IPv6 CONSORTIUM
Moonv6 Project

Application Interoperability Test Suite

Technical Document

Revision 1.1

University of New Hampshire
InterOperability Laboratory
IPv6 Consortium
http://www.iol.unh.edu

121 Technology Drive, Suite 2
Durham, NH 03824
Phone: +1-603-862-3941
Fax: +1-603-862-4181

MODIFICATION RECORD

Version 1.1	February 27, 2007: Updated for Moonv6 project
Version 1.0	September 1st, 2006: Initial Version

TABLE OF CONTENTS

University of New Hampshire
InterOperability Laboratory

ACKNOWLEDGEMENTS

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

Serena Zhao	Adobe Systems
Erica Williamsen	University of New Hampshire
Timothy Winters	University of New Hampshire
Ben Schultz	University of New Hampshire

INTRODUCTION

Overview

The University of New Hampshire's InterOperability Laboratory (IOL) is an institution designed to improve the interoperability by providing an environment where a product can be tested against other implementations and platforms. This suite of tests has been developed to help implementers evaluate their applications in an IPv6 environment. The tests do not determine if a product conforms to the specifications, nor are they purely interoperability tests. Rather, they provide a method to isolate problems within a device.

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 APP_Interop.1.2 refers to the second test of the first test group in the Application Interop 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.
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 transmitting 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 TR 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 TR'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.

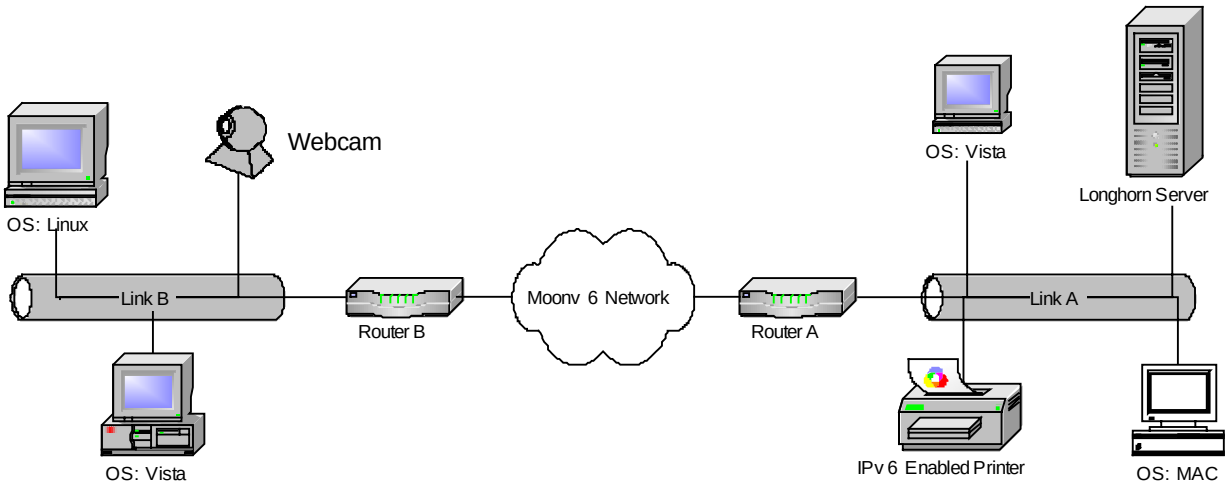
REFERENCES

The following documents are referenced in this text:

- [ADDR] Hinden, R., S. Deering, Internet Protocol Version 6 (IPv6) Addressing Architecture, RFC 3513, April 2003.
- [ADDR-Global] Hinden, R., S. Deering, Nordmark, E., IPv6 Global Unicast Address Format, RFC 3587, August 2003.
- [DNS] M. Crawford, C.Huitema. DNS Extensions to support IPv6 Address Aggregation and Renumbering. RFC 2874, July 2000.
- [FTP] J. Postel, J.K. Reynolds. File Transfer Protocol, RFC 959, October 1985.
- [FTP-IPv6] M. Allman, S. Ostermann and C. Metz. FTP Extensions for IPv6 and NATs, RFC 2428, September 1998.
- [HTTP] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee. Hypertext Transfer Protocol -- HTTP/1.1. RFC 2616, June 1999.
- [IPsec] Kent, S., Atkinson, R., IP Encapsulating Security Payload (ESP), RFC 2406, November 1998
- [IPv6-URL] Hinden, R., Carpenter, B., Masinter, L., Format for Literal IPv6 Addresses in URL's, RFC 2732, December 1999.
- [ND] Narten,T., Nordmark, E., and W. Simpson, Neighbor Discovery for IP Version 6 (IPv6), RFC 2461, December 1998
- [3363] R. Bush, et. al. Representing Internet Protocol version 6 (IPv6) Addresses in the Domain Name System (DNS), RFC 3363, August 2002.
- [3596] S. Thomson, et. al. DNS Extensions to Support IP Version 6, RFC 3596, October, 2003.

