IPv6 Support Configuration Guide, 5.2R1

November 2017

Supporting AT&T Vyatta Network Operating System

Contents

About This Guide

This guide describes IPv6 support on AT&T products that run on the AT&T Vyatta Network OS (referred to as a virtual router, vRouter, or router in the guide).

IPv6 Support Overview

IPv6 background

There are two versions of the Internet Protocol (IP) in use today. Version 4 (IPv4) is the version most commonly in use. However, there are issues with IPv4, and the Internet Engineering Task Force (IETF) has designated Version 6 (IPv6) to succeed IPv4 as the next-generation protocol for use on the Internet.

IPv6 has a number of advantages over IPv4. The following are four important ones:

- **Large address space**
  An IPv4 address consists of four bytes (32 bits). IPv6 addresses consist of 16 bytes (128 bits). The increase from 32 to 128 bits results in a huge increase in the number of available addresses: 79 billion billion billion times the addresses available in the IPv4—this is about 1038 addresses, or 1030 addresses for each person on the planet.

  The expanded address space means that IPv6 does not face the address exhaustion problems predicted imminently for IPv4. Furthermore, the availability for so many addresses means that private address spaces are not required, and that address shortage work-arounds such as Network Address Translation (NAT) can be eliminated. With no private addresses, there need be no hidden networks or hosts, and all devices can be globally reachable. A larger address space also means that features such as multihoming and aggregation are easier to implement.
- **Support for mobile devices**
  A special protocol, Mobile IP, is required to support mobility. Mobile IP is not automatic in IPv4, and there are several challenges involved in implementing Mobile IP on IPv4 networks. In contrast, Mobile IP was designed into IPv6 from its inception, and is a mandatory feature in a standards-compliant IPv6 protocol stack.
- **Flexibility**
  IPv6 includes multiple levels of hierarchy in the address space. This allows for hierarchical allocation of addressing and more efficient route aggregation. It also permits new kinds of addresses not possible in IPv4, such as link- and site-scoped addressing.
- **Security**
  Because devices can be globally reachable, end-to-end security can be employed, which is not possible on an internetwork with hidden networks and hosts.

Supported standards

The AT&T vRouter implementation of IPv6 complies with the following standards:

- RFC 2460: *Internet Protocol, version 6 (IPv6) Specification*
- RFC 4443: *Internet Control Message Protocol (ICMPv6) for the Internet protocol version 6 (IPv6)*

IPv6 addressing

IP addresses generally take the following form:

```
x:x:x:x:x:x:x:x
```

where *x* is a 16-bit hexadecimal number; for example:

```
2001:0DB8:0000:0000:51DA:27C0:E4C2:0124
```

Addresses are case-insensitive; for example, the following is equivalent to the example given above:

```
2001:0db8:0000:0000:51da:27c0:E4c2:0124
```

Leading zeros are optional; for example, the following is a valid IPv6 address:

```
2001:DB8:0:0:51DA:27C0:E4C2:124
```

IPv6 addresses often contain many bytes with a value of zero. Successive fields of zeros can be represented by replacing them with a double colon, as in the following:

```
2001:DB8::51DA:27C0:E4C2:124
```

Similarly the following:

```
2001:DB8::124
```

is equivalent to the following:

```
2001:DB8:0:0:0:0:0:0124
```

and this:

```
0:0:0:0:0:0:0:1
```

is equivalent to this:

```
::1
```

The replacement by the double colon may be made only once within an address, as using the double colon more than once can result in ambiguity. For example, the following:

```
2001:DB8::27C0::0124
```

is ambiguous between these three addresses:

```
2001:0DB8:0000:27C0:0000:0000:0000:0124
2001:0DB8:0000:0000:27C0:0000:0000:0124
2001:0DB8:0000:0000:0000:27C0:0000:0124
```

