Akeeba Backup User's Guide

Nicholas K. Dionysopoulos

Akeeba Backup User's Guide

by Nicholas K. Dionysopoulos

Abstract

This book covers the use of the Akeeba Backup site backup component for Joomla!™ -powered web sites. It does not cover any other software of the Akeeba Backup suite, including Kickstart and the desktop applications which have documentation of their own. Both the free Akeeba Backup Core and the subscription-based Akeeba Backup Professional editions are completely covered.

Table of Contents

Part I. User's Guide to Akeeba Backup for Joomla!™

Table of Contents

Chapter 1. Introduction

1. Introducing Akeeba Backup

Akeeba Backup is a complete site backup solution for your Joomla!™ powered website. As the successor to the acclaimed JoomlaPack component, Akeeba Backup builds on its strong legacy to deliver an easy to use, yet powerful, solution to backing up, restoring and moving your site between servers of the same or different architecture.

Its mission is simple: backup your entire site - including all files and database contents - inside a standalone archive. You can then restore your entire site from the contents of this archive, without the need of installing Joomla!™ prior to the restoration. You can do so in a single click manner, without the tedious work required to set up and test external utilities, without changing your server configuration and without having to dive into obscure configuration options.

If you want absolute power and flexibility, Akeeba Backup is right for you, too! It puts you in charge of fine-tuning your backup, choosing which directories, files or database tables to exclude. It can even allow you to backup non-Joomla!™ content, as long as you specify which off-site directories and databases you want to add.

Akeeba Backup has won three J.O.S.C.A.R. awards at J and Beyond. The J.O.S.C.A.R. awards are the result of a peer voting process, where the high-end Joomla! developers and web designers participating in the J and Beyond conferences pick the top extensions for Joomla!.

2. Indicative uses

Akeeba Backup can be used for much more than just backup. Some indicative uses are:

- **Security backups**. Taking a snapshot of your site should your server fail, or a hacker exploit some security hole to deface or compromise your site.
- **Template sites**. Web professionals have used Akeeba Backup in order to create "template sites". This means that you can build a site on a local server, install every component you usually do on most clients' sites and back it up. You now have a canned site that can serve as a great template for future clients. Using the same method you can have a snapshot of all the sites you have built for your clients, without the need to have them installed on your local server.
- **Build a site off-line, upload the finished site easily**. Web professionals can build a complete site off-line on a local server and when done take a snapshot with Akeeba Backup, then restore it on the production site.
- **Testing upgrades locally, without risking breaking the on-line site**. Joomla!™ updates have the potential of breaking things, especially in complex or badly written components and modules. Web masters use Akeeba Backup to get a site snapshot, restore it on a local test server, perform the upgrade there and test for any problems without the live site being at risk.
- **Debugging locally**. Almost the same as above, web professionals have used Akeeba Backup to take a snapshot of a client's Joomla!™ site in order to perform bug hunting. Using Akeeba Backup again, they can upload the fixed site back on the live server.
- **Relocating a site to a new host**. Web masters who want to take their site to a new host have found Akeeba Backup to be their saviour. Just backup the original site and restore on the new host; presto, your site is relocated with virtually no effort at all.

Akeeba Backup has the potential to save you hours of hard labor, according to our users. It is licensed under the GNU General Public License version 3 or, at your option, any later version of the license. As a result, you are free to modify it to your liking and install it on as many sites as you like without having to pay for a pricey "developer's license".

Akeeba Backup comes in two editions, Core and Professional. Akeeba Backup Core is provided free of charge and contains all the features a typical webmaster would like to have in order to easily complete backup and restoration jobs. Even if this is not enough for you, we even give away our full documentation and the comprehensive troubleshooter guide without charging a single penny! If you find something missing, or spotted a bug, don't be afraid to contact us. We have an ongoing Bug Bounty: if you're the first to help us solve a substantial bug, you'll get a free subscription.

Akeeba Backup Professional is designed to take your experience to a whole new level. Featuring advanced options, like embedded restoration, inclusion of external directories and databases, powerful filters based on regular expressions, easy exclusion of Joomla!™ extensions and support for putting your backups on compatible cloud storage services (such as Amazon's S3), it is designed to give the professional user a strong efficiency leverage. Akeeba Backup Professional is the ideal choice for professional web developers. Thanks to its liberal GNU GPL v3 license, Akeeba Backup Professional can be installed on an unlimited number of clients' websites, royalty-free! Amazing, isn't it?

3. A typical backup/restoration work flow

As stated, Akeeba Backup is designed to make your life easier. It does that by streamlining the work flow of backing up and restoring (or migrating) your site. From Akeeba Backup's perspective, restoring to the same host and location, copying your site in a subdirectory / subdomain of the same host or transferring your site to a completely new host is identical. That's right, Akeeba Backup doesn't care if you are restoring, copying, cloning or migrating your site! The process is always the same, so you only have to learn it once. The learning curve is very smooth, too!

Warning

DO NOT ATTEMPT TO RESTORE TO A DIFFERENT DATABASE TECHNOLOGY. IT WILL NOT WORK, IT IS NOT SUPPOSED TO WORK AND IT CANNOT BE MADE TO WORK. For example if you took a backup from a site using a MySQL database you CANNOT restore this database on a PostgreSQL, Microsoft SQL Server or Windows Azure SQL database.

For your information: the structure of tables is extremely different between different database server technologies. There is no one-to-one correspondence between the structures among two different database server technologies. As a result the conversion process is a very manual and tedious job which involves a lot of trial and error and knowing the code which is going to be using this database. To give you an idea, converting the tiny and easy database structure of Akeeba Backup to MS SQL Server and PostgreSQL took about 20 hours and involved making a lot of changes to our code to cater for the new databases. That was the preparatory BEFORE we started working on the actual database backup code. This is something which cannot be automated.

The typical work flow involves using two utilities from the Akeeba Backup suite: the Akeeba Backup component itself, and Akeeba Kickstart. Here is the overview:

1. Install Akeeba Backup and configure it to taste. Or use the automated Configuration Wizard to automatically configure it with the perfect settings for your server. Hit on the Backup Now button and let your site back up. When it finishes up, click on the Manage Backups button. Click on the download links on the far-right of the only backup entry from the list - or, better yet, use FTP to do that - saving all parts of the backup archive somewhere on your local PC.

2. Extract the kickstart-*VERSION*.zip file you downloaded from our Downloads repository. The only contained files are `kickstart.php` and the translation INI files. Upload them to the server on which you want to restore your site to.

3. Upload all parts of the backup archive (do not extract it yet, just upload the files) to the server on which you want to restore your site to (called here forth the *target server*). Your server's directory should now contain the `kickstart.php` and the parts of the backup archive (.jpa, .j01, etc).

4. Fire up your browser and visit the Kickstart URL on your target server, for example `http://www.example.com/kickstart.php` .

5. Change any option - if necessary - and hit the Start button. Sit back while Kickstart extracts the backup archive directly on the server! It's ultra-fast too (when compared to FTP uploading all those 4000+ files!). If it fails with an error, go back, select the `Upload using FTP` option and supply your FTP connection information, then click on Start again.

6. A new window pops up. It's the Akeeba Backup Installer (ABI), the site restoration script which was embedded inside your archive. Do not close the Kickstart window yet!

7. Follow the prompts of the Akeeba Backup Installer, filling in the details of the new server (most importantly, the new database connection and FTP connection information).

8. When the Akeeba Backup Installer is done, it prompts you to delete the installation directory. Ignore this prompt and simply close the ABI window.

9. Back to the Kickstart window, click the button titled Clean Up. Kickstart removes the installation directory, restores your .htaccess file (if you had one in the first place), removes the backup archive and itself.

10. Believe it or not, you have a working site! Honestly! Click on the View the front-end button to visit your new site.

If you are restoring to a different subdirectory on the same server as the original site, or to a whole different host, you might need to edit your .htaccess file for your site to work properly. Also note that some third party extensions which store absolute filesystem paths, absolute URLs or contain host- or directory-specific settings may require manual reconfiguration after the restoration is complete. This is all described in the restoration section of this guide. If you need help backing up your site, take a look in the Backup Now section of this guide.

4. Server environment requirements

In order to work, Akeeba Backup requires the following server software environment:

- Joomla!™ and PHP version compatibilities are detailed in our Compatibility page [https://www.akeebabackup.com/compatibility.html].

- MySQL 5.0.42 or later. MySQL 5.1 or later recommended for optimal performance. MySQL 4.x is not supported. Alternatively you may use PostgreSQL 9.1+, Microsoft SQL Server 2008+ or Microsoft Windows Azure SQL database. Akeeba Backup (since version 3.8.0) is able to backup and restore databases running on any of the above server technologies.

- Minimum 24Mb of PHP `memory_limit` (sufficient *only* for smaller web sites, without many plug-ins and modules running). More is better. 32Mb to 64Mb recommended for optimal performance on large sites. 128Mb is recommended for sites containing deep-nested directories with thousands of files.

  Even though Akeeba Backup may run on servers with a lesser memory limit, it is unlikely that it will ever finish the backup process.

- The PHP function `opendir` must be available.

- Available free space or quota limit about 75%-80% of your site's size (excluding the cache, temporary and backup directories).

- The cURL PHP module must be installed for FTP and cloud backup to work.