Common Topology

This topology is used for all tests in this test suite unless otherwise noted.



Common Test Setup

Tests in this test suite may refer to common test setup procedure defined for this section.

Common Test Setup 1.1
Summary: This minimal setup procedure provides the host devices with a default router, a global prefix and ensures the communication between Link A and Link B. This topology assumes dual stack routing.

1. Setup a static route between Router A and Router B.
2. Enable DNS services on the Server.
3. Router A transmits router advertisements with prefix “X” onto Link A.
4. Router B transmits router advertisements with prefix “Y” onto Link B.
5. Host on Link A transmits an ICMPv6 Echo Request to a Host on Link B.

Section 1: Network Protocol Interaction

Scope:

The following tests are designed to verify that a user may manipulate an application in IPv6 environments while communicating between nodes on a network. The focus of this section is on specifications for the application.

The scope of this section includes major functionality groups such as: FTP, SFTP, HTTP, SMB, DNS and Printing. These include target areas to verify address format, address stored length, and parsing.
This section does not offer an exhaustive set of tests, but rather provides test cases to verify the operation of these user actions and network protocol interaction.

Overview:

These tests are designed to verify that the application can be properly configured and used in an IPv6 environment.

Test APP_Interop.1.1: File Transfer Protocol (FTP)

Purpose: To verify the Application properly uses FTP in an IPv6 network.

References:

- [FTP]
- [FTP-IPv6]
- [ADDR]
- [ADDR-Global]

Discussion: Once IPv6 communication is established between the client and the server, both participants must be able to send and receive files accordingly.

Test Setup: **Common Test Setup 1.1** is performed.

Procedure:

Part A: Unsuppressed Literal IPv6 Link-Local Address

1. From the Application, connect to the FTP server using the unsuppressed IPv6 link-local address.
2. Observe the Application and all packets on Link A.
3. From the Application, navigate over the network through a file system at a remote location.
4. From the Application, request a file from the remote location using the “get” command provided by the FTP application.
5. Observe the Application.
6. From the Application, send a file to a remote location using the “put” command provided by the FTP application.
7. Observe the Application.

Part B: Suppressed Literal IPv6 Link-Local Address

8. From the Application, connect to the FTP server using the suppressed IPv6 link-local address.
9. Observe the Application and all packets on Link A.
10. From the Application, navigate over the network through a file system at a remote location.
11. From the Application, request a file from the remote location using the “get” command provided by the FTP application.
12. Observe the Application and all packets on Link A.
13. From the Application, send a file to a remote location using the “put” command provided by the FTP application.
14. Observe the Application and all packets on Link A.

Part C: IPv6 Link-Local Address with Upper Case

15. From the Application, connect to the FTP server using the suppressed IPv6 link-local address with upper case.
16. Observe the Application and all packets on Link A.
17. From the Application, navigate over the network through a file system at a remote location.
18. From the Application, request a file from the remote location using the “get” command provided by the FTP application.
19. Observe the Application and all packets on Link A.
20. From the Application, send a file to a remote location using the “put” command provided by the FTP application.
21. Observe the Application and all packets on Link A.

Part D: Unsuppressed Literal Global Address

22. From the Application, connect to the FTP server using the unsuppressed IPv6 on link global address.
23. Observe the Application and all packets on Link A.
24. From the Application, navigate over the network through a file system at a remote location.
25. From the Application, request a file from the remote location using the "get" command provided by the FTP application.
26. Observe the Application and all packets on Link A.
27. From the Application, send a file to a remote location using the "put" command provided by the FTP application.
28. Observe the Application and all packets on Link A.

