

DYNACCESS

DynAccessPoster documentation

zur automatisierten
DynAccess-Account-Aktualisierung

www.dynaccess.de

DynAccess is a service of

Ambos InterNet-Solutions
Inh. Heiko Ambos

Am Wellsee 59c Postfach 1233
24146 Kiel 24011 Kiel

Fon: +49 700 26267366

Fax: +49 700 26267329

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

```
#####  
#  
#               C O P Y R I G H T               #  
#      by Ambos InterNet-Solutions in 2001..2005      #  
#      http://www.ambos-is.net                        #  
#  
#      DynAccess is a professional DynDNS service      #  
#      http://www.dynaccess.com                       #  
#  
#####  
  
#####  
#  
# 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 3 years. #  
#  
# An online-help overview is available via #  
#  
#      bash:~ # perl DynAccessPoster.pl -h #  
# or #  
#      bash:~ # ./DynAccessPoster -h #  
#  
#####
```

```
#####
#                                     #
#   online-help overview             #
#                                     #
#####
#
# For the configuration and startup of the account-update program in various
# modes, we offer diverse information which you can access as follows:
#
#   bash:~ # perl DynAccessPoster.pl <online-help-parameters>
# or
#   bash:~ # ./DynAccessPoster <online-help-parameters>
#
#   -h                short help with an overview of the help topics
#   -h-all            entire help
#   -h-start          advanced start options
#   -h-configfile      configuration via /etc/DynAccessPoster.conf
#   -h-configdirect    configuration via calling parameters
#   -h-files           overview of files used by DynAccessPoster
#   -h-advanced        advanced parameters
#   -h-md5             optional MD5 for more password security
#   -h-error           error messages and their meanings
#   -h-cron            example using CRON
#   -v                screen display of version/AuthCode and build
#####

#####
#                                     #
#   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:~ # perl DynAccessPoster.pl debug
# or
#   bash:~ # ./DynAccessPoster debug
#
# 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:~ # perl DynAccessPoster.pl #
# or #
#     bash:~ # ./DynAccessPoster #
# #
# 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:~ # perl DynAccessPoster.pl force #
# or #
#     bash:~ # ./DynAccessPoster force #
# #
# you can force a new instance of DynAccessPoster. Those already running #
# will terminate automatically. #
# #
# #
# Controlled offline-switching: #
# ~~~~~ #
# If you call DynAccessPoster as follows #
# #
#     bash:~ # perl DynAccessPoster.pl offline #
# or #
#     bash:~ # ./DynAccessPoster offline #
# #
# 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. #
# #
#####
```

```
#####
#                                     #
#   configuration                     #
#   via /etc/DynAccessPoster.conf    #
#                                     #
#####
#
# The configuration takes place via the file /etc/DynAccessPoster.conf as
# long as you don't carry out the configuration directly via calling
# parameters.
# If the file /etc/DynAccessPoster does not exist, it is created
# automatically by DynAccessPoster after the first startup.
# Afterwards, DynAccessPoster terminates and requests that you configure it.
# If the file /etc/DynAccessPoster cannot be created, then depending on your
# operating-system configuration it is recommended to carry out the
# configuration of the DynAccess account-update program directly via calling
# parameters.
#
# In addition to the account name (e.g. myAccount), the domain under which
# your account is set up (e.g. dynaccess.com) and the password, you can
# configure the following:
#
# Language      = [de|en] language for the messages to the console and in the
#                  log
#                  default: en
# UseHeartBeat = [0|1] If you have a HeartBeat account, you can activate or
#                  deactivate the use of the HeartBeat.
#                  default: 1 (i.e. activated)
# LogHeartBeat = [0|1] If desired, you can record the transmission of
#                  each heartbeat in the log
#                  default: 0 (i.e. deactivated and recommended)
# LogMyIP       = [0|1] If desired, you can record the testing of your
#                  dynamic IP (myIP) in the log.
#                  For LogMyIP = 0 only errors and IP changes are
#                  recorded.
#                  default: 0 (i.e. deactivated)
# LogSetIP      = [0|1] If desired, you can list in the log the information
#                  which this client receives from our servers after
#                  an account update.
#                  By default, only the fact that the account update
#                  was successful is recorded.
#                  default: 0 (i.e. standard info)
# LogServer     = [0|1] You can record in the log the server which was
#                  accessed by the client.
#                  The IP of the DynAccess server is listed in the log
#                  in square brackets.
#                  default: 1 (i.e. activated)
#
# On Linux/UNIX systems, a change in the configuration in
# /etc/DynAccessPoster.conf is noticed by the client at run-time and taken
# into account within a few seconds, which also causes an account update.
# A restart of DynAccessPoster after a configuration change is thus not
# necessary on Linux/UNIX systems.
#
# Further DynAccess account configuration, i.e. the configuration of the
# features, should be done via http://myaccount.dynaccess.com.
# The entry field "expiration time" at http://myaccount.dynaccess.com is not
# used if DynAccessPoster is used since the expiration time is controlled
# from this program and the minimal and thus optimal value allowed by your
# tariff is set.
#####
```

```