As far as the browser is concerned, you can use:

- Internet Explorer 9, or greater (IE7 and IE6 are not supported, IE8 users may get random backup crashes on larger sites).

- Safari 4, or greater

- Opera 9, or greater. Experimental support due to lack of interest by users.

- Google Chrome 4 or greater. This is the best supported browser.

Some versions of Firefox are displaying erratic behaviour with Javascript. We cannot guarantee trouble-free operation under Firefox. Most likely it will work just fine, but if you do spot an odd behaviour please try using one of the supported browsers above before assuming there is a bug in our software. Most likely it's a bug in your version of Firefox.

In any case, you must make sure that Javascript is enabled on your browser for the backup to work. If you are using AVG antivirus, please disable its Link Checker feature (and reboot your computer) as it is known to cause problems with several Javascript-based web applications, including Akeeba Backup and its tools.

Chapter 2. Installation, updates and upgrades

1. Installing Akeeba Backup

Installing Akeeba Backup is no different than installing any other Joomla!™ extension on your site. You can read the complete instructions for installing Joomla!™ extensions on the official help page [http://help.joomla.org/content/view/1476/235/]. Throughout this chapter we assume that you are familiar with these instructions and we will try not to duplicate them.

Note

The language (translation) files are NOT installed automatically. You can download and install them from our language download page [http://cdn.akeebabackup.com/language/akeebabackup/index.html]. Do note that you will have to install both the component and the language packages for the component to work.

As noted on that page, Akeeba Ltd only produces the English and Greek language files. All other languages are contributed by third parties. If you spot an error please do not contact Akeeba Ltd; we will be unable to help you. Instead, please go to the translation project page [https://www.transifex.com/projects/p/akeebabackup/] to find the contact information of the translator. Abandoned languages will show the maintainer being our staff member "nikosdion". In this case you're out of luck; if you want to fix the language package you will need to volunteer to take over the translation project for that language.

1.1. Installing or manually updating the backup component and language files

Just like with most Joomla! extensions there are three ways to install or manually update Akeeba Backup on your site:

- Install from URL. This works only with the Professional release of our component. It is the easiest and fastest one, if your server supports it. Most servers do support this method.
- Upload and install. That's the typical extension installation method for Joomla! extensions. It rarely fails.
- Manual installation. This is the hardest, but virtually fail-safe, installation method.

Please note that installing and updating Akeeba Backup (and almost all Joomla! extensions) is actually the same thing. If you want to update Akeeba Backup please remember that you **MUST NOT** uninstall it before installing the new version! When you uninstall Akeeba Backup you will lose all your backup settings and all backup archives stored inside Akeeba Backup's directories (including the default backup output directory). This is definitely something you do not want to happen! Instead, simply install the new version on top of the old one. Joomla! will figure out that you are doing an update and will treat it as such, automatically.

Tip

If you find that after installing or updating Akeeba Backup it is missing some features or doesn't work, please try installing the same version a second time, without uninstalling the component. The reason is that very few times the Joomla! extensions installer infrastructure gets confused and fails to copy some files or entire folders. By repeating the installation you force it to copy the missing files and folders, solving the problem.

1.1.1. Install from URL

The easiest way to install Akeeba Backup Professional is using the Install from URL feature in Joomla!.

Important

This Joomla! feature requires that your server supports fopen() URL wrappers (`allow_url_fopen` is set to 1 in your server's `php.ini` file) or has the PHP cURL extension enabled. Moreover, if your server has a firewall, it has to allow TCP connections over ports 80 and 443 to `www.akeebabackup.com` and `cdn.akeebabackup.com`. If you don't see any updates or if they fail to download please ask your host to check that these conditions are met. If they are met but you still do not see the updates please file a bug report in the official Joomla! forum [http://forum.joomla.org/]. In the meantime you can use the manual update methods discussed further below this page.

First, go to our site's download page for Akeeba Backup [https://www.akeebabackup.com/downloads/akeeba-backup.html]. Make sure you are logged in. If not, log in now. These instructions won't work if you are not logged in! Click on the Take me to the downloads for this version button of the version you want to install. Please note that the latest released version is always listed *first* on the page. On that page you will find both Akeeba Backup Core and Professional. Next to the Professional edition's Download Now button you will see the DirectLink link. Right click on it and select Copy link address or whatever your browser calls this.

Now go to your site's administator page and click on Extensions, Extension Manager. If you have Joomla! 3.x click on the Install from URL tab. Clear the contents of the Install URL field and paste the URL you copied from our site's download page. Then click on the Install button. Joomla! will download and install the Akeeba Backup update.

If Joomla! cannot download the package, please use one of the methods described in this section of the documentation. If, however you get an error about copying files, folder not found or a cryptic "-1" error please follow our installation troubleshooting instructions [https://www.akeebabackup.com/documentation/troubleshooter/abinstallation.html].

1.1.2. Upload and install.

You can download the latest installation packages our site's download page for Akeeba Backup [https://www.akeebabackup.com/downloads/akeeba-backup.html]. Please note that the latest version is always on top. If you have an older version of Joomla! or PHP please consult our Compatibility page [https://www.akeebabackup.com/compatibility.html] to find the version of Akeeba Backup compatible with your Joomla! and PHP versions. In either case click on the version you want to download and install.

If you are not a subscriber, click on the Akeeba Backup Core to download the ZIP installation package of the free of charge version.

If you are a subscriber to the Professional release, please make sure that you have loged in first. You should then see an item on this page reading Akeeba Backup Professional. If you do not see it, please log out and log back in. Click on the Professional item to download the ZIP installation package.

All Akeeba Backup installation packages contain the component and all of its associated extensions. Installing it will install all of these items automatically. It can also be used to upgrade Akeeba Backup; just install it *without* uninstalling the previous release.

In any case, do not extract the ZIP files yet!

Warning

Attention Mac OS X users! Safari, the default web server provided to you by Apple, is automatically extracting the ZIP file into a directory and removes the ZIP file. In order to install the extension through Joomla!'s extensions installer you must select that directory, right-click on it and select Compress to get a ZIP file of its contents. This behaviour was changed in Mac OS X Mountain Lion, but people upgrading from older versions of Mac OS X (Mac OS X Lion and earlier) will witness the old, automatic ZIP extraction, behaviour.

Log in to your site's administrator section. Click on Extensions, Manage link on the top menu. If you are on Joomla! 3.x please click on the Upload Package File tab. Locate the Browse button next to the Package File (Joomla!

2.5, 3.0 and 3.1) or Extension package file (Joomla! 3.2 and later) field. Locate the installation ZIP file you had previously downloaded and select it. Back to the page, click on the Upload & Install button. After a short while, Joomla!™ will tell you that the component has been installed.

Warning

Akeeba Backup is a big extension (over 2Mb for the Professional release). Some servers do not allow you to upload files that big. If this is the case you can try the Manual installation or ask your host to follow our installation troubleshooting instructions [https://www.akeebabackup.com/documentation/troubleshooter/abinstallation.html] under "You get an error about the package not being uploaded to the server".

If you have WAMPServer (or any other prepackaged local server), please note that its default configuration does not allow files over 2Mb to be uploaded. To work around that you will need to modify your php.ini and restart the server. On WAMPserver left-click on the WAMP icon (the green W), click on PHP, php.ini. Find the line beginning with `upload_max_filesize`. Change it so that it reads:

```
upload_max_filesize = 6M
```

Save this file. Now, left-click on the WAMP icon, click on Apache, Service, Restart Service and you can now install the component. Editing the `php.ini` file should also work on all other servers, local and live alike.

If the installation did not work, please take a look at our installation troubleshooting instructions [https://www.akeebabackup.com/documentation/troubleshooter/abinstallation.html] or try the manual installation described below.

1.1.3. Manual installation

Sometimes Joomla!™ is unable to properly extract ZIP archives due to technical limitations on your server. In this case, you can follow a manual installation procedure.

You can download the latest installation packages our site's download page for Akeeba Backup [https://www.akeebabackup.com/downloads/akeeba-backup.html]. Please note that the latest version is always on top. If you have an older version of Joomla! or PHP please consult our Compatibility page [https://www.akeebabackup.com/compatibility.html] to find the version of Akeeba Backup compatible with your Joomla! and PHP versions. In either case click on the version you want to download and install.

If you are not a subscriber, click on the Akeeba Backup Core to download the ZIP installation package of the free of charge version.

If you are a subscriber to the Professional release, please make sure that you have loged in first. You should then see an item on this page reading Akeeba Backup Professional. If you do not see it, please log out and log back in. Click on the Professional item to download the ZIP installation package.

All Akeeba Backup installation packages contain the component and all of its associated extensions. Installing it will install all of these items automatically. It can also be used to upgrade Akeeba Backup; just install it *without* uninstalling the previous release.

Before doing anything else, you have to extract the installation ZIP file in a subdirectory named `akeeba` on your local PC. Then, upload the entire subdirectory inside your site's temporary directory. At this point, there should be a subdirectory named `akeeba` inside your site's temporary directory which contains all of the ZIP package's files.

If you are unsure where your site's temporary directory is located, you can look it up by going to the Global Configuration, click on the Server tab and take a look at the Path to Temp-folder setting. The default setting is the `tmp` directory under your site's root. Rarely, especially on automated installations using Fantastico, this might have been assigned the system-wide `/tmp` directory. In this case, please consult your host for instructions on how to upload files inside this directory, or about changing your Joomla!™ temporary directory back to the default location and making it writable.

