



# **CDRouter IPv6 User Guide**

**Version 6.2**



QA Cafe © 2010  
support@qacafe.com

# 1.Introduction

## CDRouter IPv6 Add-on for CDRouter

CDRouter IPv6 adds IPv6 test capabilities to our industry leading CPE and router test platform, CDRouter. CDRouter IPv6 has been designed from the ground up to support the most common IPv6 features found in today’s leading IPv4/IPv6 dual-stack CPE implementations.

CDRouter IPv6 supports dual-stack CPE devices that implement the following IPv6 WAN connection modes:

- Static IPv6
- [RFC 3315](#) DHCPv6, optionally with [RFC 3633](#) prefix delegation
- [RFC 4862](#) stateless address autoconfiguration, optionally with [RFC 3633](#) prefix delegation
- [RFC 3056](#), commonly referred to as “6to4” tunneling
- [RFC 5569](#), commonly referred to as “6rd” tunneling

RFCs 3056 and 5569 are commonly referred to as “6to4” and “6to4”, respectively, and are similar techniques for providing automatic IPv6 connectivity over IPv4 networks. Although drafted first, 6to4 is operationally a subset of 6rd by utilizing a fixed prefix length and globally accessible relay servers.

When CDRouter IPv6 is enabled, CDRouter will create and maintain the necessary IPv6 next-hop gateway or relay server on the WAN. CDRouter will also create one or more IPv6 clients on the LAN that support either [RFC 3315](#) DHCPv6 or IPv6 stateless address autoconfiguration as defined in [RFC 4862](#).

Support for additional IPv6 LAN and WAN modes may be included in future releases of CDRouter IPv6.

CDRouter IPv6 includes a number of test modules designed to verify a wide variety of IPv6 related functionality, as outlined in the following table:

Test Module	Module Description
basic-v6.tcl	Basic IPv6 extension header processing tests
frag-v6.tcl	IPv6 fragmentation tests
ndp.tcl	Neighbor Discovery Protocol and Router Advertisement tests for IPv6 devices
ndp-wan.tcl	Neighbor Discovery Protocol and Router Advertisement tests for the WAN side of IPv6 devices
dhcpv6-c.tcl	DHCPv6 client tests for the WAN side of the router

dhcpv6-pd.tcl	DHCPv6 prefix delegation tests for WAN to LAN IPv6 configuration
dhcpv6-s.tcl	DHCPv6 server tests for the LAN side of the router
6to4.tcl	6to4 tunnel tests for connecting IPv6 hosts over IPv6 networks
6rd.tcl	6rd tunnel tests for connecting IPv6 hosts over IPv6 networks
icmp-v6.tcl	ICMPv6 tests for baseline ICMPv6 not including Neighbor Discovery
firewall-v6.tcl	IPv6 firewall tests including port scans
apps-v6.tcl	Application tests for IPv6
forward-v6.tcl	IPv6 forwarding tests with different packet sizes and directions
scaling-v6.tcl	Scaling tests for maximum number of IPv6 clients and connections (TCP, HTTP, etc.)

CDRouter IPv6 is compatible with all CDRouter or CDRouter Multiport test setups and can be enabled on any existing CDRouter or CDRouter Multiport license. Before IPv6 testing can begin, the license file must be updated to enable CDRouter IPv6. See the [Requirements and License](#) section of this document for more information on enabling IPv6.

Once you have enabled IPv6 support within CDRouter, you can select and run any of the IPv6 specific test modules included with CDRouter IPv6. The IPv6 test modules can be run either individually or as part of a larger test run which includes other CDRouter, CDRouter Multiport, CDRouter IKE, or CDRouter TR-069 test modules.

Like all products in the CDRouter family, CDRouter IPv6 creates all of the necessary WAN-side and LAN-side network elements required to completely simulate an end-to-end IPv6 network within a single test system. CDRouter IPv6 allows the user to easily and efficiently create a variety of IPv6 network scenarios and utilize the array of included functional verification tests to thoroughly explore the IPv6 behavior of the device under test (DUT).

**NOTE:** This document covers the CDRouter IPv6 version bundled with CDRouter 6.2. The Release Notes for CDRouter 6.2 can be found here:

<http://www.qacafe.com/kb/articles/show/306>

## 2. Test Methodology

### Initial Setup

As mentioned earlier, CDRouter IPv6 is compatible with all existing CDRouter or CDRouter Multiport test setups. However, there are a few important caveats that must be taken into consideration prior to performing IPv6 testing with CDRouter IPv6.

1. IPv6 configuration within Linux

We recommend that IPv6 be disabled within the Linux operating system to avoid potential addressing conflicts when using CDRouter IPv6. Please see this [Knowledge Base article](#) for more information on disabling IPv6 within your Linux operating system.

2. Avoid using private IPv4 addresses for the primary IPv4 WAN connection when using 6to4 for IPv6 WAN connectivity

The 6to4 mechanism defined in [RFC 3056](#) relies on translating globally routable 32-bit IPv4 addresses into a unique 48-bit IPv6 prefix. 6to4 is not designed to work with private IPv4 addresses, which should be avoided when configuring and running tests with CDRouter IPv6. Please see [Example 1](#) provided in the Example Configurations section of this document for more information.

### Start-Up

During the start-up procedure, CDRouter IPv6 expects the device being tested to obtain an IPv4 address using any of the standard WAN connection modes supported by CDRouter (DHCP, PPPoE, PPPoA, PPP/T1, L2TP, PPTP, and static IP).

CDRouter will then create one or more IPv6 enabled LAN clients with IPv6 link local addresses. CDRouter's LAN clients can be configured to obtain global IPv6 addresses via DHCPv6 or stateless address autoconfiguration. When DHCPv6 is chosen, CDRouter expects the device under test to be running a DHCPv6 server. Likewise, when stateless address autoconfiguration is used, CDRouter expects the device under test to advertise a valid global prefix on the LAN.

Regardless of which LAN mode is implemented, the prefix used on the LAN is largely dependent on the IPv6 WAN mode that is selected. With certain WAN modes, LAN side prefix information is assumed to be learned or calculated dynamically, whereas with other WAN modes manual configuration of LAN side prefix information is required.

The following WAN configurations require manual configuration of LAN prefix information:

- Static IPv6
- DHCPv6 without prefix delegation
- Autoconfiguration without prefix delegation

Implementations of the following IPv6 WAN modes should dynamically update prefix information on the LAN based on information either learned or calculated from the WAN and/or WAN settings:

- DHCPv6 with prefix delegation
- Autoconfiguration with prefix delegation
- 6rd
- 6to4

If CDRouter's LAN client is successful in obtaining a global IPv6 address, basic Path MTU Discovery will be performed on the LAN and the testing procedure will begin.

## Running

IPv6 support can be enabled within CDRouter by uncommenting the testvar **supportsIPv6** and setting it to a value of "yes".

Eight test modules are included with CDRouter IPv6:

<code>basic-v6.tcl</code>	Basic IPv6 extension header processing tests
<code>frag-v6.tcl</code>	IPv6 fragmentation tests
<code>ndp.tcl</code>	Neighbor Discovery Protocol and Router Advertisement tests for IPv6 devices
<code>ndp-wan.tcl</code>	Neighbor Discovery Protocol and Router Advertisement tests for the WAN side of IPv6 devices
<code>dhcpv6-c.tcl</code>	DHCPv6 client tests for the WAN side of the router
<code>dhcpv6-pd.tcl</code>	DHCPv6 prefix delegation tests for WAN to LAN IPv6 configuration
<code>dhcpv6-s.tcl</code>	DHCPv6 server tests for the LAN side of the router
<code>6to4.tcl</code>	6to4 tunnel tests for connecting IPv6 hosts over IPv6 networks
<code>6rd.tcl</code>	6rd tunnel tests for connecting IPv6 hosts over IPv6 networks
<code>icmpv6.tcl</code>	ICMPv6 tests for baseline ICMPv6 not including Neighbor Discovery
<code>firewall-v6.tcl</code>	IPV6 firewall tests including port scans
<code>apps-v6.tcl</code>	Application tests for IPv6
<code>forward-v6.tcl</code>	IPv6 forwarding tests with different packet sizes and directions
<code>scaling-v6.tcl</code>	Scaling tests for maximum number of IPv6 clients and connections (TCP, HTTP, etc.)

