

Akeeba Remote CLI User's Guide

Nicholas K. Dionysopoulos

Akeeba Remote CLI User's Guide

Nicholas K. Dionysopoulos

Copyright © 2011-2022 Akeeba Ltd

Table of Contents

1. Welcome	1
1. Introduction	1
2. Using Remote CLI	2
1. Usage	2
2. Options	3
3. Configuration files	7
4. Walkthrough	7

Chapter 1. Welcome

1. Introduction

Remote CLI is a powerful command-line script which allows you to automate your backups (including download backup archives) with Akeeba Backup for Joomla! or WordPress; or Akeeba Solo. It is designed for power users and system administrators who want to automate these backup-related tasks. It uses our remote JSON API, which means that you can install the script on one machine (even your own PC) and control any number of sites, across different hosts, remotely.

The script is shipped as a PHAR file. This is an executable PHP package which can be used with PHP 5.5.0 or later (PHP 7.0 or 7.1 recommended). The script is available for download free of charge from our site.

Chapter 2. Using Remote CLI

1. Usage

Using the PHAR file

The traditional way to use Remote CLI is as a command line script running on your machine.

Remote CLI requires a command line version of PHP (a.k.a. PHP-CLI) to be installed on your machine. In the command line below, please substitute `php` with the full path to your PHP CLI executable file. On most Linux and Mac OS X systems, you don't need to specify a path, just **php** will do as the executable file is already in the path. On Windows, however, you most likely need to specify a full path.

Tip

The PHP project provides an installer of PHP for Windows [<http://windows.php.net/download/>] which can install PHP for you and place the `php.exe` binary in your path. For sanity's sake, we recommend you to install PHP in `c:\PHP` and put `remote.phar` in there as well. This is the easiest way to be perfectly sure where everything is installed!

Important

Remote CLI requires PHP 5.5 with the optional "phar" extension installed and activated. This is the default setup of PHP in most cases.

After downloading `remote.phar` (the Remote CLI executable PHP archive) from our site you can use Remote CLI from the command line of your operating system with a command like this:

```
php remote.phar actionName --host=hostName --secret=secretKey [options]
```

`php` is the path to your PHP CLI executable. If this is not in your path or you are not sure, please ask your host or your system administrator.

Using the Dockerized version

If you are using Docker you can make use of the Dockerized version of Remote CLI:

```
docker run --rm akeebaltd/remotecli actionName --host=hostName --secret=secretKey [options]
```

The Dockerized version is self-contained, using the latest version of PHP at the time we build the container image. It's much simpler to use since it does not require installing or configuring anything on your host machine. Moreover, it's *self-updating*. The command line above always tries to use the latest version of Remote CLI we have published in Docker Hub.

Warning

There's a caveat. You can NOT use the Dockerized version with local servers i.e. anything which resolves to an IP address only routable by the host machine. If you want to try our Remote CLI locally you have to use the PHAR file.

What are all these options?

`hostName` is the URL to your site's root, without a trailing slash, e.g. `http://www.example.com`. If you are using Akeeba Backup for WordPress the `hostName` must also include the path to Akeeba

Backup's `index.php` file tasked with handling the JSON API, e.g. `http://www.example.com/wp-content/plugins/akeebabackupwp/app/remote.php` where `http://www.example.com` is the URL to your site and `akeebabackupwp` is the name of the folder where the Akeeba Backup for WordPress plugin is installed.

`secretKey` is the Secret Key you have defined in the component Options (Akeeba Backup for Joomla!) or the System Configuration (Akeeba Backup for WordPress, Akeeba Solo) page of the backup application.

The options which are required for each action are described in the Options chapter of this documentation.

For practical examples, or if you want to find out how to automate backing up and downloading the backups of your site, please consult the Walkthrough chapter.

`actionName` can be one of:

license	Displays the license of the software and quits.
test	Tests whether Remote CLI can connect to your site. If it can not, it will tell you why.
backup	Performs a backup. It can (optionally) download the backup archive and delete it from the server afterwards.
download	Downloads a backup archive and can optionally delete it from the server afterwards
deletefile	Delete backup archives from the server
delete	Deletes backup archives and the associated backup record from the server
listbackups	Lists the latest backup records
profiles	Lists the profile IDs and descriptions
backupinfo	Shows detailed information about a backup record
upgrade	Upgrades Akeeba Backup on the server, if there is an update available

Tip

You can also use the legacy `--action=actionName` syntax for backwards compatibility with version 1.x.

