely constrained hardware (e.g., sensors) where the full IPsec Architecture is not justified.

Because most common platforms now support IPv6 and have it enabled by default, IPv6 security is an issue for networks that are ostensibly IPv4 only; see [RFC7123] for guidance on this area.

13.1. Requirements

"Security Architecture for the Internet Protocol" [RFC4301] SHOULD be supported by all IPv6 nodes. Note that the IPsec Architecture requires the implementation of both manual and automatic key management (e.g., Section 4.5 of [RFC4301]). Currently, the default automated key-management protocol to implement is IKEv2. As required in [RFC4301], IPv6 nodes implementing the IPsec Architecture MUST implement ESP [RFC4303] and MAY implement AH [RFC4302].

13.2. Transforms and Algorithms

The current set of mandatory-to-implement algorithms for the IPsec Architecture are defined in Cryptographic Algorithm Implementation Requirements for ESP and AH [RFC8221]. IPv6 nodes implementing the IPsec Architecture MUST conform to the requirements in [RFC8221]. Preferred cryptographic algorithms often change more frequently than security protocols. Therefore, implementations MUST allow for migration to new algorithms, as RFC 8221 is replaced or updated in the future.

The current set of mandatory-to-implement algorithms for IKEv2 are defined in Cryptographic Algorithm Implementation Requirements for ESP and AH [RFC8247]. IPv6 nodes implementing IKEv2 MUST conform to the requirements in [RFC8247] and/or any future updates or replacements to [RFC8247].

14. Router-Specific Functionality

This section defines general host considerations for IPv6 nodes that act as routers. Currently, this section does not discuss detailed routing-specific requirements. For the case of typical home routers, [RFC7084] defines basic requirements for customer edge routers.

14.1. IPv6 Router Alert Option - RFC 2711

The IPv6 Router Alert option [RFC2711] is an optional IPv6 Hop-by-Hop Header that is used in conjunction with some protocols (e.g., RSVP [RFC2205] or Multicast Listener Discovery (MLDv2) [RFC3810]). The Router Alert option will need to be implemented whenever such protocols that mandate its use are implemented. See Section 5.11.

14.2. Neighbor Discovery for IPv6 - RFC 4861

Sending Router Advertisements and processing Router Solicitations MUST be supported.

Section 7 of [RFC6275] includes some mobility-specific extensions to Neighbor Discovery. Routers SHOULD implement Sections 7.3 and 7.5, even if they do not implement home agent functionality.

14.3. Stateful Address Autoconfiguration (DHCPv6) - RFC 3315

A single DHCP server ([RFC3315] or [RFC4862]) can provide configuration information to devices directly attached to a shared link, as well as to devices located elsewhere within a site. Communication between a client and a DHCP server located on different links requires the use of DHCP relay agents on routers.

In simple deployments, consisting of a single router and either a single LAN or multiple LANs attached to the single router, together with a WAN connection, a DHCP server embedded within the router is one common deployment scenario (e.g., [RFC7084]). There is no need for relay agents in such scenarios.

In more complex deployment scenarios, such as within enterprise or service provider networks, the use of DHCP requires some level of configuration, in order to configure relay agents, DHCP servers, etc. In such environments, the DHCP server might even be run on a traditional server, rather than as part of a router.

Because of the wide range of deployment scenarios, support for DHCP server functionality on routers is optional. However, routers targeted for deployment within more complex scenarios (as described above) SHOULD support relay agent functionality. Note that "Basic Requirements for IPv6 Customer Edge Routers" [RFC7084] requires implementation of a DHCPv6 server function in IPv6 Customer Edge (CE) routers.

14.4. IPv6 Prefix Length Recommendation for Forwarding - BCP 198

Forwarding nodes MUST conform to BCP 198 [RFC7608]; thus, IPv6 implementations of nodes that may forward packets MUST conform to the rules specified in Section 5.1 of [RFC4632].

15. Constrained Devices

The focus of this document is general IPv6 nodes. In this section, we briefly discuss considerations for constrained devices.

In the case of constrained nodes, with limited CPU, memory, bandwidth or power, support for certain IPv6 functionality may need to be considered due to those limitations. While the requirements of this document are RECOMMENDED for all nodes, including constrained nodes, compromises may need to be made in certain cases. Where such compromises are made, the interoperability of devices should be strongly considered, particularly where this may impact other nodes on the same link, e.g., only supporting MLDv1 will affect other nodes.

The IETF 6LowPAN (IPv6 over Low-Power Wireless Personal Area Network) WG produced six RFCs, including a general overview and problem statement [RFC4919] (the means by which IPv6 packets are transmitted over IEEE 802.15.4 networks [RFC4944] and ND optimizations for that medium [RFC6775]).

