Mobile Ad Hoc Networking Working Group
INTERNET DRAFT
14 November 2001

Charles E. Perkins
Jari T. Malinen
Ryuji Wakikawa
Nokia Research Center
Elizabeth M. Belding-Royer
Yuan Sun
University of California, Santa Barbara

IP Address Autoconfiguration for Ad Hoc Networks

draft-perkins-manet-autoconf-01.txt

Status of This Memo

Abstract

If a node lacks an IP address, it cannot yet participate in ad hoc networks as currently designed, because the connectivity in an ad hoc network is typically determined by mechanisms that depend upon using the IP address as the identifier for the nodes in the ad hoc network. In this document, a mechanism by which a node in an ad hoc network may autoconfigure an IP address which is unique throughout the connected portion of the ad hoc network is specified. Specifically, mechanisms for both IPv4 and IPv6 networks, which are isolated from Internet connectivity, are described.

Contents

1. Introduction

If a node lacks an IP address, it cannot yet participate in ad hoc networks as currently designed, because the connectivity in an ad hoc network is typically determined by mechanisms that depend upon using the IP address as the identifier for the nodes in the ad hoc network. In this document, a mechanism by which a node in an ad hoc network may autoconfigure an IP address which is unique throughout the connected portion of the ad hoc network is specified. Mechanisms for configuring both IPv4 and IPv6 addresses, as appropriate, are specified.

When a node in an ad hoc network wishes to obtain an IP address, it may be difficult or impossible to contact any address allocation agency in the network. In such cases, according to the specifications given in this document, the node attempts to select a random address (on network 169.254/16 in case of IPv4, or on prefix MANET_INITIAL_PREFIX in case of IPv6). This is analogous to the way that Autonet allocations are done, and as is proposed in the zeroconf working group [3].

2. Applicability Statement

The mechanisms described in this document do not guarantee uniqueness in disconnected networks. If a network is disconnected, the process for Duplicate Address Detection (DAD) would need to be performed again when the network partition heals. This document does not specify any method for detecting when the network partition heals, nor any procedure by which such detection would cause new attempts at DAD. Any such specification would have to ensure that network healing is not accompanied by a broadcast storm of DAD messages.

The mechanisms designed in this document are designed to work independently of the protocols (i.e., routing or MAC) utilized within the protocol stack.

3. Terminology

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [1]. This section defines other terminology used with AODV that is not already defined in [2].

Duplicate Address Detection (DAD)

The process by which a node, which lacks an IP address, determines whether a candidate address it has selected is available. A node already equipped with an IP address participates in DAD in order to protect its IP address from being accidentally misappropriated for use by another node.

Address Discovery

The process by which a node in an ad hoc network discovers whether an address is already claimed within an ad hoc network.

Address Request (AREQ)

The message used during address discovery to request the address.

Address Reply (AREP)

The message used during address discovery to indicate the requested address is already utilized.

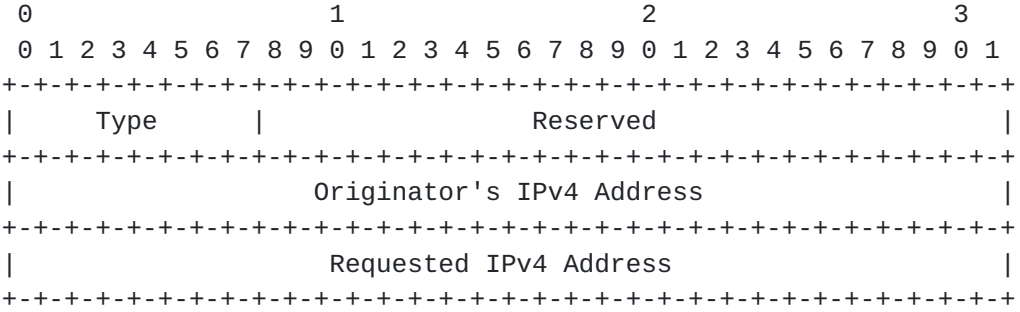
4. Overview

This protocol specifies how an IPv4 or IPv6 Manet node autoconfigures itself an address and executes a protocol exchange to check for uniqueness of this address within the reachable Manet. The protocol specifies address ranges for IPv4, and IPv6 from which to locally select the address and a message exchange for the uniqueness test of the selected address.