Warning

Before you can use Remote CLI with your site, you will need to enable the remote API in Akeeba Backup / Akeeba Solo. In order to do that, go to Akeeba Backup / Akeeba Solo and click on either the Options button in the toolbar (Akeeba Backup for Joomla!) or the System Configuration button towards the bottom of the page (Akeeba Backup for WordPress, Akeeba Solo). You will have to set Enable front-end and remote backup to Yes. You will also need to supply a Secret Key of your liking. This is the password to connect to your site over Akeeba Backup's Remote API. Due to several issues which have to do with server configuration, we strongly recommend using only alphanumeric characters (a-z, A-Z, 0-9), dash and underscore in your Secret Key. Other characters may lead to inability to connect to your site. Use something long and complex.

2. Options

Under any circumstances you can append `-m` or `--machine-readable` to the end of the command line to instruct Remote CLI to use a machine-readable output format. If you omit it, a human-readable format will be used instead (default).

Using the option `--nocolor` will skip the generation of color escape sequences. The default behaviour of Remote CLI is to color-code its output for easier reading. However, some terminal emulators, like Windows' command prompt

application, do not support them and display them as verbatim text. This can be confusing, therefore the use of `--nocolor` is strongly recommended.

Moreover, you can specify the `--license` parameter to display the text of the GNU General Public License version 3, under which the software is distributed. In this case, the action and all options will be silently ignored. This has the same effect as specifying the `license` command.

If you have a host with a self-signed certificate, or a certificate signed by a non-public certificate authority, you can use the `--certificate=/path/to/certificate.pem` option, where `/path/to/certificate.pem` is the full filesystem path to a PEM-formatted certificate. This certificate will be considered a valid certification authority for the purposes of Akeeba Remote CLI. The practical use case for that is intranets, local servers and testing environments, typically using self-signed certificates.

Each action can be used with one or several options. Some of them are mandatory for the correct operation of Remote CLI. The following sections detail the parameters which can be used with each option. All actions require the following mandatory options:

host The URL to your site's root, without a trailing slash, e.g. `http://www.example.com` Please note that if your site is located in a subdirectory, you have to give the full path to the subdirectory, e.g. `http://www.example.com/joomla`

If you are using Akeeba Backup for WordPress the `hostName` must also include the path to Akeeba Backup's `index.php` file tasked with handling the JSON API, e.g. `http://www.example.com/wp-content/plugins/akeebabackupwp/app/index.php` where `http://www.example.com` is the URL to your site and `akeebabackupwp` is the name of the folder where the Akeeba Backup for WordPress plugin is installed.

secret The secret key, as defined in Akeeba Backup's component parameters. In order to access it in the component, please log in to your site's back-end, go to Components, Akeeba Backup and click on the Component Parameters button. Make sure the Enable front-end and remote backup option is enabled. Right below it, you will find the Secret Key field.

Important

We strongly advise you to use only lower and upper case Latin letters and numbers (a-z, A-Z, 0-9) in your secret word, as many servers will refuse to work with secret words containing special characters.

Important

If you are using sh404SEF make sure you have upgraded to the latest release. Older versions conflicted with Akeeba Backup's remote API feature and will cause Remote CLI to report a JSON decoding error. No support will be provided for this error. The only workaround is to upgrade your copy of sh404SEF (which is a good idea anyway, as the older versions also contained potential security vulnerabilities).

Important

Some SEF and security components will corrupt or block the request to Akeeba Backup's Remote API. In this case you will most likely get a "JSON decoding error". If this happens you may have to instruct your component to ignore `com_akeeba` or allow full access to the `http://www.example.com/index.php?option=com_akeeba&view=json` URL. If unsure, please consult with the developer of your SEF or security component.

Below you can find the options required by each command:

test

No extra options are used

backup

- profile** The numeric profile ID you want take a backup with. If it is not specified, a backup with the default profile (#1) will be taken. Please see the profiles command for an easy way to list profile names and IDs.
- description** An optional description of your backup. Put it in double quotes, i.e. `--description="My backup description"`. If not specified, the default backup description (the one with the date and time of the backup) will be used by Akeeba Backup automatically.
- comment** An optional backup comment. Put it in double quotes, i.e. `--comment="My backup comment"`. If not specified, no comment will be stored with your backup.
- download** If you specify `--download` or `-d`, the produced backup archive will be downloaded to your computer. In this case, please refer to the next section for its required parameters. You can also specify `--delete` or `-D` together with `--download / -d` in order to delete the backup archive after it has been downloaded to your PC.

