IPv6 CONSORTIUM TEST SUITE

Internet Protocol, version 6

Multi-System Interoperability Test Suite

Technical Document

Revision 1.1

IPv6 Consortium
InterOperability Laboratory
Research Computing Center
University of New Hampshire

121 Technology Drive, Suite 2
Durham, NH 03824-3525
Phone: (603) 862-3941
Fax: (603) 862-1761
http://www.iol.unh.edu

TABLE OF CONTENTS

MODIFICATION RECORD

Revision 0.1 Completed	December 8, 1999
Revision 0.9 Completed	January 31, 2002
Revision 1.0 Completed	December 31, 2002
Revision 1.1 Completed	January 2, 2003

ACKNOWLEDGMENTS

The University of New Hampshire would like to acknowledge the efforts of the following individuals in the development of this test suite.

Dr. William Lenharth	University of New Hampshire
John Leser	University of New Hampshire
Ray LaRocca	University of New Hampshire
Jacques J. Ludman	University of New Hampshire
Ben Schultz	University of New Hampshire
Robert Wolff	University of New Hampshire

INTRODUCTION

Overview

The University of New Hampshire's InterOperability Laboratory (IOL) is an institution designed to improve the interoperability of standards based products by providing an environment where a product can be tested against other implementations of a standard. This suite of tests has been developed to help Internet Protocol, version 6 implementers evaluate the interoperability of their IPv6 products with devices from several other vendors. Successful completion of all tests contained in this suite does not guarantee that the tested device will interoperate with all other IPv6 devices in all situations. However, combined with satisfactory operation in the IOL's operation test suites, these tests provide a reasonable level of confidence that the Node Under Test (NUT) will function well in most multi-vendor IPv6 environments.

Acronyms

DR: Default Router
H: Host
HUT: Host Under Test
NUT: Node Under Test
RUT: Router Under Test

When several entities of the same type are present in a test configuration, a number is appended to the acronym to yield a label for each entity. For example, if there were three default routers in the test configuration, they would be labeled DR1, DR2 and DR3.

TEST ORGANIZATION

This document organizes tests by Section based on related test methodology or goals. Each group begins with a brief set of comments pertaining to all tests within that group. This is followed by a series of description blocks; each block describes a single test. The format of the description block is as follows:

Test Label: The test label and title comprise the first line of the test block. The test label is composed by concatenating the short test suite name, the section number, the group number, and the test number within the group. These elements are separated by periods. The Test Number is the section, group and test number, also separated by periods.

Purpose: The Purpose is a short statement describing what the test attempts to achieve. It is usually phrased as a simple assertion of the feature or capability to be tested.

References: The References section lists cross-references to the specifications and documentation that might be helpful in understanding and evaluating the test and results.

Resource Requirements: The Resource Requirements section specifies the software, hardware, and test equipment that will be needed to perform the test.

Discussion: The Discussion is a general discussion of the test and relevant section of the specification, including any assumptions made in the design or implementation of the test as well as known limitations.

Test Setup: The Test Setup section describes the configuration of all devices prior to the start of the test. Different parts of the procedure may involve configuration steps that deviate from what is given in the test setup. If a value is not provided for a protocol parameter, then the protocol's default is used for that parameter.

Procedure: This section of the test description contains the step-by-step instructions for carrying out the test. These steps include such things as enabling interfaces, unplugging devices from the network, or sending packets from a test station. The test procedure also cues the tester to make observations, which are interpreted in accordance with the observable results given for that test part.

Observable Results: This section lists observable results that can be examined by the tester to verify that the NUT is operating properly. When multiple observable results are possible, this section provides a short discussion on how to interpret them. The determination of a pass or fail for each test is usually based on how the NUT's behavior compares to the results described in this section.

Possible Problems: This section contains a description of known issues with the test procedure, which may affect test results in certain situations.

REFERENCES

The following documents are referenced in this text:

[IPv6-SPEC] Hinden, R., S. Deering, Internet Protocol, Version 6 (IPv6) Specification, RFC 2460, December 1998.

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

[ADDRCONF] Thomson, S., T. Narten, IPv6 Stateless Address Autoconfiguration, RFC 2462, December 1998.

