

DYNACCESS

DynAccessPoster Documentation

(binary version for Linux/Unix)

**for automatic
DynAccess-account update**

www.dynaccess.com

DynAccess is a service of

**Ambos InterNet-Solutions
Heiko Ambos, CEO**

Am Biberbach 23	Postfach 1233
24145 Kiel	24011 Kiel
Germany	Germany

Fon: +49 700 26267366

Fax: +49 700 26267329

<http://www.ambos-is.net>
info@ambos-is.net

foreword

This program (DynAccessPoster) updates your DynAccess account at regular intervals and ensures perfect operation.

Since the DynAccess system consists of several servers at various computing centres, if possible you should use this program or our current Windows software for account updates.

For only our account-update program ensures perfect interaction between update software and DynAccess system.

If problems occur with our servers, our programs automatically access backup servers.

Our update software is thus able to carry out successful account updates even if there are DNS problems in your network.

Third-party products or DDNS support in hardware routers do not support many of our features nor do they use our redundantly organised systems, unless there is special DynAccess support (e.g. Lancom router).

DynAccess is a B2B product which aims for the highest availability for commercial use and has an answer for many eventualities.

Our system availability is thus more than 99.9% averaged over 8 years.

QUICKstart

The complete configuration is specified via parameters when DynAccessPoster is started, thus the configuration file `/etc/DynAccessPoster` is not needed.

The program needs only the name of the account and domain with the corresponding password.

```
bash:~ # ./DynAccessPoster -account=test -domain=dynaccess.com -passwd=xyz
```

If you would prefer not to follow the usual practice of specifying the account and domain name separately, you can specify the entire host name as an alternative.

```
bash:~ # ./DynAccessPoster -hostname=test.dynaccess.de -passwd=xyz
```

Please specify your password in MD5-encoded form, available on the first page (Account) at <http://myaccount.dynaccess.com>.

No further installation is required.

The necessary files are created at run time and are described below.

Hint on calling the program

In this documentation, the program call is given by `./DynAccessPoster`.

If you copy the client into a directory which allows a direct call via the PATH variable, then `DynAccessPoster` without `./` is enough.

Hint: copy the client into the directory `/usr/local/bin`

A call via `./DynAccessPoster` is usually necessary if you start the program directly in a directory without specifying the complete path, e.g.

```
/opt/DynAccess/DynAccessPoster
```

advanced start options

You can influence the basic functioning of DynAccessPoster via diverse calling parameters.

- debug DynAccessPoster starts in debug mode (i.e. in the foreground) and prints information directly to the console
- force A new instance of DynAccessPoster is created; those already running are terminated automatically.
- offline The offline configuration is activated.
The poster sets up a connection to the DynAccess account-update server and causes the account to be configured in offline mode.
After that, DynAccessPoster terminates.

Debug- or daemon mode:

DynAccessPoster switches to daemon mode immediately after startup as long as debug mode is not forced via the corresponding parameter. Daemon mode means that DynAccessPoster switches into the background and, except for log-file entries, produces no output.

If you start DynAccessPoster in debug mode via

```
bash:~ # ./DynAccessPoster debug <account configuration>
```

then it will not switch to the background immediately.

DynAccessPoster runs in debug mode (i.e. in the foreground) and prints out directly to the screen all the information which is also written into the log file.

If an instance of DynAccessPoster is already running in daemon mode and you start another instance of DynAccessPoster in debug mode, then the DynAccessPoster which is already running in daemon mode terminates.

Two instances of DynAccessPoster never run in parallel as long as you use DynAccessPoster on a Linux/UNIX system.

Force a new instance of DynAccessPoster:

If you start the client via

```
bash:~ # ./DynAccessPoster <account configuration>
```

more than once, this has no effect on a Linux/UNIX system since the newly started DynAccessPoster notices an already running instance and terminates itself.

Via

```
bash:~ # ./DynAccessPoster force <account configuration>
```

you can force a new instance of DynAccessPoster. Those already running will terminate automatically.