Summaries for all of the test cases included in each of the modules listed above can be found at the following URL:

[http://www.qacafe.com/static/6\\_2/test-summary/test-summary-ipv6.htm](http://www.qacafe.com/static/6_2/test-summary/test-summary-ipv6.htm)

To run only the IPv6 related test modules from the buddy command-line:

```
# buddy -module basic-v6.tcl,ndp.tcl,ndp-wan.tcl,dhcpv6-c.tcl,  
dhcpv6-pd.tcl,dhcpv6-s.tcl,6to4.tcl,6rd.tcl,icmp-  
v6.tcl,firewall-v6.tcl,apps-v6.tcl,forward-v6.tcl,scaling-v6.tcl  
-trace -pt
```

For additional test execution options, see the [CDRouter User Guide](#) or the [CDRouter Quick Start Guide](#).

### 3. Requirements and License

CDRouter IPv6 is available for CDRouter 6.0 and newer releases only.

In order to use CDRouter's IPv6 functionality, your license file must include IPv6 support. For information on upgrading your license to support CDRouter IPv6, please contact [sales@qacafe.com](mailto:sales@qacafe.com). Please follow the instructions from [support@qacafe.com](mailto:support@qacafe.com) when updating your license file to enable CDRouter IPv6.

CDRouter will report the status of all available add-ons during the installation process and during startup. To verify that CDRouter IPv6 is enabled on your system, execute the command `buddy -info` as root and look for the line "IPv6 is enabled" as shown below. If this line is present, CDRouter IPv6 is enabled and ready to use.

```
root@pod5:~# buddy -info

Copyright (c) 2001-2010 by QA Cafe
Built on 2009-10-19 on fourcubed, pktsrc version 6.1 build 12

Using license installed at: /etc/cdr-mp.lic
Registered to: qacafe
Maintenance, Support and Upgrades until: 2010-05-22
Licensed to run: cdr-mp
Buddyweb is enabled
IPv6 is enabled
IKE is enabled
TR69 is enabled
TR69-EDM is enabled
```

## 4. Configuration

### Global IPv6 Configuration

CDRouter IPv6 has several IPv6 specific configuration options, also known as test variables, or “testvars” for short, that define the DUT and the test environment.

The following tables describe each available testvar. The tables below group testvars by the test interface they are applicable to or the protocol/functionality they define.

To globally enable IPv6 within CDRouter, the testvar **supportsIpv6** must be set to a value of “yes”.

Testvar	Testvar Description
<b>supportsIpv6</b>	The testvar <b>supportsIpv6</b> enables or disables IPv6 functionality within CDRouter. If enabled, CDRouter will create a new client on the LAN and attempt to obtain an IPv6 address from the DUT using the protocol specified by the testvar <b>ipv6LanMode</b> . This testvar should be set to “yes” to enable IPv6 functionality within CDRouter.

## IPv6 WAN Configuration

The following testvars describe the basic IPv6 configuration of the WAN interface.

IPv6 WAN Configuration	
<b>ipv6WanMode</b>	<p>The testvar <b>ipv6WanMode</b> specifies the IPv6 WAN protocol configured on the DUT. CDRouter will create and simulate all necessary network elements based on the chosen WAN protocol.</p> <p>CDRouter currently supports static IPv6, DHCPv6, autoconfiguration, 6rd tunneling, and 6to4 tunneling. Support for additional IPv6 WAN modes will be added to future releases of CDRouter.</p> <p>Valid values for the <b>ipv6WanMode</b> testvar are “static”, “DHCP”, “autoconf”, “6rd”, and “6to4”.</p> <p>Examples:</p> <pre>testvar ipv6WanMode static testvar ipv6WanMode DHCP testvar ipv6WanMode autoconf testvar ipv6WanMode 6rd testvar ipv6WanMode 6to4</pre>
<b>ipv6WanIspIp</b>	<p>The testvar <b>ipv6WanIspIp</b> specifies the ISP's IPv6 address. CDRouter will create a node at this address which represents the IPv6 next-hop gateway for the DUT. This testvar is only used if the <b>ipv6WanMode</b> testvar is set to “static”.</p> <p>Example:</p> <pre>testvar ipv6WanIspIp 3001::1</pre>
<b>ipv6WanIspAssignIp</b>	<p>The testvar <b>ipv6WanIspAssignIp</b> specifies the DUT's IPv6 address. This testvar is only used if the <b>ipv6WanMode</b> testvar is set to “static”.</p> <p>Example:</p> <pre>testvar ipv6WanIspAssignIp 3001::2</pre>

## IPv6 WAN Configuration

<b>ipv6WanIspNextIp</b>	<p>The testvar <b>ipv6WanIspNextIp</b> specifies a free address on the ISP's network. This address is only used during WAN renumbering tests.</p> <p>Example:</p> <pre>testvar ipv6WanIspNextIp 3001::3</pre>
<b>ipv6WanIspPrefixLen</b>	<p>The testvar <b>ipv6WanIspPrefixLen</b> specifies the WAN side IPv6 address prefix length. This must be an integer value between 1 and 128.</p> <p>Example:</p> <pre>testvar ipv6WanIspPrefixLen 64</pre>
<b>ipv6RelayServer</b>	<p>The testvar <b>ipv6RelayServer</b> specifies the IPv4 address of the 6rd or 6to4 relay server created by CDRouter on the WAN. For 6to4 configurations this address is typically the well known anycast address 192.88.99.1. For 6rd configurations this address must be manually configured and must match the relay server address used by the DUT.</p> <p>This testvar is only used if the <b>ipv6WanMode</b> testvar is set to "6rd" or "6to4".</p> <p>Example:</p> <pre>testvar ipv6RelayServer 192.88.99.1</pre>
<b>ipv6rdPrefix</b>	<p>The testvar <b>ipv6rdPrefix</b> specifies the DUT's statically configured 6rd prefix. This testvar is represented as a 16 octet IPv6 address with all bits following the 6rd prefix length set to 0.</p> <p>Example:</p> <pre>testvar ipv6rdPrefix 2010:1234::</pre>

## IPv6 WAN Configuration

<b>ipv6rdPrefixLen</b>	<p>The testvar <b>ipv6rdPrefixLen</b> specifies the length of the DUT's 6rd prefix. The prefix length must be an integer value between 1 and 128. By default the 6rd prefix length is 32.</p> <p>Example:</p> <pre>testvar ipv6rdPrefixLen 32</pre>
<b>ipv6rdIPv4MaskLen</b>	<p>The testvar <b>ipv6rdIPv4MaskLen</b> specifies the number of high order bits that are identical across all IPv4 addresses within a 6rd domain. If the IPV4 mask length is 0, the DUT's entire WAN IPV4 address is used to create the 6rd delegated prefix. If the DUT's 6rd mask length is 8, only the lowest order 24 bits of the DUT's WAN IPV4 address are used to create the 6rd prefix. The IPv4 mask length must be an integer value between 0 and 32. By default the PV4 mask length is 0.</p> <p>Example:</p> <pre>testvar ipv6rdIPv4MaskLen 0</pre>
<b>ipv6WanIspGateway</b>	<p>If a DHCPv6 relay is present on the WAN, the testvar <b>ipv6WanIspGateway</b> must be configured. This testvar should be configured to match the IPv6 address of the interface on the relay that is connected to CDRouter's WAN. This testvar should only be used when relay is present on the WAN between CDRouter and the DUT.</p>
<b>ipv6ExpectRAonWan</b>	<p>If the DUT is configured to send router advertisements on the WAN, the testvar <b>ipv6ExpectRAonWan</b> should be set to "yes". If this testvar is disabled, the test cases <code>ipv6_ndp_wan_1</code> and <code>ipv6_ndp_wan_2</code> will be automatically skipped by CDRouter. This testvar defaults to "yes".</p> <p>Example:</p> <pre>testvar ipv6ExpectRAonWan no</pre>

## WAN-Side DHCPv6 Server Configuration

The following testvars relate to the general configuration of CDRouter's DHCPv6 server on the WAN.

