



# **CDRouter IPv6 User Guide**

**Version 6.1**



QA Cafe © 2009  
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 static IPv6 or [RFC 3056](#) tunneling on the WAN. RFC 3056 is commonly referred to as “6to4” and is a technique for providing automatic IPv6 connectivity over IPv4 networks. When CDRouter IPv6 is enabled, CDRouter will create and maintain the necessary IPv6 next-hop hosts or 6to4 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 will 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:

<b>Test Module</b>	<b>Module Description</b>
<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>6to4.tcl</code>	6to4 tunnel tests for connecting IPv6 hosts over IPv6 networks
<code>icmp-v6.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.)

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 6.1.

## 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, L2TP, PPTP, static IP, etc.).

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 prefix on the LAN. This prefix should be either a manually configured prefix in the case of static IPv6 on the WAN or a 6to4 ([RFC 3056](#)) style prefix if 6to4 is used on the WAN.

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:

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
6to4.tcl	6to4 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.)

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\\_1/test-summary/test-summary-ipv6.htm](http://www.qacafe.com/static/6_1/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,6to4.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) 2001-2009 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

### IPv6 Configuration

CDRouter IPv6 has several IPv6 specific configuration options that define the DUT and the test environment.

The following table describes each option.

Test Variable (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.
<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 and 6to4 tunneling as defined by RFC 3056. 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” and “6to4”.</p> <p>Examples:</p> <pre>testvar ipv6WanMode 6to4 testvar ipv6WanMode static</pre>

<p><b>ipv6LanMode</b></p>	<p>The testvar <b>ipv6LanMode</b> 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.</p> <p>Valid values for the <b>ipv6LanMode</b> testvar are “dhcp” and “autoconf”.</p> <p>Examples:</p> <pre>testvar ipv6LanMode autoconf testvar ipv6LanMode dhcp</pre>
<p><b>ipv6RelayServer</b></p>	<p>The testvar <b>ipv6RelayServer</b> specifies the IPv4 address of the 6to4 relay server created by CDRouter on the WAN. By default CDRouter creates the 6to4 relay server at 192.88.99.1. Typically this value should not be changed unless the DUT is specifically configured for an alternate relay server address.</p> <p>This testvar is only used if the <b>ipv6WanMode</b> testvar is set to “6to4”.</p> <p>Example:</p> <pre>testvar ipv6RelayServer 192.88.99.1</pre>
<p><b>ipv6WanIspIp</b></p>	<p>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 2001::1</pre>
<p><b>ipv6WanIspAssignIp</b></p>	<p>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 2001::2</pre>

**ipv6RemoteHost**

The testvar **ipv6RemoteHost** 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.

Example:

```
testvar ipv6RemoteHost 2001:51a:cafe::1
```

**ipv6RAInterval**

The testvar **ipv6RAInterval** 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.

Example:

```
testvar ipv6RAInterval 300
```

If the value of **ipv6RAInterval** 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 **ipv6\_ndp\_2** to see the actual interval implemented by the DUT.

**ipv6MaxLanClients**

The testvar **ipv6MaxLanClients** specifies the maximum number of IPv6 LAN clients CDRouter will attempt to create during the scaling tests. The default value is 10.

If the **ipv6LanMode** 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 **ipv6MaxLanClients**. Alternatively, if the DHCPv6 pool size is greater than **ipv6MaxLanClients**, the number of LAN clients created during the scaling-v6.tcl test module will be capped at **ipv6MaxLanClients**.

Note that the maximum number of clients that can be created during the scaling-v6.tcl test module is 512.

Example:

```
testvar ipv6MaxLanClients 10
```

---

**ipv6LanInterfaceId**

The testvar **ipv6LanInterfaceId** specifies the expected 64-bit host identifier of the DUT’s global IPv6 address. Some devices use an arbitrary host identifier such as ::1001:20a:3:f4 or ::1, whereas others use a MAC based EUI64 identifier. The default value is ::%eui64%, which is a special keyword that will force CDRouter to determine the host portion of the DUT’s global IPv6 address based on the DUT’s LAN MAC address.

This testvar is only used if the **ipv6WanMode** is set to “6to4”. When 6to4 is not used, the DUT’s global LAN IPv6 address must be manually configured using the testvars **ipv6LanIp** and **ipv6LanPrefixLen**.