A Manet node performing the autoconfiguration picks two addresses, a temporary address and the actual address to use. The former is used only once in the uniqueness check to minimize the possiblity for it to be non-unique. The uniqueness check is based on sending an Address Request (AREQ) and expecting an Address Reply (AREP) back in case the address is not unique. In case no AREP is received, the uniqueness check is passed. For IPv4, the messages are ICMP [5] packets. For IPv6, on the other hand, the AREQ is a modified Neighbor Solicitation and the AREP is a modified Neighbor Advertisement, as specified below and in the Neighbor Discovery Protocol [4, 6].

5. Packet Formats

5.1. IPv4 Address Request



The format of the Address Request message is illustrated above, and contains the following fields:

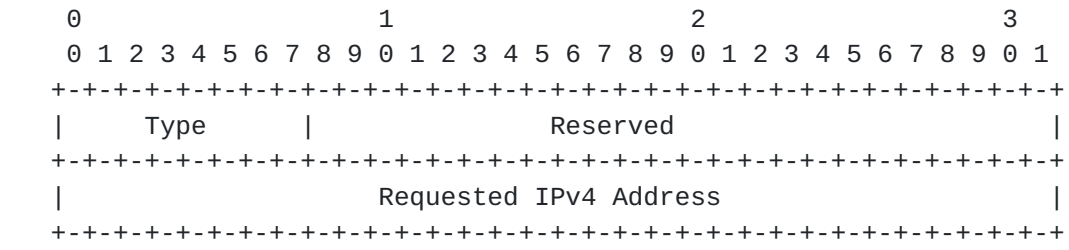
Type 1

Reserved Sent as 0; ignored upon reception.

Originator's IPv4 Address
The randomly selected temporary IP address used by the orginator of the flooded AREQ message.

Requested IPv4 Address
The randomly selected IPv4 address that is being requested by the node issuing the address request. This address may lie in the range FIRST_PERM_ADDR - 65534 from 169.254/16.

5.2. IPv4 Address Reply



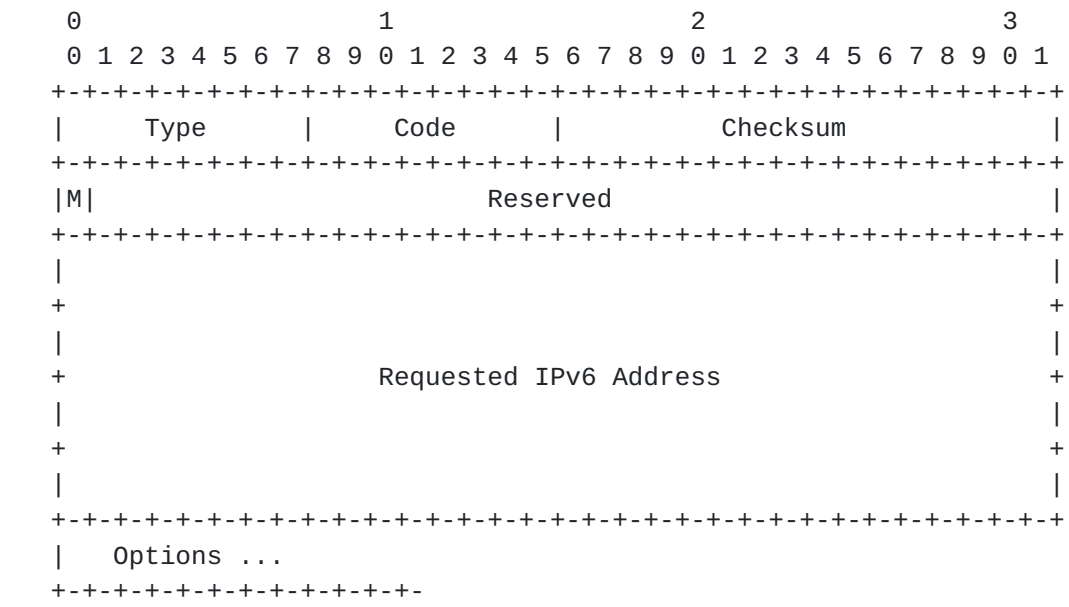
The format of the Address Reply message is illustrated above, and contains the following fields:

Type 2

Reserved Sent as 0; ignored upon reception.

Requested IPv4 Address
The randomly selected IPv4 address that is being requested by the node issuing the address request. This address may lie in the range FIRST_PERM_ADDR - 65534 from 169.254/16.

5.3. IPv6 Address Request



The format of the IPv6 Address Request message is illustrated above, it being a modified Neighbor Solicitation as defined in RFC 2461 [4] and used in RFC 2462 [6]. It contains the following modifications to NDP:

Manet Neighbor Solicitation Flag (M)
: When set, this flag indicates that this Neighbor Solicitation message should be sent over a multi-hop network. It indicates that the packet is used with a Manet Duplicate Address Detection over the Manet instead of only sending it one hop to the immediate neighbors.