### IPv6 DHCPv6 Server Configuration

<b>dhcpv6IAValidLifetime</b>	<p>The testvar <b>dhcpv6IAValidLifetime</b> defines the IA Valid Lifetime, in seconds, utilized by CDRouter's DHCPv6 server on the WAN. CDRouter's DHCPv6 server will provide this Valid Lifetime to the DUT's DHCPv6 on client on the WAN.</p> <p>To return an infinite Valid Lifetime to the WAN client, set this testvar to a value of "infinite". If an infinite lifetime is used, all tests that require the WAN client to renew will be skipped.</p> <p>Note that the Valid Lifetime must be greater than or equal to the Preferred Lifetime.</p> <p>Example:</p> <pre>testvar dhcpv6IAValidLifetime 300</pre>
<b>dhcpv6IAPreferredLifetime</b>	<p>The testvar <b>dhcpv6IAPreferredLifetime</b> defines the IA Preferred Lifetime, in seconds, utilized by CDRouter's DHCPv6 server on the WAN. CDRouter's DHCPv6 server will provide this Preferred Lifetime to the DUT's DHCPv6 on client on the WAN.</p> <p>To return an infinite Preferred Lifetime to the WAN client, set this testvar to a value of "infinite". If an infinite lifetime is used, all tests that require the WAN client to renew will be skipped.</p> <p>Note that the Preferred Lifetime must be less than or equal to the Valid Lifetime.</p> <p>Example:</p> <pre>testvar dhcpv6IAPreferredLifetime 300</pre>

## IPv6 DHCPv6 Server Configuration

<b>dhcpv6DuidType</b>	<p>The testvar <b>dhcpv6DuidType</b> specifies the DUID format utilized by CDRouter's DHCPv6 server and clients. The DUID type can be specified as Type 1 (DUID-LLT), Type 2 (DUID-EN), or Type 3 (DUID-LL) as defined in IETF RFC 3315.</p> <p>Example:</p> <pre>testvar dhcpv6DuidType 1</pre>
<b>dhcpv6ServerOptionCode&lt;N&gt;</b> <b>dhcpv6ServerOptionData&lt;N&gt;</b>	<p>CDRouter's DHCPv6 server on the WAN may be configured to return specific options to the DUT's DHCPv6 client. Up to 255 additional DHCPv6 options may be defined. The <b>dhcpv6ServerOptionCode&lt;N&gt;</b> entry should specify the DHCP option code in decimal format. The &lt;N&gt; value should be replaced with the entry number such as 1, 2, 3, etc.</p> <p>The <b>dhcpv6ServerOptionData&lt;N&gt;</b> testvar is used to define the option data for any additional DHCPv6 options configured on the WAN side DHCPv6 server. The option data must be encoded in ascii hex format. The option length will be computed automatically based on the size of the data. The &lt;N&gt; value should be replaced with the entry number such as 1, 2, 3, etc.</p> <p>Example:</p> <pre>testvar dhcpv6ServerOptionCode1 69 testvar dhcpv6ServerOptionData1 04040404 testvar dhcpv6ServerOptionCode2 71 testvar dhcpv6ServerOptionData2 04040405</pre>

## WAN-Side DHCPv6 Prefix Delegation Configuration

The following testvars relate to the configuration of DHCPv6 prefix delegation within CDRouter's WAN DHCPv6 server.

### IPv6 DHCPv6 Prefix Delegation Configuration

<b>dhcpv6WanEnablePD</b>	<p>The testvar <b>dhcpv6WanEnablePD</b> specifies whether or not the DUT supports DHCPv6 prefix delegation, as defined in RFC 3633, on the WAN.</p> <p>If prefix delegation support is enabled, CDRouter will respond to DHCPv6 IA_PD option requests from the DUT.</p> <p>Example:</p> <pre>testvar dhcpv6WanEnablePD yes</pre>
<b>dhcpv6WanAssignPrefix</b>	<p>The testvar <b>dhcpv6WanAssignPrefix</b> defines the IPv6 address prefix CDRouter's DHCPv6 server will assign to the DUT's DHCPv6 client on the WAN. It is expected that the DUT will use this prefix as the basis for its LAN side configuration.</p> <p>Example:</p> <pre>testvar dhcpv6WanAssignPrefix 3001:dddd::</pre>
<b>dhcpv6WanAssignNextPrefix</b>	<p>The testvar <b>dhcpv6WanAssignNextPrefix</b> specifies a free IPv6 address prefix on the WAN ISP network. This prefix is only used in renumbering tests.</p> <p>Example:</p> <pre>testvar   dhcpv6WanAssignNextPrefix 3001:dde::</pre>
<b>dhcpv6WanAssignPrefixLen</b>	<p>The testvar <b>dhcpv6WanAssignPrefixLen</b> specifies the IPv6 address prefix length assigned by CDRouter's DHCPv6 server on the WAN to the DUT's DHCPv6 client.</p> <p>Example:</p> <pre>testvar dhcpv6WanAssignPrefixLen 64</pre>

### IPv6 DHCPv6 Prefix Delegation Configuration

---

**dhcipv6PDLatency**

This parameter defines the expected maximum amount of time, in seconds, it should take the DUT to update its LAN-side IPv6 settings when a new prefix is learned on the WAN.

Example:

```
testvar dhcipv6PDLatency 30
```

---

### IPv6 LAN Configuration

The following testvars describe the basic IPv6 configuration of the LAN interface.

### IPv6 LAN Configuration

---

**ipv6LanMode**

The testvar **ipv6LanMode** specifies the protocol used by CDRouter to obtain IPv6 addresses for clients on the LAN. CDRouter currently supports DHCPv6 and stateless address autoconfiguration only. Support for additional IPv6 LAN modes will be added to future releases of CDRouter.

Valid values for the **ipv6LanMode** testvar are “DHCP” and “autoconf”.

Examples:

```
testvar ipv6LanMode autoconf
testvar ipv6LanMode DHCP
```

---

## IPv6 LAN Configuration

<b>ipv6LanIp</b>	<p>This testvar specifies the DUT's expected LAN side global IPv6 address.</p> <p>The keyword “%eui64%” can be used for the ID portion of this address. When “%eui64%” is used, CDRouter will automatically calculate the expected host ID portion of the DUT's LAN IPv6 address based on the EUI64 representation of the DUT's LAN MAC address.</p> <p>For IPv6 WAN modes where the prefix is deterministic, such as “6to4” or “6rd”, or when DHCPv6 prefix delegation is enabled, the prefix portion of this address may be omitted. Note that this format is suitable for 6to4, 6rd, or prefix delegation configurations only.</p> <p>Examples:</p> <pre>testvar ipv6LanIp 2004::1 testvar ipv6LanIp 2004::%eui64% testvar ipv6LanIp ::%eui64%</pre>
<b>ipv6LanPrefixLen</b>	<p>This testvar specifies the prefix length of the DUT's LAN side global IPv6 address. This value should match the value configured on the DUT and is only used in conjunction with the <b>ipv6LanIp</b> in situations where the <b>ipv6WanMode</b> is not set to “6to4” and must be manually configured.</p> <p>The prefix length must be an integer value between 8 and 64.</p> <p>Example:</p> <pre>testvar ipv6LanPrefixLen 64</pre>
<b>ipv6LanInterfaceId</b>	<p>Deprecated. This testvar is no longer supported. Please use the testvar <b>ipv6LanIp</b> instead.</p>

## IPv6 LAN Configuration

<b>ipv6MaxLanClients</b>	<p>The testvar <b>ipv6MaxLanClients</b> specifies the maximum number of IPv6 LAN clients CDRouter will attempt to create during the scaling tests. The default value is 10.</p> <p>If the <b>ipv6LanMode</b> is set to “DHCP”, the DHCPv6 client pool will be used to determine the maximum number of LAN clients to create during the scaling-v6.tcl test module if the pool size is less than <b>ipv6MaxLanClients</b>. Alternatively, if the DHCPv6 pool size is greater than <b>ipv6MaxLanClients</b>, the number of LAN clients created during the scaling-v6.tcl test module will be capped at <b>ipv6MaxLanClients</b>.</p> <p>Note that the maximum number of clients that can be created during the scaling-v6.tcl test module is 512.</p> <p>Example:</p> <pre>testvar ipv6MaxLanClients 10</pre>
<b>ipv6RAInterval</b>	<p>The testvar <b>ipv6RAInterval</b> specifies the maximum expected Router Advertisement interval supported by the DUT. CDRouter will wait up to this number of seconds for Router Advertisements on the LAN. The default value is 300 seconds.</p> <p>Example:</p> <pre>testvar ipv6RAInterval 300</pre> <p>If the value of <b>ipv6RAInterval</b> does not match the actual Router Advertisement interval on the device, CDRouter may incorrectly fail test cases. If you are unsure of the Router Advertisement interval, try setting this testvar to a large value and running test case <b>ipv6_ndp_2</b> to see the actual interval implemented by the DUT.</p>

