

**Moonv6 Test Suite**  
*IPv6 Firewall Network Level  
Interoperability Test Suite*

**Technical Document**

Revision 1.0



---

*IPv6 Consortium  
InterOperability Laboratory  
Research Computing Center  
University of New Hampshire*

*121 Technology Drive, Suite 2  
Durham, NH 03824-3525  
Phone: (603) 862-2804  
Fax: (603) 862-4181  
<http://www.iol.unh.edu>*

## **TABLE OF CONTENTS**

MODIFICATION RECORD .....	3
ACKNOWLEDGEMENTS .....	4
INTRODUCTION .....	5
TEST ORGANIZATION.....	7
REFERENCES .....	8
GROUP 1: Basic Firewall Interoperability.....	9
Test NFWW.1.1: Redundant Perimeter Defense Model .....	10
Test NFWW.1.2: Perimeter Defense Model .....	11
Test NFWW.1.3: External Router .....	12
Test NFWW.1.4: Routers.....	13

## **MODIFICATION RECORD**

Draft Version Complete August 11, 2004

Version 0.3 August 15, 2004

Version 0.5 September 9, 2004

Created a new set of test plans including Policy, Base Firewall and Firewall Interoperability.

Version 1.0 October 31, 2004

Scanned for errors and finished discussions and references. Approved for release.

## **ACKNOWLEDGEMENTS**

**The University of New Hampshire would like to acknowledge the efforts of the following individuals in the development of this test suite. This test suite belongs to the University of New Hampshire InterOperability Lab and is a collaborative effort of those listed below and the participants of Moonv6. Special thanks to Secure Computing for contributing the original test ideas for this document.**

Ankur Chadda	University of New Hampshire
Paul Meyer	Secure Computing
Jeff Pomeroy	Secure Computing
Benjamin Schultz	University of New Hampshire
L. Brad Upson	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 implementers evaluate the functioning of the policy functionality of firewall products. The tests do not determine if a product conforms to any specifications, nor are they purely interoperability tests. Rather, they provide one method to isolate problems within a device. Successful completion of all tests contained in this suite does not guarantee that the tested device will interoperate with any other devices. However, combined with satisfactory operation in the IOL's semi-production environment, these tests provide a reasonable level of confidence that the Device Under Test will function well in most multi-vendor environments.

In this test suite, when using interface oriented terms such as "accept...on the interface" or "configure ... on its interface", it is up to the equipment vendor to supply the desired functionality according to the implementation of the DUT. The term "interface" only describes the externally observable behavior, not the specifics of an internal configuration.

## Acronyms

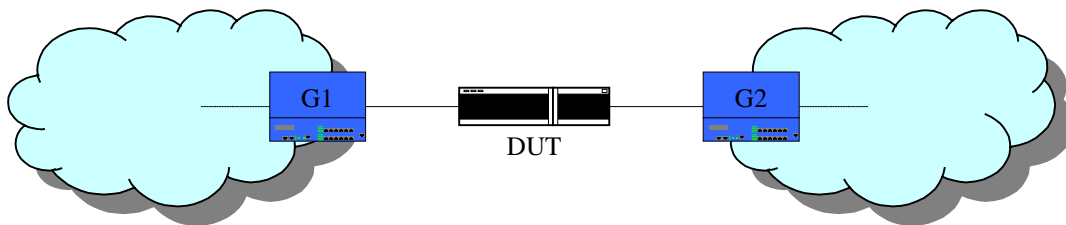
**DUT:** Device Under Test

**TR:** Testing Router

**G:** Traffic Generator

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 traffic generators in the test configuration, they would be labeled G1, G2 and G3.

## Test Configuration



Basic Test Configuration

Traffic is passed from G1 to G2 via the DUT. The DUT may be configured as a router for some of the tests that contain destination traffic to more than one network. When the term "Network Address" is used in the procedures below, it means that there is a range of IP addresses transmitted to a certain prefix that represents the destination subnet. The tests below have a variation of topologies, primarily in which a router and/or firewall is inserted between one or both of the traffic generators.

