

---

# File Transfer IPv6 Readiness

Includes FTP, FTP/S, SFTP and HTTP Interoperability Testing with Select Applications

---

Andy White, CTO

File Transfer Consulting, LLC

[www.filetransferconsulting.com](http://www.filetransferconsulting.com) – ftexperts on Twitter

608-208-1800 (US)

June 8, 2011

(ISOC's "World IPv6 Day")

## Abstract

The world is out of IPv4 addresses and the time of IPv6 is at hand. This paper explores existing file transfer technology used for bulk data exchange under the lens of IPv6 and highlights issues to consider during strategic or tactical planning. Summary results from a series of IPv6 interoperability tests illuminate the findings and recommendations.

## Table of Contents

Abstract.....	1
Table of Contents.....	2
Background.....	3
File Transfer Application Concerns.....	3
IPv6 Interoperability of File Transfer Software.....	4
Approach and Obstacles.....	4
File Transfer Client Selection.....	4
File Transfer Server Selection.....	5
File Transfer Test Environment.....	5
File Transfer Interoperability Results.....	6
Interoperability Summary.....	6
Conclusions and Recommendations.....	7
Additional Assistance.....	8
Appendix A: Detailed Client/Server Interoperability Results.....	9
Apache FTPServer 1.0.5.....	9
ProFTPD Server 1.3.3e.....	9
Pure-FTPd Server 1.0.32.....	9
CrushFTP Server 5.6.....	10
JSCAPE MFT Server 7.2.....	10
RhinoSoft.com Serv-U 11.0.....	10
Appendix B: Legal, Acknowledgements and Reference.....	11
Trademarks and Project Recognition.....	11
Disclaimer.....	11
Acknowledgments.....	11
Reference.....	11
About File Transfer Consulting.....	11

## Background

Since the 1990s the IT community has known that “IPv4” blocks of IP addresses like “55.55.55.55” would soon be exhausted. In response, an “IPv6” addressing scheme that provides almost unlimited addresses was created, formalized in a series of RFCs and rolled out in networking hardware and operating systems.

Now that IANA and APNIC have both run out of new blocks of IPv4 addresses, the time for IPv6 has arrived, but many IT administrators and architects still have no plan to adopt IPv6. There are at least three common fallacies driving the complacency:

- **“There is nothing to see over IPv6”** – Why false? As of June 8, 2011 - ISOC’s “World IPv6 Day”, the world’s top web sites (e.g., Google, Yahoo, Facebook, YouTube, etc.) will ALL expose IPv6 support, and millions of other web sites will become accessible around the world.
- **“I can use IPv4 NAT to put off IPv6”** – Why false? New blocks of IPv4 addresses have been exhausted and ISPs are already making it more difficult to get new IPv4 addresses. In the future organizations may also be forced to give up their IPv4 addresses or pay more to keep them.
- **“IPv6 will be a non-event: just turn it on and it works”** – Why false? Applications written for IPv4 addresses often have problems with address-dependent tasks such as interpreting IPv6 access lists or writing IPv6 addresses into their logs.

This paper addresses the third fallacy to help IT departments develop their IPv6 strategy by demonstrating that migrating file transfer applications to IPv6 will often require planning and testing. It explores a number of existing open source and commercial packages that can help with the transition and highlights areas to keep in mind as IT departments plan, test and migrate.

## File Transfer Application Concerns

Many applications built for IPv4 connectivity face common IPv6 concerns, including:

- IPv6 addresses in access control lists (including “IP lock-outs”)
- Logging and auditing of IPv6 addresses
- Binding to IPv6 listener sockets or creating new sockets

File transfer applications require complete answers to these concerns and more, especially when protocols that involve multiple connections are in use.

To illustrate our multiple connection concern, consider how FTP uses multiple connections for its “control” and “data” channels. To link these channels, legacy FTP uses IPv4-based PORT and PASV (passive) commands while modern FTP implementations use replacement commands (EPRT and EPSV) that work with either IPv4 or IPv6. Any differences in these commands in any combination of FTP clients and servers can lead to incompatibilities, even if the applications support most elements of IPv6.

## IPv6 Interoperability of File Transfer Software

Given our concerns with file transfer applications under IPv6, the goals of our testing and research were:

- Determine how ready file transfer applications were for IPv6.
- Find out if file transfer applications from different vendors could interoperate.

There are dozens of protocols used to transfer files today but we concentrated on just four widely available vendor-neutral protocols for this paper.