## IPv6 LAN Configuration

<b>ipv6LanSubnetId</b>	<p>When 6to4 and 6rd are used on the WAN, the subnet ID portion the DUT's LAN IPv6 prefix must be specified. In 6to4 terminology the subnet ID is referred to as the SLA ID and is fixed at 16 bits. As a result, this parameter should be four or fewer hex characters when 6to4 is used. In 6rd the subnet ID field can be variable length.</p> <p>The test cases <b>ipv6_6to4_3</b> and <b>ipv6_6rd_3</b> can be used to verify that the DUT is using the expected subnet ID.</p> <p>Example:</p> <pre>testvar ipv6LanSubnetId ffff</pre>
<b>dhcpv6ClientTimeout</b>	<p>The testvar <b>dhcpv6ClientTimeout</b> specifies the amount of time that CDRouter's DHCPv6 client(s) on the LAN wait to receive an address from the DUT's DHCPv6 server. Normally, this timeout should not be modified. If you are testing a slow DHCPv6 server, you may need to increase this timeout. The timeout value is in seconds.</p> <p>Example:</p> <pre>testvar dhcpv6ClientTimeout 5</pre>

## IPv6 LAN Configuration

<b>ipv6DhcpClientStart</b>	<p>The testvar <b>ipv6DhcpClientStart</b> specifies the host identifier portion of the first IPv6 address in the DUT's DHCPv6 address pool. This value should match the value configured on the DUT.</p> <p>This testvar is only used if the <b>ipv6LanMode</b> testvar is set to "DHCP".</p> <p>CDRouter will automatically determine the correct prefix for this address based on the IPv6 WAN mode being used. If the <b>ipv6WanMode</b> is set to "6to4", CDRouter will use the expected 6to4 prefix for the prefix portion of the DHCPv6 address pool. If the <b>ipv6WanMode</b> is set to "static", CDRouter will calculate the IPv6 prefix based on the testvars <b>ipv6LanIp</b> and <b>ipv6LanPrefixLen</b> (the DHCPv6 pool should be on the same network as <b>ipv6LanIp</b>).</p> <p>Example:</p> <pre>testvar ipv6DhcpClientStart ::2</pre>
<b>ipv6DhcpClientEnd</b>	<p>The testvar <b>ipv6DhcpClientEnd</b> specifies the host identifier portion of the final IPv6 address in the DUT's DHCPv6 address pool. This value should match the value configured on the DUT.</p> <p>This testvar is only used if the <b>ipv6LanMode</b> testvar is set to "DHCP".</p> <p>CDRouter will automatically determine the correct prefix for this address based on the IPv6 WAN mode being used. If the <b>ipv6WanMode</b> is set to "6to4", CDRouter will use the expected 6to4 prefix for the prefix portion of the DHCPv6 address pool. If the <b>ipv6WanMode</b> is set to "static", CDRouter will calculate the IPv6 prefix based on the testvars <b>ipv6LanIp</b> and <b>ipv6LanPrefixLen</b> (the DHCPv6 pool should be on the same network as <b>ipv6LanIp</b>).</p> <p>Example:</p> <pre>testvar ipv6DhcpClientEnd ::1F</pre>

## IPv6 LAN Configuration

<b>ipv6DhcpClientExclude</b>	<p>The testvar <b>ipv6DhcpClientExclude</b> specifies a list of host identifiers for IPv6 addresses that are excluded from the DHCPv6 pool as specified by the testvars <b>ipv6DhcpClientStart</b> and <b>ipv6DhcpClientEnd</b>.</p> <p>This testvar is only used if the <b>ipv6LanMode</b> testvar is set to "DHCP".</p> <p>All excluded addresses must be contained within the specified DHCPv6 pool.</p> <p>Example:</p> <pre>testvar ipv6DhcpClientExclude "::3 ::7"</pre>
<b>dhcpv6RapidCommit</b>	<p>This testvar specifies whether or not the DUT's DHCPv6 server on the LAN supports DHCPv6 option 14 (Rapid Commit).</p> <p>The test case <b>dhcpv6_server_102</b> will only be executed if support for Rapid Commit functionality is indicated with this testvar.</p> <p>Example:</p> <pre>testvar dhcpv6RapidCommit yes</pre>
<b>dhcpv6ClientOptionRequest</b>	<p>The testvar <b>dhcpv6ClientOptionRequest</b> can be used to request extra options from the DUT's DHCPv6 server on the LAN. If configured, CDRouter's LAN DHCPv6 client(s) will request these specific options from the server.</p> <p>Example, to request DHCPv6 options 21 and 22 from the server:</p> <pre>testvar dhcpv6ClientOptionRequest "21 22"</pre>

## IPv6 LAN Configuration

---

**dhcpv6ClientOptionCode<N>**  
**dhcpv6ClientOptionData<N>**

CDRouter's DHCPv6 LAN clients may be configured with up to 10 additional options using the **dhcpv6ClientOptionCode<N>** and **dhcpClientOptionData<N>** testvars. Client side options are configured the same way as the existing DHCPv6 server side options.

Each DHCPv6 option is configured using a testvar entry for the option code and another for the raw hex data. The DHCP option data must be converted into the raw ascii hex format. The total option length will be computed automatically.

Example, to add DHCPv6 option 124 and DHCP option 125:

```
testvar
dhcpv6ClientOptionCode1 124
testvar dhcpv6ClientOptionData1
000000000401020304

testvar dhcpv6ClientOptionCode2 125
testvar dhcpv6ClientOptionData2
000000000401020304
```

---

## IPv6 DNS Configuration

The following testvars relate to the basic IPv6 DNS configuration of the test environment.

IPv6 DNS Configuration	
<b>ipv6DNStoLAN</b>	<p>The testvar <b>ipv6DNStoLAN</b> specifies whether or not the DUT supports IPv6 DNS relay functionality.</p> <p>This testvar should be enabled if the DUT is configured to relay the IPV6 DNS servers on the WAN to clients on the LAN via DHCPv6 options or the Neighbor Discovery RDNSS option.</p> <p>This testvar should be disabled if the DUT announces its LAN side IPv6 address as the DNS server to all clients on the LAN.</p> <p>Example:</p> <pre>testvar ipv6DNStoLAN yes</pre>
<b>ipv6WanDnsServer</b>	<p>The testvar <b>ipv6WanDnsServer</b> is used to specify the IPv6 address of the primary DNS server created by CDRouter on the WAN.</p> <p>By default the IPv6 address 3001:51a:cafe::2 is used. Note that CDRouter's primary and backup DNS servers are dual-stack servers with both IPv6 and IPv4 addresses. The IPv4 address of the primary DNS server is configured using the test testvar <b>wanDnsServer</b>.</p> <p>Example:</p> <pre>testvar ipv6WanDnsServer 3001:51a:cafe::2</pre>

## IPv6 DNS Configuration

<b>ipv6WanBackupDnsServer</b>	<p>The testvar <b>ipv6WanBackupDnsServer</b> is used to specify the IPv6 address of the backup DNS server created by CDRouter on the WAN.</p> <p>By default the IPv6 address 3001:51a:cafe::3 is used. Note that CDRouter's primary and backup DNS servers are dual-stack servers with both IPv6 and IPv4 addresses. The IPv4 address of the backup DNS server is configured using the test testvar <b>wanDnsServer</b>.</p> <p>Example:</p> <pre>testvar ipv6WanBackupDnsServer 3001:51a:cafe::3</pre>
<b>ipv6RdnssSupport</b>	<p>The test var <b>ipv6RdnssSupport</b> specifies whether or not the DUT supports the Neighbor Discovery Protocol router advertisement option 25 (Recursive DNS Server) as defined in RFC 5006. If the DUT does not support the RDNSS option in router advertisements on the LAN, this testvar should be set to "no".</p> <p>Example:</p> <pre>testvar ipv6RdnssSupport yes</pre>

## IPv6 DNS Configuration

<b>dnsHostname&lt;N&gt;</b>	<p>The testvar <b>dnsHostname&lt;N&gt;</b> now supports configuration of specific DNS entries for IPv6 hosts. CDRouter will automatically populate all WAN side DNS servers with these user defined DNS mappings. Up to 100 DNS mappings may be defined.</p> <p>The <b>dnsHostname&lt;N&gt;</b> testvar is used to define the hostname for an entry. The <b>dnsIp&lt;N&gt;</b> testvar is used define multiple IPv4 and/or IPv6 addresses for an entry. The &lt;N&gt; should be replaced with the entry number such as 1, 2, 3, etc.</p> <p>Examples:</p> <pre>testvar dnsHostname1 foo.test.com testvar dnsIp1 5.5.5.5 testvar dnsHostname2 ipv6host.test.com testvar dnsIp2 3001:4860:0:2222::68 testvar dnsHostname3 test.qacafe.com testvar dnsIp3 "5.5.5.5 3001:cafe::1 7.7.7.7"</pre>
<b>dnsIp&lt;N&gt;</b>	<p>The <b>dnsIp&lt;N&gt;</b> entry is used to define multiple IPv4 and/or IPv6 addresses associated with a DNS hostname. The &lt;N&gt; should be replaced with the entry number such as 1, 2, 3, etc. Lists of addresses should be placed in quotes. See <b>dnsHostname&lt;N&gt;</b> for an example.</p>
<b>dnsLookupTimeout</b>	<p>This testvar specifies the DNS lookup timeout. The default value of 5 seconds can be adjusted if the DUT's DNS proxy implementation is slot to respond to DNS queries.</p> <p>Example:</p> <pre>testvar dnsLookupTimeout 10</pre>

## IPv6 General Purpose Network Configuration

The following testvars define the CDRouter's IPv6 remote host address and the IPv6 free network range.