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

[PMTU] McCann, J., S. Deering, and J. Mogul, Path MTU Discovery for IPv6, RFC 1981, August 1996.

[MLD] Deering, S., Fenner, W., Haberman, B., Multicast Listener Discovery (MLD) for IPv6, RFC 2710, October 1999.

[T/TCP] R. Braden, TCP Extensions for Transactions Functional Specification, RFC 1644, July 1994.

[FTP] J. Postel, J. Reynolds, File Transfer Protocol (FTP), RFC 959, October 1985.

[TELNET] J. Postel, J. Reynolds, TELNET Protocol Specification, RFC 854, May 1983.

[TFTP] K. Sollins, The TFTP Protocol (Revision 2), RFC 1350, July 1992.

[HTTP] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, Hypertext Transfer Protocol – HTTP/1.1, June 1999.

Section 2: Interoperability

Scope

Tests in this section verify the ability of a router or host to correctly interoperate in a heterogeneous, multi-system IPv6 environment.

Test Applicability

The device type, whether a router or host, determines the specific program of tests performed.

Router:

> *All Tests*: In each test, the RUT is substituted for DR1. The test is run once for each set of test partners (one or more hosts and zero or more routers).

Host:

> *All Tests*: In each test, the HUT is substituted for H1. The test is run once for each set of test partners (one or more hosts and one or more routers).

Group 1: Basic Interoperability

Scope

Tests in this group verify that the NUT is able to engage in basic communication in an IPv6 environment.

Overview

The following tests verify basic operations such as ICMP Echo Requests and Replies, TCP sessions, and transmissions via UDP.

Test IP6.2.1.1: ICMP Echo Interoperability

Purpose: To verify that a successful ICMPv6 Echo Request, Echo Reply exchange can be achieved in both directions.

References:

- [ICMPv6] – Section 4

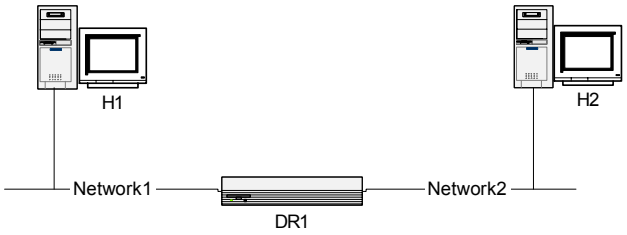
Resource Requirements:

- Monitor to capture packets
- ping6 implementations

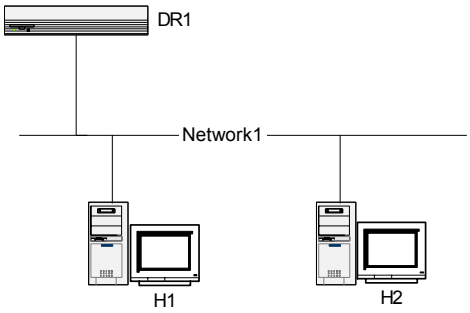
Discussion: Every node must implement an ICMPv6 Echo responder function that receives Echo Requests and sends corresponding Echo Replies. A node should also implement an application-layer interface for sending Echo Requests and receiving Echo Replies, for diagnostic purposes. The source address of an Echo Reply sent in response to a unicast Echo Request message must be the same as the destination address of that Echo Request message.

Test Setup: For each Part, connect hosts H1 and H2 and router DR1 to Network1 and Network2, per the figure below. Router DR1 routes between Network1 and Network2. Allow time for H1 and H2 to perform stateless address autoconfiguration and Duplicate Address Detection.

Part A:



Part B:



Procedure:

Part A: Off-Link Correspondence

1. Transmit ICMP Echo Requests from the global address of H1 to the global address of H2.
2. Observe the frames received by H1 and H2.
3. Transmit ICMP Echo Requests from the global address of H2 to the global address of H1.
4. Observe the frames received by H1 and H2.

Part B: On-Link Correspondence

5. Transmit ICMP Echo Requests from the link-local address of H1 to the link-local address of H2.
6. Observe the frames received by H1 and H2.
7. Transmit ICMP Echo Requests from the link-local address of H2 to the link-local address of H1.
8. Observe the frames received by H1 and H2.

