

Using FTP over FleetBroadband on Windows clients

Version 01
20 October 2007

inmarsat.com/fleetbroadband

Whilst the information has been prepared by Inmarsat in good faith, and all reasonable efforts have been made to ensure its accuracy, Inmarsat makes no warranty or representation as to the accuracy, completeness or fitness for purpose or use of the information. Inmarsat shall not be liable for any loss or damage of any kind, including indirect or consequential loss, arising from use of the information and all warranties and conditions, whether express or implied by statute, common law or otherwise, are hereby excluded to the extent permitted by English law. INMARSAT is a trademark of the International Mobile Satellite Organisation, Inmarsat LOGO is a trademark of Inmarsat (IP) Company Limited. Both trademarks are licensed to Inmarsat Global Limited. © Inmarsat Global Limited 2007. All rights reserved.

Contents

1	Introduction	1
	1.1 About this guide	1
	1.2 Other sources of information	1
2	Introduction to FTP	1
	2.1 About FTP	2
	2.2 Throughput is proportional to size of file	2
3	Using FTP clients over FleetBroadband	3
	3.1 FTP clients tested	3
	3.2 Tests carried out	3
	3.3 Test results	3
	3.4 Conclusions	4
	3.5 Recommendations	5

1 Introduction

1.1 About this guide

This document introduces Windows FTP clients that have been tested on the FleetBroadband network, advises how to optimise your FTP application for use over FleetBroadband, and provides recommendations for the use of FTP.

It is intended for first time end-users, distribution partners, service providers and anyone who wants to use a FleetBroadband terminal to connect to the FleetBroadband network and use network services. A previous knowledge of VPNs and VPN client software, and of the FleetBroadband LaunchPad application is required. A previous knowledge of satellite communications is useful, but not essential.

This document assumes that the VPN client is connected to the FleetBroadband terminal via the Ethernet interface.

The sections include:

- Introduction to FTP –what FTP does, and how it works.
- Using FTP clients on FleetBroadband – the comparative results of tests performed on the FleetBroadband network using various FTP clients, and provides recommendations based on the test results.

1.2 Other sources of information

- This is one of a series of PDF documents that make up the FleetBroadband user guides. The guides are designed to help you make the most of your FleetBroadband terminal. Other documents in the series are available for download from www.inmarsat.com/fleetbroadband. Click on **Support**, then click on **User guides**. This Web site also gives further information on the FleetBroadband service, including industry applications.
- Refer to “FleetBroadband LaunchPad help” for details on using FleetBroadband LaunchPad.
- Refer to the documentation supplied with your FTP application for details on changing configuration and settings.

2 Introduction to FTP

File Transfer Protocol (FTP) is commonly used to transfer data between client and server, and between server and client. It is one of the most effective TCP protocols for moving data, as it has very low overheads.

There are many FTP clients and servers available, which are sometimes chosen for their user interfaces. However, in tests over the FleetBroadband network and other Inmarsat networks, it has been found that the performance of the standard DOS FTP client is better than other FTP clients.

This document analyses tests performed on some of the commonly used FTP clients on the FleetBroadband network, and compares them to the DOS client.

2.1 About FTP

The FTP protocol uses two TCP connections to establish communications between the client and server.

- A command connection – used by the client to send commands to the server and by the server to send responses.
- A data connection – created when a command is given to send or receive a file. As its name suggests, this connection is concerned with the actual data transmission.

There are no modifications that can be made to the FTP protocol that can improve performance. However, note that the FTP client and server do not ordinarily compress data before transmission, and that the use of a compression program can improve performance.

Tip The most cost-effective way to send large attachments over a packet network that is billed by volume or time is to compress the file with a standard utility such as:

- WINZIPTM, available from <http://www.winzip.com> or
- Winrar, available from <http://www.rarlab.com>.

Compressing files is only of benefit if the content is not already compressed, and the recipient has a utility to de-compress the attachments.

The FTP data connection can be established in either active or passive mode:

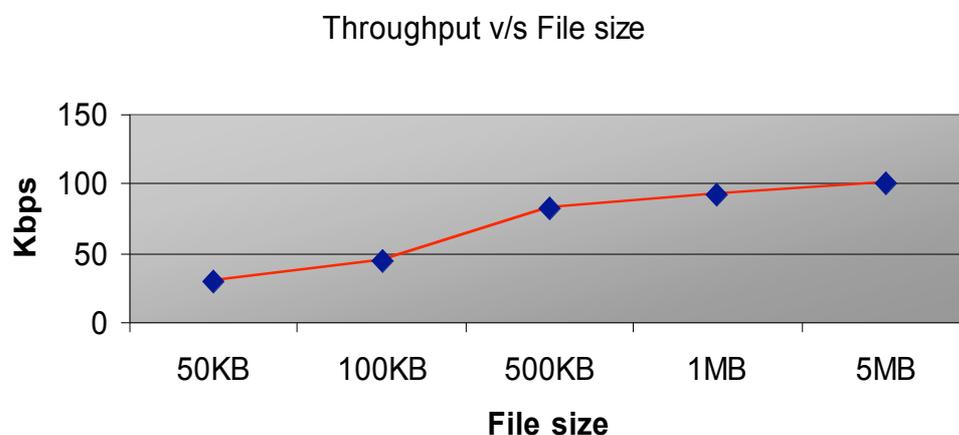
- Active mode – the client sets up a command connection from a random local port (>1024) to the command port on the server (port 21). The client informs the server that its data connection is available on the random local port +1. When a data connection is required, the server connects to the data port on the client.