Assuming that you are past this uploading step, click on Extensions, Manage link on the top menu. If you are on Joomla! 3.x please click on the Install from Directory tab. Locate the Install Directory edit box. It is already filled in with the absolute path to your temporary directory, for example `/var/www/joomla/tmp`. Please append `/akeeba` to it. In our example, it should look something like `/var/www/joomla/tmp/akeeba`. Then, click on the Install button.

If you still can't install Akeeba Backup and you are receiving messages regarding unwritable directories, inability to move files or other similar file system related error messages, please consult our installation troubleshooting instructions [https://www.akeebabackup.com/documentation/troubleshooter/abinstallation.html]. If these instructions do not help please do not request support from us; we are unlikely to be able to help you. These errors come from your site set up and can best be resolved by asking your host for assistance or by asking other users in the official Joomla!™ forums [http://forum.joomla.org].

1.1.4. The installation / update broke my site!

Some users have reported that after they have installed or updated Akeeba Backup, they were no longer able to access parts of their site, especially the back-end. This is an indication of a failed or partial installation. Should this happen, use your FTP client to remove the following directories (some of them may not be present on your site; this is normal):

```
administrator/component/com_akeeba
administrator/modules/mod_akadmin
component/com_akeeba
media/com_akeeba
plugins/quickicon/akeebabackup
plugins/system/akeebaupdatecheck
plugins/system/backuponupdate
plugins/system/oneclickaction
plugins/system/srp
```

This will do the trick! You will now be able to access your site's administrator page again and retry installing Akeeba Backup without uninstalling it first. Remember, uninstalling Akeeba Backup will remove your settings and your backups; you do not want that to happen!

2. Upgrading from Core to Professional

Upgrading from Akeeba Backup Core to Akeeba Backup Professional is by no means different than installing the component. You do not have to uninstall the previous version; in fact, you MUST NOT do that. Simply follow the installation instructions to install Akeeba Backup Professional over the existing Akeeba Backup Core installation. That's all! All your settings are preserved.

Important

When upgrading from Core to Professional you usually have to install the Professional package **twice**, without uninstalling anything in between. Sometimes Joomla! does not copy some of the files and folders the first time you install it. However, if you install the package again (without uninstalling your existing copy of Akeeba Backup) Joomla! copies all of the necessary files and performs the upgrade correctly.

3. Automatic updates

Choosing an update method

Akeeba Backup offers two update methods: Joomla! Extensions Update and Built-in.

The Joomla! Extensions Update method uses the extensions update feature that's part of Joomla! itself. Due to limitations in the early implementations of this feature in older versions of Joomla! **this method is only**

recommended and supported for Joomla! 3.2.0 and later. If you try updating Akeeba Backup Professional with the Joomla! extensions updater under Joomla! 1.6, 1.7, 2.5, 3.0 or 3.1 you will get an error. No support will be provided in this case; instead, you will be asked to read this page for further information.

The Built-in method uses our own code to retrieve update information, download the update package and extract it on your server. Joomla!'s own extensions installer code is used only to install the update on your server. **This is the only supported method for updates under Joomla! 1.6, 1.7, 2.5, 3.0 and 3.1**.

You can choose the preferred update method in Components, Akeeba Backup, Options (or Preferences in some older versions of Joomla!), under the Live Update header. Please note that this option is ignored under Joomla! 1.6, 1.7, 2.5, 3.0 and 3.1. In these Joomla! versions the Built-in method will always be used.

Manually checking for the latest version and upgrading

You can easily check for the latest published version of the Akeeba Backup component by visiting http://www.akeebabackup.com/latest. The page lists the version and release date of the latest Akeeba Backup release. You can check it against the data which appear on the right-hand pane of your Akeeba Backup Control Panel. If your release is out of date, simply click on the Download link to download the install package of the latest release to your PC.

Important

If you have the (paid) Professional edition you must enter your Download ID before trying to update the extension. Otherwise you may get a cryptic error message that downloading the update has failed. If you are using the free of charge Core edition or installing the updates manually you do not need to enter a Download ID.

Updating on Joomla! 1.x, 2.5, 3.0 and 3.1 with the Built-in update feature

Important

This is the only supported method on Joomla! 1.x, 2.5, 3.0 and 3.1.

When you select the Built-in update method it is Akeeba Backup that is responsible for retrieving the update information, downloading the update files and extracting them. However, the last part of the update (installing the updated component) is performed by Joomla!'s own code due to restrictions imposed by the Joomla! Extensions Directory (we are forbidden from writing our own extensions installer).

To access the updates go to Components, Akeeba Backup. If there is an update available you will see a yellow banner at the top of the page within a few seconds. Click the Update to X.Y.Z button on the banner (where X.Y.Z is the latest version number of Akeeba Backup). You will see a page with the summary of the update information. Click on the big, green Upgrade button to start the update process. Akeeba Backup will now begin to download, extract and install the update.

Updating on Joomla! 3.2.0 and later with the Joomla! Extensions Update feature

Important

This method is ONLY supported on Joomla! 3.2.0 and later. Do NOT try to update Akeeba Backup Professional with Joomla! Extensions Update under Joomla! 1.6, 1.7, 2.5, 3.0 or 3.1: the update will fail! If you did that, please enable the Built-in method and follow the instructions ABOVE.

Note

This Joomla! feature requires that your server supports fopen() URL wrappers (`allow_url_fopen` is set to 1 in your server's `php.ini` file) or has the PHP cURL extension enabled. Moreover, if your server has a firewall, it has to allow TCP connections over ports 80 and 443 to `www.akeebabackup.com` and `cdn.akeebabackup.com`. If you don't see any updates or if they fail to download please ask your host to check that these conditions are met. If they are met but you still do not see the updates please file a bug report in the official Joomla! forum [http://forum.joomla.org/]. In the meantime you can use the manual update methods discussed further below this page.

Important

If you have the (paid) Professional edition you must enter your Download ID before trying to update the extension. Otherwise Joomla! will return a cryptic error message that downloading the update has failed. If you are using the free of charge Core edition you do not need to enter a Download ID.

Akeeba Backup can be updated just like any other Joomla! extension, using the Joomla! extensions update feature **as long as you are using Joomla! 3.2.0 or later**. Older versions of Joomla! have limitations which do NOT allow Akeeba Backup Professional to be updated and will, instead, return an error.

When you are using the Joomla! extensions updater it is Joomla! that's responsible for finding the updates, downloading them and installing them on your server. You can access the extensions update feature in two different ways:

- From the icon your Joomla! administrator control panel page. On Joomla! 3 you will find the icon in the left-hand sidebar, under the Maintenance header. It has an icon which looks like an empty star. When there are updates found for any of your extensions you will see the Updates are available message. Clicking on it will get you to the Update page of Joomla! Extensions Manager.
- From the top menu of your Joomla! administrator click on Extensions, Extensions Manager. From that page click on the Update tab found in the left-hand sidebar. Clicking on it will get you to the Update page of Joomla! Extensions Manager.

If you do not see the updates try clicking on the Find Updates button in the toolbar. If you do not see the updates still you may want to wait up to 24 hours before retrying. This has to do with the way the update CDN works and how Joomla! caches the update information.

If there is an update available for Akeeba Backup tick the box to the left of its row and then click on the Update button in the toolbar. Joomla! will now download and install the update.

Warning

Akeeba Backup Professional needs you to set up the Download ID before you can install the updates. You can find your main download ID or create additional Download IDs on our site's Add-on Download IDs [http://akee.ba/downloadid] page. Then go to your site's administrator page and click on Components, Akeeba Backup, Options (in the toolbar). Click on the Live Update tab and paste your Download ID there. Finally, click on Save & Close.

If Joomla! cannot download the package, please use one of the manual update methods described below. If, however you get an error about copying files, folder not found or a cryptic "-1" error please follow our installation troubleshooting instructions [https://www.akeebabackup.com/documentation/troubleshooter/abinstallation.html].

If you get a white page while installing the update please try either the Built-in method (described above) or the manual update method (described below).

Updating manually

As noted in the installation section, installing and updating Akeeba Backup is actually the same thing. If the automatic update using Joomla!'s extensions update feature does not work, please install the update manually following the instructions in the installation section of this documentation.

Important

When installing an update manually you MUST NOT uninstall your existing version of Akeeba Backup. Uninstalling Akeeba Backup will always remove all your settings and any existing backup archives stored on your server. You definitely not want that to happen!

Sometimes Joomla! may forget to copy some files when updating extensions. If you find Akeeba Backup suddenly not working or if you get a warning that your installation is corrupt you need to download the latest version's ZIP file and install it *twice* on your site, *without* uninstalling it before or in-between these installations. This will most certainly fix this issue.

4. Requesting support and reporting bugs

Since July 7th, 2011, support is provided only to subscribers. If you already have an active subscription which gives you access to the support for Akeeba Backup you can request support for it through our site. You will need to log in to our site and go to Support, Akeeba Backup 3.x and click on the New Ticket button. If you can't see the button please use the Contact Us page to let us know of the ticket system problem and remember to tell us your username.

If you want to report a bug, please use the Contact Us page of our site. You don't need to be a subscriber to report a bug. Please note that unsolicited support requests sent through the Contact Us page will not be addressed. If you believe you are reporting a bug please indicate so in the contact form.

