Hierarchical Prefix Configuration for IPv6 Network

Byung-yeob Kim, Jung-soo Park, Hyoung-jun Kim
Electronics and Telecommunications Research Institute
161, Gajong-Dong, Yusong-Gu, Daejon 305-350, Korea
Tel : +82-42-860-6627 Fax : +82-42-861-5404 E-mail: {skylane, pjs, khj}@etri.re.kr

Abstract

IPv6 enables stateless address configuration of hosts. Using this technique, a host in an IPv6 network can configure its address automatically with the help of a router. This eliminates the need for a server to manage the address dispatching state in conventional IPv4 networks. In this paper, we present a stateless configuration technique for routers. By expanding IPv6 stateless address configuration, a router can configure its network with the help of an upper level router, enabling a hierarchical, router-to-router network automatic configuration

Keywords: IPv6, Automatic Configuration, Prefix Delegation

1. INTRODUCTION

Since IPv6 was emerged with huge 128-bit address space, many of features have been devised by Internet Engineering Task Force (IETF). For instance, stateless address autoconfiguration technology defined by Neighbor Discovery for IPv6 [1] and Stateless Address Autoconfiguration [2] lets a host in a network configure its address automatically with some help of a router.

With stateless autoconfiguration, a host does not need a server who manages address dispatching state anymore. An address consists of two parts: network id (prefix) and host id (postfix). In an IPv6 network, for example, a 64-bit prefix and a 64-bit host id compose a 128-bit address. Stateless autoconfiguration enabled hosts to receive a prefix from a router, configuring its address automatically, enabling network plug-and-play for a host. However, setting a prefix for a router remains unsolved. The following researches have been done to extend stateless autoconfiguration to support router-to-router prefix delegation.

2. RELATED WORKS

2.1. Automatic Prefix Delegation (APD)

APD is a router-to-router prefix delegation protocol proposed by B. Haberman and J. Martin [4]. APD defined two new ICMPv6 [3] messages for a prefix requesting router and a prefix delegating router. A requesting router use a prefix request message to request prefixes and a delegating router response with a prefix delegating message which has actual prefixes and other relevant information. Since a requesting router cannot delegate a prefix to other routers, application of APD is confined to single level hierarchy. From a hierarchical network perspective APD has room for improvement.

2.2. Router Advertisement – Prefix Delegation Option (RA-PD Option)

Nathan Lutchansky introduced a prefix delegation mechanism by defining an option message to IPv6's Router Advertisement (RA) message [5]. RA message was designed for a router to deliver a prefix to a host. By defining an option message, RA became available for router-to-router prefix delegation.

With RA-PD option, a lower level router performs prefix delegation by simply receiving RA message with PD option which contains prefixes from a higher level router. There's no prefix requesting procedure for RA-PD option, also it lacks multiple-level hierarchical prefix delivery. Due to its simplicity, RA-PD option is believed to suitable for simple environment.

The following part of this paper is composed of three sections. In section 3, we propose a hierarchical prefix delegation which is equipped with multiple-level prefix delegation capability. Basic message flow with message formats is also introduced. In section 4, Implementation issues of proposed protocol are described. Lastly, conclusions are presented in section 5.

3. HIERARCHICAL PREFIX DELEGATION

In this paper, we propose a prefix delegation protocol named Hierarchical Prefix Delegation (HPD). Based on APD protocol, HPD extends prefix delegation across multiple levels. Beside basic prefix delegation, HPD has the following extended features:

Extended Prefix Delegation – Prefix delegation is not limited to single level. Once a requesting router received a valid prefix from a delegating router, it can act as a delegating router. A prefix can be divided into several longer length prefixes to be provided to the lower level routers. Any router can derive longer prefixes from the delegated prefix, and then delegate them to lower level router.

Built-in Routing - HPD has a built-in routing capability which enables routing among HPD routers without external dynamic routing protocols such as RIP or OSPF. Built-in routing is performed by remembering interfaces where prefixes are received and sent. A HPD router set a prefix received interface as a default gateway, opening a route to the upper level router. Likewise, routes to subnets are made by registering prefix sent interfaces with a routing table.

