NAME

gpg-connect-agent – Communicate with a running agent

SYNOPSIS

gpg-connect-agent [*options*][*commands*]

DESCRIPTION

The **gpg-connect-agent** is a utility to communicate with a running **gpg-agent**. It is useful to check out the commands **gpg-agent** provides using the Assuan interface. It might also be useful for scripting simple applications. Input is expected at stdin and output gets printed to stdout.

It is very similar to running **gpg-agent** in server mode; but here we connect to a running instance.

The following options may be used:

-v
--verbose

Output additional information while running.

-q

--quiet   Try to be as quiet as possible.

--homedir *dir*

Set the name of the home directory to *dir*. If this option is not used, the home directory defaults to '*~/.gnupg*'. It is only recognized when given on the command line. It also overrides any home directory stated through the environment variable '*GNUPGHOME*' or (on Windows systems) by means of the Registry entry *HKCU\Software\GNU\GnuPG:HomeDir*.

On Windows systems it is possible to install GnuPG as a portable application. In this case only this command line option is considered, all other ways to set a home directory are ignored.

To install GnuPG as a portable application under Windows, create an empty file named '*gpg-conf.ctl*' in the same directory as the tool '*gpgconf.exe*'. The root of the installation is then that directory; or, if '*gpgconf.exe*' has been installed directly below a directory named '*bin*', its parent directory. You also need to make sure that the following directories exist and are writable: '*ROOT/home*' for the GnuPG home and '*ROOT/var/cache/gnupg*' for internal cache files.

--agent-program *file*

Specify the agent program to be started if none is running. The default value is determined by running **gpgconf** with the option **--list-dirs**. Note that the pipe symbol (|) is used for a regression test suite hack and may thus not be used in the file name.

--dirmngr-program *file*

Specify the directory manager (keyserver client) program to be started if none is running. This has only an effect if used together with the option **--dirmngr**.

--dirmngr

Connect to a running directory manager (keyserver client) instead of to the gpg-agent. If a dirmngr is not running, start it.

-S
--raw-socket *name*
> Connect to socket *name* assuming this is an Assuan style server. Do not run any special initializations or environment checks. This may be used to directly connect to any Assuan style socket server.

-E
--exec   Take the rest of the command line as a program and it's arguments and execute it as an Assuan server. Here is how you would run **gpgsm**:
> gpg-connect-agent --exec gpgsm --server

Note that you may not use options on the command line in this case.

--no-ext-connect
> When using **-S** or **--exec**, **gpg-connect-agent** connects to the Assuan server in extended mode to allow descriptor passing. This option makes it use the old mode.

--no-autostart
> Do not start the gpg-agent or the dirmngr if it has not yet been started.

-r *file*
--run *file*
> Run the commands from *file* at startup and then continue with the regular input method. Note, that commands given on the command line are executed after this file.

-s
--subst   Run the command **/subst** at startup.

--hex   Print data lines in a hex format and the ASCII representation of non-control characters.

--decode
> Decode data lines. That is to remove percent escapes but make sure that a new line always starts with a D and a space.

CONTROL COMMANDS
> While reading Assuan commands, gpg-agent also allows a few special commands to control its operation. These control commands all start with a slash (**/**).

/echo *args*
> Just print *args*.

/let *name value*
> Set the variable *name* to *value*. Variables are only substituted on the input if the **/subst** has been used. Variables are referenced by prefixing the name with a dollar sign and optionally include the name in curly braces. The rules for a valid name are identically to those of the standard bourne shell. This is not yet enforced but may be in the future. When used with curly braces no leading or trailing white space is allowed.
>
> If a variable is not found, it is searched in the environment and if found copied to the table of variables.