Examples:

```
testvar ipv6LanInterfaceId ::1
testvar ipv6LanInterfaceId ::1001:20a:3:f4
testvar ipv6LanInterfaceId ::%eui64%
```

<p><b>dnsHostname&lt;N&gt;</b></p>	<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 2001:4860:0:2222::68 testvar dnsHostname3 test.qacafe.com testvar dnsIp3 "5.5.5.5 2001:cafe::1 7.7.7.7"</pre>
<p><b>dnsIp&lt;N&gt;</b></p>	<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>
<p><b>forwardingMaxFailures</b></p>	<p>The testvar <b>forwardingMaxFailures</b> defines the maximum number of failures allowed during the forwarding tests. If this number is exceeded CDRouter will abort the running test and proceed to the next test case. The default value is 20.</p> <p>Example:</p> <pre>testvar forwardingMaxFailures 20</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 2001:cafe:1:: testvar ipv6FreeNetworkEnd 2001:ffff:ffff:: testvar ipv6FreeNetworkPrefixLen 64</pre>
<b>ipv6WanPingRespond</b>	<p>The testvar <b>ipv6WanPingRespond</b> 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.</p> <p>Example:</p> <pre>testvar ipv6WanPingRespond yes</pre>
<b>ipv6FirewallTcpClosedPorts</b> <b>ipv6FirewallUdpClosedPorts</b>	<p>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 <b>ipv6FirewallTcpClosedPorts</b> or <b>ipv6FirewallUdpClosedPorts</b> lists. The keyword “all” may also be specified for devices that default to closing all TCP or UDP IPv6 ports.</p> <p>Example:</p> <pre>testvar ipv6FirewallTcpClosedPorts "80 8080" testvar ipv6FirewallUdpClosedPorts 113</pre> <p>Note that the range of ports scanned in the firewall-v6.tcl test module can be configured using the <b>portScanStart</b> and <b>portScanStop</b> testvars. Please see the <a href="#">CDRouter User's Guide</a> for more information.</p>

<p><b>ipv6FirewallTcpOpenPorts</b> <b>ipv6FirewallUdpOpenPorts</b></p>	<p>Any TCP or UDP ports in the IPv6 firewall that are open may be configured in the <b>ipv6FirewallTcpOpenPorts</b> or <b>ipv6FirewallUdpOpenPorts</b> lists.</p> <p>Example:</p> <pre>testvar ipv6FirewallTcpOpenPorts "22 443" testvar ipv6FirewallUdpOpenPorts 161</pre> <p>Note that the range of ports scanned in the firewall-v6.tcl test module can be configured using the <b>portScanStart</b> and <b>portScanStop</b> testvars. Please see the <a href="#">CDRouter User's Guide</a> for more information.</p>
<p><b>ipv6DhcpClientStart</b></p>	<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>

<p><b>ipv6DhcpClientEnd</b></p>	<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>
<p><b>ipv6DhcpClientExclude</b></p>	<p>This 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>

<p><b>ipv6LanIp</b></p>	<p>This testvar specifies the DUT's LAN side global IPv6 address. This value should match the value configured on the DUT and is only used if the <b>ipv6WanMode</b> is not set to "6to4".</p> <p>If the DUT is using 6to4 on the WAN, the LAN side global IPv6 address will be automatically determined based on the WAN IPv4 address and the DUT's host ID as specified by the testvar <b>ipv6LanInterfaceId</b>.</p> <p>If the <b>ipv6WanMode</b> is not set to "6to4", the DUT's LAN side global IPv6 address must be manually configured and must match the value configured for this testvar.</p> <p>Example:</p> <pre>testvar ipv6LanIp 2004::1</pre>
<p><b>ipv6LanPrefixLen</b></p>	<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>

## 5. Example Configurations

In this section four basic example configurations for CDRouter IPv6 are provided. These four configurations involve the four possible combinations of the currently supported IPv6 WAN modes (6to4 and static IPv6) and IPv6 LAN modes (stateless address autoconfiguration and DHCPv6).

The examples provided cover the following configurations:

- 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 1: 6to4 on the WAN and Stateless Address Autoconfiguration 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 192.88.99.1 on the WAN. The DUT's global IPv6 address is expected to be <6to4 prefix>:<ipv6LanInterfaceId>. 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>:<ipv6LanInterfaceId> = 2002:c0c:c02:1::1**

CDRouter is also configured to use IPv6 stateless address autoconfiguration on the LAN for all clients that it creates. The maximum expected Router Advertisement interval for the DUT is set to 300 seconds. If CDRouter does not receive a Router Advertisement from

the DUT within 300 seconds, the test run will be aborted. CDRouter is also configured to create 100 LAN clients during the scaling tests.

