Secure Neighbor Discovery Working Group
Internet-Draft
Expires: July 24, 2004

J. Arkko
Ericsson
J. Kempf
DoCoMo Communications Labs USA
B. Sommerfeld
Sun Microsystems
B. Zill
Microsoft
P. Nikander
Ericsson
January 24, 2004

SEcure Neighbor Discovery (SEND)
draft-ietf-send-ndopt-03

Status of this Memo

Copyright Notice

Abstract

IPv6 nodes use the Neighbor Discovery Protocol (NDP) to discover other nodes on the link, to determine ~~each~~ the link-layer addresses of ~~the~~ nodes on the link, to find routers, and to maintain reachability information about the paths to active neighbors. If not

^L

secured, NDP is vulnerable to various attacks. This document specifies security mechanisms for NDP. Unlike ~~to~~ the original NDP specifications, these mechanisms do not make use of IPsec.

Table of Contents

^L

^L

1. Introduction

IPv6 defines the Neighbor Discovery Protocol (NDP) in RFCs 2461 [7] and 2462 [8]. Nodes on the same link use NDP to discover each other's presence, to determine each other's link-layer addresses, to find routers, and to maintain reachability information about the paths to active neighbors. NDP is used both by hosts and routers. Its functions include Neighbor Discovery (ND), Router Discovery (RD), Address Autoconfiguration, Address Resolution, Neighbor Unreachability Detection (NUD), Duplicate Address Detection (DAD), and Redirection.

Original NDP specifications called for the use of IPsec for protecting ~~the~~ NDP messages. However, the RFCs do not give detailed instructions for using IPsec ~~to secure NDP.~~ ~~It turns out that~~ in this particular application, IPsec can only be used with ~~a~~ manual configuration of security associations, due to ~~chicken-and-egg~~ problems in using IKE [20, 15]. Furthermore, the number of s manually configured security associations needed for protecting NDP can be very large [21], making that approach impractical for most purposes.

This document is organized as follows. Section 4 describes the overall approach to securing NDP. This approach involves the use of new NDP options to carry public-key based signatures. A zero-configuration mechanism is used for showing address ownership on individual nodes; routers are certified by a trust anchor [10]. The formats, procedures, and cryptographic mechanisms for the zero-configuration mechanism are described in a related specification [12].

The required new NDP options are discussed in Section 5. Section 6 describes the mechanism for distributing certificate chains to establish an authorization delegation chain to a common trust anchor.

Finally, Section 8 discusses the co-existence of secure and non-secure NDP on the same link and Section 9 discusses security considerations for Secure Neighbor Discovery.

1.1 Specification of Requirements

In this document, several words are used to signify the requirements of the specification. These words are often capitalized. The key words "MUST", "MUST NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", and "MAY" in this document are to be interpreted as described in [2].

2. Terms

Authorization Delegation Discovery (ADD)

A process through which SEND nodes can acquire a certificate chain from a peer node to a trust anchor.

Cryptographically Generated Address (CGA)

A technique [12] where the IPv6 address ~~of a node~~ is cryptographically generated using a one-way hash function from the node's public key and some other parameters.

Duplicate Address Detection (DAD)

A mechanism that assures that two IPv6 nodes on the same link are not using the same addresses.

Internet Control Message Protocol version 6 (ICMPv6)

The IPv6 control signaling protocol. Neighbor Discovery Protocol is a part of ICMPv6.

Neighbor Discovery Protocol (NDP)

The IPv6 Neighbor Discovery Protocol [7, 8].

Neighbor Discovery (ND)

The Neighbor Discovery function of the Neighbor Discovery Protocol (NDP). NDP contains also other functions but ND.

Neighbor Unreachability Detection (NUD)

This mechanism is used for tracking the reachability of neighbors.

Nonce

A random number generated by a node and used exactly once. In SEND, nonces are used to ensure that a particular advertisement is linked to the solicitation that triggered it.

Router Authorization Certificate

An X.509v3 [10] PKI certificate using the profile specified in Section 6.1.1.

^L

SEND node

An IPv6 node that implements this specification.

non-SEND node

An IPv6 node that does not implement this specification but uses the legacy RFC 2461 and RFC 2462 mechanisms.

Router Discovery (RD)

The Router Discovery function of the Neighbor Discovery Protocol.

^L

3. Neighbor and Router Discovery Overview

The Neighbor Discovery Protocol has several functions. Many of these functions are overloaded on a few central message types, such as the ICMPv6 Neighbor Advertisement message. In this section we review some of these tasks and their effects in order to understand better how the messages should be treated. This section is not normative, and if this section and the original Neighbor Discovery RFCs are in conflict, the original RFCs take precedence.

The main functions of NDP are the following.

- o The Router Discovery function allows IPv6 hosts to discover the local routers on an attached link. Router Discovery is described in Section 6 of RFC 2461 [7]. The main purpose of Router Discovery is to find neighboring routers that are willing to forward packets on behalf of hosts. Prefix discovery involves determining which destinations are directly on a link; this information is necessary in order to know whether a packet should be sent to a router or to the destination node directly.

- o The Redirect function is used for automatically redirecting a host to a better first-hop router, or to inform hosts that a destination is in fact a neighbor (i.e., on-link). Redirect is specified in Section 8 of RFC 2461 [7].

- o Address Autoconfiguration is used for automatically assigning addresses to a host [8]. This allows hosts to operate without explicit configuration related to IP connectivity. The default autoconfiguration mechanism is stateless. To create IP addresses, the hosts use any prefix information delivered to them during Router Discovery, and then test the newly formed addresses for uniqueness. A stateful mechanism, DHCPv6 [23], provides additional autoconfiguration features.

- o Duplicate Address Detection (DAD) is used for preventing address collisions [8], for instance during Address Autoconfiguration. A node that intends to assign a new address to one of its interfaces first runs the DAD procedure to verify that there is no other node using the same address. Since the rules forbid the use of an address until it has been found unique, no higher layer traffic is possible until this procedure has been completed. Thus, preventing attacks against DAD can help ensure the availability of communications for the node in question.

- The Address Resolution function ~~resolves a node's IPv6 address~~ to the corresponding link-layer address ~~for nodes on the link.~~ Address Resolution is defined in Section 7.2 of RFC 2461 [7], and

it is used for hosts and routers alike. Again, no higher level traffic can proceed until the sender knows the hardware address of the destination node or the next hop router. Note the source link layer address is not checked against the information learned through Address Resolution. This allows for an easier addition of network elements such as bridges and proxies, and eases the stack implementation requirements as less information needs to be passed from layer to layer.

- Neighbor Unreachability Detection (NUD) is used for tracking the reachability of neighboring nodes, both hosts and routers. NUD is defined in Section 7.3 of RFC 2461 [7]. NUD is security-sensitive, because an attacker could falsely claim that reachability exists when it in fact does not.

The NDP messages follow the ICMPv6 message format. All NDP functions are realized using the Router Solicitation (RS), Router Advertisement (RA), Neighbor Solicitation (NS), Neighbor Advertisement (NA), and Redirect messages. An actual NDP message includes an NDP message header, consisting of an ICMPv6 header and ND message-specific data, and zero or more NDP options. The NDP message options are formatted in the Type-Length-Value format.

```
                    <------------NDP Message---------------->
 *-------------------------------------------------------------*
 | IPv6 Header      | ICMPv6    | ND message- | ND Message     |
 | Next Header = 58 | Header    | specific    | Options        |
 | (ICMPv6)         |           | data        |                |
 *-------------------------------------------------------------*
                    <--NDP Message header-->
```

^L

4. Secure Neighbor Discovery Overview

To secure the various functions, a set of new Neighbor Discovery options is introduced. They are used ~~in~~ to protect NDP messages. This specification introduces these options, an authorization delegation discovery process, an address ownership proof mechanism, and requirements for the use of these components in NDP.