IPv6 nodes that are battery powered SHOULD implement the recommendations in [RFC7772].

16. IPv6 Node Management

Network management MAY be supported by IPv6 nodes. However, for IPv6 nodes that are embedded devices, network management may be the only possible way of controlling these nodes.

Existing network management protocols include SNMP [RFC3411], NETCONF [RFC6241], and RESTCONF [RFC8040].

16.1. Management Information Base (MIB) Modules

The obsoleted status of various IPv6-specific MIB modules is clarified in [RFC8096].

The following two MIB modules SHOULD be supported by nodes that support an SNMP agent.

16.1.1. IP Forwarding Table MIB

The IP Forwarding Table MIB [RFC4292] SHOULD be supported by nodes that support an SNMP agent.

16.1.2. Management Information Base for the Internet Protocol (IP)

The IP MIB [RFC4293] SHOULD be supported by nodes that support an SNMP agent.

16.1.3. Interface MIB

The Interface MIB [RFC2863] SHOULD be supported by nodes that support an SNMP agent.

16.2. YANG Data Models

The following YANG data models SHOULD be supported by nodes that support a NETCONF or RESTCONF agent.

16.2.1. IP Management YANG Model

The IP Management YANG Model [RFC8344] SHOULD be supported by nodes that support NETCONF or RESTCONF.

16.2.2. Interface Management YANG Model

The Interface Management YANG Model [RFC8343] SHOULD be supported by nodes that support NETCONF or RESTCONF.

17. Security Considerations

This document does not directly affect the security of the Internet, beyond the security considerations associated with the individual protocols.

Security is also discussed in Section 13 above.

18. IANA Considerations

This document has no IANA actions.

19. References

19.1. Normative References

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <https://www.rfc-editor.org/info/rfc1034>.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <https://www.rfc-editor.org/info/rfc1035>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2710] Deering, S., Fenner, W., and B. Haberman, "Multicast Listener Discovery (MLD) for IPv6", RFC 2710, DOI 10.17487/RFC2710, October 1999, <https://www.rfc-editor.org/info/rfc2710>.

[RFC2711] Partridge, C. and A. Jackson, "IPv6 Router Alert Option", RFC 2711, DOI 10.17487/RFC2711, October 1999, <https://www.rfc-editor.org/info/rfc2711>.

[RFC2863] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB", RFC 2863, DOI 10.17487/RFC2863, June 2000, <https://www.rfc-editor.org/info/rfc2863>.

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, DOI 10.17487/RFC3168, September 2001, <https://www.rfc-editor.org/info/rfc3168>.

[RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <https://www.rfc-editor.org/info/rfc3315>.

[RFC3411] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", STD 62, RFC 3411, DOI 10.17487/RFC3411, December 2002, <https://www.rfc-editor.org/info/rfc3411>.

[RFC3596] Thomson, S., Huitema, C., Ksinant, V., and M. Souissi, "DNS Extensions to Support IP Version 6", STD 88, RFC 3596, DOI 10.17487/RFC3596, October 2003, <https://www.rfc-editor.org/info/rfc3596>.

[RFC3736] Droms, R., "Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6", RFC 3736, DOI 10.17487/RFC3736, April 2004, <https://www.rfc-editor.org/info/rfc3736>.

[RFC3810] Vida, R., Ed. and L. Costa, Ed., "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, DOI 10.17487/RFC3810, June 2004, <https://www.rfc-editor.org/info/rfc3810>.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <https://www.rfc-editor.org/info/rfc4033>.

[RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005, <https://www.rfc-editor.org/info/rfc4034>.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005, <https://www.rfc-editor.org/info/rfc4035>.

[RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, DOI 10.17487/RFC4213, October 2005, <https://www.rfc-editor.org/info/rfc4213>.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, DOI 10.17487/RFC4291, February 2006, <https://www.rfc-editor.org/info/rfc4291>.

[RFC4292] Haberman, B., "IP Forwarding Table MIB", RFC 4292, DOI 10.17487/RFC4292, April 2006, <https://www.rfc-editor.org/info/rfc4292>.

[RFC4293] Routhier, S., Ed., "Management Information Base for the Internet Protocol (IP)", RFC 4293, DOI 10.17487/RFC4293, April 2006, <https://www.rfc-editor.org/info/rfc4293>.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <https://www.rfc-editor.org/info/rfc4301>.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, DOI 10.17487/RFC4303, December 2005, <https://www.rfc-editor.org/info/rfc4303>.

