Providing Network Services with Multiple Prefix Delegation

SUZUKI Shinsuke
Central Research Laboratory
Hitachi, Ltd.
1-280 Higashi-Koigakubo, Kokubunji, Tokyo, 185-8601, Japan
suz@crl.hitachi.co.jp

Abstract

With the deployment of the network services requiring special network control, such as VoIP, VPN, and streaming, it is important to configure network controlling policies according to the characteristics of the services provided on the network.

If such kind of services is provided to a SOHO network, it is difficult to configure controlling policies when the number of SOHO network users increases. This is because SOHO network users request different network services, requiring different network controlling policies to be configured on each SOHO network using its IP address.

To solve this problem, this paper proposes to allocate an address block per service and to assign prefixes to a SOHO network user from the address blocks of all the services requested by the user. With this proposal, multiple network services can be provided on an IPv6 network, taking advantage of its vast address space. It also illustrates some implementation examples of multiple prefix assignment using longest-match source-address-selection and multiple DHCPv6-based prefix-delegation.

1. Introduction

Recently, various network services, such as VoIP, VPN, and streaming, have matured and their deployment to customers has started in addition to the Internet connectivity services. Although it is theoretically possible to provide all of them within one connection, it is normally provided separately.

There are two reasons for this separation. The first reason is that customers do not want to use all the network services available to them. From the marketing point of view, network services are provided separately in order to prevent unnecessary service provisioning.

The second reason is that these services require special network controlling policies, such as packet filtering, and QoS, according to the characteristics of each of the services. From the network administrators' point of view, it is convenient to separate them from the Internet connectivity service, since these policies have to be configured independently from the Internet connectivity services. Otherwise, network administrators have to configure different network controlling policies based on the IP addresses and service port numbers. When the number of customers increases, this kind of operation would become impractical.

When such a network service connection is provided separately from the Internet connection, customers are forced to switch these connections manually to use a specific network service. For example, in case of VoIP service, a VoIP network is provided as a different network from the Internet, and the customers have to connect their phones and PCs to different networks. In case of VPN service, customers have to install a special VPN software and enable VPN on their PCs when they want to use VPN, and disable it when they do not want to use VPN. Obviously, such an awkward operation become an obstacle to the deployment of such new network services both for customers and service providers.

In this paper, a solution to overcome this obstacle by assigning multiple prefixes to customer's network is proposed. An automatic configuration mechanism of this solution, especially for SOHO customers, is also shown.

2. Multiple Prefix Assignment

To solve the problem mentioned above, this paper proposes to allocate an address block per service, and to assign prefixes to customers from the address blocks of all the services requested by the customer. Once the prefixes are given to the customers for all the requested services, customers can choose an appropriate prefix according to the service they want to use.

2.1. Example of Multiple Prefix Assignment

Suppose an ISP, having an sTLA 2001:db8::/32, provides VoIP service, streaming service, and Internet connectivity services, and there are three customers (A, B, and C) with the following contracts (see Table 1).

Table 1. an example of customer contract

Customer	VoIP service	streaming service	Internet connectivity service
A	–	x	x
B	x	–	x
C	–	–	x

With the proposed method, an address block is allocated for each services; in this case, 2001:db8:1000::/40 for VoIP service, 2001:db8:2000::/40 for streaming service, and 2001:db8:f000::/40 for the Internet connectivity service. Prefixes are assigned to customers from these address blocks based on their contract; for example, 2001:db8:1000:1::/64 (VoIP service) and 2001:db8:f000:1::/64 (Internet connectivyt service) to customer-A. ISP network administrators just have to apply the network controlling policies for the whole address block; for example, guaranteeing SLA for the traffic within 2001:db8:1000::/40, or guaranteeing at least 1Mbps bandwidth for the traffic within 2001:db8:2000::/40, etc.

VoIP service and streaming service are listed as special network services in this example, but the proposed method can be applied for other network services. One example is a VPN service for intranet connectivity. Since intranet service should only be provided for the intranet customers, it is natural to assign a prefix to VPN customers from the VPN address block for proper ingress-filtering [7].