Observable Results:

- In Part A, DR1 should forward all ICMP Echo Requests destined for a host on Network1 or Network2 to the appropriate link. H2 should receive all the ICMP Echo Requests sent from H1 and respond with ICMP Echo Replies destined for the global address of H1. H1 should receive all the ICMP Echo Requests sent from H2 and respond with ICMP Echo Replies destined for the global address of H2.
- In Part B, H2 should receive all the ICMP Echo Requests sent from H1 and respond with ICMP Echo Replies destined for the link-local address of H1. H1 should receive all the ICMP Echo Requests sent from H2 and respond with ICMP Echo Replies destined for the link-local address of H2.

Possible Problems:

- None.

Test IP6.2.1.2: TCP Interoperability

Purpose: To verify that a successful TCP session can be achieved between IPv6 implementations from various vendors.

References:

- [TELNET]

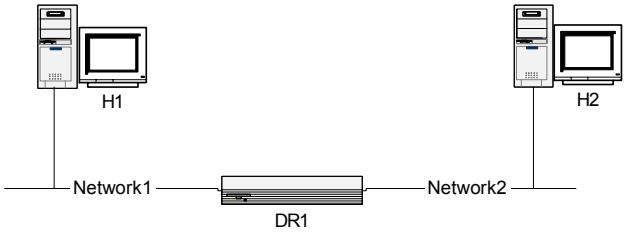
Resource Requirements:

- Monitor to capture packets
- IPv6 telnet implementations

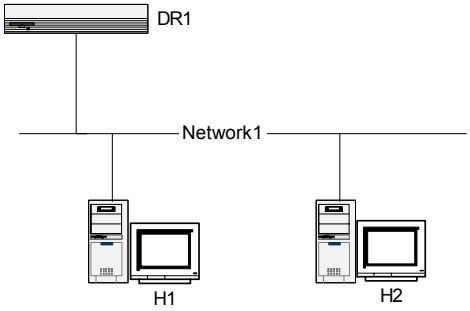
Discussion: The purpose of the telnet protocol is to provide a fairly general, bi-directional, eight-bit byte oriented communications facility. Telnet runs over the transmission control protocol (TCP).

Test Setup: For each Part, connect hosts H1 and H2 and router DR1 to Network1 and Network2, per the figure below. Router DR1 routes between Network1 and Network2. Allow time for H1 and H2 to perform stateless address autoconfiguration and Duplicate Address Detection.

Part A:



Part B:



Procedure:

Part A: Off-Link Correspondence

1. Initiate a telnet session between H1 (client) and H2 (server).
2. Initiate a telnet session between H2 (client) and H1 (server).
3. Ensure that H1 and H2 can communicate properly.
4. Terminate the telnet sessions between H1 and H2.

Part B: On-Link Correspondence

5. Initiate a telnet session between H1 (client) and H2 (server).
6. Initiate a telnet session between H2 (client) and H1 (server).
7. Ensure that H1 and H2 can communicate properly.
8. Terminate the telnet sessions between H1 and H2.

Observable Results:

- In Part A, DR1 should forward all frames destined for a host on Network1 or Network2 to the appropriate link. H2 and H1 should be able to communicate via the telnet protocol without interruption.
- In Part B, H2 and H1 should be able to communicate via the telnet protocol without interruption.

Possible Problems:

- If the NUT or test partners do not support telnet, another protocol that runs over TCP may be used (HTTP, FTP, etc.).

Test IP6.2.1.3: UDP Interoperability

Purpose: To verify that a successful UDP exchange can be achieved in both directions, between IPv6 implementations from various vendors.