download

- id** The numeric backup ID to download (see listbackups). Note: when you use the `--download` or `-d` switch with the backup action you must not use this option. In all other cases, this option is mandatory.
- dlmode** Can be one of `http`, `chunk` or `curl`
- http** instructs Remote CLI to download the backup archive through HTTP, similar to what you get by clicking the download links in the "Manage Backups" (formerly "Administer Backup Files") page of Akeeba Backup. Even though it is the simplest method, it may cause corrupted downloads of backup archives over 10-20MB on some shared hosts.
- Files are transferred unencrypted with this method. You are advised to only use this if your server is using HTTPS or with JPS (encrypted) archives only.
- chunk** works similarly to `http`, but is designed to work with larger backup archives. It tries to download 1Mb at a time, in order to work around server restrictions on HTTP download. However, on a few shared hosts this will cause the download to fail with a timeout error, memory outage error or an Internal Server Error (HTTP 500 error).
- curl** is the recommended method and can be used to download the backup archive over FTP, FTPS or SFTP. The only downside is that it needs some configuration (see the `dlurl` option below).
- If you are using plain old FTP (not FTPS or SFTP) the files and login credentials are transferred unencrypted. You are advised to never use this method with plain old FTP (not FTPS or SFTP) for security reasons.
- dlpath** Specify the path to save the downloaded file. For example, `--dlpath="c:\User\Myuser\Downloads"` on Windows, or `--dlpath="/home/myuser/Downloads"` on Linux, Mac OS X and other UNIX-based operating systems.
- dlurl** This option is only required if you are using the `curl` download method. In this case, it tells Remote CLI how to connect to your site in order to download the backup archives.

If you are using FTP, you have to specify something like `--dlurl="ftp://username:password@hostname:port/path/to/output/directory"`

Username and Password are your FTP username and password. Hostname is the FTP server's host name. Port is the numeric TCP/IP port (normally it's 21, ask your host). The `/path/to/output/directory` is the FTP path to your output directory. In order to figure out the latter, please use FileZilla to connect to your site and navigate to the backup output directory (by default that is `administrator/components/com_akeeba/backup`). Copy the path shown above the right-hand folders pane. That's the one you want.

Example:

```
--dlurl="ftp://myuser: mypassword@ftp.example.com:21/public_html/
administrator/components/com_akeeba/backup"
```

Important

FTP is an unencrypted protocol. It is trivial for a third party to snoop into your connection and find the username and password you are using to connect to your server. We very strongly recommend using SFTP which is a file transfer protocol implemented over the very secure SSH remote server connection protocol. As a second best, you can use FTPS which is an encrypted version of FTP (exactly the same way HTTPS is an encrypted version of HTTP).

delete

`id` The backup record's ID that you want deleted. Please see `listbackups` below.

deletefiles

`id` The backup record's ID that you want deleted. Please see `listbackups` below.

profiles

This command has no options. It will simply return a list of profile IDs and descriptions.

listbackups

`from` (Optional) From which record you want to start the listing. If not specified, 0 is assumed, showing the top results (latest backup records).

`to` (Optional) Up to which record you want the listing to go to. If not specified, 50 is assumed, therefore showing the latest 50 backup records.

backupinfo

`id` The backup record's ID whose info you want displayed. Please see `listbackups` above.

update

There are no options. Calling this action will check if an update is available and, if there is, will install it automatically on your site.

Important

This feature IS NOT available on Joomla!. Extension updates are handled by Joomla! itself, not Akeeba Backup. Please refer to Joomla!'s documentation for information on updating extensions.

3. Configuration files

Akeeba Remote CLI is looking for a personalized configuration file. This is true on every Operating System *except* Microsoft Windows which does not make available the necessary information to do so.

Important

Configuration files only apply to `remote.phar`. The Dockerized version does not read files from your user folder.

The file is named `.akeebaremotecli` (mind the leading dot!) and is located in your home directory. In standard Unix notation that would be `~/.akeebaremotecli`

This is a plain text file which follows the INI format. Each configuration section can define a set of options which have the same effect as being passed to the command line itself. Each section is referenced by the `--host` command line parameter. If you do not define a host, the default entry `[akeebaremotecli]` will be used instead.