Another example is a Reverse Path Forwarding (RPF) calculation in multicast network. In multicast routing, routers normally calculate the upsteam router for multicast source (RPF) by unicast routing table lookup. This calculation assumes that multicast network topology is identical to the unicast network topology. This is, however, not always true, since not all the routers support multicast routing. To overcome this topology difference, special unicast routing protocols [9, 1] are proposed to maintain unicast routing table dedicated for RPF calculation. In IPv6, however, this is not necessary if you allocate an address block dedicated for multicasting. If you allocate prefixes from this address block only to the multicast-ready network segments, RPF calculation never fails since no multicast-ready routers would have this prefix.

2.2. Advantages of Multiple Prefix Assignment and Its Limitations

The greatest advantage of this proposal is the operational simplicity; since network controlling policies do not depend on each customer's addresses, their configuration remains the same even when the number of customers increases or decreases.

Compared to the network service multiplexing by TCP/UDP port number, this proposal is much more generic and especially suited for secure communication by IPsec; since TCP/UDP header field is encrypted in IPsec technology, intermediate routers cannot apply the port-number-based network controlling policies to IPsec packets.

The only negative aspect of this proposal is the waste of IP addresses. This is, however, not a problem at least in IPv6, considering its vast address space. In other words, this kind of network service multiplexing can be an IPv6-specific solution that is almost impossible in IPv4.

Multihoming is out of scope in this proposal, although it appears similar to this proposal. This is because this proposal assumes that a prefix corresponds to a service one to one, whereas in multihoming this is not always satisfied.

3. Implementing Multiple Prefix Assignment

This section shows how to implement multiple prefix assignment with automatic configuration mechanisms especially for SOHO network users. Such an automatic configuration mechanism is indispensable when deploying multiple prefix assignment, since it is difficult for normal SOHO network users to configure their PC or CPE routers.

In the following discussion, the topology in Figure 1 is assumed, where the SOHO network is composed of a CPE router and PCs, while the ISP network is composed of PE routers and the upstream services.

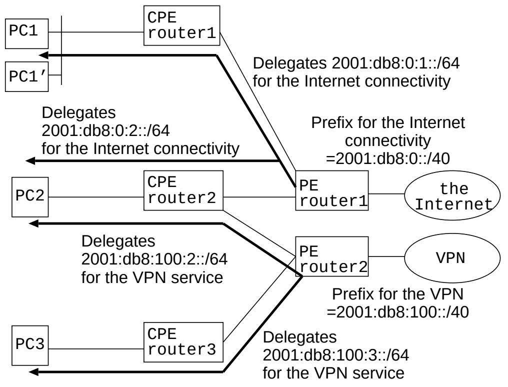
3.1. Router Requirements and Solutions

There are three requirements for PE routers and CPE routers when providing proper connectivity to the downstream nodes; Prefix Delegation, Correct Upstream Selection, and Information Integration.

1. Prefix Delegation

   In this framework, prefixes have to be provided to fulfill the contract of the customers. To meet this requirement, there should be a protocol to provide prefixes from the PE router to CPE routers based on the customers' account information.

   Prefix Delegation (PD)[8] is an answer for this requirement. PD consists of two functions for an automatic


PC1
PC1'
CPE router1
Delegates 2001:db8:0:1::/64 for the Internet connectivity
Delegates 2001:db8:0:2::/64 for the Internet connectivity
Prefix for the Internet connectivity =2001:db8:0::/40
PC2
CPE router2
PE router1
the Internet
Delegates 2001:db8:100:2::/64 for the VPN service
PE router2
VPN
Prefix for the VPN =2001:db8:100::/40
PC3
CPE router3
Delegates 2001:db8:100:3::/64 for the VPN service

PC1/1' uses 2001:db8:0:1::/64 for the Internet connectivity service.

PC2 uses 2001:db8:0:2::/64 for the Internet connectivity service,
and 2001:db8:100:2::/64 for the VPN service.

PC3 uses 2001:db8:100:3::/64 for the VPN service.