References:

- [TFTP]

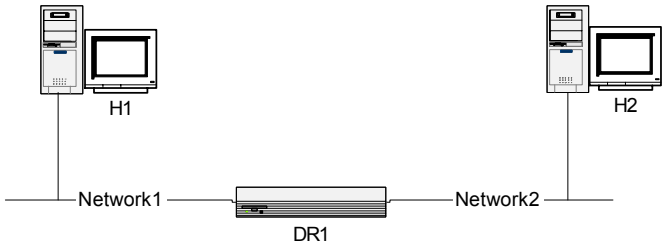
Resource Requirements:

- Monitor to capture packets
- IPv6 TFTP implementations

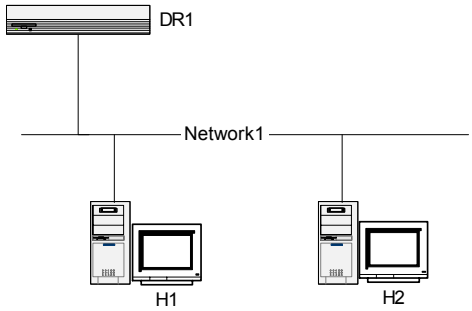
Discussion: The trivial file transfer protocol (TFTP) runs over the user datagram protocol (UDP).

Test Setup: For each Part, connect hosts H1 and H2 and router DR1 to Network1 and Network2, per the figure below. Router DR1 routes between Network1 and Network2. Allow time for H1 and H2 to perform stateless address autoconfiguration and Duplicate Address Detection.

Part A:



Part B:



Procedure:

Part A: Off-Link Correspondence

1. Initiate a TFTP session between H1 (client) and H2 (server).
2. Initiate a TFTP session between H2 (client) and H1 (server).
3. Ensure that H1 and H2 can communicate properly.
4. Terminate the TFTP sessions between H1 and H2.

Part B: On-Link Correspondence

5. Initiate a TFTP session between H1 (client) and H2 (server).
6. Initiate a TFTP session between H2 (client) and H1 (server).
7. Ensure that H1 and H2 can communicate properly.
8. Terminate the TFTP sessions between H1 and H2.

Observable Results:

- In Part A, DR1 should forward all frames destined for a host on Network1 or Network2 to the appropriate link. H2 and H1 should be able to communicate via TFTP without interruption.
- In Part B, H2 and H1 should be able to communicate via TFTP without interruption.

Possible Problems:

- If the NUT or test partners do not support TFTP, another protocol that runs over UDP may be used (DNS, NFS, SNMP, proprietary streaming media, Quake, etc.).

Group 2: Extended Interoperability

Scope

Tests in this group verify that the NUT is able to engage in various aspects of the base IPv6 protocol.

Overview

The following tests verify operations such as stateless address autoconfiguration, on-link determination, Duplicate Address Detection, path MTU discovery, fragmentation, redirects, communication when configured with multiple prefixes, network renumbering, and neighbor unreachability.

Test IP6.2.2.1: Address Autoconfiguration and Duplicate Address Detection

Purpose: To verify that an arbitrary number of hosts can properly initialize on a network and communicate with other on-link partners.

References:

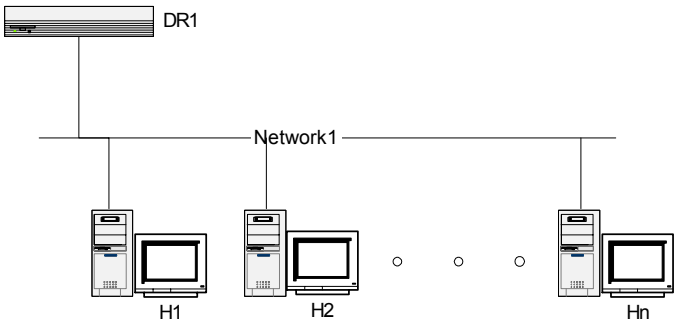
- [ADDRCONF] – Section 1, 5.4

Resource Requirements:

- Monitor to capture packets
- ping6 implementations

Discussion: When a host initializes on a given link, it performs stateless address autoconfiguration and Duplicate Address Detection. To insure that all configured addresses are likely to be unique on a given link, hosts run the Duplicate Address Detection algorithm on addresses before assigning them to an interface. The Duplicate Address Detection algorithm is performed on all addresses, independent of whether they are obtained via stateless or stateful autoconfiguration. In addition, routers perform Duplicate Address Detection on all addresses prior to assigning them to an interface.

Test Setup: Connect the n number of hosts H1 through Hn and router DR1 to Network1, per the figure below.