[RFC4311] Hinden, R. and D. Thaler, "IPv6 Host-to-Router Load Sharing", RFC 4311, DOI 10.17487/RFC4311, November 2005, <https://www.rfc-editor.org/info/rfc4311>.

[RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", STD 89, RFC 4443, DOI 10.17487/RFC4443, March 2006, <https://www.rfc-editor.org/info/rfc4443>.

[RFC4607] Holbrook, H. and B. Cain, "Source-Specific Multicast for IP", RFC 4607, DOI 10.17487/RFC4607, August 2006, <https://www.rfc-editor.org/info/rfc4607>.

[RFC4632] Fuller, V. and T. Li, "Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan", BCP 122, RFC 4632, DOI 10.17487/RFC4632, August 2006, <https://www.rfc-editor.org/info/rfc4632>.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, <https://www.rfc-editor.org/info/rfc4861>.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, DOI 10.17487/RFC4862, September 2007, <https://www.rfc-editor.org/info/rfc4862>.

[RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, DOI 10.17487/RFC4941, September 2007, <https://www.rfc-editor.org/info/rfc4941>.

[RFC5095] Abley, J., Savola, P., and G. Neville-Neil, "Deprecation of Type 0 Routing Headers in IPv6", RFC 5095, DOI 10.17487/RFC5095, December 2007, <https://www.rfc-editor.org/info/rfc5095>.

[RFC5453] Krishnan, S., "Reserved IPv6 Interface Identifiers", RFC 5453, DOI 10.17487/RFC5453, February 2009, <https://www.rfc-editor.org/info/rfc5453>.

[RFC5722] Krishnan, S., "Handling of Overlapping IPv6 Fragments", RFC 5722, DOI 10.17487/RFC5722, December 2009, <https://www.rfc-editor.org/info/rfc5722>.

[RFC5790] Liu, H., Cao, W., and H. Asaeda, "Lightweight Internet Group Management Protocol Version 3 (IGMPv3) and Multicast Listener Discovery Version 2 (MLDv2) Protocols", RFC 5790, DOI 10.17487/RFC5790, February 2010, <https://www.rfc-editor.org/info/rfc5790>.

[RFC5942] Singh, H., Beebee, W., and E. Nordmark, "IPv6 Subnet Model: The Relationship between Links and Subnet Prefixes", RFC 5942, DOI 10.17487/RFC5942, July 2010, <https://www.rfc-editor.org/info/rfc5942>.

[RFC5952] Kawamura, S. and M. Kawashima, "A Recommendation for IPv6 Address Text Representation", RFC 5952, DOI 10.17487/RFC5952, August 2010, <https://www.rfc-editor.org/info/rfc5952>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6437] Amante, S., Carpenter, B., Jiang, S., and J. Rajahalme, "IPv6 Flow Label Specification", RFC 6437, DOI 10.17487/RFC6437, November 2011, <https://www.rfc-editor.org/info/rfc6437>.

[RFC6564] Krishnan, S., Woodyatt, J., Kline, E., Hoagland, J., and M. Bhatia, "A Uniform Format for IPv6 Extension Headers", RFC 6564, DOI 10.17487/RFC6564, April 2012, <https://www.rfc-editor.org/info/rfc6564>.

[RFC6724] Thaler, D., Ed., Draves, R., Matsumoto, A., and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)", RFC 6724, DOI 10.17487/RFC6724, September 2012, <https://www.rfc-editor.org/info/rfc6724>.

[RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, DOI 10.17487/RFC6762, February 2013, <https://www.rfc-editor.org/info/rfc6762>.

[RFC6763] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", RFC 6763, DOI 10.17487/RFC6763, February 2013, <https://www.rfc-editor.org/info/rfc6763>.

[RFC6775] Shelby, Z., Ed., Chakrabarti, S., Nordmark, E., and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775, DOI 10.17487/RFC6775, November 2012, <https://www.rfc-editor.org/info/rfc6775>.

[RFC6891] Damas, J., Graff, M., and P. Vixie, "Extension Mechanisms for DNS (EDNS(0))", STD 75, RFC 6891, DOI 10.17487/RFC6891, April 2013, <https://www.rfc-editor.org/info/rfc6891>.

[RFC6946] Gont, F., "Processing of IPv6 "Atomic" Fragments", RFC 6946, DOI 10.17487/RFC6946, May 2013, <https://www.rfc-editor.org/info/rfc6946>.

[RFC7045] Carpenter, B. and S. Jiang, "Transmission and Processing of IPv6 Extension Headers", RFC 7045, DOI 10.17487/RFC7045, December 2013, <https://www.rfc-editor.org/info/rfc7045>.