#####
#                                     #                                     #
#   direct configuration via          #                                     #
#   calling parameters               #                                     #
#                                     #                                     #
#####
#
# In addition to configuring DynAccessPoster via /etc/DynAccessPoster.conf,
# you can carry out the configuration directly via calling parameters.
#
# If the configuration file /etc/DynAccessPoster.conf exists, then the
# corresponding configuration is used but aspects which are specified
# differently in the calling parameters are overwritten.
# Direct configuration via the calling parameters thus takes precedence over
# the configuration via /etc/DynAccessPoster.conf, although a mixed
# configuration is possible.
# If you wish to completely suppress the configuration via
# /etc/DynAccessPoster.conf, then additionally specify the parameter
# noCONFfile when starting DynAccessPoster. This also suppresses the
# automatic configuration of the file /etc/DynAccessPoster.conf via
# DynAccessPoster.
#
#   --Account=    Specify your account name here, without the following
#                 Domain (e.g. dynaccess.com).
#                 example: --Account=test
#
#   --Domain=     Specify your domain name (e.g. dynaccess.com) under which
#                 you have your DynAccess account.
#                 example: --Domain=dynaccess.com
#
#   --Hostname=   Optionally, instead of specifying --Account and --Domain,
#                 you can specify this via one parameter and your FQDN
#                 (Fully Qualified Domain Name).
#                 either:   --Account=test --Domain=dynaccess.com
#                 or       :   --hostname=test.dynaccess.com
#
#   --Passwd=     Specify your password here in plain text or preferably
#                 MD5-encoded, which can be obtained from
#                 http://myaccount.dynaccess.com.
#
#   --Language=[de|en] \
#   --UseHeartBeat=[0|1] \
#   --LogHeartBeat=[0|1] | Configuration of the parameters like you can
#   --LogMyIP=[0|1]      | do in the configuration file is also
#   --LogSetIP=[0|1]     | possible directly via calling parameters.
#   --LogServer=[0|1]    /
#
#
#   --PidFile=     \
#   --LogFile=     | The path to the corresponding file can be changed.
#   --MessFile=    /
#
#####
```

```
#####
#                                     #
#   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 PidFile.
#   e.g. PidFile: /etc/DynAccess.pid
#
#
#   Log file: /var/log/DynAccessPoster.log
#   ~~~~~
#   The log is written into this file. You should keep a watchfile 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.
#   If you use programs such as "eDonkey", please limit the up- and downloads
#   so that DynAccessPoster as well has enough bandwidth for problem-free
#   account updates. This is especially important when using the HeartBeat,
#   which sends connectionless UDP packets.
#   You can change the location and name of the log file via the parameter
#   LogFile.
#   e.g. LogFile: /etc/DynAccessPoster.log
#
#
#   Message-File: /var/log/DynAccessPoster.mess
#   ~~~~~
#   Important information which DynAccessPoster delivers can be overlooked
#   if your DynAccess account is running completely satisfactorily.
#   Information which the client receives from the DynAccess server during an
#   account update is also written to the normal log file.
#   This information is additionally written to a further log file, which you
#   should inspect regularly.
#   You can change the location and name of the message-log file via the
#   parameter MessFile.
#   e.g. MessFile: /etc/DynAccessPoster.mess
#
#####
```

```
#####
#                                     #
#   advanced parameters   #
#                                     #
#####
#
#   noTTY                The program tries to send important information to all
#                        consoles and SSH sessions in the hope that a server
#                        administrator will notice the information.
#                        This could perhaps be a problem for an unmanaged Linux
#                        server.
#                        Specifying the parameter noTTY prevents this.
#
#   noTTY2               In contrast to noTTY, in this case only the printing of
#                        the message information to the SSH consoles is prevented.
#                        At the main console, the information is still provided,
#                        which can be suppressed with noTTY.
#
#   noBEEP               If DynAccessPoster has received a message which is to be
#                        indicated at the console, this is signalled by a "beep"
#                        at 60-second intervals.
#                        Specifying the calling parameter noBEEP deactivates this
#                        function, as long as noTTY hasn't already been specified,
#                        which also deactivates the acoustic signalling.
#
#   noLOGfile            Here you can deactivate logging.
#
#   noMESSfile           Here you can deactivate the supplementary message log.
#
#   noCONFfile           Here you can deactivate the necessary presence of a
#                        configuration file.
#
#####

#####
#                                     #
#   optional   MD5   #
#                                     #
#####
#
# In order to prevent the password from being transmitted in plain text,
# it is possible to pass an MD5-encoded password.
# You can enter your already MD5-encoded password in the configuration. This
# can be obtained via http://myaccount.dynaccess.com.
#
# In order to increase security, DynAccessPoster supports all types of
# encoding which we have described at http://www.dynaccess.com.
# You can thus encode your password a second time against an MD5 string
# defined by a DynAccess server during an account update.
# However, this assumes that when using the Perl script you have installed
# the Perl module "Digest::MD5".
# At program start, DynAccessPoster checks this and activates the additional
# MD5 encoding.
# Since the encoding uses an additional string requested from the DynAccess
# server, this lengthens the communication to the DynAccess server.
# If your internet connection or your computers are overloaded, then it can
# happen that your account update isn't completed within 5 seconds and is
# thus terminated by the server.
#
# You can deactivate the additional MD5 encoding by calling the DynAccess
# script with the parameter "noMD5" (without quotes).
#
#####
```



```
#####
#                                     #
#   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:
#
# a) 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
#   and conserve the resources of our DynAccess servers.
#   Usually a connection is established which is terminated when a timeout
#   is reached since the operation could not be carried out quickly enough.
#
# b) 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 under a).
#   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:~ # perl DynAccessPoster.pl
# or
#   bash:~ # ./DynAccessPoster
```

```
#
# in order to activate your account at a certain time.
#
#
# Via
#
#   bash:~ # perl DynAccessPoster.pl offline
# or
#   bash:~ # ./DynAccessPoster offline
#
# your account is neutralised. In addition, all instances of DynAccessPoster
# are automatically terminated.
#
#####
```