When a packet is denied by the DUT, there are 2 options. One is to send an error back to the origin. This may be acceptable for some network configurations. The alternative is to silently discard the packet. While this is the more secure option, it may limit troubleshooting, especially if the network administrator has devices outside the firewall, such as that in a multi-site topology. In some cases, the testing may be run twice to observe the two alternate behaviors and verify they can be enabled and disabled.

## TEST ORGANIZATION

This document organizes tests by group 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 group number, and the test number within the group, separated by periods. The Test Number is the group and test number, also separated by a period. So, test label NFWF.1.2 refers to the second test of the first test group in the Network-Level Firewall test suite. The test number is 1.2.
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.
Last Modification Discussion:	The last date this test was modified.  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 packet 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 DUT 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 DUT'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:

- [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.
  
- [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.
  
- [PMTU] McCann, J., S. Deering, and J. Mogul, Path MTU Discovery for IPv6, RFC 1981, August 1996.
  
- [ADDR] Hinden, R., S. Deering, IP Version 6 Addressing Architecture, RFC 2373, July 1998.
  
- [IP] Jon Postel, Editor. Internet Protocol: DARPA INTERNET PROGRAM PROTOCOL SPECIFICATION, RFC 791, September 1981.
  
- [TCP] Jon Postel, Editor. Internet Protocol: DARPA INTERNET PROGRAM PROTOCOL SPECIFICATION, RFC 793, September 1981.
  
- [UDP] Jon Postel. User Datagram Protocol, RFC 768, August 1980.
  
- [RFC2827] P. Ferguson and D. Senie, Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing. RFC 2827, May 2000.
  
- [RFC3013] T. Killalea, Recommended Internet Service Provider Security Services and Procedures. RFC 3013, November 2000.
  
- [RFC3775] D. Johnson, C. Perkins and J. Arkko, Mobility Support in IPv6. RFC 3775, June, 2004.
  
- [FTP] Jon Postel. File Transfer Protocol (FTP), RFC 768, October 1985.
  
- [RFC1858] G. Ziemba, D. Reed and P. Traina. Security Considerations for IP Fragment Filtering, RFC 1858, October 1995.
  
- [IEEE802] IEEE Std 802-1990. IEEE Standards for Local and Metropolitan Area Networks: Overview and Architecture. December, 1990.

## **GROUP 1: Basic Firewall Interoperability**

### **Scope:**

These following tests are designed to verify basic interoperability of Firewall devices.

### **Overview:**

## Test NFWW.1.1: Redundant Perimeter Defense Model

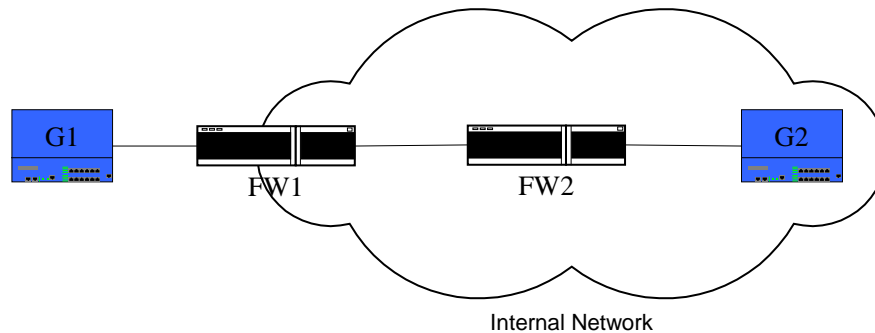
**Purpose:** To verify that a device properly operates in a redundant Perimeter Defense environment.

**References:** IP

**Resource Requirements:** Monitor to capture packets, generators

**Discussion:** Basic Firewall interoperability includes the ability forward applications traffic between two different firewall devices. In cases which the firewall has proxy capability, it is very important that both firewalls can accept and regenerate a traffic stream. This is especially true on an enterprise network in which there are two “nested” firewall implementations. These are usually different implementations to provide a higher level of security, as two implementations are less likely to have the same exploitable bugs.

**Test Setup:** Connect Devices as shown below. Configure static routing between the firewalls or enable an IP Routing protocol and properly configure it.