Procedure:

1. Configure host Hn on Network1 to have the same link-local address as the NUT.
2. Initialize the devices on Network1, powering up Hn before the NUT.
3. Allow time for all devices on Network1 to perform stateless address autoconfiguration and Duplicate Address Detection.
4. Transmit ICMP Echo Requests from H1 to the link-local address of the NUT.
5. Repeat Steps 1 through 4 for every other address of the NUT.

Observable Results:

- The NUT should perform Duplicate Address Detection on its address for Network1. It should determine that another device on Network1 already has its tentative address and use a different address or prompt for administrative configuration. Hn, and not the NUT, should not respond to the ICMP Echo Requests transmitted from H1.

Possible Problems:

- None.

Test IP6.2.2.2: Path MTU and Fragmentation

Purpose: Verify that the NUT can participate in path MTU discovery and handle fragmentation in an IPv6 network.

References:

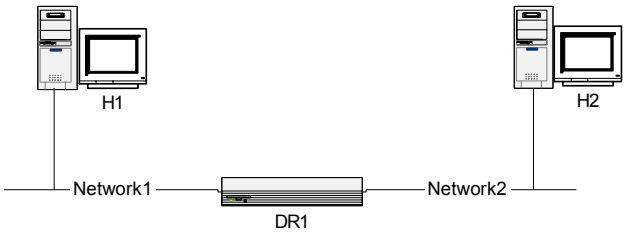
- [PMTU] – Section 3

Resource Requirements:

- Monitor to capture packets
- ping6 implementations capable of sending large packets

Discussion: IPv6 nodes should implement Path MTU Discovery in order to discover and take advantage of paths with PMTU greater than the IPv6 minimum link MTU. A source node initially assumes that the PMTU of a path is the (known) MTU of the first hop in the path. If any of the packets sent on that path are too large to be forwarded by some node along the path, that node will discard them and return ICMPv6 Packet Too Big messages. Upon receipt of such a message, the source node reduces its assumed PMTU for the path based on the MTU of the constricting hop as reported in the Packet Too Big message.

Test Setup: Connect hosts H1 and H2 and router DR1 to Network1 and Network2, per the figure below. Router DR1 routes between Network1 and Network2. Allow time for H1 and H2 to perform stateless address autoconfiguration and Duplicate Address Detection.



Procedure:

1. Configure the Network1 interface on DR1 with a path MTU of 1280 bytes.
2. Configure the Network2 interface on DR1 with a path MTU of 1500 bytes.
3. Transmit 1500 byte ICMP Echo Requests from the global address of H1 to the global address of H2.
4. Transmit 1500 byte ICMP Echo Requests from the global address of H2 to the global address of H1.
5. Increase the path MTU for Network1 to 1500 bytes.
6. Repeat Steps 3 and 4 until H1 and H2 detect that the path MTU has increased.

Observable Results:

- In Steps 3 and 4, H1 and H2 should fragment its ICMP Echo Requests and Echo Replies to fit within the minimum path MTU of Network1 of 1280 bytes. In Step 5, H1 and H2 should eventually detect that the path MTU for Network1 has increased and no longer fragment its ICMP Echo Requests and Replies.

Possible Problems:

- None.

Test IP6.2.2.3: Multiple Prefixes and Network Renumbering

Purpose: To verify that a host configured with multiple prefixes can communicate with another host on a different network when its site has been renumbered.

References:

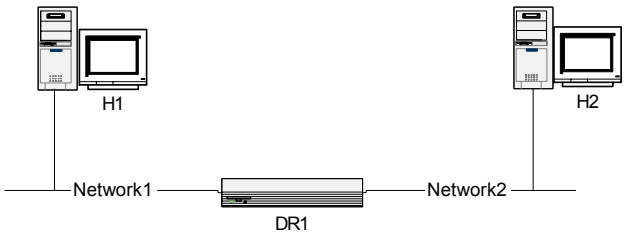
- [ND] – Section 12
- [ADDRCONF] – Section 4.1

Resource Requirements:

- Monitor to capture packets
- ping6 implementations

Discussion: Multiple prefixes can be associated with the same link. By default, hosts learn all on-link prefixes from Router Advertisements. A host with multiple on-link prefixes should be able to communicate using any configured prefix, as long as its lifetime is still valid. Further, the Neighbor Discovery protocol together with IPv6 Address Autoconfiguration provides mechanisms to aid in renumbering - new prefixes and addresses can be introduced and old ones can be deprecated and removed. Address leasing facilitates site renumbering by providing a mechanism to time-out addresses assigned to interfaces in hosts. At present, upper layer protocols such as TCP provide no support for changing end-point addresses while a connection is open. If an end-point address becomes invalid, existing connections break and all communication to the invalid address fails. Even when applications use UDP as a transport protocol, addresses must generally remain the same during a packet exchange.