3.1. Protocol Overview

In order to delegate prefixes between routers, HPD defines two new ICMPv6 messages; they are Prefix Request message and Prefix Delegation message. Prefix Request message is used by a lower level router to request a higher level router to delegate a prefix. Conversely, Prefix Delegation message is used by a higher level router to respond to a prefix request. Both messages can have one of four code values according to the prefix negotiation process. Actual prefixes delivery is made by Prefix Information Option attached to a Prefix Delegation message.

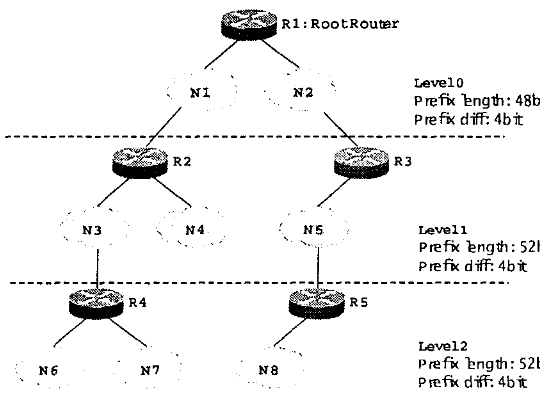

Fig. 1. A Hierarchical Network Example.

Figure 1 shows an example of a hierarchical network. In this figure the highest level (level 0) router, R1 is the root router. We presume the root router is configured manually to have 48-bit prefix by a network administrator. This prefix will be used on N1 and N2, networks belong to R1. On level 1, there are two routers (R2, R3). These routers request the root router to delegate a 52-bit prefixes which are 4-bit longer than the prefix use by the root router. These 52-bit prefixes are derived from the 48-bit prefix and used on N3, N4 and N5. In the same manner, routers on level2 will be received 4-bit longer 56-bit prefixes from their upper level routers, R2 and R3. In this scenario the 4-bit prefix difference across levels is defined by the network administrator and is inherited by prefix delegation level by level.

The message exchange procedure of HPD is depicted on figure 2. In this figure, R2 is a requesting router sending Prefix Request message while R1 is a delegating router sending Prefix Delegation message. Firstly, R2 query a router, by sending a Prefix Request message with code DELEGATOR_QUERY using multicast all over its interfaces. R1 and R4 receive the same message at the same time; however, only R1 respond to this request since it has a prefix to provide. R4 simply discards this request since it has no prefix to provide yet.

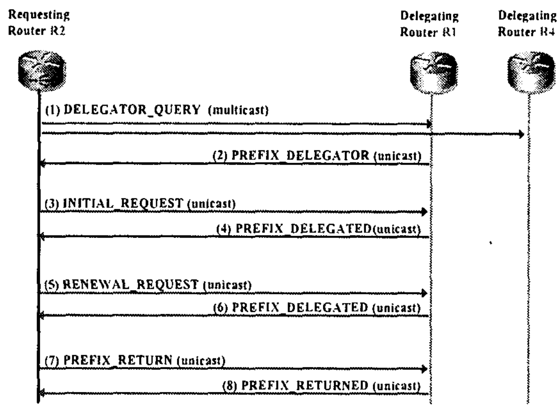

Fig. 2. HPD Message Exchange.