IPv6 addresses that are extensions of IPv4 addresses can be written in a mixed notation, where the rightmost four bytes of the IPv6 address are replaced with the four decimal octets of the IPv4 address. In mixed notation, the four hexadecimal bytes are separated by colons and the four decimal octets are separated by dots, as in the following example:

```
2001:db8:0:1::192.168.100.51
```

which is equivalent to

```
2001:db8:0:1::c0a8:6433
```

Special addresses

Like IPv4, IPv6 has some special addresses, which are used by convention for special functions. For unicast addresses, these include the following:

- The unspecified address. This address is used as a placeholder when no address is available (for example, in an initial DHCP address), or to stand for “any” address. In IPv6, the unspecified address can be represented as either of the following:

  ```
  0:0:0:0:0:0:0:0
  ::
  ```

- The localhost (loopback) interface. The loopback interface is a software interface that represents the local device itself. In IPv4, the address 127.0.0.1 is used by convention for the loopback interface. In IPv6, the loopback interface can be represented by either of the following:

  ```
  0:0:0:0:0:0:0:1
  ::1
  ```

The IPv6 address architecture is quite rich, and includes types of addressing unavailable in IPv4, such as unicast and multicast scoped addresses, aggregatable global addresses, and anycast addresses. Multicast broadcast addresses do not exist in IPv6. For more information about the IPv6 address architecture, consult RFC 4291, *IP Version 6 Addressing Architecture*.

IPv6 autoconfiguration

IPv6 supports two mechanisms for automatically configuring devices with IP addresses: stateful and stateless. Both are supported in the AT&T Vyatta vRouter.

In stateful configuration, addressing and service information is distributed by a protocol (DHCPv6) in the same way that the Dynamic Host Configuration Protocol (DHCP) distributes information for IPv4. This information is “stateful” in that both the DHCP server and the DHCP client must maintain the addressing and service information.

Stateless configuration uses the Stateless Address Autoconfiguration (SLAAC) protocol, which is a component of the larger Neighbor Discovery (ND) protocol. SLAAC has a host component and a router component.

In the host component of SLAAC, the IPv6 system constructs its own unicast global address from the system’s network prefix together with its Ethernet media access control (MAC) address. The device proposes this address to the network, without requiring approval from a server such as a DHCP server. The combination of network prefix and MAC address is assumed to be unique. Stateless autoconfiguration is performed by default by most IPv6 systems, including the AT&T Vyatta vRouter.

In the router component of SLAAC, routers respond to Router Solicitation (RS) packets from hosts with network prefix information in the form of Router Advertisement (RA) packet. Hosts receive these advertisements and use them to form globally unique IPv6 addresses. The RS and RA packets also provide the router discovery function, allowing hosts to locate routers that are configured to serve as default routers. The AT&T Vyatta vRouter fully supports router-side SLAAC and router discovery, including all required configurable parameters.

The ND protocol and the router discovery function are specified in RFC 4861. IPv6 Stateless Address Autoconfiguration is described in RFC 4862.

IPv6 forwarding

On the AT&T Vyatta vRouter, IPv6 forwarding is enabled by default. If you want to disable IPv6 forwarding, use the following command in configuration mode: `set system ipv6 disable-forwarding`. This command is described in AT&T Vyatta Network Operating System Basic System Configuration Guide.

IPv6 neighbor discovery

IPv6 Neighbor Discovery (ND) provides a layer 3 to layer 2 address resolution mechanism for IPv6 similar to the way that Address Resolution Protocol (ARP) provides for layer 3 to layer 2 address resolution for IPv4.

ND resolution is carried out in both the data plane and the control plane; however, it is primarily carried out in the data plane. Note that ND caches in the control plane and data plane are no longer synchronized because entries in the two caches are managed independently. The data plane cache contains entries for both forwarded and locally terminated traffic. The control plane maintains cache entries only for destinations with which the local stack of the control plane communicates.

The advantages of implementing the ND protocol in the data plane are as follows:

- Avoids bandwidth issues in deployments with distributed data planes, because ND resolution can be performed locally rather than on a centralized controller.
- Improves performance because ND does not need to send all ND packets to the control plane.
- Protects against scanning DOS attacks due to resolution throttling.