Important

Support cannot be provided over Twitter, Facebook, email, Skype, telephone, the official Joomla! forum, our Contact Us page or any other method except the Support section on our site. We also cannot take bug reports over any other medium except the Contact Us page and the Support section on our site. Support is not provided to non-subscribers; if you are using the Core version you can request support from other users in the official Joomla! forum or any other Joomla!-related forum in your country/region. We have to impose those restrictions in support to ensure a high level of service and quality. Thank you for your understanding.

Chapter 3. Using the Akeeba Backup component

In this chapter you are going to find detailed reference of all the pages, options and features of the Akeeba Backup components. To get things organized in a logical manner, we chose to present the individual pages in the same manner they appear on the component's Control Panel page, i.e. the first page which is presented to you when you launch the component's back-end. Some of the pages are not available as Control Panel icons, but from different areas of the component. These are discussed first.

1. Menu items

Important

This feature is only available on Joomla! 3.7.0-alpha2 and later versions.

Joomla! 3.7 and later versions allow you to create custom administrator menus. Akeeba Backup fully supports this new feature by providing custom menu item types.

Most of these custom menu item types were created with site integrators / web site agencies in mind. Typically you want to offer your client a simple, obvious way of doing backup operations (take, restore or transfer backups). Up until now you had to tell them to go to the quite busy Akeeba Backup page and click on just the one thing you want them to. As we all know, clients get distracted and start changing things they shouldn't be touching. The custom menu types below are designed to offer perfectly tailored access to the component areas that most users need. Taking and restoring a backup can become a no-brainer, reduced to simply clicking on a back-end menu item.

1.1. Control Panel

This menu item type lets you access Akeeba Backup's main page (control panel). This is the same menu item type Joomla! creates by default when you install the component.

Please remember that excluding files, folders and database tables as well as including external folders and additional databases (for the Professional edition) can only be done through the Control Panel page. It's always a good idea having a link of this type in your custom menu.

1.2. Backup

This menu item type allows the users to take backups. The default options let this work just like clicking on the Backup Now icon in Akeeba Backup's Control Panel page, i.e. the user can select an alternative backup profile, enter a backup description and/or comment and then take a backup or change their mind and return back to the Control Panel page. However the additional options let you do more interesting stuff.

The available options are:

Force backup profile
: Select the backup profile which will be pre-selected in drop-down of the Backup Now page. Selecting (None) default to the currently active backup profile, as selected in other pages of the Akeeba Backup component. By default that's profile #1. This is especially useful with the Start immediately option below.

Start immediately
: When enabled the backup will start right away, without asking the user to enter a backup description or comment and without the option to change their mind. This is equivalent to using the One Click Backup feature inside Akeeba Backup.

  We strongly recommend using this with the Force backup profile option above. Use it to set up which profile you want the backup to be taken with. This allows you to set up one-click backup menu items.

Hide toolbar — When this option is disabled the user will see the Control Panel and Help buttons at the top of the page. The former will take them back to Akeeba Backup's main page whereas the latter opens the documentation page for the Backup Now page. If you are setting up a one-click backup menu item with the options above it's a good idea to enable this option to hide these buttons. That's especially useful when you are setting up a simple menu for use by your client and you don't want them to accidentally cancel the backup by clicking on these buttons.

Return URL — Set up an internal URL to redirect the user after a successful backup. An "internal URL" is a URL pointing to a page in your site's administrator area, *without* the domain name and `/administrator/` part of it. For example, to take someone back to the Joomla! main page set this to `index.php` without anything else before or after it. To take someone back to Akeeba Backup's main page set this to `index.php?option=com_akeeba`.

Warning

Due to the way Joomla's menu manager works, it expects the URL to be URL-encoded. This means that question marks must be replaced %3F and so on. Don't worry about it. Enter the URL regularly and save the menu item **twice** in a row. We have employed a trick to force URL-encoding of the value when re-saving the menu item. Unfortunately due to a missing feature in Joomla's API we can't employ the same or a similarly clever trick the `<first>` time you save the URL.

1.3. Configuration

This menu item type allows the users to modify the main configuration of the current backup profile. It's equivalent to pressing the Configuration button in Akeeba Backup's main page.

1.4. Manage Backups

This menu item type allows the users to manage backup attempts. This includes viewing all backup attempts, viewing / changing the backup description and comments, have access to logs, download the backups, manage remotely stored backups and restore any of the past backups (as opposed to only the latest backup). It's equivalent to pressing the Manage Backups button in Akeeba Backup's main page.

1.5. Restore Latest Backup

This menu item type allows the users to restore the latest backup taken with the specified backup profile. This is especially useful if you teach your site administrators (or the clients for whom you're building sites) to take a backup right before trying to do something which could go wrong such as updating a component, changing configuration settings or doing batch operations on content.

The only option is **Backup Profile** which lets you choose which backup profile's latest backup attempt will be restored. Idea: use the same profile you've set up in a menu item of the Backup type that you've told the client to always use before any dangerous operation. This way you can offer your clients an easy way to undo their most common mistakes!

1.6. Transfer Site Wizard

This menu item type allows the users to transfer and restore the latest backup on a different server. It's equivalent to pressing the Site Transfer Wizard button in Akeeba Backup's main page.

Idea: you can train your clients to use this to deploy a site from the staging to the live server.

1.7. What to do if you don't have any menu items to Akeeba Backup

Depending on how you've set up your site's administrator menu and/or if you've hit a Joomla! bug that sometimes occures on extension update you may end up without a menu item to Akeeba Backup. Other times you may have

deliberately chosen not to display a menu to Akeeba Backup to keep clients from changing the backup settings. The question remains. How can you access Akeeba Backup and how can you restore menu items manually?

The following instructions are generic Joomla! usage tips and don't have to do with how our software works. We provide them as a courtesy. If these instructions don't work for you please do not contact Akeeba Ltd for support. We cannot offer support for generic Joomla! use. Instead please do ask for help in the Joomla support forum at `http://forum.joomla.org`.

Accessing Akeeba Backup

You can always access Akeeba Backup by visiting the `/administrator/index.php?option=com_akeeba` URL on your site, *after* logging in to your site's back-end.

That is to say, if your site's administrator URL is `http://www.example.com/administrator/index.php` enter the URL `http://www.example.com/administrator/index.php?option=com_akeeba` in your browser's address bar to access Akeeba Backup.

Restoring Joomla's default administrator menus

Important

These instructions only work on Joomla! 3.7 and later and only with the default administration template supplied with Joomla. If you have a third party administrator template please contact the template's developer for instructions regarding missing menu items or reverting to the Joomla! default administrator menu.

You need to access the `/administrator/index.php?option=com_modules` URL on your site, *after* logging in to your site's back-end.

From the drop-down that currently reads `Site` select the option `Administrator`.

Find the module which displays your administrator menu. Usually it's called Admin Menu. Click on it to edit it.

From the Menu To Show drop-down select Use System Preset. Then click on Save & Close.

2. Pages outside the Control Panel panes

2.1. Common navigation elements

All pages have their title displayed above their contents. On the tool bar there is a Control Panel icon. Clicking it will bring you back to Akeeba Backup's Control Panel (the first page of the component, with all the buttons).

On pages where editing takes place (e.g. the Configuration page, the profiles editor, etc) instead of the Control Panel icon there is a Cancel icon which discards any changes made and returns you to the previous page. On those pages you will also find a Save & Close icon which saves settings and returns you to the previous page, as well as an Apply icon which saves settings and returns you to the same editing page.

On the bottom of each page, just above the Joomla!™ footer, there is the license information. On the Control Panel page of the Akeeba Backup Core editions there is also a donation link appearing on the right sidebar; if you feel that Akeeba Backup was useful for you do not hesitate to donate any amount you deem appropriate.

2.2. The Control Panel

The main page which loads when you click on Components, Akeeba Backup is called the Control Panel screen. From here you can see if everything is in working order and access all of the component's functions and configuration options.

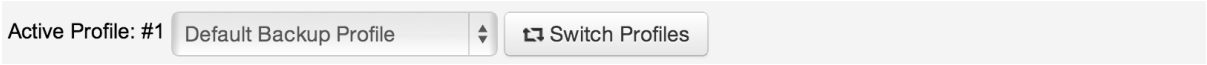
If Akeeba Backup detects a problem with loading the necessary Javascript files, it will issue a big warning message notifying you that it couldn't load the necessary Javascript files. Sometimes, depending on your server settings, this message will not be shown but the interface will behave erratically and appear different than the screen shots provided in here. In this case, you have to use your favorite FTP client and give the `media/com_akeeba` directory and all of its contained subdirectories and files 0755 permissions (read/write/execute for the owner, read/execute for group and others). If this doesn't work, one of your system plugins is killing Akeeba Backup's jQuery integration. In this case, please contact us. Even if you're not a subscriber, please drop us a line using the Contact Us [https://www.akeebabackup.com/contact-us.html] page so that we can figure out what happened and help you. That said, Akeeba Backup will try to automatically do the necessary changes for you, as long as you have provided FTP connection information to your site's Global Configuration and enabled the FTP option in that page.

Tip

Due to the way this warning works you may see a yellow or red flash in the Control Panel, Configuration or Backup Now pages. This is normal and nothing to worry about. It's just your browser being faster in rendering the page than Javascript files loading from your server.

