Transition Guide for GEOFON BREQ_FAST users

Author: GEOFON Team
Version 1.0, last updated: 2018-01-04.

Introduction

We have now shut down BREQ_FAST at the GEOFON data centre. BREQ_FAST has been running at GEOFON for many years, and its e-mail based interface [0] has been convenient for some users. Today, only a handful of users continue to use it. However it has increasingly caused difficulties to support. To better support more modern services, we have chosen to focus our limited staff on these.

This guide will help you adapt to the newer services, and in particular FDSN web services. Below we indicate some alternative methods, and attempt to answer a few questions you may have.

Please contact us at geofon_dc@gfz-potsdam.de if you have any problems with the alternatives we offer below.

How will I make requests now?

In the examples below, we show how to request the same data as with BREQ_FAST, using each of the newer services. Consider the following old-style example request:

```
.NAME Jana Seismologist
.INST TU-Grossdorf
.MAIL Hauptstrasse 10, 12555 Grossdorf, Deutschland
.EMAIL j.seismo@tu-grossdorf.de
.LABEL Test1
.END

APE GE 2010 1 1 10 0 0 2010 1 1 11 0 0 1 BHZ
```

Let's suppose the lines above are saved in a file called *my-breqfast.req*. This asks for one hour of BHZ data for GEOFON station APE on January 1 2010. There are several ways to request this data now:

1. Using WebDC. See the online help for WebDC, located at http://eida.gfz-potsdam.de/webdc3/help.html
2. Using fdsnws_fetch
3. Using arclink_fetch
4. With Direct HTTP requests to fdsnws-dataselect (*wget*, *curl* etc.)

You no longer need to identify yourself (except that WebDC asks for an e-mail address, and authentication will require it to identify you for any restricted data that you request).

Using fdsnws_fetch

This even supports the BREQ_FAST format as input. Thus the easiest way to make a request is:

1. Get *fdsnws_fetch* [1] from the SeisComP3 repository at github [SC3]. Click on

   https://raw.githubusercontent.com/SeisComP3/seiscomp3/master/src/trunk/apps/fdsnws/fdsnws_fetch.py

   and save as *fdsnws_fetch*:

```
$ mv fdsnws_fetch.py fdsnws_fetch
$ chmod +x fdsnws_fetch
```

You can get help with *fdsnws_fetch* using the "-h" option:

```
$ ./fdsnws_fetch -h
A command-line FDSN Web Service client using EIDA routing and authenticatio

Usage: fdsnws_fetch [-h|--help] [OPTIONS] -o file

Options:
  --version             show program's version number and exit
  -h, --help            show help message and exit
  -l, --longhelp        show extended help message and exit
[etc]
```

You may need to install the python-dateutil package if you are using an older Python version (<2.7).

2. Make a request. Use the "-b" option to indciate that your request is in BREQ_FAST format. The "-v" option gives more information, and "-o" lets you name the output file:

   ```
   $ ./fdsnws_fetch -v -b my-breqfast.req -o my-breqfast.seed
   ```

   The program will wait until the request is finished ("block") before terminating. That's all! Your mini-SEED data is now in *my-breqfast.seed*. As a service, *fdsnws_fetch* even prints out a little reminder about how to acknowledge use of the data appropriately.

If you are requesting restricted data, you will need either the "-a" option to supply your authentication token, or the "-c" option with a credentials file. See the Authorisation and Authentication guide at http://geofon.gfz-potsdam.de/waveform/archive/auth/ .

If you need full SEED, please try the new *fdsnws2seed* beta version [2]. This requires SeisComP be installed locally on your computer. It uses fdsnws-station to request Station XML, and converts the result to SEED for you. To install it:

```
$ curl -L https://github.com/andres-h/fdsnws_scripts/tarball/master \
    | tar xzf -
$ cd andres-h-fdsnws_scripts-*
$ for f in *.py; do chmod a+x $f; mv $f `basename $f .py`; done
```

Example usage of *fdsnws2seed* with BREQ_FAST input:

```
$ cat >req.breq
.NAME Jana Seismologist
.INST TU-Grossdorf
.END
APE GE 2010 08 01 12 00 00.0000 2010 08 01 12 10 00.0000 01 BH?

$ ./fdsnws2seed -v -r 1 -b req.breq -o req.seed
```

The .NAME and .INST lines are not required for unrestricted metadata.