[RFC7048] Nordmark, E. and I. Gashinsky, "Neighbor Unreachability Detection Is Too Impatient", RFC 7048, DOI 10.17487/RFC7048, January 2014, <https://www.rfc-editor.org/info/rfc7048>.

[RFC7112] Gont, F., Manral, V., and R. Bonica, "Implications of Oversized IPv6 Header Chains", RFC 7112, DOI 10.17487/RFC7112, January 2014, <https://www.rfc-editor.org/info/rfc7112>.

[RFC7217] Gont, F., "A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)", RFC 7217, DOI 10.17487/RFC7217, April 2014, <https://www.rfc-editor.org/info/rfc7217>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[RFC7527] Asati, R., Singh, H., Beebee, W., Pignataro, C., Dart, E., and W. George, "Enhanced Duplicate Address Detection", RFC 7527, DOI 10.17487/RFC7527, April 2015, <https://www.rfc-editor.org/info/rfc7527>.

[RFC7559] Krishnan, S., Anipko, D., and D. Thaler, "Packet-Loss Resiliency for Router Solicitations", RFC 7559, DOI 10.17487/RFC7559, May 2015, <https://www.rfc-editor.org/info/rfc7559>.

[RFC7608] Boucadair, M., Petrescu, A., and F. Baker, "IPv6 Prefix Length Recommendation for Forwarding", BCP 198, RFC 7608, DOI 10.17487/RFC7608, July 2015, <https://www.rfc-editor.org/info/rfc7608>.

[RFC8021] Gont, F., Liu, W., and T. Anderson, "Generation of IPv6 Atomic Fragments Considered Harmful", RFC 8021, DOI 10.17487/RFC8021, January 2017, <https://www.rfc-editor.org/info/rfc8021>.

[RFC8028] Baker, F. and B. Carpenter, "First-Hop Router Selection by Hosts in a Multi-Prefix Network", RFC 8028, DOI 10.17487/RFC8028, November 2016, <https://www.rfc-editor.org/info/rfc8028>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8064] Gont, F., Cooper, A., Thaler, D., and W. Liu, "Recommendation on Stable IPv6 Interface Identifiers", RFC 8064, DOI 10.17487/RFC8064, February 2017, <https://www.rfc-editor.org/info/rfc8064>.

[RFC8106] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Options for DNS Configuration", RFC 8106, DOI 10.17487/RFC8106, March 2017, <https://www.rfc-editor.org/info/rfc8106>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[RFC8201] McCann, J., Deering, S., Mogul, J., and R. Hinden, Ed., "Path MTU Discovery for IP version 6", STD 87, RFC 8201, DOI 10.17487/RFC8201, July 2017, <https://www.rfc-editor.org/info/rfc8201>.

[RFC8221] Wouters, P., Migault, D., Mattsson, J., Nir, Y., and T. Kivinen, "Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)", RFC 8221, DOI 10.17487/RFC8221, October 2017, <https://www.rfc-editor.org/info/rfc8221>.

[RFC8247] Nir, Y., Kivinen, T., Wouters, P., and D. Migault, "Algorithm Implementation Requirements and Usage Guidance for the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 8247, DOI 10.17487/RFC8247, September 2017, <https://www.rfc-editor.org/info/rfc8247>.

[RFC8343] Bjorklund, M., "A YANG Data Model for Interface Management", RFC 8343, DOI 10.17487/RFC8343, March 2018, <https://www.rfc-editor.org/info/rfc8343>.

[RFC8344] Bjorklund, M., "A YANG Data Model for IP Management", RFC 8344, DOI 10.17487/RFC8344, March 2018, <https://www.rfc-editor.org/info/rfc8344>.

19.2. Informative References

[RFC793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, DOI 10.17487/RFC0793, September 1981, <https://www.rfc-editor.org/info/rfc793>.

[RFC2205] Braden, R., Ed., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, DOI 10.17487/RFC2205, September 1997, <https://www.rfc-editor.org/info/rfc2205>.

[RFC2464] Crawford, M., "Transmission of IPv6 Packets over Ethernet Networks", RFC 2464, DOI 10.17487/RFC2464, December 1998, <https://www.rfc-editor.org/info/rfc2464>.

[RFC2491] Armitage, G., Schulter, P., Jork, M., and G. Harter, "IPv6 over Non-Broadcast Multiple Access (NBMA) networks", RFC 2491, DOI 10.17487/RFC2491, January 1999, <https://www.rfc-editor.org/info/rfc2491>.