If you see a blank page instead of the Control Panel, you may have a very old version of PHP installed on your server. Please check the minimum requirements of your currently installed Akeeba Backup version. Akeeba Backup will try to detect incompatible PHP versions but this is not always possible.

The profile selection box

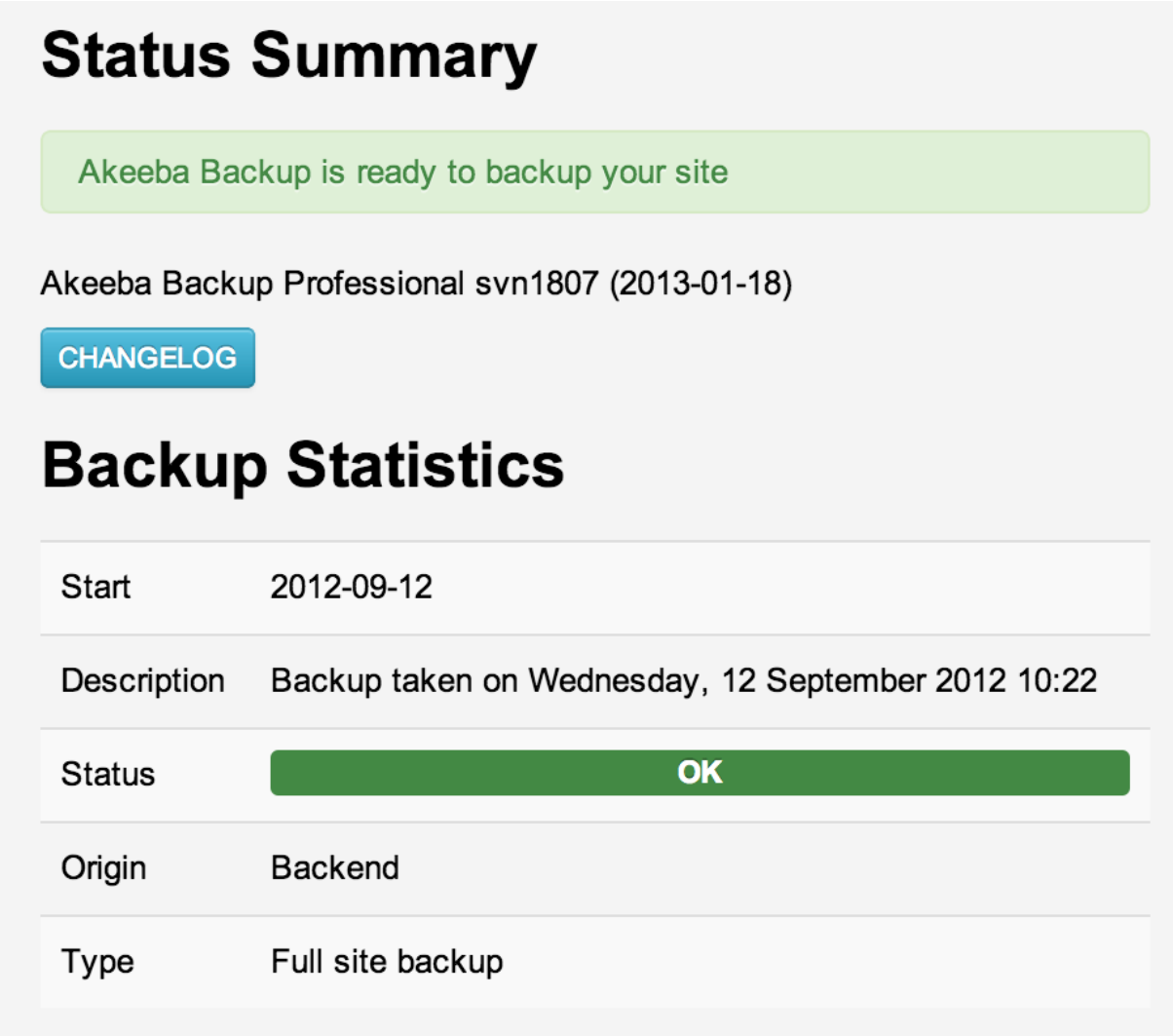


Under the quick links, there is the profile selection box. It serves a double purpose, indicating the active profile and letting you switch between available profiles. Clicking on the drop down allows you to select a new profile. Changing the selection (clicking on the drop down list and selecting a new profile) automatically makes this new profile current and Akeeba Backup notifies you about that. Should this not happen, you can manually click on the Switch Profile button on the right to forcibly make the selected profile current.

Tip

The active profile is applied in all functions of the component, including configuration, filter settings, inclusion options, etc. The only settings which are not dependent on the active profile are those accessible from the Options toolbar button. Keep this in mind when editing any of Akeeba Backup's settings!

On the right hand side of the page, you will find a column with useful information.



There are two areas:

Status Summary

In this area you can find information regarding the status of your backup output directory. Akeeba Backup will warn you if this directory is unwritable. If the text reads that there are potential problems you **must** take a look at the details below to find out what these might be!

Important

No matter what the PHP Safe Mode setting is, it is possible that your host enforces open_basedir restrictions which only allow you to have an output directory under a handful of predefined locations. On this occasion, Akeeba Backup will report the folder unwritable even though you might have enforced 0777 (read, write and execute allowed for all) permissions. These restrictions are reported in the section below the overall status text as an item entitled "open_basedir restrictions".

If any potential problems have been detected, right below the overall status you will find one or several warnings links. Just click on each warning's description to get a pop up window explaining the potential problem, its impact on your backup and precautionary or corrective steps you can take. If this section is empty, no detectable problems were found; this is a good thing, indeed!

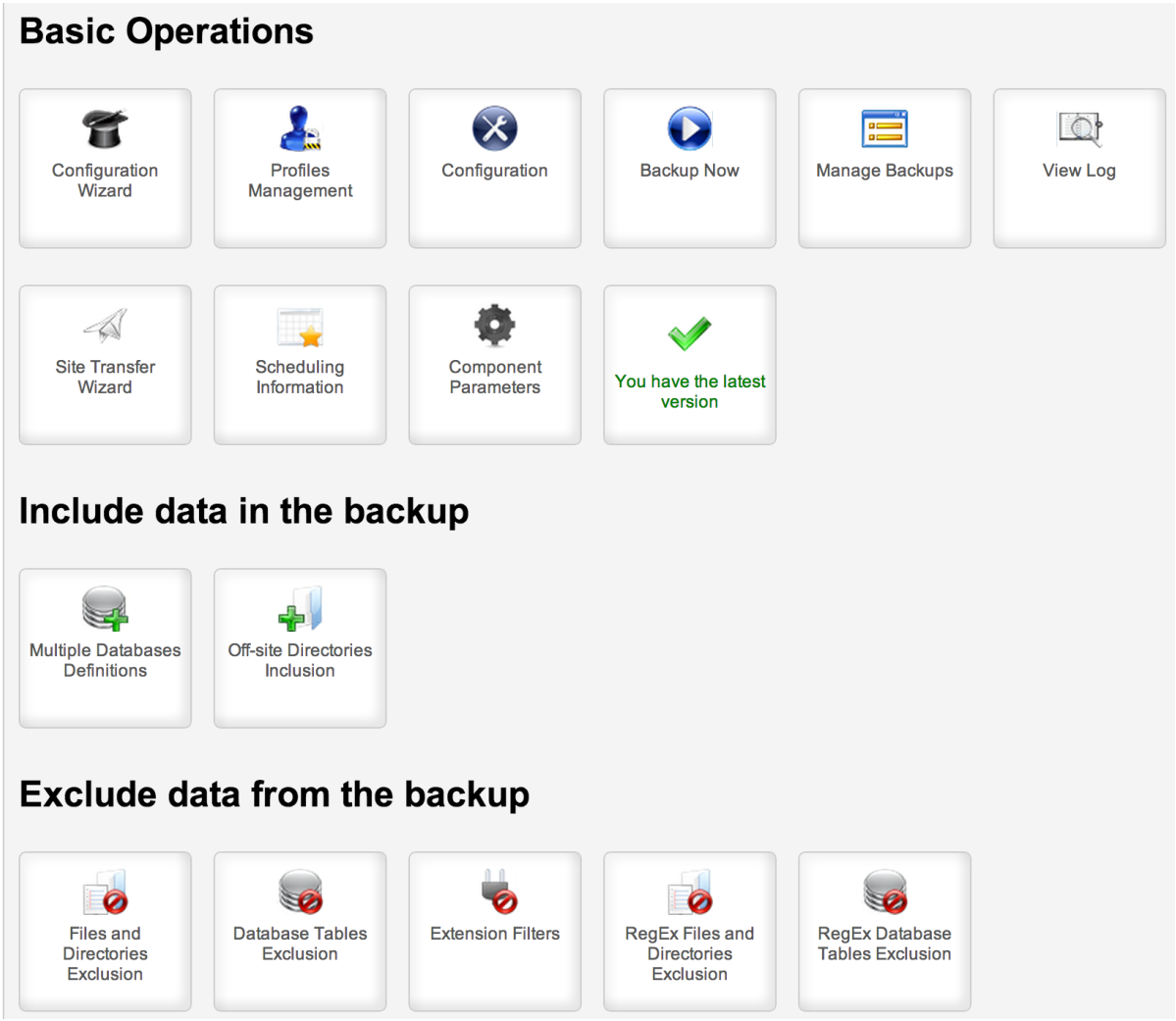
Important

You are supposed to read the full text of the warnings by clicking on each item. Quite often users post for support on our forum asking something which is already written in the full text of the warnings. Please, DO NOT seek support unless you have read the detailed descriptions of all of the potential problems appearing in this box.