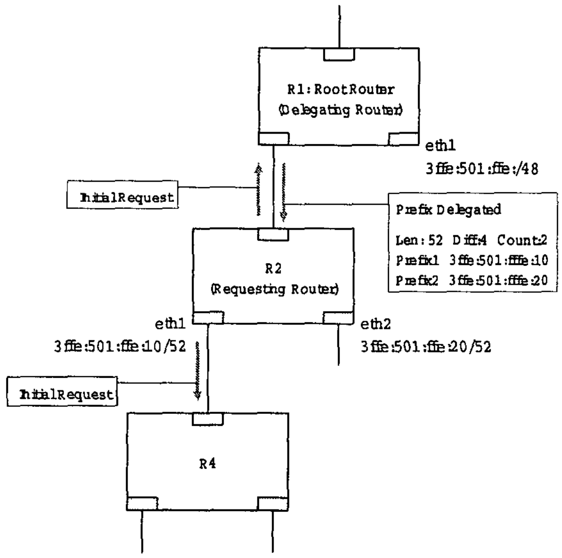

Fig. 3. HPD Prefix Delegation Result

As soon as R2 gets the presence of a delegating router (R1) by receiving code PREFIX_DELEGATOR, it unicasts a Prefix Request message with code INITIAL_REQUEST to R1, requesting the actual prefix. R2 specifies the number of prefixes it wants to be delegated on the message. R1 unicasts a Prefix Delegation message as an answer; it has a Prefix Information Option that has the actual prefixes.

A prefix may have lifetime which is specified by a Prefix Delegation message. A requesting router can request a renewal of a prefix by sending a Prefix Request message with code RENEWAL_REQUEST and a delegating router can endorse the renewal by sending Prefix Delegation message with code PREFIX_DELEGATED. The PREFIX_RETURN code is used when a requesting router returns delegated prefixes while a code PREFIX_RETURNED is used by a Delegating Router as a response.

Figure 3 shows the result of HPD prefix delegation. R2 is delegated two 52-bit prefixes from R1 and then assigns them on its interfaces eth1 and eth2. These prefixes are used on network N3 and N4. Once R2 is delegated prefixes, it acts as a Delegating Router, deriving longer length prefixes and delegating them to lower level routers. In this scenario R2 delegated a 56-bit prefix to R4.

3.2. Message Format Design

We have described the prefix delegation process by defining two ICMPv6 messages, Prefix Request and Prefix Delegation. The message format of these two messages is shown in Figure 4 and Figure 5.

```
 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             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   PrefixCnt   |   PrefixLen   |  PrefixDiff |R|   Reserved    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                  Prefix Information Option                    +
|                                                               |
```