The components of the solution specified in this document are as follows:

- Certificate chains, anchored on trusted parties, are expected to certify the authority of routers. A host and a router must have at least one common trust anchor before the host can adopt the router as its default router. Delegation Chain Solicitation and Advertisement messages are used to discover a certificate chain to the trust anchor without requiring the actual Router Discovery messages to carry lengthy certificate chains. The receipt of a protected Router Advertisement message for which no certificate chain is available triggers this ~~process.~~

- Cryptographically Generated Addresses are used to assure that the sender of a Neighbor or Router Advertisement is the "owner" of the claimed address. A public-private key pair ~~needs~~ ~~to~~ be generated by all nodes before they can claim an address. A new NDP option, the CGA option, is used to carry the public key and associated parameters.

  This specification also allows one to use non-CGA ~~addresses and to use~~ certificates to authorize their use. However, the details of such use ~~have been left for future work.~~

- A new NDP option, the Signature option, is used to protect all messages relating to Neighbor and Router discovery.

  Public key signatures ~~are used to~~ protect the integrity of the messages and ~~to~~ authenticate the identity of their sender. The authority of a public key is established either with the authorization delegation process, using certificates, or through the address ownership proof mechanism, using CGAs, or both, depending on configuration and the type of the message protected.

- In order to prevent replay attacks, two new Neighbor Discovery options, Timestamp and Nonce, are used. Given that Neighbor and Router Discovery messages are in some cases sent to multicast addresses, the Timestamp option offers replay protection without any previously established state or sequence numbers. When the messages are used in solicitation - advertisement pairs, they are protected using the Nonce option.

5. Neighbor Discovery Protocol Options

The options described in this section MUST be supported by all SEND nodes.

5.1 CGA Option

The CGA option allows the verification of the sender's CGA. The format of the CGA option is described as follows.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     | Collision Cnt |   Reserved    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                          Modifier                             |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
.                                                               .
.                       Key Information                         .
.                                                               .
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
.                                                               .
.                          Padding                              .
.                                                               .
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

~~The meaning of the fields is described as follows.~~

Type

TBD <To be assigned by IANA> for CGA.

Length

The length of the option, in units of 8 octets.

Collision Cnt

An 8-bit collision count, which can ~~get values~~ 0, 1 and 2. Its semantics are defined in [12].

Reserved

An 8-bit field reserved for future use. The value MUST be initialized to zero by the sender, and MUST be ignored by the receiver.

Modifier

A random 128-bit number used in CGA generation. Its semantics are defined in [12].

Key Information

A variable length field containing the public key of the sender, represented as an ASN.1 type SubjectPublicKeyInfo [10], encoded as described in Section 4 of [12].

This specification requires that if both the CGA option and the Signature option are present, then the publicKey field in the former option MUST be the public key referred by the Key Hash field in the latter option. Packets received with two different keys MUST be silently discarded. Note that a future extension may provide a mechanism which allows the owner of an address and the signer to be different parties.

The length of the Key Information field is determined by the ASN.1 encoding.

Padding

A variable length field making the option length a multiple of 8. ~~It begins~~ after the ASN.1 encoding of the previous field ~~has ends,~~ and continues to the end of the option, as specified by the Length field.

5.1.1 Processing Rules for Senders

The CGA option MUST be present in all Neighbor Solicitation and Advertisement messages, and in Router Solicitation messages not sent with the unspecified source address. The CGA option MAY be present in other messages.

A node sending a message using the CGA option MUST construct the message as follows.

The Modifier, Collision Cnt, and Key Information fields in the CGA option are filled in according to the rules presented above and in

[12]. The ~~used~~ public key is taken from configuration, typically from a data structure associated with the source address. The address MUST be constructed as specified in Section 4 of [12]. Depending on the type of the message, this address appears in different places:

Redirect

The address MUST be the source address of the message.

Neighbor Solicitation

The address MUST be the Target Address for solicitations sent for ~~the purpose of~~ Duplicate Address Detection, and the source address of the message otherwise.

Neighbor Advertisement

The address MUST be the source address of the message.

Router Solicitation

The address MUST be the source address of the message. Note that the CGA option is not used when the source address is the unspecified address.

Router Advertisement

The address MUST be the source address of the message.

5.1.2 Processing Rules for Receivers

Neighbor Solicitation and Advertisement messages without the CGA option MUST be silently discarded. Router Solicitation messages without the CGA option MUST be silently discarded, unless the source address of the message is the unspecified address.

A message containing a CGA option MUST be checked as follows:

If the interface has been configured to use CGA, the receiving node MUST verify the source address of the packet using the algorithm described in Section 5 of [12]. The inputs for the algorithm are the contents of the Collision Cnt, Modifier, and the Key Information fields, the claimed address in the packet (as discussed in the previous section), and the minimum acceptable Sec value. If the CGA verification is successful, the recipient proceeds with the cryptographically more time consuming check of

the signature.

Note that a receiver which does not support CGA or has not specified its use for a given interface can still verify packets using trust anchors, even if CGA ~~had been~~ used on a packet. In such a case, the CGA property of the address is simply left unverified.

5.1.3 Configuration

All nodes that support the verification of the CGA option MUST record the following configuration information:

minbits

The minimum acceptable key length for ~~the~~ public keys used in the generation of ~~the~~ CGA ~~address.~~ The default SHOULD be 1024 bits.

Implementations MAY also set an upper limit in order to limit the amount of computation they need to perform when verifying packets that use these security associations. Any implementation should follow prudent cryptographic practice in determining the appropriate key lengths.

minSec

The minimum acceptable Sec value, if CGA verification is required (see Section 2 in [12]). This parameter is intended to facilitate future extensions and experimental work. Currently, the minSec value SHOULD always be set to zero.

All nodes that support the sending of the CGA option MUST record the following configuration information:

CGA parameters

Any information required to construct CGAs, including the used Sec and Modifier values, and the CGA address itself.