[RFC2590] Conta, A., Malis, A., and M. Mueller, "Transmission of IPv6 Packets over Frame Relay Networks Specification", RFC 2590, DOI 10.17487/RFC2590, May 1999, <https://www.rfc-editor.org/info/rfc2590>.

[RFC2874] Crawford, M. and C. Huitema, "DNS Extensions to Support IPv6 Address Aggregation and Renumbering", RFC 2874, DOI 10.17487/RFC2874, July 2000, <https://www.rfc-editor.org/info/rfc2874>.

[RFC2923] Lahey, K., "TCP Problems with Path MTU Discovery", RFC 2923, DOI 10.17487/RFC2923, September 2000, <https://www.rfc-editor.org/info/rfc2923>.

[RFC3146] Fujisawa, K. and A. Onoe, "Transmission of IPv6 Packets over IEEE 1394 Networks", RFC 3146, DOI 10.17487/RFC3146, October 2001, <https://www.rfc-editor.org/info/rfc3146>.

[RFC3363] Bush, R., Durand, A., Fink, B., Gudmundsson, O., and T. Hain, "Representing Internet Protocol version 6 (IPv6) Addresses in the Domain Name System (DNS)", RFC 3363, DOI 10.17487/RFC3363, August 2002, <https://www.rfc-editor.org/info/rfc3363>.

[RFC3493] Gilligan, R., Thomson, S., Bound, J., McCann, J., and W. Stevens, "Basic Socket Interface Extensions for IPv6", RFC 3493, DOI 10.17487/RFC3493, February 2003, <https://www.rfc-editor.org/info/rfc3493>.

[RFC3542] Stevens, W., Thomas, M., Nordmark, E., and T. Jinmei, "Advanced Sockets Application Program Interface (API) for IPv6", RFC 3542, DOI 10.17487/RFC3542, May 2003, <https://www.rfc-editor.org/info/rfc3542>.

[RFC3646] Droms, R., Ed., "DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3646, DOI 10.17487/RFC3646, December 2003, <https://www.rfc-editor.org/info/rfc3646>.

[RFC3678] Thaler, D., Fenner, B., and B. Quinn, "Socket Interface Extensions for Multicast Source Filters", RFC 3678, DOI 10.17487/RFC3678, January 2004, <https://www.rfc-editor.org/info/rfc3678>.

[RFC3776] Arkko, J., Devarapalli, V., and F. Dupont, "Using IPsec to Protect Mobile IPv6 Signaling Between Mobile Nodes and Home Agents", RFC 3776, DOI 10.17487/RFC3776, June 2004, <https://www.rfc-editor.org/info/rfc3776>.

[RFC3971] Arkko, J., Ed., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, DOI 10.17487/RFC3971, March 2005, <https://www.rfc-editor.org/info/rfc3971>.

[RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", RFC 3972, DOI 10.17487/RFC3972, March 2005, <https://www.rfc-editor.org/info/rfc3972>.

[RFC4191] Draves, R. and D. Thaler, "Default Router Preferences and More-Specific Routes", RFC 4191, DOI 10.17487/RFC4191, November 2005, <https://www.rfc-editor.org/info/rfc4191>.

[RFC4294] Loughney, J., Ed., "IPv6 Node Requirements", RFC 4294, DOI 10.17487/RFC4294, April 2006, <https://www.rfc-editor.org/info/rfc4294>.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, DOI 10.17487/RFC4302, December 2005, <https://www.rfc-editor.org/info/rfc4302>.

[RFC4338] DeSanti, C., Carlson, C., and R. Nixon, "Transmission of IPv6, IPv4, and Address Resolution Protocol (ARP) Packets over Fibre Channel", RFC 4338, DOI 10.17487/RFC4338, January 2006, <https://www.rfc-editor.org/info/rfc4338>.

[RFC4380] Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", RFC 4380, DOI 10.17487/RFC4380, February 2006, <https://www.rfc-editor.org/info/rfc4380>.

[RFC4429] Moore, N., "Optimistic Duplicate Address Detection (DAD) for IPv6", RFC 4429, DOI 10.17487/RFC4429, April 2006, <https://www.rfc-editor.org/info/rfc4429>.

[RFC4584] Chakrabarti, S. and E. Nordmark, "Extension to Sockets API for Mobile IPv6", RFC 4584, DOI 10.17487/RFC4584, July 2006, <https://www.rfc-editor.org/info/rfc4584>.

[RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, DOI 10.17487/RFC4821, March 2007, <https://www.rfc-editor.org/info/rfc4821>.