Fig. 4. HPD Message Format

```
 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             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Prefix Lifetime                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                                                               |
+                            Prefix                             +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Fig. 5. Prefix Information Option Format

As stated earlier, these messages are designed as ICMPv6 messages; hence IPv6 header precedes these messages. The HPD message begins with Type field, which is to be assigned a unique value that is to identify HPD messages from other ICMPv6 messages. Since the same message format is used for both Request Prefix message and Prefix Delegation message, the two messages are distinguishable only by the Type value. The checksum field contains ICMP checksum defined in ICMPv6. PrefixCnt indicates the number of requesting or requested prefixes. PrefixLen is to show the length of the prefixes, while PrefixDiff denotes the difference of the prefixes used by the Requesting Router and the Delegating Router. R flag is not used in HPD protocol. Reserved field remains unused for the future protocol expansion.

The Code filed contains actual meaning of the messages. Each message has four different code values as described below.

DELEGATOR_QUERY: 0
DELEGATOR_QUERY code is used to query any router who can provide prefixes. The message contains this code is delivered using all-router link-local multicast. The requesting router can point out the number of prefixes it wants in the PrefixCnt field.

INITIAL_REQUEST: 1
INITIAL_REQUEST is sent to the unicast address of a delegating router. PrefixCnt filed should contiain the same or smaller number than the PrefixCnt received by the Prefix Delegation message with PREFIX_DELEGATOR code.

RENEWAL_REQUEST: 2
RENEWAL_REQUEST is used to update lifetime of a prefix. The Prefix Request message with this code is sent to the unicast address of the delegating router. At least one Prefix Information Option is attached to this message.

PREFIX_RETURN: 3
This code is used to return prefixes to the delegating router. The Prefix Request message with this code is sent to the unicast address of the delegating router. At least one Prefix Information Option is attached to this message.

Prefix Delegation message has one of the following four codes.

PREFIX_DELEGATOR: 0
The code PREFIX_DELEGATOR is used to inform the requesting router of the presence of delegating router. PrefixCnt field in this case contains the number of available prefixes and the value of 0 indicates there's no prefix available. The delegating router should put the difference of the length of prefixes in PrefixDiff field.

Authentication Required: 1
This code is to request the requesting router to use Cryptographically Generated Address (CGA) [6] as its source address. This code can be used as a response to the Prefix Request message with DELEGATOR_QUERY. The requesting router who received this code should use CGA as its source address for the consequent messages. The use of CGA addresses is up to network administrators according to local network policy for better network security.

PREFIX_DELEGATED: 2
PREFIX_DELEGATED code is for actual prefix delivery by the delegating router. Prefixes are provided in the Prefix Information Option attached to this message.

PREFIX_RETURNED: 3
The code PREFIX_RETURNED is used as an acknowledgement to the Prefix Return code. PrefixCnt message shows the number of returned prefixes.

4. IMPLEMENTATION ISSUES

We implemented the HPD by modifying RADVD [9], an IPv6 messaging daemon for LINUX. The testbed set up is based on the network topology shown in Figure 3. LINUX PCs equipped with multiple Ethernet cards are used as routers and are set up to run HPD protocol. The root router is set to get a prefix by reading configuration file provided by a network administrator. Other routers on each level get prefixes from their upper level routers, and then play the role of a delegating router for the routers on the lower level. Once a router sets its prefix, hosts connected to it get the prefix using IPv6's stateless address autoconfiguration, and then they configured their address using it.

When the built-in routing feature is on, the HPD routers perform routing among themselves without any dynamic routing protocols. The Root Router uses a static route to the outgoing interface for the global Internet connectivity.

5. CONCLUSIONS

We have discussed the Hierarchical Prefix Delegation protocol for IPv6 networks. The HPD protocol is designed to delegate prefixes between routers in hierarchical networks. With IPv6's native host address autoconfiguration capability, HPD can be used to provide address autoconfiguration over the entire network under the administrator's control.

HPD provides both network administrators and network users with plug-and-play environment. By manually configuring the prefix of the Root Router, a network administrator can configure the network prefixes all over the hierarchical network. The built-in routing capability in HPD eliminates the need for a dynamic routing protocol, decreasing the running cost of the network. Also, these prefixes provided on routers are used by hosts connected to the routers to configure their host addresses. We are streamlining HPD protocol to meet Requirements for IPv6 Prefix Delegation [10] with auxiliary automatic configuration for other network relevant information. Also, we are investigating the enhancement in security using Secured Neighbor Discovery [7].

References

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

[2] S. Thompson and T. Narten, "IPv6 Stateless Address Autoconfiguration", RFC-2462, December 1998

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

[4] B. Haberman and J. Martin, "Automatic Prefix Delegation Protocol for Internet Protocol Version 6", expired,<draft-haberman-ipngwg-auto-prefix-02.txt>, February 2002

[5] Nathan Lutchansky, "IPv6 Router Advertisement Prefix Delegation Option," expired, <draft-lutchann-ipv6-delegeta-option-00.txt>, February 2002

[6] T. Aura, "Cryptographically Generated Addresses (CGA) ", working in progress, <draft-ietf-send-cga-01.txt>

[7] J. Arkko, "SEcure Neighbor Discovery (SEND)", working in progress, <draft-arkko-send-ndopt-00.txt>

[8] Erik Guttman, "Autoconfiguration for IP networking: enabling local communication", IEEE Internet computing, June 2001

[9] Linux IPv6 Advertisement Daemon (radvd), http://v6web.litech.org/radvd/

[10] S. Miyakawa and R. Droms, "Requirements for IPv6 Prefix Delegation", RFC-3769, June 2004

[11] Byung-Yeob Kim et al., "Hierarchical Prefix Delegation Protocol for Internet Protocol Version 6 (IPv6)", <draft-bykim-ipv6-hpd-01.txt>, working in progress, February 2004