Below of all this information you can find a donation link. If you feel that Akeeba Backup has saved your day - and you do not wish or can't afford subscribing to the Professional edition - you can donate a small amount of money to help us keep the free version going!

Backup Statistics — This panel informs you about the status of your last backup attempt. The information shown is the date and time of backup, the origin (e.g. remote, backend, frontend and so on), the profile used and the backup status.

The left navigation panel set



The left navigation panel set allows access to the different functions of the component, by clicking on each icon.

You can edit the component-wide options (formerly: component parameters) by clicking on the Options button towards the top right hand of the page, in the Joomla! toolbar area.

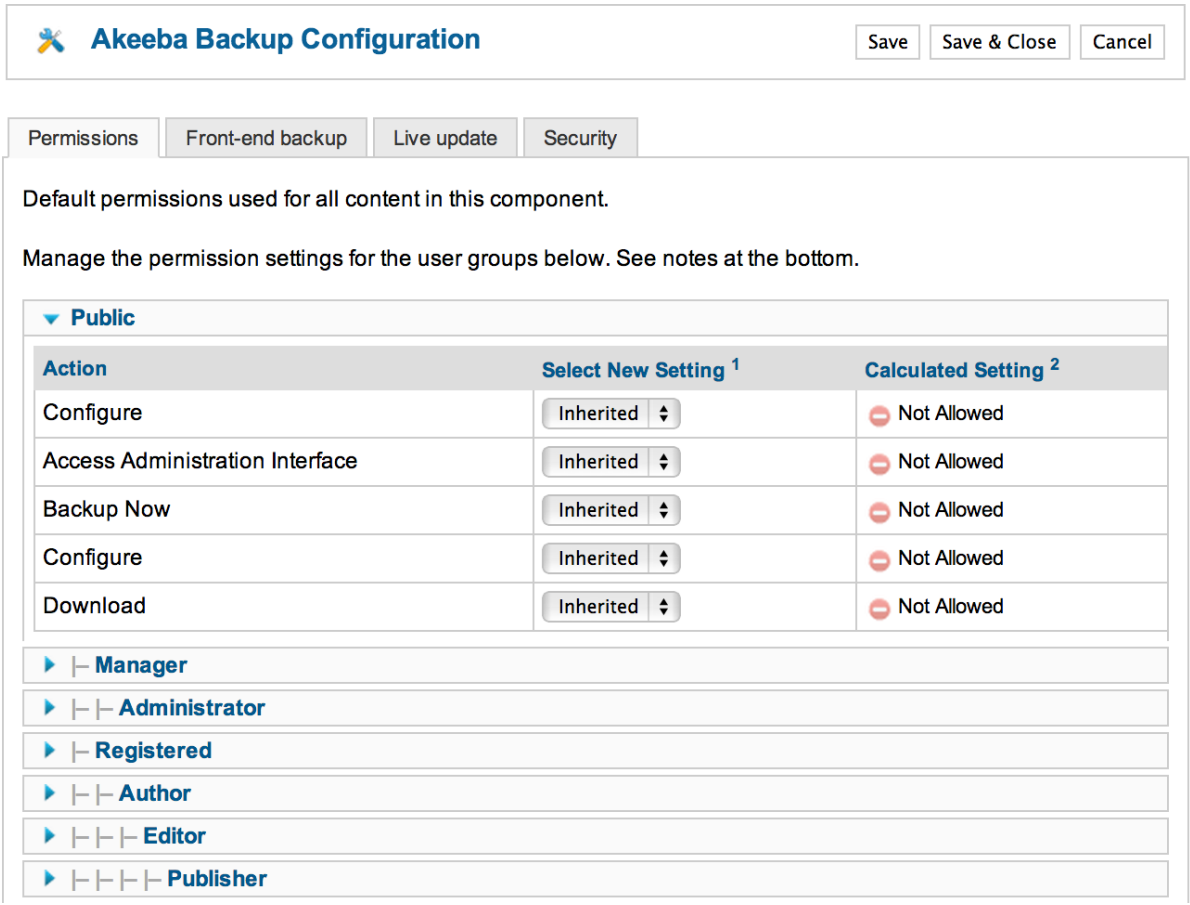
2.2.1. Editing the component's Options

You can edit the component-wide options (formerly: component parameters) by clicking on the Options button towards the top right hand of the page, in the Joomla! toolbar area. The Options editor opens in a new page.

Component options are component-wide and take effect regardless of the active profile.

There are several tabs:

Permissions



This is the standard Joomla! ACL permissions setup tab. Akeeba Backup fully supports supports Joomla! ACLs and uses the following three custom permissions:

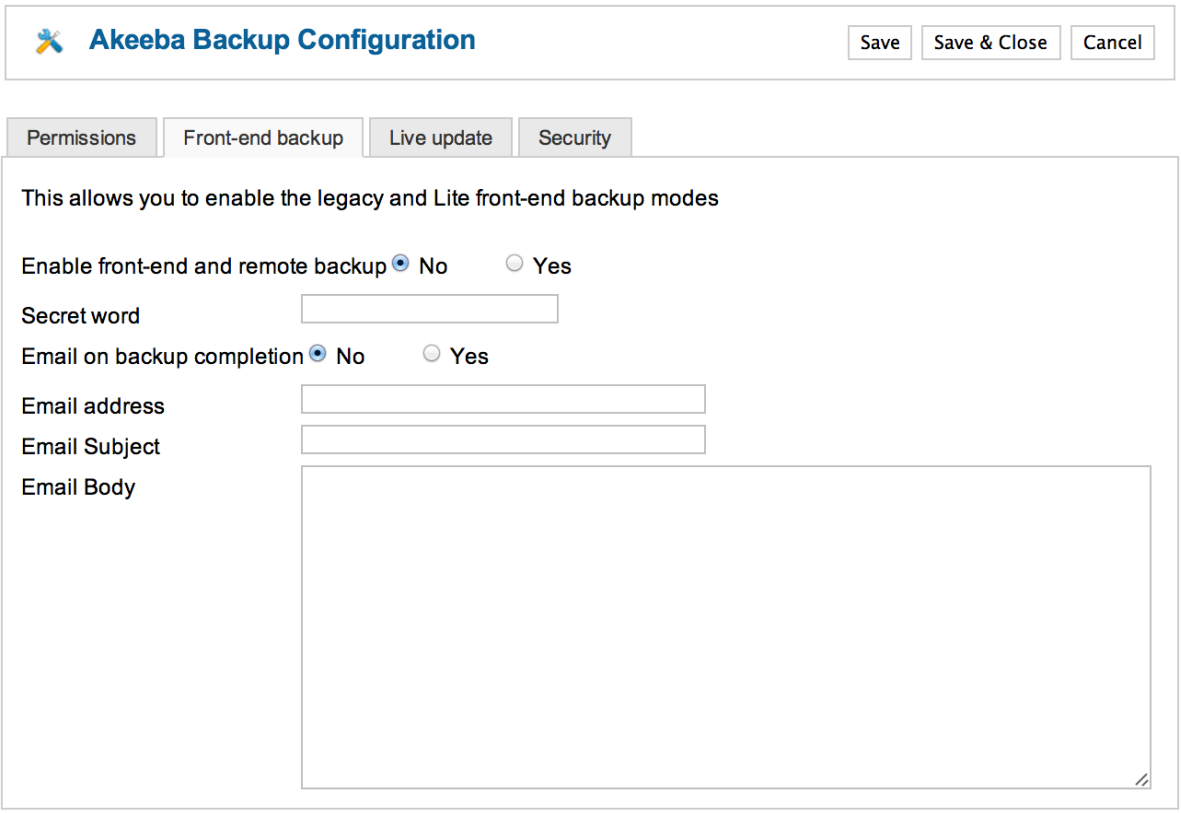
Backup Now — Allows the users of the group to take backups.

Configure — (The second one displayed in each group) Allows the users of the group to access the Configuration page, as well as all features which define what is included/excluded from the backup

Download — Allows the users of the group to download backup archives from the Manage Backups page.

Front-end backup

Here you can define options which affect front-end, CRON and remote backups.



Enable front-end and remote backup

Akeeba Backup allows you to take backups from the front-end, or from compatible remote clients (e.g. Akeeba Remote CLI and other third party products or services). In order to be able to do so, you have to enable this option.

Secret word

Whenever you need to take a front-end backup, you have to supply this secret word to let Akeeba Backup know that you really have access to its functions and you're not an impostor, or a hacker attempting to cause a massive denial of service attack by overloading your server with backup operations.

Please note that if you use any character other than a-z, A-Z and 0-9 you MUST NOT use the secret word verbatim in the front-end backup URL. Instead, you have to URL-encode it. The Schedule Automatic Backups page does that automatically for you. Just go to Components, Akeeba Backup, click Schedule Automatic Backups, scroll all the way down and use one of the tabs to get the URL or command line you need to use with the secret word properly encoded in the URL.