### IPv6 General Purpose Network Configuration

<b>ipv6RemoteHost</b>	<p>The testvar <b>ipv6RemoteHost</b> specifies the IP address of an IPv6-enabled remote host created by CDRouter on the WAN side of the network. This host will be used for verification of LAN to WAN and WAN to LAN traffic through the DUT, and is commonly referred to as the IPv6 Remote Host. The default address for the IPv6 Remote Host is shown in the example below.</p> <p>Example:</p> <pre>testvar ipv6RemoteHost 3001:51a:cafe::1</pre>
<b>ipv6FreeNetworkStart</b> <b>ipv6FreeNetworkEnd</b> <b>ipv6FreeNetworkPrefixLen</b>	<p>These testvars are used to configure a range of free IPv6 addresses available to CDRouter. At certain points during a test case, addresses from this range may be allocated to a test case. These addresses should not conflict with any IPv6 addresses that are configured and should not be within the designated 6to4 prefix range (2002::/16).</p> <p>Example:</p> <pre>testvar ipv6FreeNetworkStart 3001:cafe:1:: testvar ipv6FreeNetworkEnd 3001:ffff:ffff:: testvar ipv6FreeNetworkPrefixLen 64</pre>

## IPv6 Firewall Configuration

The following testvars define the configuration of the DUT's IPv6 firewall and the associated test environment.

### IPv6 Firewall Configuration

---

**ipv6FirewallTcpClosedPorts**  
**ipv6FirewallUdpClosedPorts**

By default, all TCP and UDP ports in the IPv6 firewall are assumed to be operating in “stealth” mode. Any TCP or UDP ports that are closed may be configured in the **ipv6FirewallTcpClosedPorts** or **ipv6FirewallUdpClosedPorts** lists. The keyword “all” may also be specified for devices that default to closing all TCP or UDP IPv6 ports. Port ranges are supported with a hyphen: 22-24 signifies 22, 23, and 24.

Example:

```
testvar ipv6FirewallTcpClosedPorts "80 8080"  
testvar ipv6FirewallUdpClosedPorts 113  
testvar ipv6FirewallUdpClosedPorts "22-24"
```

Note that the range of ports scanned in the firewall-v6.tcl test module can be configured using the **portScanStart** and **portScanStop** testvars. Please see the [CDRouter User's Guide](#) for more information.

---

**ipv6FirewallTcpOpenPorts**  
**ipv6FirewallUdpOpenPorts**

Any TCP or UDP ports in the IPv6 firewall that are open may be configured in the **ipv6FirewallTcpOpenPorts** or **ipv6FirewallUdpOpenPorts** lists. Port ranges are supported with a hyphen: 22-24 signifies 22, 23, and 24.

Example:

```
testvar ipv6FirewallTcpOpenPorts "22 443"  
testvar ipv6FirewallUdpOpenPorts 161  
testvar ipv6FirewallUdpOptnPorts "22-24"
```

Note that the range of ports scanned in the firewall-v6.tcl test module can be configured using the **portScanStart** and **portScanStop** testvars. Please see the [CDRouter User's Guide](#) for more information.

---

---

**IPv6 Firewall Configuration**

---

**ipv6WanPingRespond**

The testvar **ipv6WanPingRespond** indicates whether or not the DUT will respond to ICMPv6 Echo Requests on its WAN interface. The default value is 'no', indicating the DUT will not answer a ping from the WAN.

Example:

```
testvar ipv6WanPingRespond yes
```

---

## 5.Example Configurations

In this section eight basic example configurations for CDRouter IPv6 are provided. The examples provided cover the following modes of operation:

- Example 1: 6to4 on the WAN and autoconf on the LAN
- Example 2: 6to4 on the WAN and DHCPv6 on the LAN
- Example 3: Static IPv6 on the WAN and autoconf on the LAN
- Example 4: Static IPv6 on the WAN and DHCPv6 on the LAN
- Example 5: DHCPv6 without prefix delegation on the WAN and autoconf on the LAN
- Example 6: DHCPv6 with prefix delegation on the WAN and DHCPv6 on the LAN
- Example 7: Autoconf on the WAN and DHCPv6 on the LAN
- Example 8: 6rd on the WAN and autoconf on the LAN

### Example 1: 6to4 on the WAN and autoconf on the LAN

In this example, CDRouter is configured to create a basic dual-stack IPv4/IPv6 test setup for the DUT using 6to4 for IPv6 connectivity on the WAN and autoconfiguration for IPv6 address assignment on the LAN. This is the most basic IPv6 test setup and requires minimal configuration of CDRouter and the DUT.

From an IPv4 perspective, CDRouter will create a DHCP server on the WAN which will terminate the DUT's WAN DHCP connection. CDRouter is configured to provide the global IPv4 address 12.12.12.2 and DNS servers 1.1.1.1 and 1.1.1.2 to the DUT. The DUT is also configured with a LAN IPv4 address of 192.168.1.1 and a DHCP pool range of 192.168.1.2 through 192.168.2.7. CDRouter is configured to match these LAN-side settings on the DUT. Please see the [CDRouter User's Guide](#) for more information on the IPv4 specific configuration options listed below.

From an IPv6 perspective, IPv6 is enabled and it is expected that the DUT will provide IPv6 connectivity to all connected clients via an automatic 6to4 tunnel on the WAN. As a result, CDRouter is configured to create a 6to4 relay server at the default anycast address of 192.88.99.1 on the WAN.

The DUT's global IPv6 address is expected to be <6to4 prefix>:<ipv6LanSubnetId>:<host ID portion of ipv6LanIp>. In this particular example, assuming the 16-bit Site Level Aggregator (SLA) used by the DUT is simply 1, the expected global IPv6 address of the DUT is:

**<6to4 prefix>:<ipv6LanSubnetId>:<host ID portion of ipv6LanIp> =  
2002:c0c:c02:1::1**

CDRouter is also configured to use IPv6 stateless address autoconfiguration on the LAN for all clients that it creates. CDRouter will create 100 LAN clients during the scaling tests.

This example also assumes that the DUT supports the NDP RDNSS option (25) and is configured to provide its global IP address as the DNS server to all clients on the LAN.

#### IPv4 Configuration:

```
testvar wanInterface eth2
testvar wanMode DHCP
testvar wanIspIp 12.12.12.1
testvar wanIspAssignIp 12.12.12.2
testvar wanNatIp 12.12.12.2
testvar wanIspNextIp 12.12.12.254
testvar wanIspMask 255.255.255.0
testvar wanMac 00:0b:0b:00:00:01
testvar wanDnsServer 1.1.1.1
testvar wanBackupDnsServer 1.1.1.2

testvar lanInterface eth1
testvar lanIp 192.168.1.1
testvar lanMask 255.255.255.0
testvar lanMac 00:0a:0a:00:00:01
testvar dhcpClientStart 192.168.1.2
testvar dhcpClientEnd 192.168.1.7
```

#### IPv6 Configuration:

```
testvar supportsIPv6 yes

# -- WAN configuration
testvar ipv6WanMode 6to4
testvar ipv6RelayServer 192.88.99.1

# -- LAN configuration
testvar ipv6LanMode autoconf
testvar ipv6LanIp ::1
testvar ipv6LanSubnetId 1
testvar ipv6SupportsRdnss yes
testvar ipv6DNStoLAN no
testvar ipv6RAInterval 300

# -- other configuration options
testvar ipv6RemoteHost 3001:51a:cafe::1
testvar ipv6MaxLanClients 100
testvar ipv6FreeNetworkStart 3001:cafe:1::
testvar ipv6FreeNetworkEnd 3001:ffff:ffff::
testvar ipv6FreeNetworkPrefixLen 64
```

#### IPv6 Firewall Configuration:

```
testvar ipv6FirewallTcpClosedPorts "80 8080"
testvar ipv6FirewallUdpClosedPorts all
testvar ipv6FirewallTcpOpenPorts "443"
testvar ipv6FirewallUdpOpenPorts 113
testvar ipv6WanPingRespond no
```

Note that the range of ports scanned in the firewall-v6.tcl test module can be configured using the **portScanStart** and **portScanStop** testvars. Please see the [CDRouter User's Guide](#) for more information.