Figure 1. Example network topology

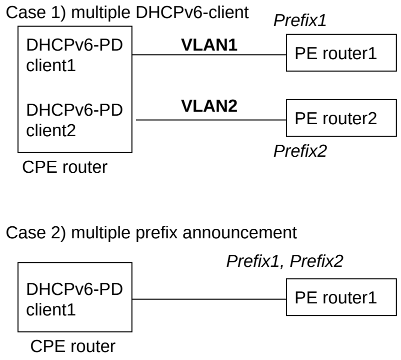


Figure 2. DHCPv6 clients on CPE router

prefix assignment; an automatic prefix assignment to CPE router from PE router, and an automatic Router-Advertisement to PCs from the CPE router using that prefix.

DHCPv6-PD[10], which is the most popularly implemented PD protocol, works on DHCPv6 protocol framework. Hence, DHCPv6-PD can delegate prefixes based on the customers' account information, like DHCPv4 addressing.

DHCPv6 client has to choose one DHCPv6 server in its protocol handshake even when multiple DHCPv6 servers are available. So when there are multiple DHCPv6-PD servers to support different services, the CPE router needs to run as many DHCPv6-PD clients as the servers, and gather prefixes from each server with each client (Figure 2). When there is only one DHCPv6-PD server to support different services, the CPE router need only one DHCPv6-PD client. The DHCPv6-PD server, however, has to provide multiple services, which can be troublesome considering the service provisioning.

2. Correct Upstream Selection

Since there are multiple upstream routers at a CPE router, it has to choose an appropriate router to forward packets from a PC; i.e. providing only one default route to an upstream router is not enough.

There are many techniques to configure this routing automatically, but the most solid method is to use a normal unicast routing protocol, like RIPng, to deliver appropriate routing information from PE routers to CPE routers. In the case of Figure 1, PE router1 should advertise the default route and PE router2 should advertise 2001:db8:100::/40 to their downstream CPE routers.

When a PE router delegates a prefix to a CPE router and the CPE router is connected to another PE router, the CPE router should not announce the delegated prefix to the second PE router; otherwise, unexpected packet forwarding occurs (see Figure 3). The simplest way to meet this condition is to make CPE routers listen only to the advertisement from the upstream PE routers. The PE routers can install a static route of the delegated prefix with the nexthop as the CPE router to which the prefix was delegated. In the case of Figure 1, PE router1 installs a static route to 2001:db8:0:2::/64 with the nexthop as CPE router2, and PE router2 installs a static route to 2001:db8:100:2::/64 with the nexthop as CPE router2.

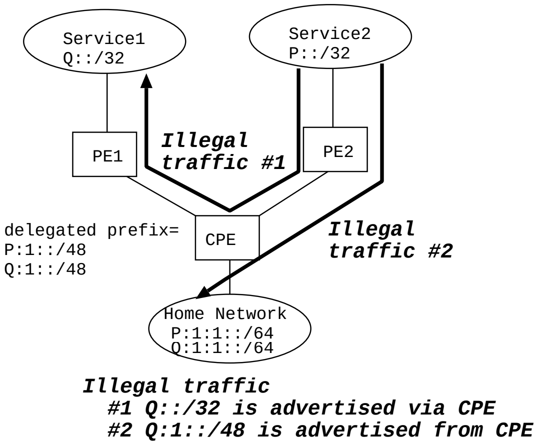

Service1 Q::/32
Service2 P::/32
PE1
Illegal traffic #1
PE2
delegated prefix= P:1::/48 Q:1::/48
CPE
Illegal traffic #2
Home Network P:1:1::/64 Q:1:1::/64
Illegal traffic
#1 Q::/32 is advertised via CPE
#2 Q:1::/48 is advertised from CPE

Figure 3. unexpected packet forwarding

3. Information Integration

   Each networking service can provide the same server function (providing different contents) to customers. A typical example is DNS; the Internet DNS and intranet DNS may be provided at the same time from different network services. In such a case, these resembling server functions have to be integrated within CPE routers, so that PCs need not be aware of their existence. Or they should be conveyed to PCs so that PCs can select which service is to be used. These two are same in the information integration's point of view; only difference is the place to integrate information.