Since Akeeba Backup 5.5.2 the Secret Word is stored encrypted in the database by default. It will be stored unencrypted with you have either explicitly set the Use encryption option to No or if your server does not support settings encryption. When the Secret Word is stored encrypted the text in this field will begin with ###AES128### or ###CTR128###. This is intentional and has to do with the security of your settings. For this reason we urge you to **NEVER** copy the Secret Word directly from this field. Instead, please use the Scheduling Information page in the Control Panel.

Clarification: if you see a Secret Word beginning with ###AES128### or ###CTR128### and you wish *to change it* you can still do so! Just type your new Secret Word. The next time you visit an Akeeba Backup page or run a front-end or remote backup it will be automatically encrypted.

Important

For security reasons you are recommended to use a "secret word" consisting of at least 16 random, mixed case alphanumeric characters. It should not be a dictionary word or based off a dictionary word. One good resource for truly random secret words is Random.org's password generator [https://www.random.org/passwords/?num=1&len=24&format=html&rnd=new]. A secret key returned by this generator would require several quadrillions of trillions years to brute force using the available technology in the foreseeable future, i.e. it's really secure to use.

DO NOT USE SHORT OR SIMPLE SECRET WORDS such as "p@ssw0rd", "secret", "admin", "1234", "unicorn", "Morpheus" or "supercalifragilisticexpialidocious". An attacker would try these simple passphrases first and take control over your backups very easily.

Warning

As of Akeeba Backup 4.5.0, the front-end backup feature and the JSON API will be **DISABLED** if you are using a Secret Word with a low complexity. The complexity is calculated based on password best practice criteria. As a rule of thumb use a 16 character secret word consisting of mixed case alphanumeric characters.

Note

Why is this field not a password field? The Secret word is transmitted in the clear when you load the page and is also visible when you view the source of the page or right click on the field and choose Inspect Element. In other words, as long as someone has access to the component configuration page they can trivially find out the secret word. Not to mention that the secret work is also plainly visible in the Schedule Automatic Backups page.

Email on backup completion
: When enabled, Akeeba Backup will send an email regarding the backup status every time a front-end or remote backup is complete or failed.

Email address
: When the above option is enabled, the email will be sent to this email address. If you leave it blank, Akeeba Backup will send a copy of the email to all Super Administrators of the site.

Email subject
: This option lets you customise the subject of the email message which will be sent when a remote, CRON or front-end backup succeeds. You can use the same variables you can use in file names, i.e. [HOST] for the domain name of your site and [DATE] for the current date and time stamp. Leave blank to use the generic default option.

Email body
: This option lets you customise the body of the email message which will be sent when a remote, CRON or front-end backup succeeds. Leave blank to use the generic default option. The email is delivered as plain text; you may not use any HTML to format it. You can use the same variables you can use in file names, i.e. [HOST] for the domain name of your site and [DATE] for the current date and time stamp, inside the body text. Moreover, you may also use any or all of the following variables in order to enhance the clarity of your message:

  [PROFILENUMBER] The numeric ID of the current backup profile

  [PROFILENAME] The description of the current backup profile

  [PARTCOUNT] The number of archive parts of the backup archive which was just generated

  [FILELIST] A list of filenames of the archive parts of the backup archive which was just generated

[REMOTESTATUS] Available since Akeeba Backup 3.5.3. Shows the status of post-processing, e.g. uploading the file to remote storage like Amazon S3. If you are not using post-processing, this is always empty. If the transfer to the remote storage was successful it will output "Post-processing (upload to remote storage) was successful". If the transfer fails it will output "Post-processing (upload to remote storage) has FAILED".

The options under Check for failed backups are used with the feature for checking for failed backups automatically.

Stuck backup timeout — A backup will be considered stuck (failed) after this many seconds of inactivity. Please note that uploading backup archives to remote storage, such as Amazon S3, using the native CRON mode might take substantially longer than that. We advise you to leave this value as is and schedule the backup failure checks to take place a substantial amount of time (e.g. 1 hour) after the expected end time of your scheduled backups. If a backup failure check takes place before a backup has finished it is very possible that you will end up with a failed backup!

Email address — The email address which will be notified for failed backups

Email subject — Leave blank to use the default. You can use all of Akeeba Backup's variables you can use for naming archive files, e.g. [HOST] and [DATE]

Email body — Leave blank to use the default. You can use all of Akeeba Backup's variables you can use for naming archive files, e.g. [HOST] and [DATE].

Live update

These options define how Akeeba Backup will notify you regarding available updates

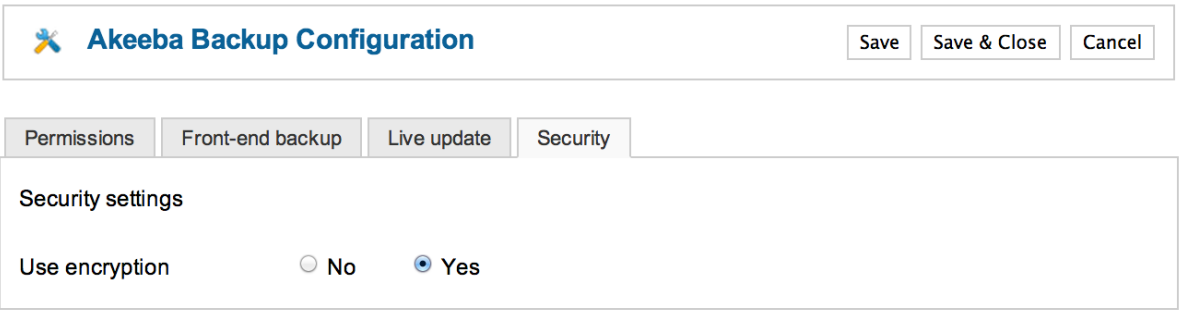
Download ID — If and only if you are using the Professional release you have to specify your Download ID for the live update feature to work properly. You can get your Download ID by visiting AkeebaBackup.com and clicking My Subscriptions. Your Download ID is printed below the list of subscriptions. Filling in this field is required so that only users with a valid Professional subscription can download update packages, just as you'd expect from any commercial software.

Note

Users of Akeeba Backup Core do not need to supply this information. Akeeba Backup Core is provided free of charge to everybody, therefore there is no need to validate the update against a username and a password.

Security

These options define how Akeeba Backup will secure your settings



Use Encryption — Your settings can be automatically stored encrypted using the industry standard AES-128 encryption scheme. This will protect your passwords and settings from prying eyes. If,

however, you do not want to use this feature, please set this option to No and reload the Control Panel page to apply this setting. Do note that your server must have either the mcrypt or the OpenSSL PHP extension installed for this feature to work. Please keep in mind that even if your site is using HTTPS this doesn't mean that you have the OpenSSL *PHP extension* installed. You usually have to ask your host to enable it for you.

Tip

For security reasons, we recommend always having this option turned on

Please note that you may have to go to the Configuration page and click on the Save & Close button before Akeeba Backup can successfully detect if your server supports encryption or not. Before doing that, Akeeba Backup might always report that your server does not support encryption.

Back-end

These options define how Akeeba Backup will display its administration interface

Date format — Defines how the Start time of backups will display in the Manage Backups page. Leave blank to use the default date format. The date format follows the conventions of the PHP date() function [http://www.php.net/date].

Push notifications

Akeeba Backup 4.2.2 and later can notify you on backup start, finish and –sometimes– on backup failure using push notifications delivered through the third party application Pushbullet. Push messages are delivered to all your devices running the Pushbullet client software including smartphones and tablets (iOS, Android, Windows) as well as laptops and desktops (Windows, Linux, Mac OS X).

Please note that backup failure notifications are only delivered for backups started through the back-end. For technical reasons beyond our control these notifications can not be delivered for remote (JSON API) and scheduled (CRON job) backups: if the backup fails the PHP executable stops working, therefore our PHP code to send notifications can not work.

Push notifications — Select the push notifications type. Currently only Pushbullet and None are supported. If you choose None the push notifications are disabled.

Pushbullet Access Token — Enter your Pushbullet Access Token. You can find it in your Pushbullet account page [https://www.pushbullet.com/account]. Do note that this token gives full access to your Pushbullet account and is visible by everyone who can view and edit Akeeba Backup's settings.

3. Basic Operations

The Basic Operations group contains the most common functions you will need on your daily Akeeba Backup usage. In fact, you will only use the other pages sparingly, mostly when you create a backup profile or want to update it after doing significant changes to your site.

3.1. Profiles Management

Profiles Management page

The Profiles Management page is the central place from where you can define and manage *backup profiles* . Each backup profiles can be regarded as a container holding Akeeba Backup configuration values and filter settings. Each one uniquely and completely defines the way Akeeba Backup will perform its backup process.

The main page consists of a list of all backup profiles. On the left hand column there is a check box allowing the selection of a backup profile so that one of the toolbar operations can be applied. The other column displays the description of the backup profile. Clicking on it leads you to the editor page, where you can change this description.

On the page's toolbar you can find the operations buttons:

New
: Creates a new, empty profile. Clicking on this button will lead you to the editor page, where you can define the name of the new profile, or cancel the operation if you've changed your mind.

Copy
: Creates a prostine copy of the selected backup profile. The copy will have the same name and include all of the configuration options and filter settings of the original.

Delete
: Permanently removes all selected backup profiles. All associated configuration options and filter settings are removed as well. This is an irreversible operation; once a profile is deleted, it's gone forever.

  You can only delete one profile at a time. If you select multiple profiles, only the first one (topmost) will be removed.