5.2 Signature Option

The Signature option allows public-key based signatures to be attached to NDP messages. Both trust anchor authentication and CGAs ~~can be used.~~ The format of the Signature option is described in the following:

```
Arkko, et al.            Expires July 24, 2004                [Page 14]
^L
Internet-Draft      SEcure Neighbor Discovery (SEND)        January 2004
```

```
      0                   1                   2                   3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |     Type      |    Length     |  Pad Length   |   Reserved    |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                                                               |
     |                          Key Hash                             |
     |                                                               |
     |                                                               |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                                                               |
     .                                                               .
     .                       Digital Signature                       .
     .                                                               .
     |                                                               |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                                                               |
     .                                                               .
     .                          Padding                              .
     .                                                               .
     |                                                               |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

~~The meaning of the fields is described below:~~

Type

TBD <To be assigned by IANA> for Signature.

Length

The length of the option, in units of 8 octets.

Pad Length

An 8-bit integer field, giving the length of the Pad field in units of an octet.

Reserved

An ~~an~~ 8-bit field reserved for future use. The value MUST be initialized to zero by the sender, and MUST be ignored by the receiver.

Key Hash

A 128-bit field contains the most significant (leftmost) 128-bits of a SHA1 hash of the public key used for ~~the~~ constructing the

^L

signature. The SHA1 is taken over the presentation used in the Key Information field of the CGA option. Its purpose is to associate the signature to a particular key known by the receiver. Such a key can be either stored in the certificate cache of the receiver, or be received in the CGA option in the same message.

Digital Signature

A variable length field contains the signature constructed using the sender's private key, over the the following sequence of octets:

1. The 128-bit CGA Type Tag [12] value for SEND, 0x086F CA5E 10B2 00C9 9C8C E001 6427 7C08 (generated randomly).

2. The 128-bit Source Address field from the IP header.

3. The 128-bit Destination Address field from the IP header.

4. The 32-bit ICMP header.

5. The NDP message header.

6. All NDP options preceding the Signature option.

The signature is constructed using the RSA algorithm and MUST be encoded as private key encryption in PKCS#1 format [13]. The signature value is computed with the RSASSA-PKCS1-v1_5 algorithm and SHA-1 hash as defined in [13].

This field starts after the Key Hash field. The length of the Digital Signature field is determined by the length of the Signature option minus the length of the other fields (including the variable length Pad field).

This variable length field contains padding, as many bytes as is given by the Pad Length Field.

5.2.1 Processing Rules for Senders

Neighbor Solicitation, Neighbor Advertisement, Router Advertisement, and Redirect messages MUST contain the Signature option. Router Solicitation messages not sent with the unspecified source address MUST contain the Signature option.

A node sending a message using the Signature option MUST construct the message as follows:

^L

- o The message is constructed in its entirety, without the Signature option.
- o The Signature option is added as the last option in the message.
- o For the purpose of constructing a signature, the following data items are concatenated:
  - * The 128-bit CGA Type Tag.
  - * The source address of the message.
  - * The destination address of the message.
  - * The contents of the message, starting from the ICMPv6 header, up to but excluding the Signature option.
- o The message, in the form defined above, is signed using the configured private key, and the resulting PKCS#1 signature is put to the Digital Signature field.

5.2.2 Processing Rules for Receivers

Neighbor Solicitation, Neighbor Advertisement, Router Advertisement, and Redirect messages without the Signature option MUST be silently discarded. Router Solicitation messages without the Signature option MUST be silently discarded, unless the source address of the message is the unspecified address.

A message containing a Signature option MUST be checked as follows:

- o The Signature option MUST appear as the last option.

o The Key Hash field MUST indicate the use of a known public key, either one learned from a preceding CGA option, or one known by other means.

o The Digital Signature field MUST have correct encoding, and not exceed the length of the Signature option.

o The Digital Signature verification MUST show that the signature has been calculated as specified in the previous section.

o If the use of a trust anchor has been configured, a valid authorization delegation chain MUST be known between the receiver's trust anchor and the sender's public key.

^L

Note that the receiver may verify just the CGA property of a packet, even if, in addition to CGA, the sender has used a trust anchor.

Messages that do not pass all the above tests MUST be silently discarded. The receiver MAY silently discard packets ~~also otherwise,~~ e.g., as a response to an apparent CPU exhausting DoS attack.

5.2.3 Configuration

All nodes that support the reception of the Signature options MUST ~~record the following configuration~~ information for each separate NDP message type:

authorization method

This parameter determines the method through which the authority of the sender is determined. It can have four values:

trust anchor

The authority of the sender is verified as described in Section 6.1. The sender may claim additional authorization through the use of CGAs, but that is neither required nor verified.

CGA

The CGA property of the sender's address is verified as described in [12]. The sender may claim additional authority through a trust anchor, but that is neither required nor verified.

trust anchor and CGA

Both the trust anchor and the CGA verification is required.

trust anchor or CGA

Either the trust anchor or the CGA verification is required.

anchor

The public keys and names of the allowed trust anchor(s), if authorization method is not set to CGA.

All nodes that support the sending of Signature options MUST record the following configuration information:

^L

keypair

A public-private key pair. If authorization delegation is in use, there must exist a delegation chain from a trust anchor to this key pair.

CGA flag

A flag that indicates whether CGA is used or ~~is~~ not ~~used.~~ This flag may be per interface or per node.

5.2.4 Performance Considerations

The construction and verification of this option is computationally expensive. In the NDP context, however, the hosts typically have the need to perform only a few signature operations as they enter a link, and a few operations as they find a new on-link peer with which to communicate.

Routers are required to perform a larger number of operations, particularly when the frequency of router advertisements is high due to mobility requirements. Still, the number of required signature operations is on the order of a few dozen ~~ones~~ per second, some of which can be precomputed as discussed below. A large number of router solicitations may cause higher demand for performing asymmetric operations, although RFC 2461 limits the rate at which responses to solicitations can be sent.

Signatures can be precomputed for unsolicited (multicast) Neighbor and Router ~~Advertisements, if~~ the timing of such future advertisements is known. Typically, solicited advertisements are sent to the unicast address from which the solicitation was sent. Given that the IPv6 header is covered by the signature, it is not possible to precompute solicited-~~for~~ advertisements.

5.3 Timestamp and Nonce options

5.3.1 Timestamp Option

The purpose of the Timestamp option is to ~~ensure~~ that unsolicited advertisements and redirects have not been replayed. The format of this option is described in the following:

^L


```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     Type      |    Length     |          Reserved             |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               |
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    +                          Timestamp                            +
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

~~Where the fields are as follows:~~

Type

TBD <To be assigned by IANA> for Timestamp.

Length

The length of the option, in units of 8 octets, i.e., 2.

Reserved

A 48-bit field reserved for future use. The value MUST be initialized to zero by the sender, and MUST be ignored by the receiver.

Timestamp

A 64-bit unsigned integer field containing a timestamp. The value indicates the number of seconds since January 1,, 1970 00:00 UTC, using a fixed point format. In this format the integer number of seconds is contained in the first 48 bits of the field, and the remaining 16 bits indicate the number of 1/64K fractions of a second.

5.3.2 Nonce Option

The purpose of the Nonce option is to ensure that an advertisement is a fresh response to a solicitation sent earlier by the ~~receiving~~ ~~same~~ node. The format of this option is described in the following:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |  Nonce ...                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               |
|                                                               |
.                                                               .
.                                                               .
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

~~Where the fields are as follows:~~

Type

TBD <To be assigned by IANA> for Nonce.

Length

The length of the option, in units of 8 octets.

Nonce

A field containing a random number selected by the sender of the solicitation message. The length of the random number MUST be at least 6 bytes.

5.3.3 Processing rules for senders

All solicitation messages MUST include a Nonce. All solicited-~~for~~ advertisements MUST include a Nonce, ~~copying the nonce value~~ from the received solicitation. When sending a solicitation, the sender MUST store the nonce internally so that it can recognize any replies containing that particular nonce.

All solicitation, advertisement, and redirect messages MUST include a Timestamp. Senders SHOULD set the Timestamp field to the current time, according to their real time clock.

If a message has both Nonce and Timestamp options, the Nonce option SHOULD precede the Timestamp option ~~in order.~~

5.3.4 Processing rules for receivers

The processing of the Nonce and Timestamp options depends on whether a packet is a solicited-~~for~~ advertisement ~~or not.~~ A system may implement the distinction in various means. Section 5.3.4.1 defines

the processing rules for solicited-~~for~~ advertisements. Section 5.3.4.2 defines the processing rules for all other messages.

In addition, the following rules apply in ~~any~~ ~~case:~~

- o Messages received without the Timestamp option MUST be silently discarded.

- o Solicitation messages received without the Nonce option MUST be silently discarded.

- o Advertisements sent to a unicast destination address without a Nonce option MUST be silently discarded.

- o An implementation may utilize some mechanism such as a timestamp cache to strengthen resistance to replay attacks. When there is a very large number of nodes on the same link, or when a cache filling attack is in progress, it is possible that the cache holding the most recent timestamp per sender becomes full. In this case the node MUST remove some entries from the cache or refuse some new requested entries. The specific policy as to which entries are preferred over the others is left as an implementation decision. However, typical policies may prefer existing entries over new ones, CGAs with a large Sec value over smaller Sec values, and so on. The issue is briefly discussed in Appendix C.

- o The receiver MUST be prepared to receive the Timestamp and Nonce options in any order, as per RFC 2461 [7] Section 9.

5.3.4.1 Processing solicited-~~for~~ advertisements

The receiver MUST verify that it has recently sent a matching solicitation, and that the received advertisement contains a copy of the Nonce sent in the solicitation.

If the message contains a Nonce option, but the Nonce value is not recognized, the message MUST be silently discarded.

Otherwise, if the message does not contain a Nonce option, it MAY be considered as a ~~non-solicited-for~~ advertisement, and processed according to Section 5.3.4.2.