Controlled offline-switching:

If you call DynAccessPoster as follows

```
bash:~ # ./DynAccessPoster offline <account configuration>
```

then this leads to the neutralisation of your DynAccess account, i.e. your account is configured with a neutral IP.

In addition, all instances of DynAccessPoster terminate.

This makes sense when you want to go offline in a controlled manner and would like to neutralise your DynAccess account.

further DynAccess files

The script writes the following files:

Pid file: /var/run/DynAccessPoster.pid

The current process ID is written into this file.

This is important since you can start DynAccessPoster more than once.

Each additional instance terminates automatically when it notices that DynAccessPoster is already running.

Deleting this file causes the automatic termination of DynAccessPoster.

You can change the location of the Pid file via the parameter FILEpid.

e.g. -FILEpid=/root/DynAccess.pid

alive file: /tmp/DynAccessPoster.alive

This file is needed for the identification of other instances of the program which are already running. You can change the name and location of the file via the parameter -FILEalive when starting the program.

e.g. -FILEalive=/root/DynAccess.alive

Log file: /var/log/DynAccessPoster.log

The log is written into this file. You should keep a watchful eye on this when initially using DynAccessPoster.

If your internet connection is near its maximum load, then it is possible that the log is filled with numerous error messages. In the short term, this is not a problem.

In case of errors, the script tries, at short intervals, to carry out a successful update. Compare this log with the log of your DynAccess account at <http://myaccount.dynaccess.com>

In no case should repeated errors in the log file be a permanent situation.

You can change the location of the Pid file via the parameter FILElog.

e.g. -FILElog=/tmp/DynAccess.log

Restricting the logging

Depending on the area of application, it can be necessary to reduce or even deactivate the logging.

For example, you can log the sending of every HeartBeat which however needs a lot of storage space. You can suppress the logging of account updates (setIP) and requesting your own IP (myIP). Since the client interacts with the DynAccess server, the server can transmit individual plain-text messages to the DynAccessPoster which are recorded in the log. You can deactivate these as well.

If you deactivate all types of logging, only errors are recorded.

If this is not desired, you can completely deactivate logging by specifying "-FILElog=" as a parameter.

If you call the program with no further parameters or with the switch -h, you obtain a list of the parameters and the corresponding default values.

error messages and their meaning

The script informs you via the log file with detailed information. You should always keep an eye on the additional message file, even if your account is functioning well.

However, we would like to explain the following log-file entries:

There is a problem to do a myIP

DynAccessPoster obtains your current IP at fixed (short) intervals from a DynAccess server.

If DynAccessPoster detects an IP change, a setIP, i.e. an account update, is performed.

If you receive the error message that no myIP could be carried out, then you can assume that the internet connection is overloaded or non-existent.

The script could not establish a connection to a DynAccess server within a fixed given time span. The attempt was terminated and will be carried out again after a short time.

If the problem is due to an unreachable DynAccess server, DynAccessPoster automatically accesses another server.

The occasional occurrence of such an entry is not unusual.

In case of a large number of such error messages, you should take appropriate action in order to allow DynAccessPoster to operate properly.

Usually a connection is established which is terminated when a timeout is reached since the operation could not be carried out quickly enough.

There is a problem to do a setIP

DynAccessPoster could not carry out an account update.

The same causes are to be investigated as described above.

In the case of regular problems or even occasional neutralisation of your DynAccess account, you should take action to solve the problem.

CRON example

Via CRON, you can execute timed scripts and programs under Linux.

If you would not like to set up a quasi-permanent connection with DynAccess but rather would like to make your resources available only at certain times, then you can achieve this very simply.

Start the script via CRON with the command

```
bash:~ # ./DynAccessPoster <account configuration>
```

in order to activate your account at a certain time.

Via

```
bash:~ # ./DynAccessPoster offline <account configuration>
```

your account is neutralised. In addition, all instances of DynAccessPoster are automatically terminated.