- **FTP** – Insecure channels; updated for IPv6 as per RFC 2428.
- **FTP/S** – Secure channels; using SSL/TLS as per RFC 4217.
- **SFTP** – File transfer over an SSH-secured channel.
- **HTTP** – World’s most common file transfer protocol.

### Approach and Obstacles

While there are a number of IPv6 certifications for network equipment and IPv6 certifications for network engineers, there are currently no generally acknowledged IPv6 certification processes for applications or software. Nonetheless we wanted our tests to be accessible and repeatable, so we adopted much of our testing procedure from well-established product interoperability procedures.

Locating IPv6-ready network equipment and operating systems was not an issue. Many routers already advertise IPv6 compatibility and modern Windows and Linux platforms provide good IPv6 compatibility out of the box.

However, we did encounter significant problems trying to pass IPv6 traffic through ISPs using consumer-grade networking (e.g., DSL and cable modems) and more problems with IPv6 in virtual server environments. In the first case we compensated using IPv6-to-IPv4 “tunnels” (both “6on4” and “6in4”); in the second we switched virtual machine providers until we found one that worked. (*Additional compatibility notes are on our web site – <http://www.filetransferconsulting.com>.*)

### File Transfer Client Selection

We initially selected six file transfer clients that advertised IPv6 compliance for testing. We chose two common open source clients (FileZilla and FireFTP), a free client that shipped with the popular Microsoft Windows operating system (Microsoft FTP) and three leading commercial clients (JSCAPE AnyClient, GlobalSCAPE CuteFTP and SmartSoft SmartFTP).

Commercial clients and projects that do not yet claim IPv6 support were not included in our study; we look forward to including these clients in a future study.

We first vetted all potential clients for FTP and FTP/S connectivity against a “neutral” server (Apache FTPServer 1.0.5) using IPv4 connectivity. All clients passed, but it should be noted that we included Microsoft FTP even though it does not support FTP/S under either IPv4 or IPv6.

Next, we tested potential clients using the same server using IPv6 connectivity. At this point we screened Globalscape CuteFTP out because it could not resolve IPv6 hostnames, and therefore could not connect. We also considered omitting JSCAPE AnyClient due to version 2.3 issues performing file transfer functions and directory listings over FTP and FTPS. However, we ultimately accepted an advance version of AnyClient (v3.0) into our final test matrices because it included significant IPv6 enhancements and worked well against our test servers.

After reducing our client pool to five, we tested for SFTP functionality over SSH v2.0 against OpenSSH 5.8p1 using IPv6 connectivity and found no additional problems.

Finally, we added three popular web browser clients (Microsoft Internet Explorer – or “IE”, Mozilla Firefox and Google Chrome) to perform HTTP testing. (Browser FTP support was not tested.)

## File Transfer Server Selection

We selected six file transfer servers that offer IPv6 support to test against our selected file transfer clients and web browsers. We chose three common open source servers (Apache FTPServer, ProFTPD and Pure-FTPd) and three commercial servers (RhinoSoft.com Serv-U, JSCAPE MFT Server and CrushFTP).

Commercial servers and projects that do not offer IPv6 support were not included in our study; we look forward to including these servers in a future study. We also excluded “managed file transfer” servers from this round of interoperability testing, but intend to include this technology in a future study.

No servers were screened out of the process during initial testing: all reliably supported some or all of the IPv6 attributes and services required for our tests.

## File Transfer Test Environment

The test environment for our interoperability testing consisted of a simple platform architecture, predefined test sets with minimal client configuration. The platform architecture behind the test was a single Windows machine for client testing, a single Linux machine running under a virtual machine for server testing and an IPv6 capable router for network connectivity.

Each test set involved establishing a connection to the server using an IPv6 hostname, signing on, uploading and downloading files, changing directories and requesting directory listings. These tests were repeated for FTP, FTP/S and SFTP, and similar procedures were used for HTTP to access our commercial servers over their web-based transfer interfaces.

Some clients required explicit configuration settings to properly connect and operate using IPv6 over certain protocols; these options are detailed on our web site at <http://www.filetransferconsulting.com>.

If a client successfully performed all tests against a server using a particular protocol, it earned a “PASS” result. If it failed one or more tests, it earned a “FAIL” result.

