



```
-v | --version      Display version information.
-h | -? | --help    Display this help message
```

This is the local administration tool. Some options are only available as a superuser/system administrator.

Administration accounts can also be tied to a user account. This allows for multiple administrators per installation (One active connection).

gilitool:

```
usage: gilitool [-c <path> [-v] | -t <path> | -h]
```

options:

```
-h, --hwid          Generate device ID.
-c, --check         Check license file.
-t, --transfer      Transfer license to this machine (3 day minimum
                    between transfers).
-v, --verbose       Enable verbose output.
-ver, --version     Show version information.
-?, --help         Show this help message.
```

This is the license/metric reporting tool. It can be used to determine your device's hardware identification string, check the license file for errors, or to transfer your license to a new machine (remote licenses only).

To transfer a license:

```
Stop GIservr.
Run `gilitool --transfer`.
Install GIservr on the target machine with your current license.
Start GIservr.
```

```
----- * * * * * Configuration * * * * * -----
```

Logging:

```
logLevel=(off|severe|warning|info|config|fine|finer|finest), default info
Sets logging level from least->most verbose.
```

```
logAdmin=(true|false), default false
Sets if admin interface events should be logged.
```

```
log=(true|false), default false
Sets if any events should be logged.
```

```
logSize=(integer), default 0 (infinite)
Sets the maximum log size before rotation in bytes (cumulative).
```

```
logSizeKB=(integer), default 0 (infinite)
Sets the maximum log size before rotation in kilobytes (cumulative).
```

```
logSizeMB=(integer), default 5
Sets the maximum log size before rotation in megabytes (cumulative).
```

```
maxLogRotations=(integer), default 5
Sets the maximum number of log rotations to keep (0 is infinite).
```

General Configuration:

```
useActiveDirectory=(true|false), default false
Sets if the server should authenticate using Active Directory or built
in authentication.
```

```
activeDirectoryDomain=(string), default is Undefined
Sets the active directory domain for authentication, e.g. my_domain.com.
```

```
activeDirectoryServer=(string), default is Undefined
Sets the active directory server for authentication,
e.g. local.my_domain.com or the IP address.
```

```
serverName=(string), default is from system
```

Sets the name of the server.

serverAddress=(string), default is from system  
Sets the address for the server to listen on.

idleout=(integer), default 0 (never)  
Sets how long before a session is automatically disconnected due to being idle (in seconds).

adminPort=(short), default 4450  
Sets the port the admin interface can use to connect.

admin=(true|false), default true  
Sets whether the admin account is active for this server.

remoteShutdown=(true|false), default false  
Sets if remote shutdowns are allowed from the administrative interface.

maxLoginAttempts=(integer), default 0 (infinite)  
Sets how many failed logins (GIclient) before automatic disconnect.

doesNotExpire=(true|false), default false  
Sets if passwords expire or not (Can be set in the user's \*.cfg file)

exec=(string), default fpdaemon  
Sets the path to the fpdaemon program for the server to use.

restrictClientVersion=(true|false), default false  
Sets strict mode for client version verification, prevents outdated clients from logging in.

maxLoginsPerUser=(integer), default 0 (infinite)  
(6.0.01.08) Sets the maximum number of sessions that can be in use by a single user at once.

expireWarnDays=(integer), default 7  
(6.0.01.11) Sets the number of days prior to expiration that the end user will be warned. Max value 120 days.

heartbeatInterval=(integer), default 60  
(6.0.01.12) Sets the number of seconds before a wakeup message is sent to the client application. Setting this value to 0 will disable the feature.

#### Password Configuration:

minimumLength=(integer), default 5  
Sets minimum length of a password.

maximumLength=(integer), default 48  
Sets maximum length of a password.

expireDays=(integer), default 0 (does not expire)  
Sets how many days until a password is marked as expired. Overrides general configuration.

reuseDays=(integer), default 0  
Sets how many days until a password can be reused.

reuseInstances=(integer), 3  
Sets how many password changes until a password can be reused.

requireUpper=(true|false), default false  
Sets if a password requires at least one uppercase letter.

requireLower=(true|false), default false  
Sets if a password requires at least one lowercase letter.

requireNumeric=(true|false), default false  
Sets if a password requires at least one number.

requireSymbol=(true|false), default false  
Sets if a password requires at least one symbol.

complexityErrorMsg=(string)  
defaults to "Entered Password is not sufficiently complex."

Sets the error message to display on a complexity check failure.

```
reuseErrorMessage=(string)
  defaults to "Too soon to reuse this password."
  Sets the error message to display on a reuse failure.
```

GIservr server configuration file format.

The configuration files can take advantage of comments by prefixing a line with either '#' or 'REM'.

There are also a few tags that can be used to augment the behavior of these config files.

```
[GLOBAL]
  This is the default tag. It suggests to GIservr to include this variable
  for all client types.
```

```
[GI_ONLY]
  This tag suggests to GIservr that it only sets this variable for a
  fileProGI client.
```

```
[WEB_ONLY]
  This tag suggests to GIservr that it only sets this variable for a
  fileProWeb client.
```

A blank line will also reset the tag to GLOBAL

NOTE: These tags are ignored in the 'giservr.cfg' and 'password.cfg' files.

Example:

```
# set startcmd to open the menu 'mainmenu_gui' if launched with fileProGI
[GI_ONLY]
startcmd="p mainmenu_gui"

# set startcmd to open the menu 'mainmenu_web' if launched with fileProWeb
[WEB_ONLY]
startcmd="p mainmenu_web"

# these are in the global namespace
# the space between the last [WEB_ONLY] tag and the next line reset the
# namespace
somevar1=definition
somevar2=definition
somevar3=definition
```