If the message is accepted, the receiver SHOULD store the receive time of the message and the time stamp time in the message, as specified in Section 5.3.4.2

5.3.4.2 Processing all other messages

Receivers SHOULD be configured with an allowed timestamp Delta value, a "fuzz factor" for comparisons, and an allowed clock drift parameter. The recommended default value for the allowed Delta is 3,600 seconds (1 hour), for fuzz factor 1 second, and for clock drift 1% (0.01).

How about moving these values into the constants section at the end of the paper and assigning symbolic constants?

To facilitate timestamp checking, each node SHOULD store the following information for each peer:

The receive time of the last ~~received and accepted~~ SEND message. This is called RDlast.

The time stamp in the last ~~received, accepted~~ SEND message. This is called TSlast.

Receivers SHOULD then check the Timestamp field as follows:

o When a message is received from a new peer, i.e., one that is not stored in the cache, the received timestamp, TSnew, is checked and the packet is accepted if the timestamp is recent enough with respect to the reception time of the packet, RDnew:

    -Delta < (RDnew - TSnew) < +Delta

  The RDnew and TSnew values SHOULD be stored into the cache as RDlast and TSlast.

o If the timestamp is NOT within the boundaries but the message is a Neighbor Solicitation message that should be ~~responded to~~ by the receiver, the receiver MAY respond to the message. However, if it does respond to the message, it MUST NOT create a ~~neighbor cache~~ entry. This allows nodes that have large difference in their clocks to still communicate with each other, by exchanging NS/NA pairs.

o When a message is received from a known peer, i.e., one that already has an entry in the cache, the time stamp is checked against the previously received SEND message:

    TSnew + fuzz > TSlast + (RDnew - RDlast) x (1 - drift) - fuzz

o If TSnew < TSlast, which is possible if packets arrive rapidly and out of order, TSlast MUST NOT be updated, i.e., the stored TSlast for a given node MUST NOT ever decrease. Otherwise TSlast SHOULD be updated. ~~Independent on whether TSlast is updated or not, RDlast is updated in any case.~~

6. Authorization Delegation Discovery

~~Several protocols (NDP included) allow~~ a node to automatically configure itself based on information ~~it learns~~ shortly after connecting to a new link. It is particularly easy to configure "rogue" routers on an unsecured link, and it is particularly difficult for a node to distinguish between valid and invalid sources of information, when the node needs this information before being able to communicate with nodes outside ~~of~~ the link.

Since the newly-connected node cannot communicate off-link, it cannot be responsible for searching information to help ~~validating~~ the router(s); however, given a chain of appropriately signed certificates, it can check someone else's search results and conclude that a particular message comes from an authorized source. In the typical case, a ~~router, which is already~~ connected ~~to~~ beyond the link, can (if necessary) communicate with off-link nodes and construct such a certificate chain.

The Secure Neighbor Discovery Protocol mandates a certificate format and introduces two new ICMPv6 messages that are used between hosts and routers to allow the host to learn a certificate chain with the assistance of the router.

6.1 Certificate Format

The certificate chain of a router terminates in a Router Authorization Certificate that authorizes a specific IPv6 node to act as a router. Because authorization chains are not a common practice in the Internet at the time this specification is being written, the chain MUST consist of standard Public Key Certificates (PKC, in the sense of [18]). The certificate chain MUST start from the identity of a trust anchor that is shared by the host and the router. This allows the host to anchor trust for the router's public key in the trust anchor. Note that there MAY be multiple certificates issued by a single trust anchor.

6.1.1 Router Authorization Certificate Profile

Router Authorization Certificates are X.509v3 certificates, as defined in RFC 3280 [10], and MUST contain at least one instance of the X.509 extension for IP addresses, as defined in [11]. The parent certificates in the certificate chain MUST contain one or more X.509 IP address extensions, back up to a trusted party (such as the user's ISP) that configured the original IP address space block for the router in question, or delegated the right to do so ~~for someone.~~ The certificates for intermediate delegating authorities MUST contain X.509 IP address extension(s) for subdelegations. The router's

certificate is signed by the delegating authority for the prefixes the router is authorized to to advertise.

The X.509 IP address extension MUST contain at least one addressesOrRanges element. This element MUST contain an addressPrefix element with an IPv6 address prefix for a prefix the router or the intermediate entity is authorized to route. If the entity is allowed to route any prefix, the used IPv6 address prefix is the null prefix, 0/0. The addressFamily element of the containing IPAddrBlocks sequence element MUST contain the IPv6 Address Family Identifier (0002), as specified in [11] for IPv6 prefixes. Instead of an addressPrefix element, the addressesOrRange element MAY contain an addressRange element for a range of prefixes, if more than one prefix is authorized. The X.509 IP address extension MAY contain additional IPv6 prefixes, expressed either as an addressPrefix or an addressRange.

A SEND node receiving a Router Authorization Certificate MUST first check whether the certificate's signature was generated by the delegating authority. Then the client MUST check whether all the addressPrefix or addressRange entries in the router's certificate are contained within the address ranges in the delegating authority's certificate, and whether the addressPrefix entries match any addressPrefix entries in the delegating authority's certificate. If an addressPrefix or addressRange is not contained within the delegating authority's prefixes or ranges, the client MAY attempt to take an intersection of the ranges/prefixes, and use that intersection. If the addressPrefix in the certificate is the null prefix, 0/0, such an intersection SHOULD be used. (In that case the intersection is the parent prefix or range.) If the resulting intersection is empty, the client MUST NOT accept the certificate.

The above check SHOULD be done for all certificates in the chain. If any of the checks fail, the client MUST NOT accept the certificate. The client also needs to perform validation of advertised prefixes as discussed in Section 7.3.

Care should be taken if the certificates used in SEND are re-used to provide authorization in other circumstances, for example with routing gateway protocols. It is necessary to ensure that the authorization information is appropriate for all applications. SEND certificates may authorize a larger set of prefixes than the router is really authorized to advertise on a given interface. For instance, SEND allows the use of the null prefix. This prefix might cause verification or routing problems in other applications. It is RECOMMENDED that SEND certificates containing the null prefix are only used for SEND.

Since it is possible that some PKI certificates used with SEND do not immediately contain the X.509 IP address extension element, an implementation MAY contain facilities that allow the prefix and range checks to be relaxed. However, any such configuration options SHOULD be off by default. That is, the system SHOULD have a default configuration that requires rigorous prefix and range checks.

The following is an example of a certificate chain. Suppose that ispgroup.com is the trust anchor. The host has this ~~certificate for it:~~

```
Certificate 1:
  Issuer: isp_group.com
  Validity: Jan 1, 2004 through Dec 31, 2004
  Subject: isp_group.com
  Extensions:
    IP address delegation extension:
       Prefixes: P1, ..., Pk
    ... possibly other extensions ...
  ... other certificate parameters ...
```

When the host attaches ~~then~~ to a ~~link~~ served by router_x.isp_foo.com, it receives the following certificate chain:

```
Certificate 2:
  Issuer: isp_group.com
```

```
          Validity: Jan 1, 2004 through Dec 31, 2004
          Subject: isp_foo.com
          Extensions:
            IP address delegation extension:
              Prefixes: Q1, ..., Qk
            ... possibly other extensions ...
          ... other certificate parameters ...

        Certificate 3:
          Issuer: isp_foo.com
          Validity: Jan 1, 2004 through Dec 31, 2004
          Subject: router_x.isp_foo.com
          Extensions:
            IP address delegation extension:
              Prefixes R1, ..., Rk
            ... possibly other extensions ...
          ... other certificate parameters ...
```

When processing the three certificates, the usual RFC 3280 certificate path validation is performed, for instance by checking for revoked certificates. In addition, the IP addresses in the delegation extension must be ~~subsumed by~~ the IP addresses in the

delegation extension in the issuer's certificate. So in this example, R1, ..., Rs must be ~~subsumed by~~ Q1,...,Qr, and Q1,...,Qr must be ~~subsumed by~~ P1,...,Pk. If the certificate chain is valid, then router_foo.isp_foo_example.com is authorized to route the prefixes R1,...,Rs.