Requested IPv6 Address
: The locally selected IPv6 address that is being requested by the node issuing the address request. This address may be chosen by selecting e.g., at random a host number from a suitable prefix, by default from a site-local prefix MANET_PREFIX.

IP Fields:

Hop Limit
: This field SHOULD be set to NET_DIAMETER to allow for the modified NDP packet to traverse multiple hops up to the edge of the Manet.

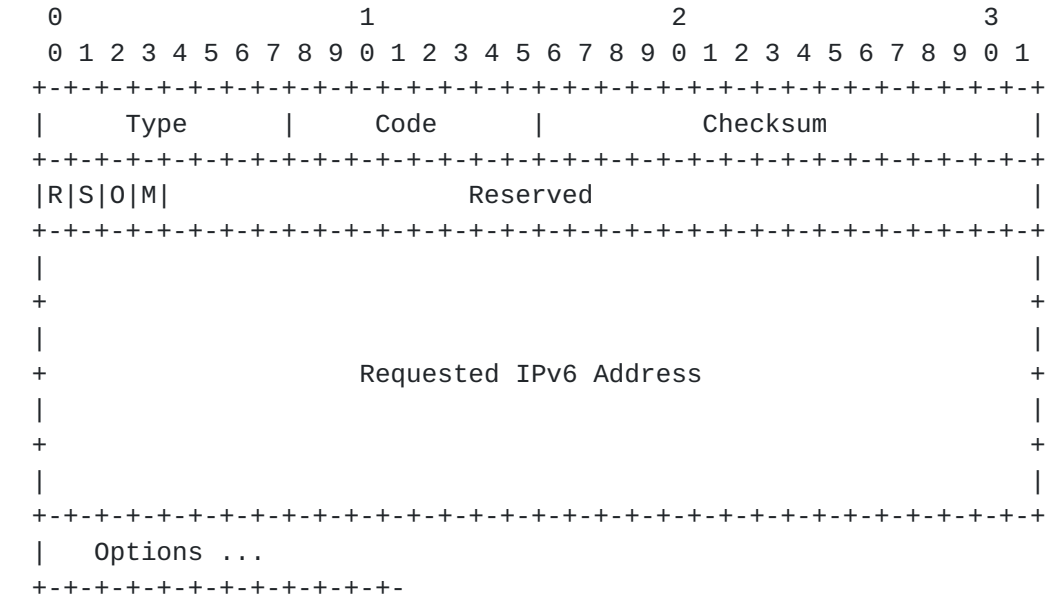
Source Address

This is a site-local, short-lived temporary address

from the site-local prefix MANET_INITIAL_PREFIX, with a host number randomly picked by the sender of this AREQ, used only once in the address uniqueness messages. This address is also used as the IPv6 UID.

Destination Address
This is the all-nodes multicast address, 0xffQQ::QQQQ.

5.4. IPv6 Address Reply



The format of the IPv6 Address Reply message is illustrated above. It is a modified Neighbor Advertisement as defined in RFC 2461 [4] and used in RFC 2462 [6]. It contains the following modifications to NDP:

Manet Neighbor Advertisement Flag (M)
When set, this flag indicates that this Neighbor Advertisement message should be sent over a multi-hop network. It indicates that the packet is used with a Manet Duplicate Address Detection over the Manet instead of only sending it one hop to the immediate neighbors.

Requested IPv6 Address
The IPv6 address that is being requested by the node issuing the address request. This address was chosen by the sender of the AREQ selecting e.g., at random a host number from a suitable prefix, by default from a site-local prefix MANET_PREFIX.

IP Fields:

Hop Limit
: This field SHOULD be set to NET_DIAMETER to allow for the modified NDP packet to traverse multiple hops up to the edge of the Manet.

Source Address
: This address is not link-local and belongs to the sender of the AREP. It SHOULD BE the same as the Requested IPv6 Address.

Destination Address
: This is the site-local, short-lived temporary address from the site-local prefix MANET_INITIAL_PREFIX, with a host number randomly picked by the sender of the AREQ, used only once in the address uniqueness messages. This address is also used as the IPv6 UID.

6. IPv4 Address Autoconfiguration

6.1. Address Request (AREQ)

Following the suggestions for DAD as with IPv6 Stateless Address Autoconfiguration [6] and zeroconf [3], the node first picks a random IP address in the range FIRST_PERM_ADDR - 65534 from 169.254/16. This is the IP address for which it will issue the address request. The node then selects a random, temporary IP address in the range 0 - LAST_TMP_ADDR. This ID will serve as a source IP address for the short period while the node performs the address discovery. Then, the node issues an Address Request (AREQ) for that randomly selected address. The packet format for the AREQ is given in section 5. The node places its randomly selected source IP address in the the Originator's IPv4 Address field. The node then broadcasts this request to its neighbors. It then sets a timer for ADDRESS_DISCOVERY milliseconds, and proceeds as described in section 6.3.