Commands for IPv6

In addition to the general IPv6 information found in this document, information specific to major functions of the AT&T Vyatta vRouter are found within the applicable documents for that function; for example, the following:

- Commands for enabling and disabling IPv6 on the system are located in AT&T Vyatta Network Operating System Basic System Configuration Guide.
- Commands for configuring IPv6 on a given interface are located in the guide that describes the interface. For example, commands for configuring IPv6 on an Ethernet interface are located in AT&T Vyatta Network Operating System LAN Interfaces Configuration Guide.
- Static IPv6 routing information can be found in AT&T Vyatta Network Operating System Basic Routing Configuration Guide.
- RIPng-related dynamic IPv6 routing information can be found in AT&T Vyatta Network Operating System RIPng Configuration Guide.
- BGP-related dynamic IPv6 routing information can be found in AT&T Vyatta Network Operating System BGP Configuration Guide.
- DHCPv6-related information can be found in AT&T Vyatta Network Operating System Services Configuration Guide as well as AT&T Vyatta Network Operating System LAN Interfaces Configuration Guide.
- Tunneling IPv6 over IPv4 is discussed in AT&T Vyatta Network Operating System Tunnels Configuration Guide.
- Multicast routing for IPv6 is discussed in AT&T Vyatta Network Operating System IGMP and MLD Configuration Guide.

IPv6 Configuration Examples

Configure an IPv6 address on an interface

The following figure shows a simple network with two IPv6 nodes.

Figure 1: IPv6 address on an interface

IPv6 addresses are configured on data-plane interfaces in the same way that IPv4 addresses are. To configure dp0p1p3 on R1, perform the following steps in configuration mode.

Table 1: Add an IPv6 address to dp0p1p3 on R1

Step	Command
Add the IPv6 address to the dp0p1p3 interface.	`vyatta@R1# set interfaces dataplane dp0p1p3 address 2001:db8:2::1/64`
Commit the change.	`vyatta@R1# commit`
Verify the configuration.	`vyatta@R1# show interfaces dataplane dp0p1p3 duplex auto hw-id b6:71:6b:8a:c9:3c mtu 1500 speed auto`
Change to operational mode.	`vyatta@R1# exit exit vyatta@R1:~$`

Step	Command
Show the status of the interfaces on R1.	vyatta@R1:~$ show interfaces Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down Interface IP Address S/L Description dp0p1p1 - u/u dp0p1p2 - u/u dp0p1p3 2001:DB8:2::1/64 u/u dp0p1p4 - u/u lo 127.0.0.1/8 u/u lo ::1/128 u/u

To configure dp0p1p1 on R2, perform the following steps in configuration mode.

Table 2: Add an IPv6 address to dp0p1p1 on R2

Step	Command
Add the IPv6 address to the dp0p1p1 interface.	vyatta@R2# set interfaces dataplane dp0p1p1 address 2001:db8:2::2/64
Commit the change.	vyatta@R2# commit
Verify the configuration.	vyatta@R2# show interfaces dataplane dp0p1p1 address 2001:db8:2::2/64 duplex auto hw-id 3a:26:db:4d:63:a2 speed auto
Change to operational mode.	vyatta@R2# exit exit vyatta@R2:~$

Step	Command
Show the status of the interfaces on R2.	vyatta@R2:~$ show interfaces Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down Interface IP Address S/L Description dp0p1p1 2001:DB8:2::2/64 u/u dp0p1p2 - u/u dp0p1p3 - u/u lo 127.0.0.1/8 u/u lo ::1/128 u/u

Verify IPv6 support

A simple step to verify that IPv6 support is available is to configure the loopback interface with an IPv6 address and then ping it. To verify IPv6 support, perform the following step in operational mode.