#### 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
testvar ipv6WanMode 6to4
testvar ipv6LanMode autoconf
testvar ipv6RelayServer 192.88.99.1
testvar ipv6RemoteHost 2001:51a:cafe::1
testvar ipv6RAInterval 300
testvar ipv6LanInterfaceId ::1
testvar ipv6MaxLanClients 100
testvar forwardingMaxFailures 20
testvar ipv6FreeNetworkStart 2001:cafe:1::
testvar ipv6FreeNetworkEnd 2001:ffff:ffff::
testvar ipv6FreeNetworkPrefixLen 64
testvar ipv6WanPingRespond yes
```

#### IPv6 Firewall Configuration:

```
testvar ipv6FirewallTcpClosedPorts "80 8080"
testvar ipv6FirewallUdpClosedPorts all
testvar ipv6FirewallTcpOpenPorts "443"
testvar ipv6FirewallUdpOpenPorts 113
```

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>:<ipv6LanInterfaceId>. 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>:<ipv6LanInterfaceId> = 2002:c0c:c02:1::1**

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

**<6to4 prefix>:<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>:<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.

IPv6 Configuration:

```
testvar supportsIPv6 yes
testvar ipv6WanMode 6to4
testvar ipv6LanMode dhcp
testvar ipv6RelayServer 192.88.99.1
testvar ipv6RemoteHost 2001:51a:cafe::1
testvar ipv6DhcpClientStart ::2
testvar ipv6DhcpClientEnd ::1F
testvar ipv6DhcpClientExclude "::3 ::7"
testvar ipv6RAInterval 300
testvar ipv6LanInterfaceId ::1
testvar ipv6MaxLanClients 100
testvar forwardingMaxFailures 20
testvar ipv6FreeNetworkStart 2001:cafe:1::
testvar ipv6FreeNetworkEnd 2001:ffff:ffff::
testvar ipv6FreeNetworkPrefixLen 64
testvar ipv6WanPingRespond yes
```

### **Example 3: Static IPv6 on the WAN and Stateless Address Autoconfiguration 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 2001::2, and CDRouter's IPv6 address (representing the ISP next-hop gateway) is 2001::1.

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.

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

IPv6 Configuration:

```
testvar supportsIPv6 yes
testvar ipv6WanMode static
testvar ipv6LanMode autoconf
testvar ipv6WanIspIp 2001::1
testvar ipv6WanIspAssignIp 2001::2
testvar ipv6LanIp 2004::1
testvar ipv6LanPrefixLen 64
testvar ipv6RemoteHost 2001:51a:cafe::1
testvar ipv6RAInterval 300
testvar ipv6MaxLanClients 100
testvar forwardingMaxFailures 20
testvar ipv6FreeNetworkStart 2001:cafe:1::
testvar ipv6FreeNetworkEnd 2001:ffff:ffff::
testvar ipv6FreeNetworkPrefixLen 64
testvar ipv6WanPingRespond yes
```

#### **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.

In this example, 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 2001::2, and CDRouter's IPv6 address (representing the ISP next-hop gateway) is 2001::1.

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**

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

IPv6 Configuration:

```
testvar supportsIPv6 yes
testvar ipv6WanMode static
testvar ipv6LanMode dhcp
testvar ipv6WanIspIp 2001::1
testvar ipv6WanIspAssignIp 2001::2
testvar ipv6LanIp 2004::1
testvar ipv6LanPrefixLen 64
testvar ipv6DhcpClientStart ::2
testvar ipv6DhcpClientEnd ::1F
testvar ipv6DhcpClientExclude ::12
testvar ipv6RemoteHost 2001:51a:cafe::1
testvar ipv6RAInterval 300
testvar ipv6LanInterfaceId ::1
testvar forwardingMaxFailures 20
testvar ipv6FreeNetworkStart 2001:cafe:1::
testvar ipv6FreeNetworkEnd 2001:ffff:ffff::
testvar ipv6FreeNetworkPrefixLen 64
testvar ipv6WanPingRespond yes
```

## 6. Testing Exercises

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

### 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, PPTP, L2TP, etc., 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 increasing the maximum expected Router Advertisement interval as defined by the testvar **ipv6RAInterval**. Setting this testvar to a larger value, 600 seconds for example, will force CDRouter to wait this amount of time for Router Advertisements on the LAN. If no

Router Advertisements are received, the test run will be aborted and one may conclude that the device under test does not unofficially support IPv6.

## 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, PPTP, L2TP, etc.).

## 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 `ipv6RAInternal` 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 to ensure 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.

## 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 2461](#) “Neighbor Discovery for IPv6”
3. IETF [RFC 2463](#) “Internet Control Message Protocol for IPv6” aka ICMPv6
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)”