## File Transfer Interoperability Results

The top file transfer clients in our tests were FileZilla and SmartFTP: all had perfect “PASS” ratings against all six of our servers. FireFTP followed next with some occasional FTP/S issues and Microsoft FTP came next with a single FTP issue. Following close behind was JSCAPE AnyClient which was perfect when using FTP/S or SFTP but had problems using FTP with IPv6 hostnames (connections using explicit IP addresses generally worked fine).

Three of our six file transfer servers achieved perfect IPv6 “PASS” ratings: Apache FTP, ProFTPD and RhinoSoft.com Serv-U. The top server group is followed closely by ProFTPD (one set of “MIXED” results), then CrushFTP (one set of “FAIL” results) and finally by JSCAPE MFT Server (one set each of “FAIL” and “MIXED” results).

### Interoperability Summary

Client (down) \ Server (across)	Apache FTP	ProFTPD	Pure-FTPd	CrushFTP	JSCAPE MFT	RhinoSoft Serv-U
<b>FileZilla 3.4.0</b>	PASS	PASS	PASS	PASS	PASS	PASS
<b>FireFTP 1.0.10</b>	PASS	MIXED <sup>1</sup>	PASS	PASS	MIXED <sup>2</sup>	PASS
<b>Microsoft FTP (FTP only)</b>	PASS	PASS	PASS	FAIL	PASS	PASS
<b>SmartFTP 4.0</b>	PASS	PASS	PASS	PASS	PASS	PASS
<b>Web Browsers (HTTP only)</b>	N/A	N/A	N/A	PASS	FAIL	PASS
JSCAPE AnyClient 3.0 <sup>3</sup>	PASS <sup>3</sup>	PASS <sup>3</sup>	PASS <sup>3</sup>	PASS <sup>3</sup>	PASS <sup>3</sup>	PASS <sup>3</sup>

1 = Passed using FTP; failed using FTP/S

2 = Passed using FTP and SFTP; failed using FTP/S

3 = Passed using SFTP and FTP/S, failed using FTP (w/ hostnames) elsewhere

Overall our interoperability results are good news for IT. They show that IPv6 adopters already have a number of open source and commercial file transfer clients and servers they can deploy today.

In the process of working with these vendors and projects we also saw promising signs that compatibility will only improve. For example, JSCAPE addressed many of the issues we initially discovered in AnyClient 3.0 patches, and CrushFTP worked with us on the identified Microsoft FTP issue.

Furthermore, in the process of researching IPv6-ready software, a number of vendors and projects shared specific future plans to support IPv6. These commitments include:

- RhinoSoft.com: IPv6 support in next client release (FTP Voyager 16.0)
- GlobalSCAPE: IPv6 support in next server release (EFT 6.4)
- FileZilla: IPv6 support in next server release (Server 0.9.38)

Detailed results for each tested client/server combination are found in Appendix A.

## Conclusions and Recommendations

Our interoperability tests prove that it is unlikely that anyone's existing file transfer servers, clients and partner connections will transparently migrate to IPv6. To avoid surprises IT departments must plan and prepare for IPv6 migration.

To assist with IPv6 migration planning, we offer the following recommendations.

- **Inventory and classify existing file transfer applications noting their level of IPv6 support:**  
For example: "none", "none but planned", "claimed but incomplete" or "complete." (All of the tested clients and servers in our interoperability tests would be classified as either "claimed but incomplete" or "complete".)
  - If your existing file transfer applications do not support IPv6, ask your vendor or open source project about their specific plans ("roadmap") to implement IPv6.
  - If a vendor or project does not plan to support IPv6, prepare to find a replacement for the file transfer software they currently provide.
    - If possible, plan to replace Internet-facing applications that communicate with partners first. Migration of these systems may take significant lead-time.
    - Also consider the placement, role and/or upgrade of FTP proxies, firewalls and other application-aware technology when planning a migration.
- When new software claiming IPv6 support is released, **put new IPv6 software to a complete test using all its supported protocols** before committing to it.
  - Please consider using the software packages profiled in this paper as a baseline for your own tests, and consider evaluating the vendors and projects listed here your own use. (Though flaws may exist in profiled packages' current implementations, their early adoption of IPv6 could provide them an interoperability advantage over late adopters.)
- **Understand your corporate IPv6 plan** and communicate widely.
  - If you manage file transfer applications, be aware that these applications can be affected by IT changes and that file transfer application changes often affects back-end processing. All affected groups should understand ramifications of an IPv6 address change or application change and should approve the pacing and phasing of each step.
  - Tell your data exchange partners you are developing an IPv6 plan and share what you can about the plan when you have it. This outreach both reassures your partners that you will be ready for the future and reminds them to devise and share their plan too.