6.2. Address Request Processing

When a node receives an AREQ message, the node first notes the Requested_IP_Address and Originator's IPv4 Address. It checks its buffered list of AREQ message identifiers to determine whether it has seen this request before. If it has already seen this request, it discards the packet. Otherwise, the node enters the values of these fields into a temporary buffer. These two values serve to uniquely identify the request. If the node receives this request again as the packet is rebroadcast by its neighbors, it will note that it

has already received the request, and hence will not reprocess the packet.

Next, the node creates a reverse route entry for the node indicated by the Originator's IPv4 Address field. It enters this destination address in its route table, and uses the node from which it received the AREQ as the next hop towards the source node. The node enters a lifetime for this route of REVERSE_ROUTE_LIFETIME. In this way, if the node later receives an AREP, the node will have a current route to the source node. Hence it will be able to forward the AREP towards the source node.

The node then checks whether its own IP address matches the requested address in the AREQ. If the node's IP address does not match the requested address, it rebroadcasts the packet to its neighbors.

On the other hand, if the node has the same IP address as that in the AREQ, the node MUST reply to the packet. To do so, it creates an Address Reply (AREP) packet. The packet format for the AREP is given in section 5. It copies the Requested_IP_Address from the AREQ message, and places them in the AREP. It then unicasts this packet to the source node, as indicated by the source IP address in the IP header of the received AREQ message. The reverse route that was created by the AREQ broadcast is used to route the AREP back to the source node.

6.3. Address Reply Processing

When a node originates an AREQ, it sets a timer for ADDRESS_DISCOVERY milliseconds. During that time, it waits for the reception of an AREP. If no AREP is returned for the selected address within a timeout period, the node retries the AREQ up to AREQ_RETRIES times. If, after all retries, no AREP is still received, the node assumes that the address is not already in use, and that the address can safely be taken for its own.

On the other hand, if the node does receive an AREP within the discovery period, and if the Requested_IP_Address match its recorded values, then this indicates that another node within the ad hoc network is currently using that Requested_IP_Address. In this case, the node randomly picks another address from the same FIRST_PERM_ADDR - 65534 range and begins the ad hoc DAD procedure again.

7. IPv6 Address Autoconfiguration

7.1. Overview

This section describes the steps specific to an IPv6 Manet node taken when autoconfiguring an address to its interface. The steps described here adhere to the principles in IPv6 stateless Address Autoconfiguration [6], but with changes as specified below.

When an IPv6 node performs Manet address autoconfiguration, it first obtains a non-link-local prefix from which to configure an address. The prefix cannot have link-local scope because the address is valid over a multiple hop distance, not only to the immediate neighbors. The method to obtain a globally routable prefix may be the Internet Gateway Discovery, as described in Global Connectivity for Mobile Ad Hoc Networks [7].

In case the node does not know any suitable prefix, it uses the MANET_PREFIX, with prefix length 64, reserved for this purpose.

The node also acquires another temporary address for the sole use of an AREQ-AREP protocol message exchange for the uniqueness check of the chosen address. This second, temporary address is chosen from the MANET_INITIAL_PREFIX. The non-temporary actual address to configure is from the part of the MANET_PREFIX not overlapping with MANET_INITIAL_PREFIX.

Hence, unlike in NDP, stateless Manet autoconfiguration occurs whether Manet router advertisements are present or not.

The node selects a host number as in IPv6 stateless address autoconfiguration and concatenates this to the prefix. After this, the node performs a uniqueness check to this address, as specified below.

7.2. Address Request and Reply

To check for address uniqueness, the node sends an Address Request (AREQ) and expects to receive an Address Reply (AREP) if the tentative address is already in use within the reachable Manet. The AREQ is a modified Neighbor Solicitation containing the tentative address. The AREP is a modified Neighbor Advertisement response to the request. Message formats for IPv6 AREQ and AREP are given in section 5.

The IPv6 node broadcasts the AREQ to the all-nodes multicast address as the destination address in the IPv6 header. The source address is a temporary one. It MUST be picked at random from the

MANET_INITIAL_PREFIX. This address is only used in this message exchange and discarded thereafter. Uniqueness of this address is based on its short lifetime. A node does the uniqueness check only once for an address and the domain for these short-lived addresses is as large as the IPv4 address space.