6.2 Certificate Transport

The Delegation Chain Solicitation (DCS) message is sent by a host when it wishes to request a certificate chain between a router and ~~the~~ one of the host's trust anchors. The Delegation Chain Advertisement (DCA) message is sent ~~as an answer to~~ the DCS message. These messages are separate from the rest of Neighbor and Router Discovery, in order to reduce the effect of the potentially voluminous certificate chain information on other messages.

The Authorization Delegation Discovery (ADD) process does not exclude other forms of discovering certificate chains. For instance, during fast movements mobile nodes may learn information - including the certificate chains - of the next router from a previous router.

Where hosts themselves are certified by a trust anchor, these messages MAY also optionally be used between hosts to acquire the peer's certificate chain. However, the details of such usage are ~~left for future specification.~~

6.2.1 Delegation Chain Solicitation Message Format

Hosts send Delegation Chain Solicitations in order to prompt routers to generate Delegation Chain Advertisements.

```
     0                   1                   2                   3
```

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Type      |     Code      |          Checksum             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |          Identifier           |          Reserved             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   Options ...
   +-+-+-+-+-+-+-+-+-+-+-+-+-
```

IP Fields:

Source Address

A link-local unicast address assigned to the sending interface, or the unspecified address if no address is assigned to the sending interface.

^L

Destination Address

Typically the All-Routers multicast address, the Solicited-Node multicast address, or the address of the host's default router.

Hop Limit

255

ICMP Fields:

Type

TBD <To be assigned by IANA> for Delegation Chain Solicitation.

Code

0

Checksum

The ICMP checksum [9].

Identifier

A 16-bit unsigned integer field, acting as an identifier to help matching advertisements to solicitations. The Identifier field MUST NOT be zero, and its value SHOULD be randomly generated. ~~(~~This randomness does not need to be cryptographically hard, though ~~Its~~ purpose is to avoid ~~collisions.)~~

Reserved

An unused field. It MUST be initialized to zero by the sender and MUST be ignored by the receiver.

Valid Options:

Trust Anchor

One or more trust anchors that the client is willing to accept. The first (or only) Trust Anchor option MUST contain a DER Encoded X.501 Name; see Section 6.2.3. If there is more than one Trust Anchor option, the options past the first one may contain any types of ~~Trust Anchors~~.

Future versions of this protocol may define new option types. Receivers MUST silently ignore any options they do not recognize and continue processing the message. All included options MUST have a length that is greater than zero.

ICMP length (derived from the IP length) MUST be 8 or more octets.

6.2.2 Delegation Chain Advertisement Message Format

Routers send out Delegation Chain Advertisement messages in response to a Delegation Chain Solicitation.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Code      |          Checksum             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identifier            |          Component            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Reserved                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Options ...
+-+-+-+-+-+-+-+-+-+-+-+-
```

IP Fields:

Source Address

A link-local unicast address assigned to the interface from which this message is sent. Note that routers may use multiple addresses, and therefore this address ~~no~~t sufficient for the unique identification of routers.

Destination Address

Either the Solicited-Node multicast address of the receiver or the link-scoped All-Nodes multicast address.

Hop Limit

255

ICMP Fields:

Type

TBD <To be assigned by IANA> for Delegation Chain

Advertisement.

Code

0

Checksum

The ICMP checksum [9].

Identifier

A 16-bit unsigned integer field, acting as an identifier to help matching advertisements to solicitations. The Identifier field MUST be zero for advertisements sent to the All-Nodes multicast address and MUST NOT be zero for others.

Component

A 16-bit unsigned integer field, used for informing the receiver which certificate is being sent, and how many are still left to be sent in the whole chain.

A single advertisement MUST be broken into separately sent components if there is more than one Certificate option, in order to avoid excessive fragmentation at the IP layer. Unlike the fragmentation at the IP layer, individual components of an advertisement may be stored and used before all the components have arrived; this makes them slightly more reliable and less prone to Denial-of-Service attacks.

The first message in a N-component advertisement has the Component field set to N-1, the second set to N-2, and so on. Zero indicates that there are no more components coming in this advertisement.

The components MUST be ordered so that the trust anchor ~~end of the chain~~ is the one sent first. Each certificate sent after ~~it~~ can be verified with the previously sent certificates. The certificate of the sender comes last.

Reserved

An unused field. It MUST be initialized to zero by the sender and MUST be ignored by the receiver.

Valid Options:

Certificate

One certificate is provided in each Certificate option, to establish a (part of a) certificate chain to a trust anchor.

The certificate of the trust anchor itself SHOULD NOT be included.

Trust Anchor

Zero or more Trust Anchor options may be included to help receivers decide which advertisements are useful for them. If present, these options MUST appear in the first component of a multi-component advertisement.

Future versions of this protocol may define new option types. Receivers MUST silently ignore any options they do not recognize and continue processing the message. All included options MUST have a length that is greater than zero.

ICMP length (derived from the IP length) MUST be 8 or more octets.

6.2.3 Trust Anchor Option

The format of the Trust Anchor option is described in the following:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |  Name Type    |  Pad  Length  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Name ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

~~Where the fields are as follows:~~

Type

TBD <To be assigned by IANA> for Trust Anchor.

Length

The length of the option, (including the Type, Length, Name Type, Name Length, and Name fields,) in units of 8 octets.

Name Type

The type of the name included in the Name field. This specification defines ~~only~~ one legal value for this field:

```
        1         DER Encoded X.501 Name
        2         FQDN
```

Pad Length

The number of padding octets beyond the end of the Name field but within the length specified by the Length field. Padding octets MUST be set to zero by senders and ignored by receivers.

Name

When the Name Type field is set to 1, the Name field contains a DER encoded X.501 certificate Name, represented and encoded exactly as in the matching X.509v3 trust anchor certificate.

When the Name Type field is set to 2, the Name field contains a Fully Qualified Domain Name of the trust anchor, for example, "trustanchor.example.com". The name is stored as a string, in the "preferred name syntax" DNS format, as specified in RFC 1034 [1] Section 3.5. Additionally, the restrictions discussed in RFC 3280 [10] Section 4.2.1.7 apply.

All systems MUST ~~implement~~ support the DER Encoded X.501 Name. Implementations MAY support the FQDN name type.

6.2.4 Certificate Option

The format of the certificate option is described in the following:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |  Cert Type    |  Pad Length   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Certificate ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

~~Where the fields are as follows:~~

Type

TBD <To be assigned by IANA> for Certificate.

^L

Length

The length of the option, (including the Type, Length, Cert Type, Pad Length, and Certificate fields,) in units of 8 octets.

Cert Type

The type of the certificate included in the Certificate field. This specification defines only one legal value for this field:

1 X.509v3 Certificate, as specified below

Pad Length

The number of padding octets beyond the end of the Certificate field but within the length specified by the Length field. Padding octets MUST be set to zero by senders and ignored by receivers.

Certificate

When the Cert Type field is set to 1, the Certificate field contains an X.509v3 certificate [10], as described in Section 6.1.1.

6.2.5 Processing Rules for Routers

Routers SHOULD possess a key pair and a certificate from at least one certificate authority.

A router MUST silently discard any received Delegation Chain Solicitation messages that do not satisfy all of the following validity checks:

- o All requirements listed in Section 6.2.1 are fulfilled.
- o If the message includes an IP Authentication Header, the message authenticates correctly.

The contents of the Reserved field, and of any unrecognized options, MUST be ignored. Future, backward-compatible changes to the protocol may specify the contents of the Reserved field or add new options; backward-incompatible changes may use different Code values. The contents of any defined options that are not specified to be used with Router Solicitation messages MUST be ignored and the packet processed in the normal manner. The only defined option that may appear is the Trust Anchor option. A solicitation that passes the

validity checks is called a "valid solicitation".