- **Learn about areas of special concern** with file transfer under IPv6
  - Be wary of FTP or FTP/S compatibility over IPv6. Our tests demonstrated good interoperability across different packages when using SFTP or HTTP, but using FTP and FTP/S surfaced a number of different problems.
    - Single-port protocols like SFTP and HTTP generally present fewer interoperability problems under IPv6 than FTP or FTPS.
  - Packages based on Java present at least two challenges with IPv6. Java 7 changes some of the cipher suites available to file transfer applications and may affect SSL/TLS interoperability. Windows servers with Java based server software will have to upgrade to the forthcoming (Sun) Java 7 release to allow Java-based servers to bind to IPv6 addresses. *(Notes on this online at <http://www.filetransferconsulting.com> )*
  - Proxy servers, load balancers and firewalls that function perfectly with file transfer protocols under IPv4 may not work the same way under IPv6. While an examination of these technologies was beyond the scope of this paper, anecdotal evidence (primarily vendor feedback) suggests that many of these technologies will be problematic. If you use them, be sure to include these technologies in your pre-deployment testing too.

## Additional Assistance

File Transfer Consulting provides both strategic consulting (e.g., if and when you should introduce IPv6 into your data exchanges) and tactical consulting (e.g., how you should introduce IPv6).

Please also contact us with any questions or clarifications about this white paper; we intend to update it and republish it periodically.

## Appendix A: Detailed Client/Server Interoperability Results

### Apache FTPServer 1.0.5

(SFTP and HTTP interfaces were not tested because this server does not support those protocols)

Client (down) \ Protocol (across)	Plain	FTP/S
FileZilla 3.4.0	PASS	PASS
FireFTP 1.0.10	PASS	PASS
JSCAPE AnyClient 3.0	FAIL <sup>1</sup>	PASS
Microsoft FTP	PASS	N/A
SmartFTP 4.0	PASS	PASS

1 - Connects but cannot reliably produce a directory listing when using IPv6 hostnames.

### ProFTPD Server 1.3.3e

(SFTP and HTTP interfaces were not tested because this server does not support those protocols)

Client (down) \ Protocol (across)	FTP	FTP/S
FileZilla 3.4.0	PASS	PASS
FireFTP 1.0.10	PASS	FAIL <sup>1</sup>
JSCAPE AnyClient 3.0	FAIL <sup>2</sup>	PASS
Microsoft FTP	PASS	N/A
SmartFTP 4.0	PASS	PASS

1 = Connects but cannot reliably produce a directory listing when using IPv6 hostnames.

2 = Displays an "invalid number of arguments" error.

### Pure-FTPd Server 1.0.32

(SFTP and HTTP interfaces were not tested because this server does not support those protocols)

Client (down) \ Protocol (across)	FTP	FTP/S
FileZilla 3.4.0	PASS	PASS
FireFTP 1.0.10	PASS	PASS
JSCAPE AnyClient 3.0	FAIL <sup>1</sup>	PASS
Microsoft FTP	PASS	N/A
SmartFTP 4.0	PASS	PASS

1 - Connects but cannot reliably produce a directory listing when using IPv6 hostnames

## CrushFTP Server 5.6

Client (down) \ Protocol (across)	FTP	FTP/S	SFTP	HTTP
IE 9.0	PASS	N/A	N/A	PASS
FireFox 3.6.17	PASS	N/A	N/A	PASS
Chrome 11.0	PASS	N/A	N/A	PASS
FileZilla 3.4.0	PASS	PASS	PASS	N/A
FireFTP 1.0.10	PASS	PASS	PASS	N/A
JSCAPE AnyClient 3.0	FAIL <sup>1</sup>	PASS	PASS	N/A
Microsoft FTP	FAIL <sup>1</sup>	N/A	N/A	N/A
SmartFTP 4.0	PASS	PASS	PASS	N/A

<sup>1</sup> = Both clients attempt "PORT" or "PASV" instead of "EPRT" and "EPSV"

## JSCAPE MFT Server 7.2

Client (down) \ Protocol (across)	FTP	FTP/S	SFTP	HTTP
IE 9.0	PASS	N/A	N/A	FAIL <sup>2</sup>
FireFox 3.6.17	PASS	N/A	N/A	FAIL <sup>2</sup>
Chrome 11.0	PASS	N/A	N/A	FAIL <sup>2</sup>
FileZilla 3.4.0	PASS	PASS	PASS	N/A
FireFTP 1.0.10	PASS	FAIL <sup>1</sup>	PASS	N/A
JSCAPE AnyClient 3.0	FAIL <sup>1</sup>	PASS	PASS	N/A
Microsoft FTP	PASS	N/A	N/A	N/A
SmartFTP 4.0	PASS	PASS	PASS	N/A

<sup>1</sup> = Connects but cannot reliably get a directory listing.