Configuration file example:

```
[akeebaremotecli]
host=https://www.example.com
secret=0123456789abcdef0123456789abcdef

[mybackup]
host=https://www.example.com
secret=0123456789abcdef0123456789abcdef
action=backup
profile=2

[example]
host=https://www.example.com
secret=0123456789abcdef0123456789abcdef
profile=3
```

Example using the default section (tests the connection):

```
remote.phar test
```

Example using a named section, without having to specify a command (takes a backup with profile #2):

```
remote.phar backup --host=mybackup
```

Example using a named section, specifying commands and overriding options (takes a backup with profile #1):

```
remote.phar backup --host=example --profile=1
```

4. Walkthrough

Below we will see how to get Remote CLI interface your site, make sure Akeeba Backup is up to date, run a backup and automatically download the backup archive to your PC.

In this fictional example, the following assumptions are made: Your site can be found at `http://www.example.com` and the secret word is `0Lsp-MjOSLvUL8VuM1h9T2zWfM4EGqMA`. The backup archives are stored in the default output directory (`administrator/components/com_akeeba/backup`). The site's FTP host is `ftp.example.com` and its username and password are `example` and `test` respectively. The site is under the `public_html` directory. We also assume that the PHP CLI executable is already in your path.

Connection test

We begin by making sure that Remote CLI can access our site:

```
php remote.phar --action=test --host=http://www.example.com --secret=0Lsp-MjOSLvUL8VuM1h9T
```

You should see something like this:

```
Akeeba Remote Control CLI svn724 (2011-06-14)
Copyright ©2008-2014 Nicholas K. Dionysopoulos / Akeeba Ltd
```

```
-----
This program comes with ABSOLUTELY NO WARRANTY. This is Free Software and you
are welcome to redistribute it under certain conditions. Use command line
option --license for details.
-----
```

```
Successful connection to site
```

The last line tells you that the connection worked. However, if something goes wrong, you might get one of these error messages:

Error 401 - Invalid login credentials : Your secret word is wrong. Go to your site's back-end, Components, Akeeba Backup, click on Component Parameters and enter a secret key of your liking. We strongly recommend using only a-z, A-Z and 0-9 to work around limitations with some hosts.

Error 503 - Access denied: You forgot to enable the remote API or enter a secret word in Akeeba Backup. Go to your site's back-end, Components, Akeeba Backup, click on Component Parameters and set Enable remote and front-end backups to `Yes`. Then enter a secret key of your liking. We strongly recommend using only a-z, A-Z and 0-9 to work around limitations with some hosts.

Your server does not seem to be compatible with Remote Control : Check that you are using the correct host name, including the `http/https` prefix. Also make sure that Akeeba Backup 3.3.0 or later is installed and activated. Also note that some SEF and security extensions may be blocking the Remote API calls to your site. In this case, you won't be able to use Remote CLI.

Important

If your host name, secret key or any other option contains non-alphanumeric characters (anything except 0-9, a-z, A-Z, slash, backslash, dash, dot and underscore) you **MUST** enclose the option value in double quotes. For example, `--secret=ab=gt%6` will not work, but `--secret="ab=gt%6"` will do. If you get connection errors despite having set the correct host name and secret key, try using double quotes around all of your option values.

Taking a backup

The following command will take a backup (without downloading it to your local PC), using backup profile #1:

```
php remote.phar --action=backup --host=http://www.example.com --secret=0Lsp-MjOSLvUL8VuM1h
```

You should see a long list of output coming through, detailing the backup progress. In the end it should read:

Backup finished successfully

And that's it! You now have a backup. Since it's useless without downloading, let's try...

Taking and downloading a backup

There are many ways to download a backup archive. We'll explore only two, using HTTP (not recommended on most sites) and FTP.

First, let's try HTTP. The command line to achieve that is the same as the backup command line, with the `--download`, `--dlmethod` and `--dlpath` added:

```
php remote.phar --action=backup --host=http://www.example.com
--secret=0Lsp-MjOSLvUL8VuM1h9T2zWfM4EGqMA --profile=1
--download --dlmode=http --dlpath="c:\Downloads"
```

Warning

The command line above should be entered ON A SINGLE LINE. We have added the line break for typesetting purposes only.

This will take a backup and then download it to `C:\Downloads` (or anything else you specify with `--dlpath`) using the HTTP download method. After the "Backup finished successfully" messages you will also see many download lines and a "Archive downloaded successfully". If that fails, you'll have to use the FTP mode.