Routers SHOULD send advertisements in response to valid solicitations received on an advertising interface. If the source address in the solicitation was the unspecified address, the router MUST send the response to the link-scoped All-Nodes multicast address. If the source address was a unicast address, the router MUST send the response to the Solicited-Node multicast address corresponding to the source address. Routers SHOULD NOT send Delegation Chain Advertisements more than MAX_DCA_RATE times within a second. When there are more solicitations ~~than this,~~ the router SHOULD send the response to the All-Nodes multicast address regardless of the source address that appeared in the solicitation.

In an advertisement, the router SHOULD include suitable Certificate options so that a delegation chain to the solicited trust anchor can be established. The anchor is identified by the Trust Anchor option. If the Trust Anchor option is represented as a DER Encoded X.501 Name, then the Name must be equal to the Subject field in the anchor's certificate. If the Trust Anchor option is represented as an FQDN, the FQDN must be equal to an FQDN in the subjectAltName field of the anchor's certificate. The router SHOULD include the Trust Anchor option(s) in the advertisement for which the delegation chain was found.

If the router is unable to find a chain to the requested anchor, it SHOULD send an advertisement without any certificates. In this case the router SHOULD include the Trust Anchor options which were solicited.

6.2.6 Processing Rules for Hosts

Hosts SHOULD possess the public key and trust anchor name of at least one certificate authority, they SHOULD possess their own key pair, and they MAY posses a certificate from the above mentioned certificate authority.

A host MUST silently discard any received Delegation Chain Advertisement messages that do not satisfy all of the following validity checks:

- o All requirements listed in Section 6.2.2 are fulfilled.

- o If the message includes an IP Authentication Header, the message authenticates correctly.

The contents of the Reserved field, and of any unrecognized options, MUST be ignored. Future, backward-compatible changes to the protocol

may specify the contents of the Reserved field or add new options; backward-incompatible changes may use different Code values. The contents of any defined options that are not specified to be used with Delegation Chain Advertisement messages MUST be ignored and the packet processed in the normal manner. The only defined options that may appear are the Certificate and Trust Anchor options. An advertisement that passes the validity checks is called a "valid advertisement".

Hosts SHOULD store certificate chains retrieved in Delegation Chain Discovery messages if they start from an anchor trusted by the host. The certificate chains SHOULD be verified, as defined in Section 6.1, before storing them. Routers MUST send the certificates one by one, starting from the trust anchor end of the chain. Except for temporary purposes to allow for message loss and reordering, hosts SHOULD NOT store certificates received in a Delegation Chain Advertisement unless they contain a certificate which can be immediately verified either to the trust anchor or to a certificate which has been verified earlier.

Note that ~~it may be useful to cach~~ this information and implied verification results for use over multiple attachments to the network.

The host has a need to retrieve a delegation chain when a Router Advertisement has been received with a public key that is not stored in the hosts' cache of certificates, or there is no authorization delegation chain to the host's trust anchor. In these situations, the host MAY transmit up to MAX_DCS_MESSAGES Delegation Chain Solicitation messages, each separated by at least DCS_INTERVAL seconds.

Delegation Chain Solicitations SHOULD NOT be sent if the host has a currently valid certificate chain from a reachable router to a trust anchor.

When soliciting certificates for a router, a host MUST send Delegation Chain Solicitations either to the All-Routers multicast address, if it has not selected a default router yet, or to the default router's IP address, if it has already been selected.

If two hosts want to establish trust with the DCS and DCA messages, the DCS message SHOULD be sent to the Solicited-Node multicast address of the receiver. The advertisements SHOULD be sent as specified above for routers. However, the exact details are ~~left for a future specification.~~

When processing possible advertisements sent as responses to a

solicitation, the host MAY prefer to process first those advertisements with the same Identifier field value as in the solicitation. This makes Denial-of-Service attacks against the mechanism harder (see Section 9.3).

^L

7. Addressing

7.1 ~~CGA Addresses~~

Nodes that use stateless address autoconfiguration, SHOULD generate a new CGA as specified in Section 4 of [12] for each new autoconfiguration run. The nodes MAY continue to use the same public key and modifier, and start the process from Step 4.

By default, a SEND-enabled node SHOULD use only CGAs as its own addresses. Other types of addresses MAY be used in testing, diagnostics or other purposes. However, this document does not describe how to choose between different types of addresses for different communications. A dynamic selection can be provided by an API, such as the one defined in [22].

7.2 Redirect Addresses

If the Target Address and Destination Address fields in the ICMP Redirect message are equal, then this message is used to inform hosts that a destination is in fact a neighbor. In this case the receiver MUST verify that the given address falls within the range defined by the router's certificate. Redirect messages failing this check MUST be silently discarded.

Note that RFC 2461 rules prevent a ~~bogus router from sending a Redirect message when the host is not using the bogus router as a default router.~~

7.3 Advertised Prefixes

The router's certificate defines the address range(s) that it is allowed to advertise. Upon processing a Prefix Information option within a Router Advertisement, nodes SHOULD verify that the prefix specified in this option falls within the range defined by the certificate, if the certificate contains a prefix extension. Options failing this check MUST be silently discarded.

Nodes SHOULD use one of the certified prefixes for stateless autoconfiguration. If none of the advertised prefixes match, then either there is a configuration problem or the advertising router is an attacker, and the host MUST use a different advertising router as its default router (if available). If the node is performing stateful autoconfiguration, it SHOULD check the address provided by the DHCP server against the certified prefixes and MUST NOT use the address if the prefix is not certified.

In any case, the user should inform the network operator upon

^L

receiving an address or prefix outside the certified range, since this is either a misconfiguration or an attack.

If the network operator wants to constrain which routers are allowed to route particular prefixes, routers SHOULD be configured with certificates having prefixes listed in the prefix extension. Routers so configured MUST advertise the prefixes which they are certified to route, or a subset thereof.

Network operators that do not want to constrain routers this way SHOULD configure routers with certificates containing either the null prefix or no prefix extension at all.

7.4 Limitations

This specification does not address the protection of NDP packets for nodes that are configured with a static address (e.g., PREFIX::1). Future certificate ~~chain-based~~ authorization specifications are needed for such nodes.

It is outside the scope of this specification to describe the use of trust anchor authorization between nodes with dynamically changing addresses. Such dynamically changing addresses may be the result of stateful or stateless address autoconfiguration, or through the use of RFC 3041 [17] addresses. If the CGA method is not used, nodes would be required to exchange certificate chains that terminate in a certificate authorizing a node to use an IP address having a particular interface identifier. This specification does not specify the format of such certificates, since there are currently a few cases where such certificates are required by the link layer and it is up to the link layer to provide certification for the interface identifier. This may be the subject of a future specification. It is also outside the scope of this specification to describe how stateful address autoconfiguration works with the CGA method.

The Target Address in Neighbor Advertisement is required to be equal to the source address of the packet, except in the case of proxy Neighbor Discovery. Proxy Neighbor Discovery is not supported by this specification; ~~it is planned to be specified in a future document.~~

^L

8. Transition Issues

During the transition to secure links or as a policy consideration, network operators may want to run a particular link with a mixture of secure and insecure nodes. Nodes that support SEND SHOULD support the use of SEND and the legacy NDP at the same time.

In a mixed environment, SEND nodes receive both secure and insecure messages but give priority to "secured" ones. Here, the "secured" messages are ones that contain a valid signature option, as specified above, and "insecure" messages are ones that contain no signature option.

SEND nodes send only secured messages. Legacy Neighbor Discovery nodes will obviously send only insecure messages. Per RFC 2461 [7], such nodes will ignore the unknown options and will treat secured messages in the same way as they treat insecure ones. Secured and insecure nodes share the same network resources, such as prefixes and address spaces.

In a mixed environment SEND nodes follow the protocols defined in RFC 2461 and RFC 2462 with the following exceptions:

- o All solicitations sent by SEND nodes MUST be secured.

- o Unsolicited advertisements sent by a SEND node MUST be secured.

- o A SEND node MUST send a secured advertisement in response to a secured solicitation. Advertisements sent in response to an insecure solicitation MUST be secured as well, but MUST NOT contain the Nonce option.