[RFC4877] Devarapalli, V. and F. Dupont, "Mobile IPv6 Operation with IKEv2 and the Revised IPsec Architecture", RFC 4877, DOI 10.17487/RFC4877, April 2007, <https://www.rfc-editor.org/info/rfc4877>.

[RFC4884] Bonica, R., Gan, D., Tappan, D., and C. Pignataro, "Extended ICMP to Support Multi-Part Messages", RFC 4884, DOI 10.17487/RFC4884, April 2007, <https://www.rfc-editor.org/info/rfc4884>.

[RFC4890] Davies, E. and J. Mohacsi, "Recommendations for Filtering ICMPv6 Messages in Firewalls", RFC 4890, DOI 10.17487/RFC4890, May 2007, <https://www.rfc-editor.org/info/rfc4890>.

[RFC4919] Kushalnagar, N., Montenegro, G., and C. Schumacher, "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals", RFC 4919, DOI 10.17487/RFC4919, August 2007, <https://www.rfc-editor.org/info/rfc4919>.

[RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, DOI 10.17487/RFC4944, September 2007, <https://www.rfc-editor.org/info/rfc4944>.

[RFC5014] Nordmark, E., Chakrabarti, S., and J. Laganier, "IPv6 Socket API for Source Address Selection", RFC 5014, DOI 10.17487/RFC5014, September 2007, <https://www.rfc-editor.org/info/rfc5014>.

[RFC5072] Varada, S., Ed., Haskins, D., and E. Allen, "IP Version 6 over PPP", RFC 5072, DOI 10.17487/RFC5072, September 2007, <https://www.rfc-editor.org/info/rfc5072>.

[RFC5121] Patil, B., Xia, F., Sarikaya, B., Choi, JH., and S. Madanapalli, "Transmission of IPv6 via the IPv6 Convergence Sublayer over IEEE 802.16 Networks", RFC 5121, DOI 10.17487/RFC5121, February 2008, <https://www.rfc-editor.org/info/rfc5121>.

[RFC5555] Soliman, H., Ed., "Mobile IPv6 Support for Dual Stack Hosts and Routers", RFC 5555, DOI 10.17487/RFC5555, June 2009, <https://www.rfc-editor.org/info/rfc5555>.

[RFC6275] Perkins, C., Ed., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, DOI 10.17487/RFC6275, July 2011, <https://www.rfc-editor.org/info/rfc6275>.

[RFC6563] Jiang, S., Conrad, D., and B. Carpenter, "Moving A6 to Historic Status", RFC 6563, DOI 10.17487/RFC6563, March 2012, <https://www.rfc-editor.org/info/rfc6563>.

[RFC6980] Gont, F., "Security Implications of IPv6 Fragmentation with IPv6 Neighbor Discovery", RFC 6980, DOI 10.17487/RFC6980, August 2013, <https://www.rfc-editor.org/info/rfc6980>.

[RFC7066] Korhonen, J., Ed., Arkko, J., Ed., Savolainen, T., and S. Krishnan, "IPv6 for Third Generation Partnership Project (3GPP) Cellular Hosts", RFC 7066, DOI 10.17487/RFC7066, November 2013, <https://www.rfc-editor.org/info/rfc7066>.

[RFC7084] Singh, H., Beebee, W., Donley, C., and B. Stark, "Basic Requirements for IPv6 Customer Edge Routers", RFC 7084, DOI 10.17487/RFC7084, November 2013, <https://www.rfc-editor.org/info/rfc7084>.

[RFC7123] Gont, F. and W. Liu, "Security Implications of IPv6 on IPv4 Networks", RFC 7123, DOI 10.17487/RFC7123, February 2014, <https://www.rfc-editor.org/info/rfc7123>.

[RFC7278] Byrne, C., Drown, D., and A. Vizdal, "Extending an IPv6 /64 Prefix from a Third Generation Partnership Project (3GPP) Mobile Interface to a LAN Link", RFC 7278, DOI 10.17487/RFC7278, June 2014, <https://www.rfc-editor.org/info/rfc7278>.

[RFC7371] Boucadair, M. and S. Venaas, "Updates to the IPv6 Multicast Addressing Architecture", RFC 7371, DOI 10.17487/RFC7371, September 2014, <https://www.rfc-editor.org/info/rfc7371>.

[RFC7421] Carpenter, B., Ed., Chown, T., Gont, F., Jiang, S., Petrescu, A., and A. Yourtchenko, "Analysis of the 64-bit Boundary in IPv6 Addressing", RFC 7421, DOI 10.17487/RFC7421, January 2015, <https://www.rfc-editor.org/info/rfc7421>.