After sending the AREQ the node then sets a timer to ADDRESS_DISCOVERY milliseconds, and proceeds as described in section 6.3. An AREP is a unicast sent back to the originator of the AREQ in case the tentative address was in use by sender of the AREP. In case no AREP was received within the timer wait, the tentative address is considered valid.

Address Request and Reply Processing follow the logic for the processing of the corresponding IPv4 messages.

In the case of IPv6, the node unicasts the AREP ADDRESS_RETRIES times, to increase robustness, back to the originator. The return path for AREP unicast MAY be the short lived state of AREQ source IPv6 address, link-layer source address, originating interface triplet which was stored in the intermediate nodes when forwarding the AREQ. When this state exists, a node forwarding a data packet first looks for the IP destination from that state. If found, the node forwards the packet to the respective link using the stored link layer address.

8. Security Considerations

This document does not define any method for secure operation of the autoconfiguration protocol. The danger exists that a malicious node may pretend to have any given IP address, so that another node would receive AREP messages apparently denying it the use of whatever address it might choose. This lack of security is problematic for many approaches to IP address autoconfiguration. It is symptomatic of the basic conflict between security, and operation in any mode where preconfigured information (including security association data) is not available.

9. Configuration Parameters

This section gives default values for some important values associated with address discovery protocol operations.

Parameter Name	Value
ALL_MANET_NODES	ff05:ffff::/64
ADDRESS_DISCOVERY	3 * NODE_TRAVERSAL_TIME * NET_DIAMETER / 2
REVERSE_ROUTE_LIFETIME	ADDRESS_DISCOVERY * 2
ADDRESS_RETRIES	3
FIRST_PERM_ADDR	2048
LAST_TMP_ADDR	FIRST_PERM_ADDR - 1
MANET_INITIAL_PREFIX	fec0:0:0:ffff::/96
MANET_PREFIX	fec0:0:0:ffff::/64
NET_DIAMETER	10
NODE_TRAVERSAL_TIME	40
UID_TIMEOUT	2 * ADDRESS_DISCOVERY

References

[1] S. Bradner. Key words for use in RFCs to Indicate Requirement Levels. Request for Comments (Best Current Practice) 2119, Internet Engineering Task Force, March 1997.

[2] J. Manner et al. Mobility Related Terminology (work in progress). draft-manner-seamoby-terms-02.txt, July 2001.

[3] E. Guttman and S. Cheshire (chairs). Zero Configuration Networking (zeroconf), June 1999. http://www.ietf.org/html.charters/zeroconf-charter.html.

[4] T. Narten, E. Nordmark, and W. Simpson. Neighbor Discovery for IP Version 6 (IPv6). Request for Comments (Draft Standard) 2461, Internet Engineering Task Force, December 1998.

[5] J. Postel. Internet Control Message Protocol. Request for Comments (Standard) 792, Internet Engineering Task Force, September 1981.

[6] S. Thomson and T. Narten. IPv6 Stateless Address Autoconfiguration. Request for Comments (Draft Standard) 2462, Internet Engineering Task Force, December 1998.

[7] R. Wakikawa, J. Malinen, C. Perkins, A. Nilsson, and A. Tuominen. Global connectivity for Mobile Ad Hoc Networks (work in progress). Internet Draft, Internet Engineering Task Force, November 2001.

Author's Addresses

Questions about this memo can be directed to:

Charles E. Perkins
Communications Systems Laboratory
Nokia Research Center
313 Fairchild Drive
Mountain View, CA 94303
USA
+1 650 625 2986
+1 650 625 2502 (fax)
charliep@iprg.nokia.com

Ryuji Wakikawa
Communications Systems Laboratory
Nokia Research Center
313 Fairchild Drive
Mountain View, CA 94303
USA
+1 650 625 2000
+1 650 625 2502 (fax)
rwakikaw@iprg.nokia.com

Jari T. Malinen
Communications Systems Laboratory
Nokia Research Center
313 Fairchild Drive
Mountain View, CA 94303
USA
+1 650 625 2355
+1 650 625 2502 (fax)
jmalinen@iprg.nokia.com

Elizabeth M. Belding-Royer
Dept. of Computer Science
University of California, Santa Barbara
Santa Barbara, CA 93106
+1 805 893 3411
+1 805 893 8553 (fax)
eroyer@cs.ucsb.edu

Yuan Sun
Dept. of Computer Science
University of California, Santa Barbara
Santa Barbara, CA 93106
+1 805 893 8981
+1 805 893 8553 (fax)
suny@cs.ucsb.edu