

Accessing beamtime data using FTP

How it works

A **FTP** server provides interactive access for DOOR users to beamtime directories using the FTP protocol. It allows you to browse and download beam time data using any suitable FTP client. Access is granted according to the working group definitions in the Gamma Portal. As we are using a secure implementation of the FTP protocol, you might have to configure your client. You can find configuration examples below, see *Configuring FTP clients*. The FTP server uses a port range of 30000 - 32000 to transfer data. These ports need to be open for outgoing traffic at the receiving site, which is usually the case.

To access your data:

1. Use the 'FTP Configuration' menu entry of the [Gamma Portal](#) to [select a specific beam time you want to access](#) and click 'FTP register'. If you do not select a beam time, the FTP server will point you to a README file. See 'testing the service' below.
2. Use a suitable FTP client to connect to [psftp.desy.de](#). We tested our setup using *filezilla*, *lftp* and *fireftp*. See *Configuring FTP clients* below for some information on these clients.
3. Use your [DOOR](#) account to authenticate to the FTP server. DOOR user names and passwords are *case sensitive*.
4. If you want to access a different beam time, simply log out of the FTP server, use the 'FTP Configuration' menu again to de-register the old beam time and register for the new one, then log in to the FTP server and thus to the new beam time.

Testing the service without having done a beam time yet:

1. If you have a DOOR or DESY account, you can verify that the service works for you.
2. Just connect to the service using a FTP client e.g. [filezilla](#).
3. Right-click on the file named [README.txt](#)
4. A text editor should open and display a short [hint](#).

Configuring FTP clients

Access to [psftp.desy.de](#) is encrypted to protect your privacy. To connect to the server for the first time, you need to accept the server's certificate and - depending on the client you use - the certificates of the signing organizations. We tested the following clients and describe the steps to take.

Note: On windows systems, we recommend using *filezilla* instead of *winSCP* due to lower performance and frequent reconnects of *win SCP*.

filezilla (Linux, Windows)

Features

- *filezilla* is a GTK based GUI.
- It works out of the box (you must accept the certificate) on at least Fedora 22, Ubuntu 12.10, Ubuntu 15.04 (14.xx ??), Debian 7.3, SuSE 13.2, Windows (7 professional and Server 2012) and EL 7.
- It is known to be broken on SL 6 (Bugzilla 1029485).
- You can browse directory structures on both the FTP server and your local machine
- You can view files like text files or JPEGs. In fact you can configure it to display the contents of any file for which you have a display program
- It has a queue concept allowing you to start and stop transfers
- It can recursively download directory trees
- It can do several transfers simultaneously

Configuration

- Configure it to [use explicit TLS](#) when connecting to [psftp.desy.de](#).
- [Accept the certificate](#) by clicking *ok*.

Usage example

- When you are logged in to your beamtime, you can use *filezilla* to browse directory structures, recursively download directory trees and even look at single files. [Here is an example of the GUI](#).

lftp (Linux)

Features

- *lftp* is a command line client available with most Linux distributions.
- You can browse directory structures
- It can be used in batch mode or within scripts and is able to copy or sync **whole directory trees**.
- You can use the `cat` command to view text files
- It is very robust when doing transfers

Configuration

lftp needs to be configured by the user to trust the certificate chain:

```
mkdir ~/.lftp/certs
# Add the trust chain for DFN issued certificates
# NOTE: This can only be done from a IPv4 aware system.
#       From a IPv6 only system, this currently does not work.
curl https://pki.pca.dfn.de/dfn-ca-global-g2/pub/cacert/chain.txt > ~/.lftp/certs/desy.pem
# Activate them:
echo 'set ssl:ca-file "~/.lftp/certs/desy.pem" ' >> ~/.lftp/rc
```

The servers welcome message might hint to maintenances or other events of interest. If you do not see the message, you can add a line containing `debug 3` to your lftp configuration file, `~/.lftp/rc`. An example file might look as follows:

```
set ssl:ca-file "~/.lftp/certs/desy.pem"
debug 3
```

Usage example

- lftp can be used to [synchronize whole directory trees](#).

Notes

- Debian 8 needs another line in the configuration file: "`set ftp:ssl-protect-list no`" (omit the "s..")
- The server is now IPV6 enabled. On some client OSes (e.g. Redhat 6, openSUSE at least with 42.3), lftp gets stuck after the login prompt. Here is how you can [check](#) if your client is affected. The [solution](#) is to put the following line to your `lftp` configuration file (`~/.lftp/rc`):

```
set dns:order 'inet inet6'
```

which would then look like

```
set ssl:ca-file "~/.lftp/certs/desy.pem"
set dns:order 'inet inet6'
debug 3
```

fireftp (firefox plugin; works on all systems supported by firefox)

NOTE: fireFTP is not available with the latest versions of firefox, starting with Quantum (v 57.0). The developer has discontinued the product (see <http://fireftp.net/>).

Features

- `fireftp` is a GUI implemented as firefox plugin and should work on any system supporting firefox.
- You can browse directory structures on both the FTP server and your local machine
- You can view files like text files or JPEGs.
- It has a queue concept allowing you to start and stop transfers
- It can recursively download directory trees
- It can do several transfers simultaneously

Configuration

To use it with `psftp.desy.de`, you need to import the certificate chain into your browser. To do this, go to https://pki.pca.dfn.de/desy-ca/cgi-bin/pub/pki?cmd=getStaticPage;name=download_cacert and import the certificates presented there into firefox by simply clicking on them. Then start `fireftp`.

Note

With large number of files and directories, `fireftp` will take a huge amount of time setting up and use up large amounts of RAM. With a 8.7 million file beamtime, the setup phase took more than 24 hours with more than 7 GB memory consumption.