### Example 2: 6to4 on the WAN and DHCPv6 on the LAN

In this example, the IPv4 configuration and IPv6 firewall configuration are identical to those provided in Example 1 above.

The primary difference between this configuration and the configuration provided in Example 1 is the IPv6 LAN mode. As in Example 1, the DUT will provide IPv6 connectivity to all connected clients via an automatic 6to4 tunnel on the WAN. As a result, CDRouter is configured to create a 6to4 relay server at 192.88.99.1 on the WAN. The DUT's global IPv6 address is expected to be <6to4 prefix>:<ipv6LanSubnetId>:<host ID portion of ipv6LanIp>. In this particular example, assuming the 16-bit Site Level Aggregator (SLA) used by the DUT is simply 1, the expected global IPv6 address of the DUT is again:

**<6to4 prefix>:<ipv6LanSubnetId>:<host ID portion of ipv6LanIp> =  
2002:c0c:c02:1::1**

Likewise, the DHCPv6 pool range of the DUT is expected to be:

**<6to4 prefix>:<ipv6LanSubnetId>:<ipv6DhcpClientStart> = 2002:c0c:c02:1::2  
<6to4 prefix>:<ipv6DhcpClientEnd> = 2002:c0c:c02:1::1F**

With the following LAN clients excluded from the DHCPv6 pool:

**<6to4 prefix>:<ipv6LanSubnetId>:<ipv6DhcpClientExclude> = 2002:c0c:c02:1::3,  
2002:c0c:c02:1::7**

In this example, 28 LAN clients (the size of the DHCPv6 pool) will be created during the scaling tests.

This example also assumes that the DUT does not support the NDP RDNSS option (25) and that the DUT is configured to provide the specific IPv6 DNS server addresses 3001:51:cafe::2 and 3001:51:cafe::3 to clients on the LAN via DHCPv6 options.

IPv6 Configuration:

```
testvar supportsIPv6 yes

# -- WAN configuration
testvar ipv6WanMode 6to4
testvar ipv6RelayServer 192.88.99.1

# -- LAN configuration
testvar ipv6LanMode DHCP
```

```

testvar ipv6LanIp ::1
testvar ipv6LanSubnetId 1
testvar ipv6DhcpClientStart ::2
testvar ipv6DhcpClientEnd ::1F
testvar ipv6DhcpClientExclude ":::3 :::7"
testvar ipv6SupportsRdnss no
testvar ipv6DNStoLAN yes
testvar ipv6RAInterval 300

# -- other configuration options
testvar ipv6RemoteHost 3001:51a:cafe::1
testvar ipv6WanDnsServer 3001:51a:cafe::2
testvar ipv6WanBackupDnsServer 3001:51:cafe::3
testvar ipv6MaxLanClients 100
testvar ipv6FreeNetworkStart 3001:cafe:1::
testvar ipv6FreeNetworkEnd 3001:ffff:ffff::
testvar ipv6FreeNetworkPrefixLen 64

```

### Example 3: Static IPv6 on the WAN and autoconf on the LAN

In this example, the IPv4 configuration and IPv6 firewall configuration are identical to those provided in Example 1 above.

The primary difference between this configuration and the configuration provided in Example 1 is the IPv6 WAN mode. In this example, it is expected that the DUT will provide IPv6 connectivity to all connected clients via a static IPv6 connection on the WAN. The DUT's static IPv6 address is 3001::2, and CDRouter's IPv6 address (representing the ISP next-hop gateway) is 3001::1. CDRouter will create IPv6 DNS servers at the addresses 3001:51a:cafe::2 and 3001:51a:cafe::3.

On the LAN side, the DUT is configured for autoconf. Since the IPv6 WAN mode is static, the DUT's global LAN IPv6 address must be manually configured. In this example the DUT's global LAN IPv6 address is configured to 2004::1 with a prefix length of 64. This example assumes that IPv6 DNS information will be provided to clients on the LAN via DHCPv6 information requests.

In this example, 100 LAN clients will be created during the scaling tests.

IPv6 Configuration:

```

testvar supportsIPv6 yes

# -- WAN configuration
testvar ipv6WanMode static
testvar ipv6WanIspIp 3001::1
testvar ipv6WanIspAssignIp 3001::2

# -- LAN configuration
testvar ipv6LanMode autoconf
testvar ipv6LanIp 2004::1
testvar ipv6LanPrefixLen 64
testvar ipv6DNStoLAN yes
testvar ipv6RAInterval 300

```

```
# -- other configuration options
testvar ipv6RemoteHost 3001:51a:cafe::1
testvar ipv6WanDnsServer 3001:51a:cafe::2
testvar ipv6WanBackupDnsServer 3001:51a:cafe::3
testvar ipv6MaxLanClients 100
testvar ipv6FreeNetworkStart 3001:cafe:1::
testvar ipv6FreeNetworkEnd 3001:ffff:ffff::
testvar ipv6FreeNetworkPrefixLen 64
```

#### Example 4: Static IPv6 on the WAN and DHCPv6 on the LAN

In this example, the IPv4 configuration and IPv6 firewall configuration are identical to those provided in Example 1 above. Additionally, both the IPv6 WAN mode and IPv6 LAN mode differ from that of Example 1.

On the WAN side, it is expected that the DUT will provide IPv6 connectivity to all connected clients via a static IPv6 connection. In this example, the DUT's static IPv6 address is 3001::2, and CDRouter's IPv6 address (representing the ISP next-hop gateway) is 3001::1. CDRouter will create IPv6 DNS servers at the addresses 3001:51a:cafe::2 and 3001:51a:cafe::3.

On the LAN side, the DUT's global IPv6 address is configured to be 2004::1 with a prefix length of 64. The LAN side prefix is automatically determined by CDRouter based on these values.

Any LAN clients created by CDRouter are configured to obtain global IPv6 addresses via DHCPv6. The configured DHCPv6 pool is ::2 through ::1F, where the prefix is automatically determined as stated above. There is one address excluded from the DHCPv6 pool range (::12). This pool range must match the pool range configured on the DUT. With the prefix included, the DHCPv6 pool range is:

```
<LAN prefix>:<ipv6DhcpClientStart> = 2004::2
< LAN prefix>:<ipv6DhcpClientEnd> = 2004::1F
```

With the following LAN clients excluded from the DHCPv6 pool:

```
<LAN prefix>:<ipv6DhcpClientExclude> = 2004::12
```

This example assumes that IPv6 DNS server address will be provided to clients on the LAN via DHCPv6 options.

In this example, 29 LAN clients (the size of the DHCPv6 pool) will be created during the scaling tests.

IPv6 Configuration:

```
testvar supportsIPv6 yes
```

```

# -- WAN configuration
testvar ipv6WanMode static
testvar ipv6WanIspIp 3001::1
testvar ipv6WanIspAssignIp 3001::2
testvar ipv6WanIspNextIp 3001::3
testvar ipv6WanIspPrefixLen 64

# -- LAN configuration
testvar ipv6LanMode DHCP
testvar ipv6LanIp 2004::1
testvar ipv6LanPrefixLen 64
testvar ipv6DhcpClientStart ::2
testvar ipv6DhcpClientEnd ::1F
testvar ipv6DhcpClientExclude ::12
testvar ipv6DNStoLAN yes
testvar ipv6RAInterval 300

# -- other configuration options
testvar ipv6RemoteHost 3001:51a:cafe::1
testvar ipv6WanDnsServer 3001:51a:cafe::2
testvar ipv6WanBackupDnsServer 3001:51a:cafe::3
testvar ipv6FreeNetworkStart 3001:cafe:1::
testvar ipv6FreeNetworkEnd 3001:ffff:ffff::
testvar ipv6FreeNetworkPrefixLen 64

```

### Example 5: DHCPv6 without prefix delegation on the WAN and autoconf on the LAN

In this example, the IPv4 configuration and IPv6 firewall configuration are identical to those provided in Example 1 above. Additionally, it is expected that the DUT will implement DHCPv6 on the WAN for acquisition of a non-temporary IPv6 address. DHCPv6 prefix delegation is not used in this configuration. CDRouter's DHCPv6 server (representing the ISP next-hop gateway as well) is 3001::2. CDRouter will assign the non-temporary IPv6 address 3001::2 and the IPv6 DNS server addresses 3001:51a:cafe::2 and 3001:51a:cafe::3 to the DUT via DHCPv6. The prefix length utilized by CDRouter on the WAN is configured to 64 bits. CDRouter will use the address 3001::3 as a temporary address for renumbering tests on the WAN.