[RFC7721] Cooper, A., Gont, F., and D. Thaler, "Security and Privacy Considerations for IPv6 Address Generation Mechanisms", RFC 7721, DOI 10.17487/RFC7721, March 2016, <https://www.rfc-editor.org/info/rfc7721>.

[RFC7739] Gont, F., "Security Implications of Predictable Fragment Identification Values", RFC 7739, DOI 10.17487/RFC7739, February 2016, <https://www.rfc-editor.org/info/rfc7739>.

[RFC7772] Yourtchenko, A. and L. Colitti, "Reducing Energy Consumption of Router Advertisements", BCP 202, RFC 7772, DOI 10.17487/RFC7772, February 2016, <https://www.rfc-editor.org/info/rfc7772>.

[RFC7844] Huitema, C., Mrugalski, T., and S. Krishnan, "Anonymity Profiles for DHCP Clients", RFC 7844, DOI 10.17487/RFC7844, May 2016, <https://www.rfc-editor.org/info/rfc7844>.

[RFC7934] Colitti, L., Cerf, V., Cheshire, S., and D. Schinazi, "Host Address Availability Recommendations", BCP 204, RFC 7934, DOI 10.17487/RFC7934, July 2016, <https://www.rfc-editor.org/info/rfc7934>.

[RFC8087] Fairhurst, G. and M. Welzl, "The Benefits of Using Explicit Congestion Notification (ECN)", RFC 8087, DOI 10.17487/RFC8087, March 2017, <https://www.rfc-editor.org/info/rfc8087>.

[RFC8096] Fenner, B., "The IPv6-Specific MIB Modules Are Obsolete", RFC 8096, DOI 10.17487/RFC8096, April 2017, <https://www.rfc-editor.org/info/rfc8096>.

[RFC8273] Brzozowski, J. and G. Van de Velde, "Unique IPv6 Prefix per Host", RFC 8273, DOI 10.17487/RFC8273, December 2017, <https://www.rfc-editor.org/info/rfc8273>.

[POSIX] IEEE, "Information Technology -- Portable Operating System Interface (POSIX(R)) Base Specifications, Issue 7", IEEE Std 1003.1-2017, DOI: 10.1109/IEEESTD.2018.8277153, January 2018, <https://ieeexplore.ieee.org/document/8277153>.

[USGv6] National Institute of Standards and Technology, "A Profile for IPv6 in the U.S. Government - Version 1.0", NIST SP500-267, July 2008, <https://www.nist.gov/programs-projects/usgv6-program>.

Appendix A. Changes from RFC 6434

There have been many editorial clarifications as well as significant additions and updates. While this section highlights some of the changes, readers should not rely on this section for a comprehensive list of all changes.

1. Restructured sections.

2. Added 6LoWPAN to link layers as it has some deployment.

3. Removed the Downstream-on-Demand (DoD) IPv6 Profile as it hasn't been updated.

4. Updated MLDv2 support to a MUST since nodes are restricted if MLDv1 is used.

5. Required DNS RA options so SLAAC-only devices can get DNS; RFC 8106 is a MUST.

6. Required RFC 3646 DNS Options for DHCPv6 implementations.

7. Added RESTCONF and NETCONF as possible options to network management.

8. Added a section on constrained devices.

9. Added text on RFC 7934 to address availability to hosts (SHOULD).

10. Added text on RFC 7844 for anonymity profiles for DHCPv6 clients.

11. Added mDNS and DNS-SD as updated service discovery.

12. Added RFC 8028 as a SHOULD as a method for solving a multi-prefix network.

13. Added ECN RFC 3168 as a SHOULD.

14. Added reference to RFC 7123 for security over IPv4-only networks.

15. Removed Jumbograms (RFC 2675) as they aren't deployed.

16. Updated obsoleted RFCs to the new version of the RFC, including RFCs 2460, 1981, 7321, and 4307.

17. Added RFC 7772 for power consumptions considerations.

18. Added why /64 boundaries for more detail -- RFC 7421.

19. Added a unique IPv6 prefix per host to support currently deployed IPv6 networks.

20. Clarified RFC 7066 was a snapshot for 3GPP.

21. Updated RFC 4191 as a MUST and the Type C Host as a SHOULD as they help solve multi-prefix problems.

22. Removed IPv6 over ATM since there aren't many deployments.

23. Added a note in Section 6.6 for Rule 5.5 from RFC 6724.

24. Added MUST for BCP 198 for forwarding IPv6 packets.

25. Added a reference to RFC 8064 for stable address creation.

26. Added text on the protection from excessive extension header options.

27. Added text on the dangers of 1280 MTU UDP, especially with regard to DNS traffic.

28. Added text to clarify RFC 8200 behavior for unrecognized extension headers or unrecognized ULPs.

29. Removed dated email addresses from design team acknowledgements for [RFC4294].

Appendix B. Changes from RFC 4294 to RFC 6434

There have been many editorial clarifications as well as significant additions and updates. While this section highlights some of the changes, readers should not rely on this section for a comprehensive list of all changes.