Note that because the client does not establish a connection with the data port of the server, the client's firewall often sees the server as an external system trying to establish a connection, and blocks it. Therefore, you must configure any firewall on the client to allow traffic from the server.

- Passive mode – the client sets up a command connection from a random local port (>1024) to the command port on the server (port 21). The client sends the PASV command. The server allocates a random port for the data connection, and informs the client. The client then initiates the data connection directly to this port on the server.

2.2 Throughput is proportional to size of file

The graph below shows the typical relationship between the throughput realised and the size of the file being exchanged.



This behaviour is noticed across all networks. It happens due to the presence of TCP Slow Start algorithm which operates for the first part of the transmission – which therefore comprises a greater part of the transmission for smaller files. As a result transfer protocols produce larger overheads for small files.

3 Using FTP clients over FleetBroadband

3.1 FTP clients tested

The following Windows FTP clients have been tested on the FleetBroadband network:

- CuteFTP from Globalscape (www.cuteftp.com)
- WS_FTP (www.ipswitch.com)
- SmartFTP (www.smartftp.com)
- Command line FTP client, Windows XP
- 16k buffers were used (ftp -w:16384)

3.2 Tests carried out

One 500KB (4mbits) file was sent and received three times to and from the test server over a standard IP connection in an operational environment (multiple terminals in a single spot beam).

TCP-PEP (Inmarsat's performance enhancing proxy) was switched off. Different buffer sizes were used and the average taken. Inmarsat's FleetBroadband test tool was used to measure Simple Network Management Protocol (SNMP) time, packet and byte count. For the purposes of clarity, logon, changing directory and logging off are not included in the figures.

The DOS FTP client was set to a 16KB buffer size as, from previous studies this has proved to be slightly more efficient than the standard buffer size of 4KB.

All the other FTP applications were used with default settings initially, then advanced settings were changed, where possible, as detailed below.

3.3 Test results

FTP PUT

The results for sending data were as follows:

Client	Time (seconds)	% Overhead
CuteFTP	86.52	8
WS_FTP	65.16	57
SmartFTP	111.09	14
DOS FTP	47.0	8

- On further analysis, it was seen that a combination of bad packet filling and timing issues caused the reduced performance. In addition, the DOS FTP application forces TCP time-stamping, which can aid the efficient flow of data over high latency networks.
- CuteFTP has options to change the buffer size used. In tests with large buffers set (128KB), the PUT time was reduced to 54 seconds. This still falls short of the DOS FTP performance.

FTP GET

The results for receiving data were as follows:

Client	Time (seconds)	% Overhead
CuteFTP	44.12	7
WS_FTP	59.07	38
SmartFTP	73.38	9
DOS FTP	32.05	8

- Again, CuteFTP, WS_FTP and SmartFTP do not match the performance of DOS FTP. In the GET direction, both WS_FTP and CuteFTP come closest to matching DOS FTP.
- The CuteFTP buffers were changed again, and this reduced performance in the GET direction.

3.4 Conclusions

Different FTP clients gave very different results over the FleetBroadband network. Note that most graphical FTP clients are designed for their ease of use, rather than for the optimal speed over high latency networks.

It is also worth noting the overheads for the different applications; WS_FTP showed overheads of 38% or 57%. Whilst this does not greatly impact throughput, it significantly affects the cost – using this FTP client can add up to 50% to the cost.

The DOS client out-performed some of the most well known graphical FTP clients. Due to this, Inmarsat's solutions team undertook a search for a graphical FTP client that worked as well as the DOS client. In this process, the following clients were tested informally:

- FTP Explorer
- LeechFTP
- FileZilla
- FTPRush
- CoreFTP

Of these clients, CoreFTP & Filezilla performed best, and also had the ability to configure the send and receive buffers with minute precision.

CoreFTP set-up

The default buffers for CoreFTP are:

- Send – 512B
- Receive – 4096B

Changing the receive buffer does not greatly affect performance, although setting this to a higher value may show a small improvement when using large TCP window sizes.

The send buffer is very important, and determines the size of packets used by TCP in the transfers. For best general usage set the buffer to 32768.

Tip

- Set the receive buffer to 32768.
- Buffers can be changed on a per site basis, so ensure that you change these options either before creating any site profiles, or edit those that you already have using the advanced button.

3.5 Recommendations

When using FTP over FleetBroadband, check that your chosen FTP client matches the performance of DOS FTP. If you are unsure, use the CoreFTP package, and modify the buffers.