On the LAN side, the DUT is configured for autoconf. The DUT's global LAN IPv6 address must be manually configured. In this example the DUT's global LAN IPv6 address is configured to 2004::1 with a prefix length of 64. This configuration also assumes that the DUT passes the IPv6 DNS server addresses learned on the WAN to clients on the LAN via the NDP RDNSS option (25).

In this example, 100 LAN clients will be created during the scaling tests.

IPv6 Configuration:

```

testvar supportsIPv6 yes

# -- WAN configuration
testvar ipv6WanMode DHCP
testvar ipv6WanIspIp 3001::1
testvar ipv6WanIspAssignIp 3001::2

```

```

testvar ipv6WanIspNextIp 3001::3
testvar ipv6WanIspPrefixLen 64
testvar dhcpv6WanenabledPD no

# -- LAN configuration
testvar ipv6LanMode autoconf
testvar ipv6LanIp 2004::1
testvar ipv6LanPrefixLen 64
testvar ipv6SupportsRdnss yes
testvar ipv6DNStoLAN yes
testvar ipv6RAInterval 300

# -- other configuration options
testvar ipv6RemoteHost 3001:51a:cafe::1
testvar ipv6WanDnsServer 3001:51a:cafe::2
testvar ipv6WanBackupDnsServer 3001:51a:cafe::3
testvar ipv6MaxLanClients 100
testvar ipv6FreeNetworkStart 3001:cafe:1::
testvar ipv6FreeNetworkEnd 3001:ffff:ffff::
testvar ipv6FreeNetworkPrefixLen 64

```

### Example 6: DHCPv6 with prefix delegation on the WAN and DHCPv6 on the LAN

In this example, the IPv4 configuration and IPv6 firewall configuration are identical to those provided in Example 1 above. Additionally, it is expected that the DUT will implement DHCPv6 on the WAN for acquisition of a non-temporary IPv6 address and an IPv6 address prefix for the LAN. CDRouter's DHCPv6 server (representing the ISP next-hop gateway as well) is 3001::2. CDRouter will assign the non-temporary IPv6 address 3001::2 and the IPv6 DNS server addresses 3001:51a:cafe::2 and 3001:51a:cafe::3 to the DUT via DHCPv6. The prefix length utilized by CDRouter on the WAN is configured to 64 bits. CDRouter will use the address 3001::3 as a temporary address for renumbering tests on the WAN.

The prefix assigned by CDRouter to the DUT is 3001:dddd:: with a prefix length of 48 bits. The prefix 3001:ddde:: will be used by CDrouter for renumbering tests on the WAN. CDRouter is also configured with a prefix delegation latency of 30 seconds. This latency specifies the expected amount of time, in seconds, that it takes for the DUT to propagate prefix changes on the WAN to clients on the LAN.

On the LAN side, the DUT is configured to provide addresses to clients via DHCPv6, as in Example 2. This example also assumes that the DUT is configured to provide the IPv6 DNS server addresses learned on the WAN, 3001:51:cafe::2 and 3001:51:cafe::3, to clients on the LAN via DHCPv6 options.

In this example, 100 LAN clients will be created during the scaling tests.

IPv6 Configuration:

```

testvar supportsIPv6 yes

# -- WAN configuration

```

```

testvar ipv6WanMode DHCP
testvar ipv6WanIspIp 3001::1
testvar ipv6WanIspAssignIp 3001::2
testvar ipv6WanIspNextIp 3001::3
testvar ipv6WanIspPrefixLen 64
testvar dhcpv6WanenabledPD yes
testvar dhcpv6WanAssignPrefix 2011:dddd::
testvar dhcpv6WanAssignNextPrefix 3001:ddde::
testvar dhcpv6WanAssignPrefixLen 48
testvar dhcpv6PDLatency 30

# -- LAN configuration
testvar ipv6LanMode DHCP
testvar ipv6DhcpClientStart ::2
testvar ipv6DhcpClientEnd ::1F
testvar ipv6DhcpClientExclude "::3 ::7"
testvar ipv6DNStoLAN yes
testvar ipv6RAInterval 300

# -- other configuration options
testvar ipv6RemoteHost 3001:51a:cafe::1
testvar ipv6WanDnsServer 3001:51a:cafe::2
testvar ipv6WanBackupDnsServer 3001:51a:cafe::3
testvar ipv6MaxLanClients 100
testvar ipv6FreeNetworkStart 3001:cafe:1::
testvar ipv6FreeNetworkEnd 3001:ffff:ffff::
testvar ipv6FreeNetworkPrefixLen 64

```

### Example 7: Autoconf on the WAN and DHCPv6 on the LAN

In this example, the IPv4 configuration and IPv6 firewall configuration are identical to those provided in Example 1 above. Additionally, it is expected that the DUT will implement stateless address autoconfiguration for IPv6 address acquisition on the WAN. The prefix and prefix length advertised by CDRouter are automatically determined based on WAN ISP address and prefix length. In this example the prefix advertised on the WAN will be 3001::/64.

On the LAN side, the DUT is configured to provide addresses to clients via DHCPv6, as in Example 2. This example also assumes that the DUT is configured to provide the IPv6 DNS server addresses learned on the WAN, 3001:51:cafe::2 and 3001:51:cafe::3, to clients on the LAN via DHCPv6 options.

In this example, 100 LAN clients will be created during the scaling tests.

IPv6 Configuration:

```

testvar supportsIPv6 yes

# -- WAN configuration
testvar ipv6WanMode autoconf
testvar ipv6WanIspIp 3001::1
testvar ipv6WanIspPrefixLen 64
testvar dhcpv6WanenabledPD no

# -- LAN configuration
testvar ipv6LanMode DHCP

```

```
testvar ipv6DhcpClientStart ::2
testvar ipv6DhcpClientEnd ::1F
testvar ipv6DhcpClientExclude "::3 ::7"
testvar ipv6DNStoLAN yes
testvar ipv6RAInterval 300

# -- other configuration options
testvar ipv6RemoteHost 3001:51a:cafe::1
testvar ipv6WanDnsServer 3001:51a:cafe::2
testvar ipv6WanBackupDnsServer 3001:51a:cafe::3
testvar ipv6MaxLanClients 100
testvar ipv6FreeNetworkStart 3001:cafe:1::
testvar ipv6FreeNetworkEnd 3001:ffff:ffff::
testvar ipv6FreeNetworkPrefixLen 64
```

### **Example 8: 6rd on the WAN and autoconf on the LAN**

Please see this [Application Note](#) for more information on configuring CDRouter for 6rd support on the WAN.

## 6. Testing Exercises

Aside from running each test case in CDRouter IPv6, other testing scenarios can be created. The following test scenarios are recommended.

Our [whitepaper on IPv6 testing with CDRouter](#) also contains a number of common issues associated with residential IPv6 implementations.

### Vary the IPv4 Configuration

IPv6 functionality within the device under test should be consistent regardless of its IPv4 configuration. CDRouter IPv6 can be used to efficiently verify that the IPv6 functionality of the device under test continues to perform as expected across a wide range of IPv4 WAN modes and configurations.

With CDRouter multiple configuration files can be created that each have a different IPv4 WAN configuration. For example, separate configuration files for all of the supported IPv4 WAN connection modes, including static IP, DHCP, PPPoE, PPPoA, PPP/T1, PPTP, and L2TP, can be easily created and tested. Other parameters such as the default IPv4 addresses assigned by CDRouter to the device under test can also be modified and tested. Running CDRouter IPv6's test modules against the device under test in all of the above configuration scenarios will quickly reveal any inconsistent behavior.