Test Setup: Connect hosts H1 and H2 and router DR1 to Network1 and Network2, per the figure below. Initially, Network1 has one prefix, Prefix1, associated with it. Router DR1 routes between Network1 and Network2. Allow time for H1 and H2 to perform stateless address autoconfiguration and Duplicate Address Detection.



Procedure:

1. Network1 is configured with a new prefix Prefix2. The old prefix, Prefix1, is configured to time out such that the old and new prefix lifetimes overlap.
2. Allow time for H1 to be configured with the new prefix and for Duplicate Address Detection to be performed.
3. Transmit ICMP Echo Requests from a global address of H1 to the global address of H2.

4. Transmit an ICMP Echo Request from the global address of H2 to the global address of H1 associated with Prefix1.
5. Transmit an ICMP Echo Request from the global address of H2 to the global address of H1 associated with Prefix2.
6. Allow enough time to elapse so that Prefix1 has timed out.
7. Transmit ICMP Echo Requests from a global address of H1 to the global address of H2.
8. Transmit an ICMP Echo Request from the global address of H2 to the global address of H1 associated with Prefix1.
9. Transmit an ICMP Echo Request from the global address of H2 to the global address of H1 associated with Prefix2.

Observable Results:

- In Step 2, H1 should configure the new prefix Prefix1. In Step 3, H2 should respond to ICMP Echo Requests from H1 with Echo Replies sent to the appropriate global address of H1. In Steps 4 and 5, H1 should respond to ICMP Echo Requests from H2 with Echo Replies sent from the appropriate global address. In Step 7, H2 should respond to ICMP Echo Requests from H1 with Echo Replies sent to the appropriate global address of H1. In Steps 8 and 9, H1 should only respond to ICMP Echo Requests sent to the global address associated with Prefix2.

Possible Problems:

- None.

Test IP6.2.2.4: Redirect Function

Purpose: Verify the correct interoperability between the NUT's redirect handling with that of various IPv6 router implementations.

References:

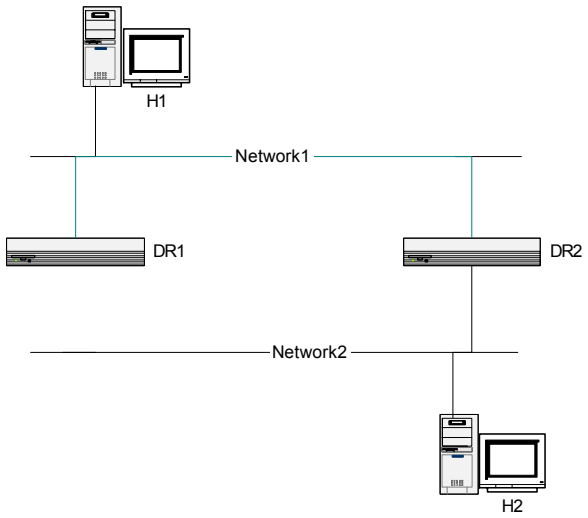
- [ND] – Section 4.5, 4.6.3, 8

Resource Requirements:

- Monitor to capture packets
- ping6 implementations

Discussion: Routers send redirect packets to inform a host of a better first-hop node on the path to a destination. Hosts can be redirected to a better first-hop router, but can also be informed by a redirect that the destination is in fact a neighbor. The latter is accomplished by having the ICMP Target Address be equal to the ICMP Destination Address in the redirect message.