Variable functions are available: The name of the function must be followed by at least one space and the at least one argument. The following functions are available:

get   Return a value described by the argument. Available arguments are:

> **cwd**   The current working directory.
>
> **homedir**
> > The gnupg homedir.
>
> **sysconfdir**
> > GnuPG's system configuration directory.
>
> **bindir**   GnuPG's binary directory.
>
> **libdir**   GnuPG's library directory.
>
> **libexecdir**
> > GnuPG's library directory for executable files.
>
> **datadir**
> > GnuPG's data directory.
>
> **serverpid**
> > The PID of the current server. Command **/serverpid** must have been given to return a useful value.

unescape *args*
> Remove C-style escapes from *args*. Note that **\0** and **\x00** terminate the returned string implicitly. The string to be converted are the entire arguments right behind the delimiting space of the function name.

unpercent *args*
unpercent+ *args*
> Remove percent style escaping from *args*. Note that **%00** terminates the string implicitly. The string to be converted are the entire arguments right behind the delimiting space of the function name. **unpercent+** also maps plus signs to a spaces.

percent *args*
percent+ *args*
> Escape the *args* using percent style escaping. Tabs, formfeeds, linefeeds, carriage returns and colons are escaped. **percent+** also maps spaces to plus signs.

errcode *arg*
errsource *arg*
errstring *arg*
> Assume *arg* is an integer and evaluate it using **strtol**. Return the gpg-error error code, error source or a formatted string with the error code and error source.

+
-
*
/
%   Evaluate all arguments as long integers using **strtol** and apply this operator. A division by zero yields an empty string.

!

|

&   Evaluate all arguments as long integers using **strtol** and apply the logical operators NOT, OR or AND. The NOT operator works on the last argument only.

/definq *name var*

Use content of the variable *var* for inquiries with *name*. *name* may be an asterisk (*****) to match any inquiry.

/definqfile *name file*

Use content of *file* for inquiries with *name*. *name* may be an asterisk (*****) to match any inquiry.

/definqprog *name prog*

Run *prog* for inquiries matching *name* and pass the entire line to it as command line arguments.

/datafile *name*

Write all data lines from the server to the file *name*. The file is opened for writing and created if it does not exists. An existing file is first truncated to 0. The data written to the file fully decoded. Using a single dash for *name* writes to stdout. The file is kept open until a new file is set using this command or this command is used without an argument.

/showdef

Print all definitions

/cleardef

Delete all definitions

/sendfd *file mode*

Open *file* in *mode* (which needs to be a valid **fopen** mode string) and send the file descriptor to the server. This is usually followed by a command like **INPUT FD** to set the input source for other commands.

/recvfd Not yet implemented.

/open *var file* [*mode*]

Open *file* and assign the file descriptor to *var*. Warning: This command is experimental and might change in future versions.

/close *fd*

Close the file descriptor *fd*. Warning: This command is experimental and might change in future versions.

/showopen

Show a list of open files.

/serverpid
 Send the Assuan command **GETINFO pid** to the server and store the returned PID for internal purposes.

/sleep Sleep for a second.

/hex
/nohex Same as the command line option **--hex**.

/decode
/nodecode
 Same as the command line option **--decode**.

/subst
/nosubst
 Enable and disable variable substitution. It defaults to disabled unless the command line option **--subst** has been used. If /subst as been enabled once, leading whitespace is removed from input lines which makes scripts easier to read.

/while *condition*
/end These commands provide a way for executing loops. All lines between the **while** and the corresponding **end** are executed as long as the evaluation of *condition* yields a non-zero value or is the string **true** or **yes**. The evaluation is done by passing *condition* to the **strtol** function. Example:

```
/subst
/let i 3
/while $i
  /echo loop counter is $i
  /let i ${- $i 1}
/end
```

/if *condition*
/end These commands provide a way for conditional execution. All lines between the **if** and the corresponding **end** are executed only if the evaluation of *condition* yields a non-zero value or is the string **true** or **yes**. The evaluation is done by passing *condition* to the **strtol** function.

/run *file*
 Run commands from *file*.

/bye Terminate the connection and the program.

/help Print a list of available control commands.

SEE ALSO
 gpg-agent(1), **scdaemon**(1)

 The full documentation for this tool is maintained as a Texinfo manual. If GnuPG and the info program are properly installed at your site, the command

```
info gnupg
```

should give you access to the complete manual including a menu structure and an index.