Part E: Suppressed Literal Global Address

29. From the Application, connect to the FTP server using the suppressed IPv6 on link global address.
30. Observe the Application and all packets on Link A.
31. From the Application, navigate over the network through a file system at a remote location.
32. From the Application, request a file from the remote location using the "get" command provided by the FTP application.
33. Observe the Application and all packets on Link A.
34. From the Application, send a file to a remote location using the "put" command provided by the FTP application.
35. Observe the Application and all packets on Link A.

Part F: Host Name

36. From the Application, connect to the FTP server using a host name.
37. Observe the Application and all packets on Link A.
38. From the Application, navigate over the network through a file system at a remote location.
39. From the Application, request a file from the remote location using the "get" command provided by the FTP application.
40. Observe the Application and all packets on Link A.
41. From the Application, send a file to a remote location using the "put" command provided by the FTP application.
42. Observe the Application and all packets on Link A.

Part G: Invalid IPv6 Address

43. From the Application, connect to the FTP server using the invalid IPv6 address fe80:0000:20c:1111:2222:3333.
44. Observe the application and all packets on Link A.

Observable Results:

- *Part A*

Step 2: The Application was able to successfully connect to the ftp server. The Application was able to obtain the IPv6 address of the destination host through DNS services. The Application was able to view the file.
Step 5: The Application was able to receive an uncorrupted copy of the requested file.
Step 7: The remote location was able to receive an uncorrupted copy of the requested file.

- *Part B*
  Step 9: The Application was able to successfully connect to the ftp server. The Application was able to obtain the IPv6 address of the destination host through DNS services. The Application was able to view the file.
  Step 12: The Application was able to receive an uncorrupted copy of the requested file.
  Step 14: The remote location was able to receive an uncorrupted copy of the requested file.
- *Part C*
  Step 16: The Application was able to successfully connect to the ftp server. The Application was able to obtain the IPv6 address of the destination host through DNS services. The Application was able to view the file.
  Step 19: The Application was able to receive an uncorrupted copy of the requested file.
  Step 21: The remote location was able to receive an uncorrupted copy of the requested file.
- *Part D*
  Step 23: The Application was able to successfully connect to the ftp server. The Application was able to obtain the IPv6 address of the destination host through DNS services. The Application was able to view the file.
  Step 26: The Application was able to receive an uncorrupted copy of the requested file.
  Step 28: The remote location was able to receive an uncorrupted copy of the requested file.
- *Part E*
  Step 30: The Application was able to successfully connect to the ftp server. The Application was able to obtain the IPv6 address of the destination host through DNS services. The Application was able to view the file.
  Step 33: The Application was able to receive an uncorrupted copy of the requested file.
  Step 35: The remote location was able to receive an uncorrupted copy of the requested file.
- *Part F*
  Step 37: The Application was able to successfully connect to the ftp server. The Application was able to obtain the IPv6 address of the destination host through DNS services. The Application was able to view the file.
  Step 40: The Application was able to receive an uncorrupted copy of the requested file.
  Step 42: The remote location was able to receive an uncorrupted copy of the requested file.
- *Part G*
  Step 44: The Application was unable to connect to the ftp server.

Possible Problems:

- The Application may not support the FTP application.
- The host or OS does not support a DNS client.

Test APP_Interop.1.2: Server Message Block (SMB)

Purpose: To verify the Application properly uses SMB in an IPv6 network.

References:
- [ADDR]
- [ADDR-Global]
- [3363]
- [3596]

Discussion: SMB is a network application level protocol commonly used to communicate and share between nodes on a network. The following tests ensure the ability to exchange information over the network using SMB in IPv6 environments.

Test Setup: **Common Test Setup 1.1** is performed.

Procedure:

Part A: Unsuppressed Literal IPv6 Link-Local Address
1. From the Application, browse files over the network using the unsuppressed IPv6 link-local address of the remote location.
2. Observe the Application and all packets on Link A.
3. From the Application, open a file over the network using the unsuppressed IPv6 link-local address of the remote location.
4. Observe the Application and all packets on Link A.
5. Alter and save the file at the remote location.
6. Observe the application and all packets on Link A.