- o A SEND node that uses the CGA authorization method for protecting Neighbor Solicitations SHOULD perform Duplicate Address Detection as follows. If Duplicate Address Detection indicates the tentative address is already in use, generate a new tentative CGA ~~address.~~ If after 3 consecutive attempts no non-unique address was generated, log a system error and give up attempting to generate an address for that interface.

  When performing Duplicate Address Detection for the first tentative address, accept both secured and insecure Neighbor

Advertisements and Solicitations received as response to the Neighbor Solicitations. When performing Duplicate Address Detection for the second or third tentative address, ignore insecure Neighbor Advertisements and Solicitations.

o The node SHOULD have a configuration option that causes it to

^L

ignore insecure advertisements even when performing Duplicate Address Detection for the first tentative address. This configuration option SHOULD be disabled by default. This is a recovery mechanism, in case attacks against the first address become common.

o The Neighbor Cache, Prefix List and Default Router list entries MUST have a secured/insecure flag that indicates whether the message that caused the creation or last update of the entry was secured or insecure. Received insecure messages MUST NOT cause changes to existing secured entries in the Neighbor Cache, Prefix List or Default Router List. Received secured messages cause an update of the matching entries and flagging of them as secured.

o The conceptual sending algorithm is modified so that an insecure router is selected only if there is no reachable SEND router for the prefix. That is, the algorithm for selecting a default router favors reachable SEND routers over reachable non-SEND ones.

o A SEND node SHOULD have a configuration option that causes it to ignore all insecure Neighbor Solicitation and Advertisement, Router Solicitation and Advertisement, and Redirect messages. This can be used to enforce SEND-only networks.

9. Security Considerations

9.1 Threats to the Local Link Not Covered by SEND

SEND does not provide confidentiality for NDP communications.

SEND does not compensate for an insecure link layer. For instance, there is no assurance that payload packets actually come from the same peer that the NDP was run against.

There may be no cryptographic binding in SEND between the link layer frame address and the IPv6 address. On an insecure link layer that allows nodes to spoof the link layer address of other nodes, an attacker could disrupt IP service by sending out a Neighbor Advertisement having the source address on the link layer frame of a victim, a valid CGA address and a valid signature corresponding to itself, and a Target Link-layer Address extension corresponding to the victim. The attacker could then proceed to cause a traffic stream to bombard the victim in a DoS attack. This attack cannot be prevented just by securing the link layer.

Even on a secure link layer, SEND does not require that the addresses on the link layer and Neighbor Advertisements correspond to each other. However, it is RECOMMENDED that such checks be performed where this is possible on the given link layer technology.

Prior to participating in Neighbor Discovery and Duplicate Address Detection, nodes must subscribe to the link-scoped All-Nodes Multicast Group and the Solicited-Node Multicast Group for the address that they are claiming for their addresses; RFC 2461 [7]. Subscribing to a multicast group requires that the nodes use MLD [16]. MLD contains no provision for security. An attacker could send an MLD Done message to unsubscribe a victim from the Solicited-Node Multicast address. However, the victim should be able to detect such an attack because the router sends a Multicast-Address-Specific Query to determine whether any listeners are still on the address, at which point the victim can respond to avoid being dropped from the group. This technique will work if the router on the link has not been compromised. Other attacks using MLD are possible, but they primarily lead to extraneous (but not overwhelming) traffic.

9.2 How SEND Counters Threats to NDP

The SEND protocol is designed to counter the threats to NDP, as outlined in [25]. The following subsections contain a regression of the SEND protocol against the threats, to illustrate what aspects of the protocol counter each threat.

9.2.1 Neighbor Solicitation/Advertisement Spoofing

This threat is defined in Section 4.1.1 of [25]. The threat is that a spoofed message may cause a false entry in a node's Neighbor Cache. There are two cases:

1. Entries made as a side effect of a Neighbor Solicitation or Router Solicitation. A router receiving a Router Solicitation with a firm IPv6 source address and a Target Link-Layer Address extension inserts an entry for the IPv6 address into its Neighbor Cache. Also, a node performing Duplicate Address Detection (DAD) that receives a Neighbor Solicitation for the same address regards the situation as a collision and ceases to solicit for the address.

   In either case, SEND counters these treats by requiring the Signature and CGA options to be present in such solicitations.

   SEND nodes can send Router Solicitation messages with a CGA source address and a CGA option, which the router can verify, so the Neighbor Cache binding is correct. If a SEND node must send a Router Solicitation with the unspecified address, the router will not update its Neighbor Cache, as per RFC 2461.

2. Entries made as a result of a Neighbor Advertisement message. SEND counters this threat by requiring the Signature and CGA options to be present in these advertisements.

See also Section 9.2.5, below, for discussion about replay protection and timestamps.

9.2.2 Neighbor Unreachability Detection Failure

This attack is described in Section 4.1.2 of [25]. SEND counters this attack by requiring a node responding to Neighbor Solicitations sent as NUD probes to include a Signature option and proof of authorization to use the interface identifier in the address being probed. If these prerequisites are not met, the node performing NUD discards the responses.

9.2.3 Duplicate Address Detection DoS Attack

This attack is described in Section 4.1.3 of [25]. SEND counters this attack by requiring the Neighbor Advertisements sent as responses to DAD to include a Signature option and proof of authorization to use the interface identifier in the address being tested. If these prerequisites are not met, the node performing DAD discards the responses.

When a SEND node is used on a link that also connects to non-SEND nodes, the SEND node ignores any insecure Neighbor Solicitations or Advertisements that may be sent by the non-SEND nodes. This protects the SEND node from DAD DoS attacks by non-SEND nodes or attackers simulating to non-SEND nodes, at the cost of a potential address collision between a SEND node and non-SEND node. The probability and effects of such an address collision are discussed in [12].

9.2.4 Router Solicitation and Advertisement Attacks

These attacks are described in Sections 4.2.1, 4.2.4, 4.2.5, 4.2.6, and 4.2.7 of [25]. SEND counters these attacks by requiring Router Advertisements to contain a Signature option, and that the signature is calculated using the public key of a node that can prove its authorization to route the subnet prefixes contained in any Prefix Information Options. The router proves its authorization by showing a certificate containing the specific prefix or the indication that the router is allowed to route any prefix. A Router Advertisement without these protections is discarded.

SEND does not protect against brute force attacks on the router, such as DoS attacks, or compromise of the router, as described in Sections 4.4.2 and 4.4.3 of [25].

9.2.5 Replay Attacks

This attack is described in Section 4.3.1 of [25]. SEND protects against attacks in Router Solicitation/Router Advertisement and Neighbor Solicitation/Neighbor Advertisement transactions by including a Nonce option in the solicitation and requiring the advertisement to include a matching option. Together with the signatures this forms a challenge-response protocol. SEND protects against attacks from unsolicited messages such as Neighbor Advertisements, Router Advertisements, and Redirects by including a Timestamp option. A window of vulnerability for replay attacks exists until the timestamp expires.

When timestamps are used, SEND nodes are protected against replay attacks as long as they cache the state created by the message containing the timestamp. The cached state allows the node to protect itself against replayed messages. However, once the node flushes the state for whatever reason, an attacker can re-create the state by replaying an old message while the timestamp is still valid. Since most SEND nodes are likely to use fairly coarse grained timestamps, as explained in Section 5.3.1, this may affect some nodes.

9.2.6 Neighbor Discovery DoS Attack

This attack is described in Section 4.3.2 of [25]. In this attack, the attacker bombards the router with packets for fictitious addresses on the link, causing the router to busy itself with performing Neighbor Solicitations for addresses that do not exist. SEND does not address this threat because it can be addressed by techniques such as rate limiting Neighbor Solicitations, restricting the amount of state reserved for unresolved solicitations, and clever cache management. These are all techniques involved in implementing

Neighbor Discovery on the router.

9.3 Attacks against SEND Itself

The CGAs have a 59-bit hash value. The security of the CGA mechanism has been discussed in [12].