### Procedure:

Part A: Traffic forwarding between the two firewalls

1. Configure FW1 and FW2 to deny traffic from a range of destination IPv4 unicast addresses on the interfaces connected towards the external network (to G1).
2. From G1, transmit traffic to G2 containing...
3. From G1, transmit traffic to G2 containing....
4. Observe the packets transmitted by the DUT on G2.

Part B: Application forwarding between the two firewalls

5. Configure FW1 and FW2 to deny traffic from a range of destination IPv4 unicast addresses on the interfaces connected towards the external network (to G1).
6. From G1, transmit traffic to G2 containing...
7. From G1, transmit traffic to G2 containing....
8. Observe the packets transmitted by the DUT on G2.

### Observable Results:

- In Parts A and B, the traffic transmitted from G1 should be forwarded by the DUT and observed on G2.

**Possible Problems:** None.

## Test NFWF.1.2: Perimeter Defense Model

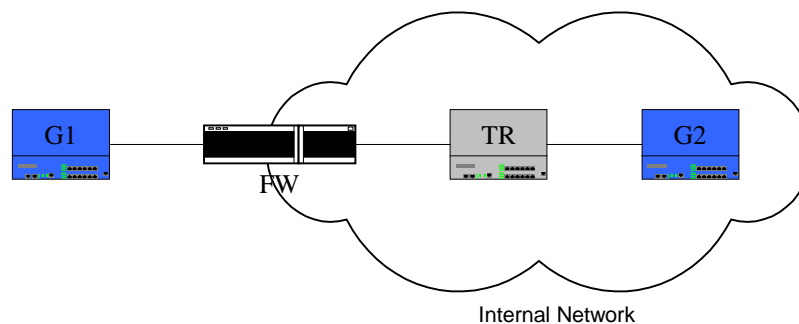
**Purpose:** To verify that a Firewall properly operates in a Perimeter Defense environment.

**References:** IPv6-SPEC

**Resource Requirements:** Monitor to capture packets, packet generators

**Discussion:** The most basic firewall model, Perimeter Defense protects a network from external attacks such as Denial of Service, as well as machines that scan servers and workstations for open ports and other vulnerabilities.

**Test Setup:** Connect Devices as shown below.



### Procedure:

#### Part A: *Deny traffic originating from a globally defined IPv6 Address*

1. Configure the DUT to deny traffic from a globally defined source IPv6 address on its interface connected to G1.
2. From G1, transmit traffic to G2 containing the source IPv6 address configured in the previous step.
3. From G1, transmit traffic to G2 containing a global source address that is not denied per the configuration.
4. Observe the packets transmitted by the DUT on G2.

#### Part B: *Deny traffic originating from a link local IPv6 Address*

5. Configure the DUT to deny traffic from a link local defined source IPv6 address on its interface connected to G1.
6. From G1, transmit traffic to G2 containing the source IPv6 address configured in the previous step.
7. From G1, transmit traffic to G2 containing a global source address that is not denied per the configuration.
8. Observe the packets transmitted by the DUT on G2.

### Observable Results:

- In all Parts, the traffic transmitted from denied addresses/networks on G1 should not be forwarded by the DUT nor observed on G2. The traffic transmitted from the accepted address/networks at G1 should be forwarded by the DUT and observed on G2.

**Possible Problems:** None.

### Test NFWF.1.3: External Router

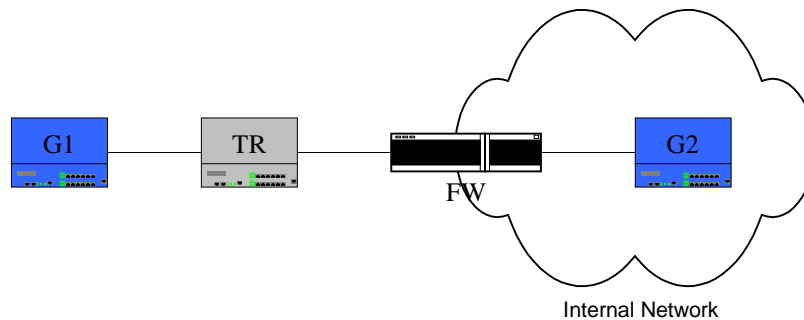
**Purpose:** To verify that a device properly operates with an external router.