1. Updated the Introduction to indicate that this document is an applicability statement and is aimed at general nodes.

2. Significantly updated the section on mobility protocols; added references and downgraded previous SHOULDs to MAYs.

3. Changed the Sub-IP Layer section to just list relevant RFCs, and added some more RFCs.

4. Added a section on SEND (it is a MAY).

5. Revised the section on Privacy Extensions [RFC4941] to add more nuance to the recommendation.

6. Completely revised the IPsec/IKEv2 section, downgrading the overall recommendation to a SHOULD.

7. Upgraded recommendation of DHCPv6 to a SHOULD.

8. Added a background section on DHCP versus RA options, added a SHOULD recommendation for DNS configuration via RAs (RFC 6106), and cleaned up the DHCP recommendations.

9. Added the recommendation that routers implement Sections 7.3 and 7.5 of [RFC6275].

10. Added a pointer to subnet clarification document [RFC5942].

11. Added text that "IPv6 Host-to-Router Load Sharing" [RFC4311] SHOULD be implemented.

12. Added reference to [RFC5722] (Overlapping Fragments), and made it a MUST to implement.

13. Made "A Recommendation for IPv6 Address Text Representation" [RFC5952] a SHOULD.

14. Removed the mention of delegation name (DNAME) from the discussion about [RFC3363].

15. Numerous updates to reflect newer versions of IPv6 documents, including [RFC3596], [RFC4213], [RFC4291], and [RFC4443].

16. Removed discussion of "Managed" and "Other" flags in RAs. There is no consensus at present on how to process these flags, and discussion of their semantics was removed in the most recent update of Stateless Address Autoconfiguration [RFC4862].

17. Added many more references to optional IPv6 documents.

18. Made "A Recommendation for IPv6 Address Text Representation" [RFC5952] a SHOULD.

19. Updated the MLD section to include reference to Lightweight MLD [RFC5790].

20. Added a SHOULD recommendation for "Default Router Preferences and More-Specific Routes" [RFC4191].

21. Made "IPv6 Flow Label Specification" [RFC6437] a SHOULD.

Acknowledgments

- Acknowledgments (Current Document)

  The authors would like to thank Brian Carpenter, Dave Thaler, Tom Herbert, Erik Kline, Mohamed Boucadair, and Michayla Newcombe for their contributions and many members of the 6man WG for the inputs they gave.

- Authors and Acknowledgments from RFC 6434

  RFC 6434 was authored by Ed Jankiewicz, John Loughney, and Thomas Narten.

  The authors of RFC 6434 thank Hitoshi Asaeda, Brian Carpenter, Tim Chown, Ralph Droms, Sheila Frankel, Sam Hartman, Bob Hinden, Paul Hoffman, Pekka Savola, Yaron Sheffer, and Dave Thaler for their comments. In addition, the authors thank Mark Andrews for comments and corrections on DNS text and Alfred Hoenes for tracking the updates to various RFCs.

- Authors and Acknowledgments from RFC 4294

  RFC 4294 was written by the IPv6 Node Requirements design team, which had the following members: Jari Arkko, Marc Blanchet, Samita Chakrabarti, Alain Durand, Gerard Gastaud, Jun-ichiro Itojun Hagino, Atsushi Inoue, Masahiro Ishiyama, John Loughney, Rajiv Raghunarayan, Shoichi Sakane, Dave Thaler, and Juha Wiljakka.

  The authors of RFC 4294 thank Ran Atkinson, Jim Bound, Brian Carpenter, Ralph Droms, Christian Huitema, Adam Machalek, Thomas Narten, Juha Ollila, and Pekka Savola for their comments.

Authors' Addresses

Tim Chown
Jisc
Lumen House, Library Avenue
Harwell Oxford, Didcot OX11 0SG
United Kingdom

Email: tim.chown@jisc.ac.uk

John Loughney
Intel
Santa Clara, CA
United States of America

Email: john.loughney@gmail.com

Timothy Winters
University of New Hampshire, Interoperability Lab (UNH-IOL)
Durham, NH
United States of America

Email: twinters@iol.unh.edu