Table 3: Confirm IPv6 support

Step	Command
Ping the loopback interface.	vyatta@R1:~$ ping ::1 PING ::1(::1) 56 data bytes 64 bytes from ::1: icmp_seq=1 ttl=64 time=2.13 ms 64 bytes from ::1: icmp_seq=2 ttl=64 time=0.086 ms ^C --- ::1 ping statistics --- 2 packets transmitted, 2 received, 0% packet loss, time 1006ms rtt min/avg/max/mdev = 0.086/1.112/2.138/1.026 ms

Display the IPv6 routing table

When an IPv6 address is added to an interface, a connected network for it appears in the routing table. To display the routing table, perform the following step in operational mode.

Table 4: Display the IPv6 routing table

Step	Command

Show the routing table.	`vyatta@R1:~$ show ipv6 route`
	`IPv6 Routing Table`
	`Codes: K - kernel route, C - connected, S -`
	` static, R - RIP, O - OSPF, IA - OSPF inter`
	` area, E1 - OSPF external type 1,`
	`  E2 - OSPF external type 2, I - IS-IS,`
	` B - BGP`
	`  > - selected route, * - FIB route, p -`
	` stale info`
	`Timers: Uptime`
	`C>* ::1/128 is directly connected, lo`
	`C>* 2001:db8:2::/64 is directly connected,`
	` dp0p1p3`
	`C * fe80::/64 is directly connected, dp0p1p3`
	`C * fe80::/64 is directly connected, dp0p1p2`
	`C>* fe80::/64 is directly connected, dp0p1p1`
	`K>* ff00::/8 is directly connected, dp0p1p3`

Confirm connectivity

To confirm that R1 and R2 can communicate, use the `ping` command. To confirm connectivity, perform the following step in operational mode.

Table 5: Confirm connectivity between R1 and R2

Step	Command

Ping R2 from R1.	`vyatta@R1:~$ ping 2001:db8:2::2`
	`PING 2001:db8:2::2(2001:db8:2::2) 56 data`
	` bytes`
	`64 bytes from 2001:db8:2::2: icmp_seq=1`
	` ttl=64 time=6.52 ms`
	`64 bytes from 2001:db8:2::2: icmp_seq=2`
	` ttl=64 time=0.333 ms`
	`^C`
	`--- 2001:db8:2::2 ping statistics ---`
	`2 packets transmitted, 2 received, 0% packet`
	` loss, time 1013ms`
	`rtt min/avg/max/mdev =`
	` 0.333/3.427/6.522/3.095 ms`

Display IPv6 Neighbor Discovery (ND) cache

To display a list of neighbors in the Neighbor Discovery (ND) caches in both the data plane and the controller, use the `show ipv6 neighbors` command. To display the ND cache in the data plane only use the `show ipv6 neighbors`. Perform the following step in operational mode.

Table 6: Display the ND cache

Step	Command

Display the list of known neighbors in both the data plane and the controller.	`vyatta@R1:~$ show ipv6 neighbors`
	
	`IPv6 Address HW address`
	` Dataplane Controller Device`
	`2001:db8:2::2 52:54:0:9b:6a:3f`
	` VALID [REACHABLE] VALID [REACHABLE] dp0p1p3`
	`fe80::20c:29ff:fe4e:fcb6 52:54:0:9b:6a:3f`
	` VALID [DELAY] dp0p1p3`

Display the list of known neighbors only in the data plane.	`vyatta@R1:~$ show dataplane nd`
	
	`IPv6 Address HW address`
	` Flags State Device`
	`2001:db8:2::2 52:54:0:9b:6a:3f`
	` VALID REACHABLE dp0p1p3`
	`fe80::20c:29ff:fe4e:fcb6 52:54:0:9b:6a:3f`
	` VALID DELAY dp0p1p3`

Clear ND cache

To clear the Neighbor Discovery (ND) cache, use the `reset ipv6 neighbors` command. To clear the ND cache on interface dp0p1p3, perform the following step in operational mode.

Table 7: Clear the ND cache

Step	Command
Clear the list of known neighbors on dp0p1p3.	`vyatta@R1:~$ reset ipv6 neighbors interface dp0p1p3`