**References:** IPv6-SPEC

**Resource Requirements:** Monitor to capture packets, generators

**Discussion:**

**Test Setup:** Connect Devices as shown below.



#### Procedure:

Part A: Deny traffic with a global IPv6 Address Destination

1. Configure the DUT to deny traffic from a globally defined destination IPv6 address on the interface connected to G1.
2. From G1, transmit traffic to G2 containing the destination IPv6 address configured in the previous step.
3. From G1, transmit traffic to G2 containing a destination address that is not denied per the configuration.
4. Observe the packets transmitted by the DUT on G2.

#### Observable Results:

- In all Parts, the traffic transmitted from denied addresses/networks on G1 should not be forwarded by the DUT nor observed on G2. The traffic transmitted from the accepted address/networks at G1 should be forwarded by the DUT and observed on G2.

**Possible Problems:** None.

## Test NFWW.1.4: Routers

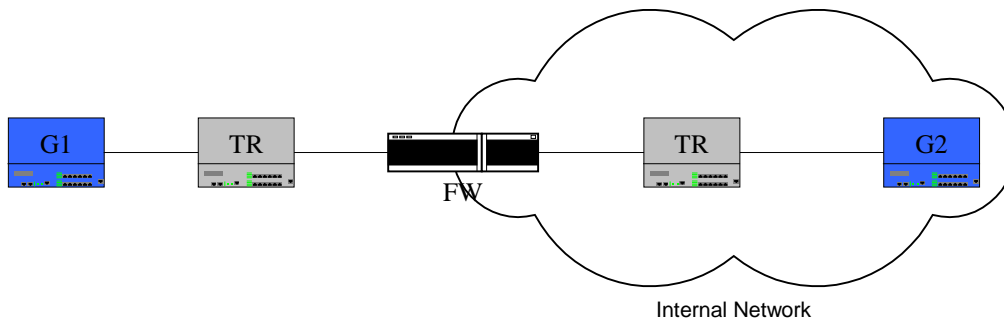
**Purpose:** To verify that a device properly accepts and denies UDP port numbers based on policy.

**References:** UDP

**Resource Requirements:** Monitor to capture packets, generators

**Discussion:**

**Test Setup:** Connect Devices as shown below



### Procedure:

Part A: Deny traffic with a specific UDP Port Destination

1. Configure the DUT to deny traffic containing a destination UDP port on the interface connected to G1.
2. From G1, transmit traffic containing the destination UDP port configured in the previous step.
3. From G1, transmit traffic to G2 containing a valid destination UDP port.
4. Observe the packets transmitted by the DUT on G2.

Part B: Deny traffic with a specific UDP Port Source

5. Configure the DUT to deny traffic containing a source UDP port on the interface connected to G1.
6. From G1, transmit traffic to G2 containing the source UDP port configured in the previous step.
7. From G1, transmit traffic to G2 containing a valid source UDP port.
8. Observe the packets transmitted by the DUT on G2.

Part C: Accept all traffic with a specific UDP Port Destination

9. Configure the DUT to accept all traffic containing a destination UDP port on the interface connected to G1. Would you want a destination for G2?
10. From G1, transmit traffic to G2 containing the destination UDP port configured in the previous step.
11. Observe the packets transmitted by the DUT on G2.

Part D: Accept all traffic with a specific UDP Port Source

12. Configure the DUT to accept all traffic containing a source UDP port on the interface connected to G1.
13. From G1, transmit traffic to G2 containing the source UDP port configured in the previous step.
14. Observe the packets transmitted by the DUT on G2.

**Observable Results:**

- In Parts A and B, the traffic transmitted from denied UDP port on G1 should not be forwarded by the DUT nor observed on G2. The traffic transmitted from the valid UDP port at G1 should be forwarded by the DUT and observed on G2.
- In Parts C and D, the traffic transmitted from G1 should be forwarded by the DUT and observed on G2.

**Possible Problems:** None.