Part B: Suppressed Literal IPv6 Link-Local Address
7. From the Application, browse files over the network using the suppressed IPv6 link-local address of the remote location.
8. Observe the Application and all packets on Link A.
9. From the Application, open a file over the network using the suppressed IPv6 link-local address of the remote location.
10. Observe the Application and all packets on Link A.
11. Alter and save the file at the remote location.
12. Observe the Application and all packets on Link A.

Part C: IPv6 Link-Local Address with Upper Case
13. From the Application, browse files over the network using the suppressed IPv6 link-local address with upper case of the remote location.
14. Observe the Application and all packets on Link A.
15. From the Application, open a file over the network using the suppressed IPv6 link-local address with upper case of the remote location.

16. Observe the Application and all packets on Link A.
17. Alter and save the file at the remote location.
18. Observe the Application and all packets on Link A.

Part D: Unsuppressed Literal Global Address

19. From the Application, browse files over the network using the unsuppressed IPv6 global address of the remote location.
20. Observe the Application and all packets on Link A.
21. From the Application, open a file over the network using the unsuppressed IPv6 global address of the remote location.
22. Observe the Application and all packets on Link A.
23. Alter and save the file at the remote location.
24. Observe the Application and all packets on Link A.

Part E: Suppressed Literal Global Address

25. From the Application, browse files over the network using the suppressed IPv6 global address of the remote location.
26. Observe the Application and all packets on Link A.
27. From the Application, open a file over the network using the suppressed IPv6 global address of the remote location.
28. Observe the Application and all packets on Link A.
29. Alter and save the file at the remote location.
30. Observe the Application and all packets on Link A.

Part F: Host Name

31. From the Application, browse files over the network using a host name of the remote location.
32. Observe the Application and all packets on Link A.
33. From the Application, open a file over the network using the host name of the remote location.
34. Observe the Application and all packets on Link A.
35. Alter and save the file at the remote location.
36. Observe the Application and all packets on Link A.

Part G: Invalid IPv6 Address

37. From the Application, browse files over the network using the fe80:0000:20c:1111:2222:3333 address of the remote location.
38. Observe the Application and all packets on Link A.

Observable Results:

- *Part A*
  Step 2: The Application was able to view the file.
  Step 4: The Application was able to open the file from the remote location.
  Step 6: The Application was able to save the file to the remote location.
- *Part B*
  Step 8: The Application was able to view the file.
  Step 10: The Application was able to open the file from the remote location.

Step 12: The Application was able to save the file to the remote location.

- *Part C*
  Step 14: The Application was able to view the file.
  Step 16: The Application was able to open the file from the remote location.
  Step 18: The Application was able to save the file to the remote location.
- *Part D*
  Step 20: The Application was able to view the file.
  Step 22: The Application was able to open the file from the remote location.
  Step 24: The Application was able to save the file to the remote location.
- *Part E*
  Step 26: The Application was able to view the file.
  Step 28: The Application was able to open the file from the remote location.
  Step 30: The Application was able to save the file to the remote location.
- *Part F*
  Step 32: The Application was able to obtain the IPv6 address of the destination host through DNS services. The Application was able to view the file. The Application was able to view the file.
  Step 34: The Application was able to open the file from the remote location.
  Step 36: The Application was able to save the file to the remote location.
- *Part G*
  Step 38: The Application was unable to view the file.

Possible Problems:

- The Application may not allow a literal IPv6 address to be entered into the interface.
- The OS on the server or client may not support the SMB application over IPv6.
- The host or OS does not support a DNS client.

Test APP_Interop.1.3: HTTP/URL Parsing

Purpose: To verify the Application properly uses HTTP and parses URL information in an IPv6 network.

References:
- [HTTP]
- [IPv6-URL]
- [ADDR]
- [ADDR-Global]

Discussion: The Hypertext Transfer Protocol (HTTP) is an application-level protocol. This application is generally used for communicating between users and other Internet systems. IPv6 environments must not hinder this communication.

The literal IPv6 address is represented in a URL using the literal address enclosed in "[" and "]" characters. This format has been implemented in widely deployed browsers for IPv6 environments.

Test Setup: **Common Test Setup 1.1** is performed.