Some Denial-of-Service attacks against NDP and SEND itself remain. For instance, an attacker may try to produce a very high number of packets that a victim host or router has to verify using asymmetric methods. While safeguards are required to prevent an excessive use of resources, this can still render SEND non-operational.

When CGA protection is used, SEND deals with the DoS attacks using the verification process described in Section 5.2.2. In this process, a simple hash verification of the CGA property of the address is performed before performing the more expensive signature verification.

When trust anchors and certificates are used for address validation in SEND, the defenses are not quite as effective. Implementations SHOULD track the resources devoted to the processing of packets received with the Signature option, and start selectively discarding packets if too many resources are spent. Implementations MAY also first discard packets that are not protected with CGA.

The Authorization Delegation Discovery process may also be vulnerable to Denial-of-Service attacks. An attack may target a router by requesting a large number of delegation chains to be discovered for different trust anchors. Routers SHOULD defend against such attacks by caching discovered information (including negative responses) and by limiting the number of different discovery processes they engage in.

Attackers may also target hosts by sending a large number of unnecessary certificate chains, forcing hosts to spend useless memory and verification resources for them. Hosts can defend against such attacks by limiting the amount of resources devoted to the

certificate chains and their verification. Hosts SHOULD also prioritize advertisements that sent as a response to their solicitations above unsolicited advertisements.

^L

10. Protocol Constants

Host constants:

MAX_DCS_MESSAGES	3 transmissions
DCS_INTERVAL	4 seconds

Router constants:

MAX_DCA_RATE	10 times per second

^L

11. IANA Considerations

This document defines two new ICMP message types, used in Authorization Delegation Discovery. These messages must be assigned ICMPv6 type numbers from the informational message range:

- o The Delegation Chain Solicitation message, described in Section 6.2.1.
- o The Delegation Chain Advertisement message, described in Section 6.2.2.

This document defines six new Neighbor Discovery Protocol [7] options, which must be assigned Option Type values within the option numbering space for Neighbor Discovery Protocol messages:

- o The CGA option, described in Section 5.1.
- o The Signature option, described in Section 5.2.
- o The Timestamp option, described in Section 5.3.1.
- o The Nonce option, described in Section 5.3.2.
- o The Trust Anchor option, described in Section 6.2.3.
- o The Certificate option, described in Section 6.2.4.

This document defines a new 128-bit value under the CGA Message Type [12] namespace, 0x086F CA5E 10B2 00C9 9C8C E001 6427 7C08.

This document defines a new name space for the Name Type field in the Trust Anchor option. Future values of this field can be allocated using standards action [6]. The current values for this field are:

1 DER Encoded X.501 Name

2 FQDN

Another new name space is allocated for the Cert Type field in the Certificate option. Future values of this field can be allocated using standards action [6]. The current values for this field are:

1 X.509v3 Certificate

Normative References

[1] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, November 1987.

[2] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

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

[4] Kent, S. and R. Atkinson, "IP Authentication Header", RFC 2402, November 1998.

[5] Piper, D., "The Internet IP Security Domain of Interpretation for ISAKMP", RFC 2407, November 1998.

[6] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

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

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

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

[10] Housley, R., Polk, W., Ford, W. and D. Solo, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 3280, April 2002.

[11] Lynn, C., Kent, S. and K. Seo, "X.509 Extensions for IP Addresses and AS Identifiers", draft-ietf-pkix-x509-ipaddr-as-extn-03 (work in progress), September 2003.

[12] Aura, T., "Cryptographically Generated Addresses (CGA)", draft-ietf-send-cga-03 (work in progress), December 2003.

[13] RSA Laboratories, "RSA Encryption Standard, Version 2.1", PKCS 1, November 2002.

[14] National Institute of Standards and Technology, "Secure Hash Standard", FIPS PUB 180-1, April 1995, <http://

^L

www.itl.nist.gov/fipspubs/fip180-1.htm>.

^L

Informative References

[15] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

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

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

[18] Farrell, S. and R. Housley, "An Internet Attribute Certificate Profile for Authorization", RFC 3281, April 2002.

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

[20] Arkko, J., "Effects of ICMPv6 on IKE and IPsec Policies", draft-arkko-icmpv6-ike-effects-02 (work in progress), March 2003.

[21] Arkko, J., "Manual SA Configuration for IPv6 Link Local Messages", draft-arkko-manual-icmpv6-sas-01 (work in progress), June 2002.

[22] Nordmark, E., Chakrabarti, S. and J. Laganier, "IPv6 Socket API for Address Selection", draft-chakrabarti-ipv6-addrselect-02 (work in progress), October 2003.

[23] Droms, R., "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", draft-ietf-dhc-dhcpv6-28 (work in progress), November 2002.

[24] Kent, S., "IP Encapsulating Security Payload (ESP)", draft-ietf-ipsec-esp-v3-06 (work in progress), July 2003.

[25] Nikander, P., "IPv6 Neighbor Discovery trust models and threats", draft-ietf-send-psreq-00 (work in progress), October 2002.

[26] International Organization for Standardization, "The Directory - Authentication Framework", ISO Standard X.509, 2000.

[27] Institute of Electrical and Electronics Engineers, "Local and Metropolitan Area Networks: Port-Based Network Access Control", IEEE Standard 802.1X, September 2001.

^L

Authors' Addresses

Jari Arkko
Ericsson

Jorvas 02420
Finland

EMail: jari.arkko@ericsson.com

James Kempf
DoCoMo Communications Labs USA
181 Metro Drive
San Jose, CA 94043
USA

EMail: kempf@docomolabs-usa.com

Bill Sommerfeld
Sun Microsystems
1 Network Drive UBUR02-212
Burlington, MA 01803
USA

EMail: sommerfeld@east.sun.com

Brian Zill
Microsoft

USA

EMail: bzill@microsoft.com

Pekka Nikander
Ericsson

Jorvas 02420
Finland

EMail: Pekka.Nikander@nomadiclab.com

Appendix A. Contributors

Tuomas Aura contributed the transition mechanism specification in Section 8. Jonathan Trostle contributed the certificate chain example in Section 6.1.1.

Appendix B. Acknowledgments

The authors would like to thank Tuomas Aura, Erik Nordmark, Gabriel Montenegro, Pasi Eronen, Greg Daley, Jon Wood, Julien Laganier, Francis Dupont, and Pekka Savola for interesting discussions in this problem space and feedback regarding the SEND protocol.

Appendix C. Cache Management

In this section we outline a cache management algorithm that allows a node to remain partially functional even under a cache filling DoS attack. This appendix is informational, and real implementations SHOULD use different algorithms in order to avoid he dangers of mono-cultural code.

There are at least two distinct cache related attack scenarios:

1. There are a number of nodes on a link, and someone launches a cache filling attack. The goal here is clearly make sure that the nodes can continue to communicate even if the attack is going on.

2. There is already a cache filling attack going on, and a new node arrives to the link. The goal here is to make it possible for the new node to become attached to the network, in spite of the attack.

From this point of view, it is clearly better to be very selective in how to throw out entries. Reducing the timestamp Delta value is very discriminative against those nodes that have a large clock difference, while an attacker can reduce its clock difference into arbitrarily small. Throwing out old entries just because their clock difference is large seems like a bad approach.

A reasonable idea seems to be to have a separate cache space for new entries and old entries, and under an attack more eagerly drop new cache entries than old ones. One could track traffic, and only allow those new entries that receive genuine traffic to be converted into old cache entries. While such a scheme will make attacks harder, it will not fully prevent them. For example, an attacker could send a little traffic (i.e. a ping or TCP syn) after each NS to trick the victim into promoting its cache entry to the old cache. Hence, the node may be more intelligent in keeping its cache entries, and not just have a black/white old/new boundary.

It also looks like a good idea to consider the sec parameter when forcing cache entries out, and let those entries with a larger sec a higher chance of staying in.

Intellectual Property Statement

Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.