   In general, such an information integration is a difficult task. Regarding DNS, however, it can be solved by a DNS relay on CPE router, which queries to the correct upstream DNS server using the target domain-name. Since DNS server information and DNS domain-name information can be provided to CPE router via DHCPv6[4, 6] from PE router, CPE routers just have to reflect the DNS-related information into its DNS relay configuration (Figure 4).

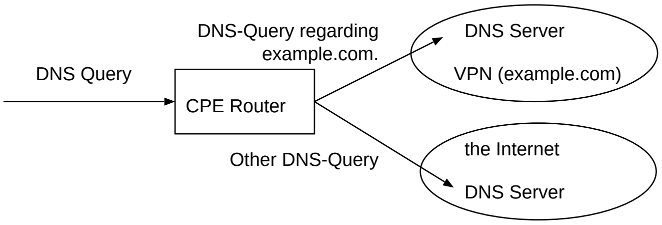


Figure 4. Switching DNS relay

3.2. Node Requirements and Solutions

Multiple prefixes are assigned automatically by PD and packets for these prefixes are properly routed as depicted in the previous section, so the only requirements on node are the acceptance of the integrated information and the proper source address selection.

1. Acceptance of the Integrated Information

   As described in the previous section, CPE routers integrate information from upstream routers, so the information has to be informed to the downstream nodes in some manner, and the nodes have to receive that information. In case of DNS, the CPE router have to announce that the CPE router is the DNS recursive name server and the nodes have to make use of this server information. To inform these information, stateless-DHCPv6 [5] can be used.

2. Source Address Selection

   When a node has multiple addresses, it has to choose a right source address depending on the destination where it tries to send a packet. Otherwise, it has to try all the possible addresses as a source address. This, however, leads to a UDP communication failure, or a long delay in TCP connection establishment.

   The longest-match source address selection algorithm [3] is sufficient, since the prefix of each service is allocated from its address block; for example, when PC2 in Figure 1 tries to send a packet to 2001:db8:101::1 in VPN, the source address is 2001:db8:100:2:xxx, instead of 2001:db8:0:2:xxx, since it matches best with the destination address 2001:db8:101::1.

   However, there is a case where this longest-match source address selection is not sufficient; suppose ISP-A and ISP-B provides VPN service and the Internet connectivity service, and a customer has ISP-A's VPN service and ISP-B's Internet connectivity service. In this case a customer cannot connect to the Internet (non-VPN) site on ISP-A, since the customer's node chooses the VPN prefix as the source address and in general it is impossible to connect from VPN to the Internet.

   To prevent such a case, a criteria for source address selection should be provided to nodes. Because only the service provider can determine such a criteria, it is natural to inform such information from PE router to nodes via CPE. It can be achieved using a DHCPv6 option containing a set of source address selection policy table [3] entries. (Figure 5).

3.3. Security Consideration

There are some security issues that have to be taken care of in the proposed automatic configuration method.

First issue is the improper use of a prefix; if someone uses a prefix from the address block of a network service without any permission from the PE routers, he/she can make use of the network service that corresponds to that prefix. This can be protected, however, by routing protocol filter at PE routers; even if someone announces such bogus prefixes to the PE router, the PE router can filter them. Once they are filtered, he/she cannot make use of the network services using that prefix, since his/her packets cannot be routed.

Second issue is a faked PE router; if someone installs a faked PE router, and this PE router delegates and routes prefixes in the proposed manner, ISP's network service can be overridden by that router. This can happen only when