<sup>2</sup> = Web server cannot bind to IPv6 ports.

## RhinoSoft.com Serv-U 11.0

Client (down) \ Protocol (across)	FTP	FTP/S	SFTP	HTTP
IE 9.0	PASS	N/A	N/A	PASS
FireFox 3.6.17	PASS	N/A	N/A	PASS
Chrome 11.0	PASS	N/A	N/A	PASS
FileZilla 3.4.0	PASS	PASS	PASS	N/A
FireFTP 1.0.10	PASS	PASS	PASS	N/A
JSCAPE AnyClient 3.0	FAIL <sup>1</sup>	PASS	PASS	N/A
Microsoft FTP	PASS	N/A	N/A	N/A
SmartFTP 4.0	PASS	PASS	PASS	N/A

<sup>1</sup> = Connects but cannot reliably produce a directory listing when using IPv6 hostnames.

## Appendix B: Legal, Acknowledgements and Reference

### Trademarks and Project Recognition

AnyClient and “MFT Server” are trademarks of JSCAPE, LLC.

Chrome is trademark of Google, Inc.

CrushFTP is a product owned by Ben Spink.

CuteFTP is a trademark of GlobalSCAPE, Inc.

FTPServer is trademark and an open source project governed by the Apache Software Foundation.

FireFTP is an open source project led by Mime Čuvalo.

FileZilla is an open source project led by Tim Kosse.

Internet Explorer is a registered trademark of Microsoft, Inc.

ProFTPD is an open source project led by John Morrissey, Michael Renner, Daniel Roesen and TJ Sanders.

Pure-FTPd is an open source project led by Frank Denis.

Serv-U and “FTP Voyager” are registered trademarks of Rhino Software, Inc.

SmartFTP is a trademark of SmartSoft, Ltd.

Supranet is a trademark of Supranet Communications, Inc.

### Disclaimer

File Transfer Consulting, LLC (“FTC”) provides this white paper “as is” and without any warranty of any kind, expressed or implied. FTC is not responsible for and expressly disclaims all warranties of any kind with respect to third party content, products and services. FTC is not responsible for any loss, costs or damages incurred due to your access to or use of any third party content, products or services mentioned in herein or due to your dealings with any such third party. If you choose to form any products or services from a third party, your relationship shall be with that third party and FTC shall not be responsible for the quality of any third party products or services, or fulfilling any of the terms of your agreement with a third party, or for any other aspect of your relationship with a third party.

### Acknowledgments

JSCAPE ([www.jscape.com](http://www.jscape.com)), RhinoSoft.com ([www.rhinosoft.com](http://www.rhinosoft.com)), CrushFTP ([www.crushftp.com](http://www.crushftp.com)), FileZilla ([www.filezilla.net](http://www.filezilla.net)), GlobalSCAPE ([www.globalscape.com](http://www.globalscape.com)), Frank Denis of Pure-FTPd ([www.pureftpd.org](http://www.pureftpd.org)) and Mime Čuvalo of FireFTP ([fireftp.mozdev.org](http://fireftp.mozdev.org)) for addition technical information and access.

Supranet ([www.supranet.net](http://www.supranet.net)) for providing IPv6 connectivity and hosting services in support of this project. ISOC (“Internet Society” - [www.isoc.org](http://www.isoc.org)) – and especially Phil Roberts - for organizing “World IPv6 Day”, leading ongoing efforts to promote acceptance of IPv6 and other assistance.

### Reference

RFC 2428 (FTP over IPv6): <https://tools.ietf.org/html/rfc2428>

RFC 4217 (FTP over SSL/TLS): <https://tools.ietf.org/html/rfc4217>

Additional testing notes can be found at <http://www.filetransferconsulting.com>

### About File Transfer Consulting

File Transfer Consulting is a vendor-independent firm that aims to simplify data exchange. We provide industry thought leadership, strategic consulting, integration services, training and technology.