Using arclink_fetch

Arclink_fetch is an older client which is part of SeisComP 3. It can also use BREQ_FAST format input files, with the "-f breqfast" option. You must provide a user name, with the "-u" option:

```
$ ~/seiscomp3/bin/seiscomp exec arclink_fetch -v -u me@some.where.org \
    -f breqfast my-breq_fast.req -o req.seed
```

```
parsed 1 lines from breqfast message
launching request thread (eida.gfz-potsdam.de:18002)

the following data requests were sent:
 datacenter name: GFZ
 request ID: 105709494, Label: , Type: WAVEFORM, Encrypted: False, Args: resp_
 status: READY, Size: 78770, Info:
     volume ID: GFZ, dcid: GFZ, Status: OK, Size: 78770, Encrypted: False, Inf
         request: 2010,1,1,10,0,0 2010,1,1,11,0,0 GE APE BHZ .
         status: OK, Size: 91648, Info:

saved file: req.seed

in case of problems with your request, please contact eida@gfz-potsdam.de
```

(Note the *arclink_fetch* command above is continued with a backslash ("\"), but it can all be on one line too.) By default, *arclink_fetch* queries the ArcLink server at GEOFON, though requests can be sent to other EIDA nodes too, with the "-a" option. Again, *arclink_fetch* waits until its request is completed before returning to the shell. You can get help with *arclink_fetch* using the "-h" option:

```
Usage: arclink_fetch [-h|--help] [OPTIONS] -u USER -o OUTPUTFILE [REQUEST]

Options:
  --version             show program's version number and exit
  -h, --help
  -l, --longhelp
  -w SSLPASSWORDFILE, --password-file=SSLPASSWORDFILE
                        file containing passwords used for decryption of
                        encrypted data (default dcidpasswords.txt)
  -a ADDRESS, --address=ADDRESS
                        address of primary ArcLink node (default eida.gfz-
                        potsdam.de:18001)
```

You can, but don't have to, also use the native ArcLink request format. Instead of the line above, this would be:

```
2010,01,01,10,00,00 2010,01,01,11,00,00 GE APE BHZ .
```

The start and end dates use commas as separators. They come before the channel identifiers. And network code comes before station code. You would then request using

```
$ arclink_fetch -v -u me@some.where.org my-breq_fast.req -o req.seed
```

i.e. without the '-f breqfast' option.

Direct HTTP requests to fdsnws-dataselect

Some examples are provided on our web page, Using the GEOFON/FDSN web services. You can use any web browser or a tool such as wget or curl to make requests. The above example request could be made using *wget* as:

```
wget -v "http://geofon.gfz-potsdam.de/fdsnws/dataselect/1/query?net=GE&station
    -O my-dataselect.seed
```

Remember to use quotation marks around the URL to protect the shell from the ampersand ("&") characters between parts of the query string. By replacing "dataselect" by "station" in the above request, and adding "&level=response", you may obtain metadata for this station as Station XML.

The webservice URL builder tool http://geofon.gfz-potsdam.de/waveform/builder-dataselect.php might help you get started.

ObsPy is a convenient seismological toolbox which is a good alternative for Python users. It offers clients for both the web service and ArcLink.

Questions

1. When will the service stop?

   A: We plan to accept no BREQ_FAST e-mail messages after 5 January 2018. This date may change.

2. Why are you doing this?

   A: Old services on which BREQ_FAST is based (e-mail and FTP) are becoming difficult to support securely. In particular some mail providers place restrictions on automated e-mail services which have caused difficulties. Also we had to maintain a relatively large storage are for anonymous FTP. There is considerable ability for a few users to inconvenience many others.

3. What about other data centres running BREQ_FAST?

   A: IRIS's BREQ_FAST service (mail sent to addresses *@iris.washington.edu*, or using the web form at http://ds.iris.edu/ds/nodes/dmc/manuals/breq_fast/) is unaffected by this change at GEOFON. We are not aware of any other major data centres running BREQ_FAST, but they will not be affected by this either.

4. But I have an old script to automate everything...

   A: Probably it can easily be modified to use (i) *fdsnws_fetch*, (ii) *arclink_fetch*, or (iii) web services directly. In particular, (i) and (ii) can accept requests in BREQ_FAST format.

5. What about full SEED / dataless SEED?

   A: Try the *fdsnws2seed* service described above. *arclink_fetch* can provide full SEED or dataless SEED with the "-k" option ("-k fseed", or "-k dseed" respectively). WebDC can also provide full SEED.

6. What about restricted data?

   A: Currently BREQ_FAST requests for restricted data are accepted, and produce files for download which are encrypted. These require a password which you have been sent previously.

   If you use *arclink_fetch* or ObsPy, you will receive the same sort of encrypted files, but these tools are able to decrypt them for you locally using your password for the GEOFON data centre (by default these expect your password is stored on your computer only, in a file called *dcidpasswords.txt*; see http://geofon.gfz-potsdam.de/news/index.php?id=7)

   fdsnws_fetch uses a newer token-based system suitable for web services, and after authenticating to the web server, delivers data if you are authorised. In this case, data currently travels across the network *unencrypted*. See http://geofon.gfz-potsdam.de/waveform/archive/auth/ .