Procedure:

Part A: Parse IPv6 Literal Unsuppressed Link-Local Address
1. The Application browses to a webpage on a remote box using the unsuppressed link-local address.
2. Observe the Application and all packets on Link A.

Part B: Parse IPv6 Literal Suppressed Link-Local Address
3. The Application browses to a webpage on a remote box using the suppressed link-local address.
4. Observe the Application and all packets on Link A.

Part C: Parse IPv6 Link-Local Address with Upper Case
5. The Application browses to a webpage on a remote box using the suppressed link-local with upper case address.
6. Observe the Application and all packets on Link A.

Part D: Unsuppressed Literal Global Address
7. The Application browses to a webpage on a remote box using the unsuppressed global address.
8. Observe the Application and all packets on Link A.

Part E: Suppressed Literal Global Address
9. The Application browses to a webpage on a remote box using the suppressed global address.
10. Observe the Application and all packets on Link A.

Part F: Host Name
11. The Application browses to a webpage on a remote box using a host name address.
12. Observe the Application and all packets on Link A.

Part G: Invalid URL IPv6 Address

13. The Application browses to a webpage on a remote box using an invalid URL IPv6 address syntax.
14. Observe the Application and all packets on Link A.

Observable Results:

- *Part A*
  Step 2: The Application is able to view the website.
- *Part B*
  Step 4: The Application is able to view the website.
- *Part C*
  Step 6: The Application is able to view the website.
- *Part D*
  Step 8: The Application is able to view the website.
- *Part E*
  Step 10: The Application is able to view the website.
- *Part F*
  Step 12: The Application was able to obtain the IPv6 address of the destination host through DNS services. The Application is able to view the website.
- *Part G*
  Step 14: The Application is unable to view the website.

Possible Problems:

- The Application does not support URL parsing.
- The host or OS does not support a DNS client.

Test APP_Interop.1.4: IPv6 Address Stored Length

Purpose: To verify that the application properly stores an IPv6 address length.

References:

- [ADDR]
- [ADDR-Global]

Discussion: IPv6 addresses have an increased size from the IPv4 address 32 bits to 128 bits. These addresses should be stored with the minimum character set to 39.

Test Setup: **Common Test Setup 1.1** is performed.

Procedure:

Part A: Unsuppressed Literal Link-Local IPv6 address

1. The Application is configured to store an unsuppressed link-local address.
2. Observe the Application and all packets on Link A.

Part B: Suppressed Literal Link-Local IPv6 address

3. The Application is configured to store a suppressed link-local address.
4. Observe the Application and all packets on Link A.

Part C: Unsuppressed Literal Global IPv6 address

5. The Application is configured to store an unsuppressed global address.
6. Observe the Application and all packets on Link A.

Part D: Suppressed Literal Global IPv6 address

7. The Application is configured to store a suppressed link-local address.
8. Observe the Application.

Observable Results:

- *Part A*
  Step 2: The Application must be able to store an IPv6 address.
- *Part B*
  Step 4: The Application must be able to store an IPv6 address.
- *Part C*
  Step 6: The Application must be able to store an IPv6 address.
- *Part D*
  Step 8: The Application must be able to store an IPv6 address.

Possible Problems:

- The Application might not have the ability or need to store IPv6 information.

Test APP_Interop.1.5: Printing

Purpose: To verify the application properly prints over an IPv6 network.

References:

- [ADDR]
- [ADDR-Global]

Discussion: This test ensures that applications are able to discover a printer and print a file in an IPv6 office environment.

Test Setup: **Common Test Setup 1.1** is performed.

Procedure:

Part A: Print File using Link-Local Address

1. From the Application, open a file over the network.
2. Configure a printer's link-local address as the default printer of the application.
3. Select to print the file.
4. Observe the Application and all packets on Link A.

Part B: Print File using Global Address

5. From the Application, open a file over the network.
6. Configure a printer's global address as the default printer of the application.
7. Select to print the file.
8. Observe the Application and all packets on Link A.

Observable Results:

- *Part A*
  Step 4: The Application should print the document on the printer.
- *Part B*
  Step 8: The Application should print the document on the printer.

Possible Problems:

- The Application may not support printing over IPv6.

Section 2: Application Over Networks

Scope:

The following tests are designed to verify that a user may manipulate an IPv6 environment while communicating between nodes over an IPv6 network. The focus of this section is on specifications for the application.