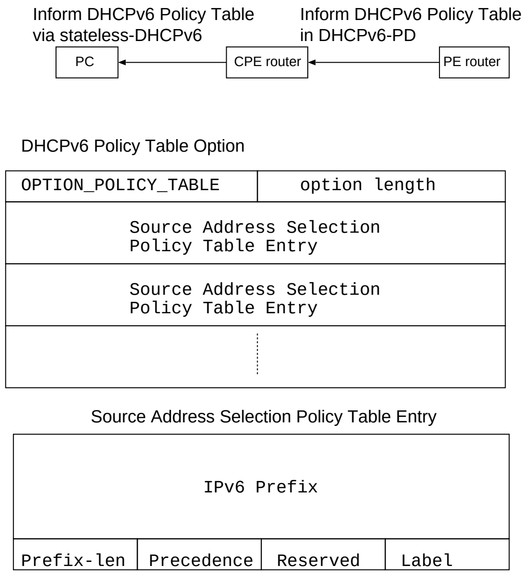


Figure 5. DHCPv6 policy table option

PE router and CPE routers are connected on a shared media like Ethernet (Figure 6).

Considering network services are normally provided on a separate connection, practically speaking this is not a serious issue. If necessary, you can detect it by monitoring protocol messages within the shared media; if these protocol messages do not arrive from the proper PE router's address, it means there is a faked PD server. It is easy to monitor these protocol packets even when MLD snooping[2] is used, since these protocol messages are sent to linklocal multicast address.

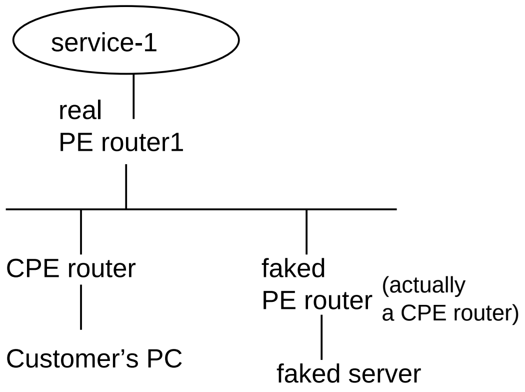


Figure 6. faked PD server

4. Conclusions

This paper proposes a multiple-address-assignment operation, which simplifies network service multiplexing. It also clarifies the requirements and solutions for routers and hosts. Furthermode, it introduces an implementation example for SOHO network to satisfy these requirements automatically, using DHCPv6-based techniques such as prefix-delegation, DNS-information integration, and the announcement of source-address-selection policy. Some security issues in this automatic configuration are also analyzed.

5. Acknowledgment

This work is supported by Telecommunications Advanced Organization (TAO) of Japanese Government.

I would like to show my special thanks to ATARASHI Yoshifumi, IKEDA Naoya, MIZUTANI Mika, and OHNO Shuji for their valuable suggestion for this proposal.

References

[1] T. Bates, Y. Rekter, R. Chandra, and D. Katz. “Multiprotocol Extensions for BGP-4”. *IETF RFC2858*, June 2000.

[2] M. J. Christensen, K. Kimball, and F. Solensky. “Considerations for IGMP and MLD Snooping Switches”. *IETF draft-ietf-magma-snoop-09.txt*, August 2003.

[3] R. Draves. “Default Address Selection for Internet Protocol version 6 (IPv6)”. *IETF RFC3484*, February 2003.

[4] R. Droms. “DNS Configuration options for DHCPv6”. *IETF draft-ietf-dhc-dhcpv6-opt-dnsconfig-04.txt*, August 2003.

[5] R. Droms. “Guide to Implementing Stateless DHCPv6 Service”. *IETF draft-ietf-dhc-dhcpv6-stateless-01.txt*, October 2003.

[6] R. Droms, J. Bound, B. Voltz, T. Lemon, C. E. Perkins, and M. Carney. “Dynamic Host Configuration Protocol for IPv6 (DHCPv6)”. *IETF RFC3315*, July 2003.

[7] P. Ferguson and D. Senie. “Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing”. *IETF RFC2827*, May 2000.

[8] S. Miyakawa and R. Droms. “Requirements for IPv6 prefix delegation”. *IETF draft-ietf-ipv6-prefix-delegation-03.txt*, January 2002.

[9] J. Moy. “Multicast Extensions for OSPF”. *IETF RFC1584*, March 1994.

[10] O. Troan and R. Droms. “IPv6 Prefix Options for DHCPv6”. *IETF draft-ietf-dhc-dhcpv6-opt-prefix-delegation-05.txt*, October 2003.