First, you need to know the following information about your site:

- FTP hostname. We'll assume it's `ftp.example.com` for this example
- FTP username and password. We'll assume it's `foo` and `bar`, respectively
- The FTP path to your backup output directory. We'll assume it's `/public_html/administrator/components/com_akeeba/backup`.

Tip

In order to find this directory, connect to your site using FileZilla. Navigate inside the folder Joomla! is installed in. Usually it's a directory named `public_html`, `htdocs`, `www` or something similar. If unsure don't ask us, ask your host. Then navigate to your backup output directory. The default is `administrators/components/com_akeeba/backup`. If unsure, check your Akeeba Backup Configuration page. Now, on the right-hand pane you will find the FTP path, right above the folders list. Copy this; it's the path we need.

Then, modify the command line to read:

```
php remote.phar --action=backup --host=http://www.example.com
--secret=0Lsp-MjOSLvUL8VuM1h9T2zWfM4EGqMA --profile=1
--download --dlmode=curl --dlpath="c:\Downloads"
--dlurl="ftp://foo:bar@ftp.example.com/public_html/administrator/components/com_akeeba/bac
```

Warning

The command line above should be entered ON A SINGLE LINE. We have added the line break for typesetting purposes only.

One final note. After you download the backup archive, it's a good idea to delete the copy on the server for security reasons. Just append `--delete` to the command line to achieve that. For example, the command line above will become:

```
php remote.phar --action=backup --host=http://www.example.com
--secret=0Lsp-MjOSLvUL8VuM1h9T2zWfM4EGqMA --profile=1
--download --dlmode=curl --dlpath="c:\Downloads" --delete
--dlurl="ftp://foo:bar@ftp.example.com/public_html/administrator/components/com_akeeba/b
```

Warning

The command line above should be entered ON A SINGLE LINE. We have added the line break for typesetting purposes only.

That's it!

Automating the whole lot

If you want to automate updating Akeeba Backup, taking and downloading backups, you can simply create a batch/script file and use your Operating System's scheduling features to run it at predefined intervals. We suggest taking a look at the following tutorials:

- [HowTo: Add Jobs To cron Under Linux or UNIX?](http://www.cyberciti.biz/faq/how-do-i-add-jobs-to-cron-under-linux-or-unix-oses/) [<http://www.cyberciti.biz/faq/how-do-i-add-jobs-to-cron-under-linux-or-unix-oses/>]
- [Using the 'cron' scheduler in Mac OS X](http://hints.macworld.com/article.php?story=2001020700163714) [<http://hints.macworld.com/article.php?story=2001020700163714>]
- [How to schedule tasks on Windows XP](http://support.microsoft.com/kb/308569) [<http://support.microsoft.com/kb/308569>]
- [How to schedule tasks on Windows 7](http://windows.microsoft.com/en-US/windows7/Schedule-a-task) [<http://windows.microsoft.com/en-US/windows7/Schedule-a-task>] (also works on Vista)

As for the batch/script file, all you have to do is to include the commands above in a `.BAT` (Windows) or `.sh` (Linux, Mac OS X) file. For example, a Windows batch file (`backup.bat`) looks like that:

```
REM IMPORTANT! Before running Remote CLI, you have to change the current
REM directory to wherever remote.phar is located in. In this example, we
REM will assume that you followed our advice and put it in c:\PHP
C:
CD c:\PHP
```

```
REM The following line takes and downloads a backup - REMOVE THE LINE
REM BREAKS WHEN COPYING THIS SCRIPT TO YOUR SYSTEM, OTHERWISE IT WILL
REN NOT WORK!!!
```

```
php-cli.exe remote.phar --action=backup --host=http://www.example.com
--secret=0Lsp-MjOSLvUL8VuM1h9T2zWfM4EGqMA --profile=1 --download
--dlmode=curl --dlpath="c:\Downloads"
--dlurl="ftp://foo:bar@ftp.example.com/public_html/administrator/components/com_akeeba/ba
--delete
```

Note that we assume that the path to PHP is `c:\PHP\php-cli.exe` and you followed our advice and also copied Remote CLI (`remote.phar`) into `C:\PHP`. That's it! You can now use the Scheduled Tasks feature of Windows to make this file run whenever you want, essentially having an always up-to-date copy of Akeeba Backup installed on your site, take automated backups of your site, download them to your PC and remove them from the server. All you need to do once you set this up is to remember to leave your PC turned on for the scheduled task to run. Easy peasy!