The scope of this section includes major functionality groups such as: Off-Link, VPN and IPsec Tunneling. These include target areas to verify FTP, SMB, HTTP, and Printing.
This section does not offer an exhaustive set of tests, but rather provides test cases to verify the operation of these user actions and network protocol interaction.

Overview:

These tests are designed to verify that the application can be properly used over various IPv6 network implementations.

Test APP_Interop.2.1: Off-Link

Purpose: To verify the application properly uses off-link addresses for FTP, SMB, HTTP and Printing.

References:

- [ADDR-Global]
- [FTP]
- [HTTP]

Discussion: An application must be able to operate off link over an IPv6 network. The following parts concentrate on an application communicating to an off link location through ftp, smb, http and printing.

Test Setup: **Common Test Setup 1.1** is performed. The Application is moved onto Link B.

Procedure:

Part A: FTP

1. From the Application, connect to the FTP server using the IPv6 off-link global address on Link A.
2. Observe the Application and all packets on Link A and Link B.

Part B: SMB

3. From the Application, browse files over the network using the IPv6 off-link global address on Link A of the remote location.
4. Observe the Application and all packets on Link A and Link B.
5. From the Application, open a file over the network using the IPv6 off-link global address on Link A of the remote location.
6. Observe the Application and all packets on Link A and Link B.
7. Alter and save the file at the remote location.
8. Observe the Application and all packets on Link A and Link B.

Part C: HTTP

9. The Application browses to a webpage on a remote box using the IPv6 off-link global address on Link A.
10. Observe the Application and all packets on Link A and Link B.

Part D: Printing

11. From the Application, open a file over the network.
12. Configure a printer's global address on Link A as the default printer of the application.
13. Select to print the file.
14. Observe the Application and all packets on Link A and Link B.

Observable Results:

- *Part A*
  Step 2: The Application was able to successfully connect to the ftp server.
- *Part B*
  Step 4: The Application was able to view the file.
  Step 6: The Application was able to open the file from the remote location.
  Step 8: The Application was able to save the file to the remote location.
- *Part C*

Step 10: The Application was able to view the website.

- *Part D*

  Step 14: The Application should print the document on the printer.

Possible Problems:

- The Application may not support all these features.

Test APP_Interop.2.2: IPsec Tunnel

Purpose: To verify the application properly uses IPsec Tunnel.

References:

- [ADDR-Global]
- [FTP]
- [HTTP]
- [IPsec]

Discussion: An application must be able to operate over an IPv6 IPsec Tunnel. The following parts concentrate on an application communicating over an IPsec tunnel to a remote location through ftp, smb, http and printing.

Test Setup: **Common Test Setup 1.1** is performed. The Application is moved onto Link B. An IPsec Tunnel between Link A and Link B is setup.

Procedure:

Part A: FTP

1. From the Application, connect to the FTP server on Link A over the IPsec Tunnel.
2. Observe the Application and all packets on Link A and Link B.

Part B: SMB

3. From the Application, browse files on the FTP server on Link A using the over the IPsec Tunnel.
4. Observe the Application and all packets on Link A and Link B.
5. From the Application, open a file over the network using the address on Link A over the IPsec Tunnel.
6. Observe the Application and all packets on Link A and Link B.
7. Alter and save the file at the remote location.
8. Observe the Application and all packets on Link A and Link B.

Part C: HTTP

9. The Application browses to a webpage on a remote box using the address on Link A over the IPsec Tunnel.
10. Observe the Application and all packets on Link A and Link B.

Part D: Printing

11. From the Application, open a file over the network.
12. Configure a printer's global address on Link A as the default printer of the application.
13. Print the file over the IPsec Tunnel.
14. Observe the Application and all packets on Link A and Link B.

Observable Results:

- *Part A*
  Step 2: The Application was able to successfully connect to the FTP server.
- *Part B*

Step 4: The Application was able to view the file.
Step 6: The Application was able to open the file from the remote location.
Step 8: The Application was able to save the file to the remote location.

- *Part C*
  Step 10: The Application was able to view the website.
- *Part D*
  Step 14: The Application should be print the document on the printer.

Possible Problems:

- The Application may not support all these features.