### Add a Wireless LAN Client

IPv6 functionality should be consistent for both wired and wireless LAN clients. Running the same set of test cases with both wired and wireless LAN clients will quickly reveal any discrepancies or inconsistencies in the device behavior. Note that certain test cases will be automatically skipped by CDRouter when using only a wireless LAN interface for testing. In these scenarios any test cases that require the creation of multiple LAN hosts, such as the scaling modules, will be skipped due to limitations inherent in the wireless drivers used.

### Testing for Unofficial Support IPv6

Some devices do not advertise support for or include any means to configure IPv6, yet do actually provide IPv6 connectivity for LAN clients. Such devices will often automatically create an IPv6 6to4 connection on the WAN and begin sending IPv6 Router Advertisements on the LAN.

CDRouter IPv6 can be used to quickly determine if a device unofficially supports IPv6 by utilizing a basic IPv6 configuration similar to Example 1 above. If CDRouter successfully passes startup with this IPv6 configuration enabled, one can assume that some level of IPv6 functionality is supported within the DUT.

### **Test all IPv6 Modes Supported by the DUT**

Devices that support IPv6 often have a wide range of configuration options. A great testing exercise is to test all of the different combinations of IPv6 related options supported by the DUT. If a DUT supports four different IPv6 WAN connection modes (assume DHCPv6 with and without prefix delegation, static IPv6, and 6to4) and two LAN modes (assume autoconf and DHCPv6) there are essentially eight different baseline configurations that can be tested. Mixing in various other options, such as DNS or prefix delegation configuration options, will grow the possible test matrix further.

This type of testing is easy to perform with CDRouter and provides a great baseline for a device's overall IPv6 behavior across a wide range of possible configurations. Any inconsistent or broken behavior can be easily and quickly identified this way. This same methodology can be extended to include IPv4 configuration options as well, as mentioned earlier in this section.

## 7. Possible Problems

### IPv6 Conflict Between CDRouter and the Linux Kernel

We recommend that IPv6 be disabled within the Linux operating system to avoid potential addressing conflicts when using CDRouter IPv6. Please see this [Knowledge Base article](#) for more information on disabling IPv6 within your Linux operating system.

CDRouter IPv6 will automatically detect if any of the allocated test interfaces have IPv6 addresses that were assigned by the Linux kernel. If pre-existing IPv6 addresses are detected, CDRouter will produce a warning message and abort the test run. Please see the above Knowledge Base article for information on disabling IPv6 within your Linux operating system.

It is also possible to skip CDRouter's IPv6 address check. Please see this [Knowledge Base article](#) for more information.

### 6to4 Tunnel Not Being Created by DUT

The 6to4 mechanism defined in [RFC 3056](#) relies on translating globally routable 32-bit IPv4 addresses into a unique 48-bit IPv6 prefix. 6to4 is not designed to work with private IPv4 addresses, which should be avoided when configuring and running tests with CDRouter IPv6. Please see the [example configuration](#) provided in the Configuration section of this document for more information.

Some devices may not initiate a 6to4 tunnel on the WAN if the primary IPv4 address obtained on the WAN is not a global address. In these cases configuring CDRouter to assign a global IPv4 address to the DUT may force initiation of the 6to4 tunnel on the WAN as expected.

### Inconsistent IPv6 Behavior for Different IPv4 WAN Modes

The IPv6 behavior of some devices varies significantly based on the IPv4 WAN connection mode that is used. For example, the behavior of a dual-stack CPE configured for static IPv6 and DHCP IPv4 connectivity on the WAN may be different than the results for the same basic setup but using PPPoE for IPv4 connectivity on the WAN instead. In rare cases some devices and combinations of IPv4 and IPv6 WAN connection modes essentially do not work all. Ideally, the behavior should be consistent regardless of the IPv4 connection type used. As mentioned in Section 6, a good testing exercise is to verify IPv6 functionality across all IPv4 WAN connection types supported by the DUT (DHCP, PPPoE, PPPoA, PPP/T1, PPTP, L2TP, and static IP).

## **No IPv6 Firewall**

Many devices that include IPv6 support have no firewalling capabilities for IPv6 connections. This poses a significant threat to overall security as IPv6 connections from the WAN are simply forwarded blindly to hosts on the LAN.

## **Limited IPv6 Firewall Support**

In IPv6 enabled devices IPv6 firewall support is often limited when compared to IPv4 firewall support. As a result, devices may handle certain applications differently over IPv4 and IPv6 connections. For example, FTP and TFTP are widely supported applications that generally work through IPv4 firewalls. However, FTP and TFTP ALG functionality for IPv6 is less common. The end result is that some applications, such as FTP and TFTP, may work properly through the device under test over IPv4 but not IPv6.

## **Missing or Incorrect Default Routes for IPv6**

Some devices that claim support for IPv6 do not properly establish default routes for IPv6 traffic. As a result, these devices are unable to forward IPv6 traffic from LAN to WAN, resulting in restricted access clients on the LAN attempting to communicate with IPv6 resources on the WAN.

## **Unofficial Support for IPv6**

As mentioned in the previous section, some devices that do not officially support IPv6 actually do. This is problematic because often there is no IPv6 firewall support within these devices and no mechanism for configuring or disabling IPv6 support.

## **Incorrect Router Advertisement Interval Configured**

If the value of the testvar `ipv6RAInterval` does not match the actual Router Advertisement interval configured on the device under test, CDRouter may incorrectly fail test cases or fail to obtain global IPv6 addresses via stateless address autoconfiguration. If you are unsure of the Router Advertisement interval, try setting this testvar to a large value and running test case `ipv6_ndp_2` to see the actual interval implemented by the DUT.

## **IPv6 Subnet Issues**

When static IPv6 is used on the WAN, it is important that the LAN side configuration utilizes a different IPv6 network to ensure proper routing from LAN to WAN and WAN to LAN through the DUT. If both the LAN and WAN interfaces of the DUT are on the same IPv6 network, packets may not be routed properly.

### **IPv6 DNS Information when DHCPv6 Prefix Delegation is Enabled**

Some devices may not request IPv6 DNS information when DHCPv6 is used only for prefix delegation on the WAN. This issue can be quickly identified with CDRouter by running tests using autoconf with prefix delegation as the primary IPv6 connection WAN mode.

### **IPv6 Routes Not Updated After Renumbering Event on WAN**

When a new address or prefix is assigned by the ISP to the DUT, the DUT must update its routing table to ensure that traffic continues to flow properly from LAN to WAN and vice-versa. Some implementations fail to update their IPv6 routes following an address or prefix change on the WAN – this often results in dropped packets and a general loss of connectivity for all clients on the LAN.

### **Not All IPv6 Configurations Work as Expected**

CDRouter makes it easy to run a baseline set of tests against a device using a wide variety of IPv6 configuration options. This is an excellent test exercise, as mentioned in the previous section, and often reveals inconsistencies in IPv6 behavior. A common example is a device that works well when a tunneling protocol, such as 6to4 is used on the WAN, but fails to set up the proper IPv6 routes when native IPv6 or DHCPv6 is used on the WAN instead. In this scenario is that the DUT fails to forward IPv6 packets for any WAN mode other than 6to4.

## 8. More Help

For more information on CDRouter and a listing of CDRouter IPv6 test cases, please visit the CDRouter IPv6 page at:

<http://www.qacafe.com/show/ipv6>

For general help running CDRouter, please check the CDRouter User Guide on-line at:

<http://www.qacafe.com/cdrouter>

Additional support notes on IPv6 can be found in the QA Cafe knowledge base at:

<http://www.qacafe.com/help>

For additional help, please contact [support@qacafe.com](mailto:support@qacafe.com)

## 9. References

1. IETF [RFC 2460](#) “Internet Protocol, Version 6”
2. IETF [RFC 4861](#) “Neighbor Discovery for IP version 6”
3. IETF [RFC 4443](#) “Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification”
4. IETF [RFC 4862](#) “IPv6 Stateless Address Autoconfiguration”
5. IETF [RFC 3056](#) “Connection of IPv6 Domains via IPv4 Clouds” aka 6to4
6. IETF [RFC 3315](#) “Dynamic Host Configuration Protocol for IPv6 (DHCPv6)”
7. IETF [RFC 3633](#) “IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6”
8. IETF [RFC 5569](#) “IPv6 Rapid Deployment on IPv6 Infrastructures (6rd)”
9. IETF [RFC 5006](#) “IPv6 Router Advertisement Option for DNS Configuration”
10. IETF [RFC 3646](#) “DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)”