

NAME

dbiproxy - A proxy server for the DBD::Proxy driver

SYNOPSIS

```
dbiproxy <options> --localport=<port>
```

DESCRIPTION

This tool is just a front end for the [DBI::ProxyServer](#) package. All it does is picking options from the command line and calling `DBI::ProxyServer::main()`. See [DBI::ProxyServer](#) for details.

Available options include:

--chroot=dir

(UNIX only) After doing a `bind()`, change root directory to the given directory by doing a `chroot()`. This is useful for security, but it restricts the environment a lot. For example, you need to load DBI drivers in the config file or you have to create hard links to Unix sockets, if your drivers are using them. For example, with MySQL, a config file might contain the following lines:

```
my $rootdir = '/var/dbiproxy';
my $unixsockdir = '/tmp';
my $unixsockfile = 'mysql.sock';
foreach $dir ($rootdir, "$rootdir$unixsockdir") {
  mkdir 0755, $dir;
}
link("$unixsockdir/$unixsockfile",
"$rootdir$unixsockdir/$unixsockfile");
require DBD::mysql;

{
  'chroot' => $rootdir,
  ...
}
```

If you don't know `chroot()`, think of an FTP server where you can see a certain directory tree only after logging in. See also the `--group` and `--user` options.

--configfile=file

Config files are assumed to return a single hash ref that overrides the arguments of the new method. However, command line arguments in turn take precedence over the config file. See the "CONFIGURATION FILE" section in the [DBI::ProxyServer](#) documentation for details on the config file.

--debug

Turn debugging mode on. Mainly this asserts that logging messages of level "debug" are created.

--facility=mode

(UNIX only) Facility to use for Sys::Syslog. The default is **daemon**.

--group=gid

After doing a `bind()`, change the real and effective GID to the given. This is useful, if you want your server to bind to a privileged port (<1024), but don't want the server to execute as root. See also the `--user` option.

GID's can be passed as group names or numeric values.

--localaddr=ip

By default a daemon is listening to any IP number that a machine has. This attribute allows one to restrict the server to the given IP number.

--localport=port

This attribute sets the port on which the daemon is listening. It must be given somehow, as there's no default.

--logfile=file

By default logging messages will be written to the syslog (Unix) or to the event log (Windows NT). On other operating systems you need to specify a log file. The special value "STDERR" forces logging to stderr. See `Net::Daemon::Log` for details.

--mode=modename

The server can run in three different modes, depending on the environment.

If you are running Perl 5.005 and did compile it for threads, then the server will create a new thread for each connection. The thread will execute the server's `Run()` method and then terminate. This mode is the default, you can force it with "`--mode=threads`".

If threads are not available, but you have a working `fork()`, then the server will behave similar by creating a new process for each connection. This mode will be used automatically in the absence of threads or if you use the "`--mode=fork`" option.

Finally there's a single-connection mode: If the server has accepted a connection, he will enter the `Run()` method. No other connections are accepted until the `Run()` method returns (if the client disconnects). This operation mode is useful if you have neither threads nor `fork()`, for example on the Macintosh. For debugging purposes you can force this mode with "`--mode=single`".

--pidfile=file

(UNIX only) If this option is present, a PID file will be created at the given location. Default is to not create a pidfile.

--user=uid

After doing a `bind()`, change the real and effective UID to the given. This is useful, if you want your server to bind to a privileged port (<1024), but don't want the server to execute as root. See also the `--group` and the `--chroot` options.

UID's can be passed as group names or numeric values.

--version

Suppresses startup of the server; instead the version string will be printed and the program exits immediately.

AUTHOR

Copyright (c) 1997 Jochen Wiedmann
Am Eisteich 9
72555 Metzingen
Germany

Email: joe@ispsoft.de
Phone: +49 7123 14881

The [DBI::ProxyServer](#) module is free software; you can redistribute it and/or modify it under the same terms as Perl itself. In particular permission is granted to Tim Bunce for distributing this as a part of the DBI.

SEE ALSO

[DBI::ProxyServer](#), [DBD::Proxy](#), [DBI](#)