Test Setup: Connect hosts H1 and H2 and routers DR1 and DR2 to Network1 and Network2, per the figure below. Configure router DR2 to not transmit Router Advertisements on Network1. Router DR1 is not connected to Network2. Router DR2 routes between Network1 and Network2. Allow time for H1 and H2 to perform stateless address autoconfiguration and Duplicate Address Detection.



Procedure:

1. Configure a static route on DR1 indicating DR2 as the next hop for network Network2.
2. Transmit an ICMP Echo Request from the global address of H1 to the global address of H2.
3. Allow time for DR1 to send an ICMP Redirect message to H1 specifying DR2 as a better first hop.
4. Transmit an ICMP Echo Request from the global address of H1 to the global address of H2.

5. Remove the static route on DR1 configured in Step 1.

Observable Results:

- In Step 2, H2 should respond to the ICMP Echo Request with an ICMP Echo Reply. In Step 3, DR1 should send an ICMP Redirect message to H1 indicating DR2 as a better first hop to network Network2. In Step 4, H2 should respond to the ICMP Echo Request with an ICMP Echo Reply. H1 should use DR2 as its first hop.

Possible Problems:

- None.

Test IP6.2.2.5: Neighbor Unreachability Detection: Loss of Default Router

Purpose: To verify that a host can determine that its default router is no longer reachable, so that it may switch to another default router.

References:

- [ND] – Section 6.3.5, 7.3

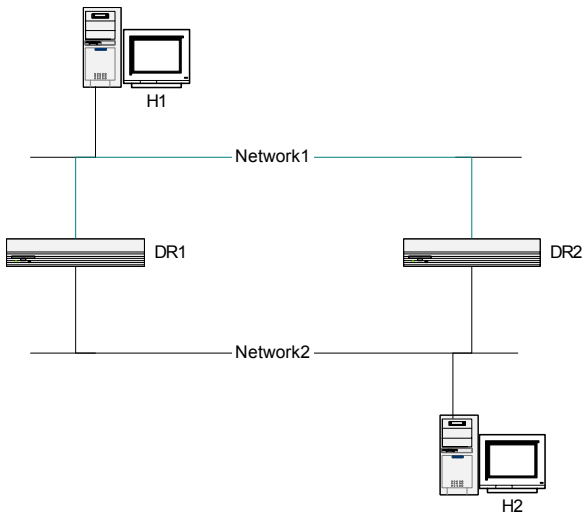
Resource Requirements:

- Monitor to capture packets
- ping6 implementations

Discussion: Communication to or through a neighbor may fail for numerous reasons at any time, including hardware failure, hot-swap of an interface card, etc. If the destination has failed, no recovery is possible and communication fails. On the other hand, if it is the path that has failed, recovery may be possible. Thus, a node actively tracks the reachability "state" for the neighbors to which it is sending packets. Neighbor Unreachability Detection is used for all paths between hosts and neighboring nodes, including host-to-host, host-to-router, and router-to-host communication. Neighbor Unreachability Detection may also be used between routers, but is not required if an equivalent mechanism is available, for example, as part of the routing protocols.

When a path to a neighbor appears to be failing, the specific recovery procedure depends on how the neighbor is being used. If the neighbor is the ultimate destination, for example, address resolution should be performed again. If the neighbor were a router, however, attempting to switch to another router would be appropriate.

Test Setup: Connect hosts H1 and H2 and routers DR1 and DR2 to Network1 and Network2, per the figure below. Routers DR1 and DR2 route between Network1 and Network2. Allow time for H1 and H2 to perform stateless address autoconfiguration and Duplicate Address Detection.



Procedure:

1. Transmit an ICMP Echo Request from the global address of H1 to the global address of H2
2. Disconnect the link between Network1 and the router that H1 uses as a first hop in Step 1.
3. Transmit an ICMP Echo Request from the global address of H1 to the global address of H2.
4. Allow time for H1 to determine that its first hop in Step 3 is unreachable and switch to the other router as its default.
5. Transmit an ICMP Echo Request from the global address of H1 to the global address of H2.

Observable Results:

- In Step 4, H1 should perform Neighbor Unreachability Detection and determine that its first hop is no longer available. In Step 5, the ICMP Echo Request should be received and replied to by H2.

Possible Problems:

- None.