When you create a new profile or copy an existing profile, the newly generated profile becomes current. This means that you can work on your new profile as soon as you're finished creating it, without the need to manually make it current from the Control Panel page.

To the left of each profile's name you will find two buttons:

Configure...
: Clicking this button makes that profile current and opens the Configuration page. This is equivalent to going back to the Control Panel, selecting that profile in the list, waiting for the page to reload and clicking on Configuration. We figured that having to click to just one button is much faster – and simpler!

Export
: Since Akeeba Backup 3.6.6 you can export a profile in JSON format. Clicking this button will ask you to download a file with all of the profile settings. You will be able to import that file on the same or a different site using the Import feature further down the page.

  ### Warning

  Please note that the file you are downloading contains all of the configuration information **UNENCRYPTED**. We strongly advise you to NEVER, EVER use

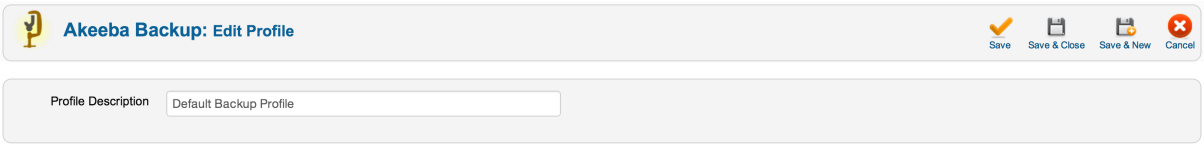
this feature on a shared connection (e.g. library, airport, Internet cafe, etc), over an unencrypted Wi-Fi network, when using a self-signed SSL certificate or in any other case where the security of your data is not guaranteed. It's fine using it over a secure connection, i.e. an HTTPS connection using a commercially signed (not self signed!) SSL certificate. We also strongly advise against storing exported profile files in media or services which are not encrypted and could be compromised, e.g. USB keys you use at your clients' office (their PC may be laden with malware unbeknownst to you) or unencrypted cloud storage. Something like an IronKey, an encrypted ZIP archive (using AES encryption, not the legacy ZIP encryption) or a hard disk protected with full-disk encryption software is always the preferred storage method. This may sound paranoid –and it is– but remember that your configuration data may contain sensitive information like your access credentials to an FTP server, an Amazon S3 account, a Dropbox account, database connection credentials and so on.

You can also find an Import area below the list of profile. Use the file browser field to select a previously exported profile file from the same *or a different* site. Then click the Import button. Akeeba Backup will import the profile at the end of the profiles list.

Important

We strongly advise you to review your settings after importing a profile. If the profile comes from another site and you have used an absolute path or overridden the database connection information you will have to change those settings to reflect the new site's parameters.

The Edit Profile page



The editor page which appears when creating or editing a profile is trivial. The only changeable parameter is the profile's description. Clicking on Save & Close applies the settings and gets you to the main Profiles Management page. Clicking on Apply applies the settings and returns you to the editor page. Finally, clicking on Cancel will disregard any changes made and get you to the main Profiles Management page.

Tip

In order to configure the settings of the profile click on the Configure button next to it in the profiles list.

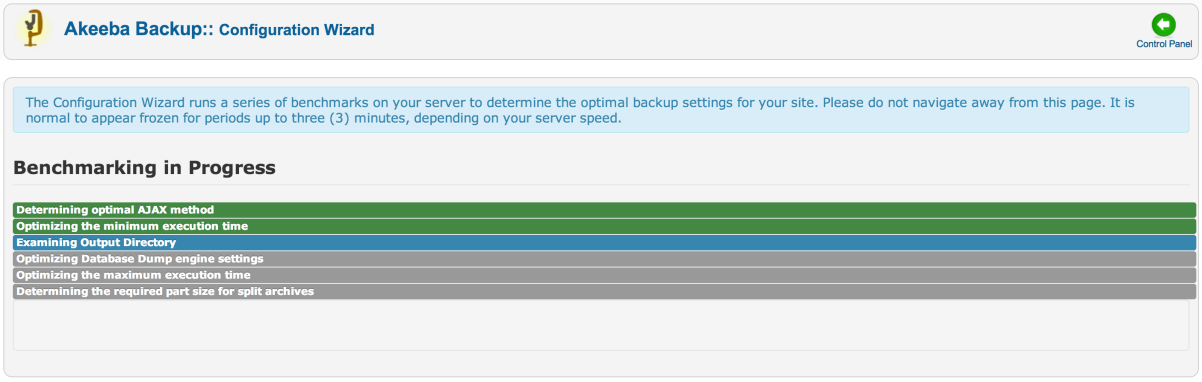
3.2. Configuration Wizard

Akeeba Backup 3.1.5 and later include the Configuration Wizard feature. This is an automated process which will benchmark your server's performance and try to fine tune common configuration variables for optimal backup performance. The Configuration Wizard settings are applied to the current profile only. If you want to fine tune a different profile, you have to select it from the drop-down list in the Control Panel page before clicking on the Configuration Wizard button. Do note that using the Configuration Wizard has the following effects:

- Your backup type is switched to "Full site backup"
- The archiver engine is switched to "JPA (Recommended)"

If you want to use a different backup type and/or archive type, you can review the configuration changes after the wizard is finished.

The Configuration Wizard page



The Configuration Wizard will automatically fine tune the following configuration parameters:

- AJAX method (use AJAX or IFrames)
- Optimize the minimum execution time so as to make the backup as fast as possible without your server throwing 403 Forbidden errors
- Adjust the location and/or permissions of the output directory. Useful if you just transferred your site to a new server or location.
- Optimize the database dump engine settings to make database dump as fast as possible, while avoiding memory outage errors
- Optimize the maximum execution time so that as few steps as possible are performed during the backup, without causing a timeout
- Automatically determines if your server needs archive splitting.

Important

The Configuration Wizard does not address the archive splitting required when you are using a post-processing engine (such as backup-to-email, S3, Dropbox, etc). If you are using post-processing you may have to manually set the Part Size for Split Archives to a different value manually.

At the end of the wizard process, you can either try taking a backup immediately or review and possibly modify the configuration parameters.

3.3. Configuration

Note

Some options discussed below may be only available in the for-a-fee Professional edition!

The Configuration page is split in many sections - or panes, if you like - each one serving as a group for closely related options. Each of those panes displays a title and below it you can find all of the options. Hovering your mouse of the label - the left hand part of each row - you will be presented with a quite big tooltip providing short documentation of the setting and its available options. This way you won't have to refer to this document constantly when configuring Akeeba Backup.

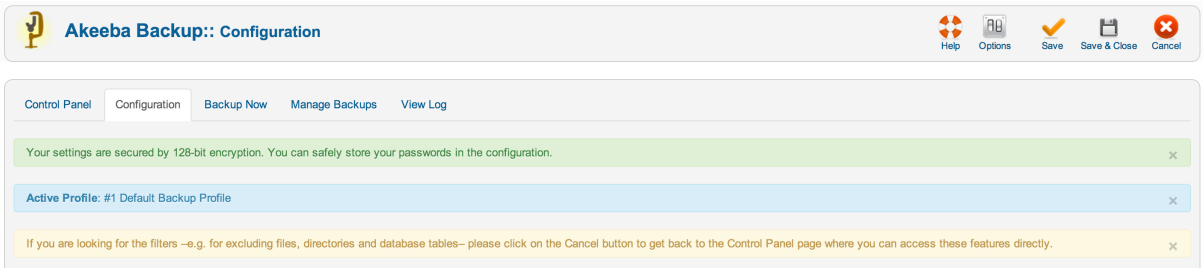
Some of the settings also feature a button. They can either do some action, like browsing for a folder and testing connection parameters, or it may be labeled Configure.... This latter case is mostly interesting, as pressing the button will toggle the display of a sub-pane which contains options pertaining to this specific option. This GUI pattern is primarily used for "engines" type settings.

Another interface element worth mentioning are the composite drop-downs. Whenever you are supposed to enter a number, Akeeba Backup presents you with a drop-down menu of the most common options. You can either select a value from the list, or select "Custom...". In the latter case, a text box appears to the right of the drop-down. You can now type in your desired value, even if it's not on the list. Do note that all of these elements have preset minimum/maximum values. If you attempt to enter a value outside those boundaries, or an invalid number, they will automatically revert to the closest value which is within the presents bounds.

Note

If you had been using earlier releases of Akeeba Backup, you will remember that these values used to use a draggable slider. Since the slider was rather "jumpy" and hard to configure, we reverted to using composite drop-downs in order to make entry of settings easier and faster.

The top of the Configuration page



On the top of the page you can see the numeric ID and title of the active backup profile. This acts as a reminder, so that you know which profile's settings you are editing. The toolbar also contains a Parameters button. Clicking on it will launch the profile-independent, component-wide parameters editor. It's the same as clicking the Options button in the toolbar area of the Control Panel page.

The toolbar also has the following buttons:

- Save. Saves all changes and comes back to the same configuration page.
- Save & Close. Saves all changes and returns to the main Control Panel page of the component.
- Save & New. Saves all changes and creates a new backup profile with the new (saves) changes. Then it switches to this new backup profile and opens its Configuration page. This allows you to create multiple variations of the same backup profile very easily. It is equivalent to using Copy in the backup profiles page and then clicking on the profile's Configuration button, only with less clicks.

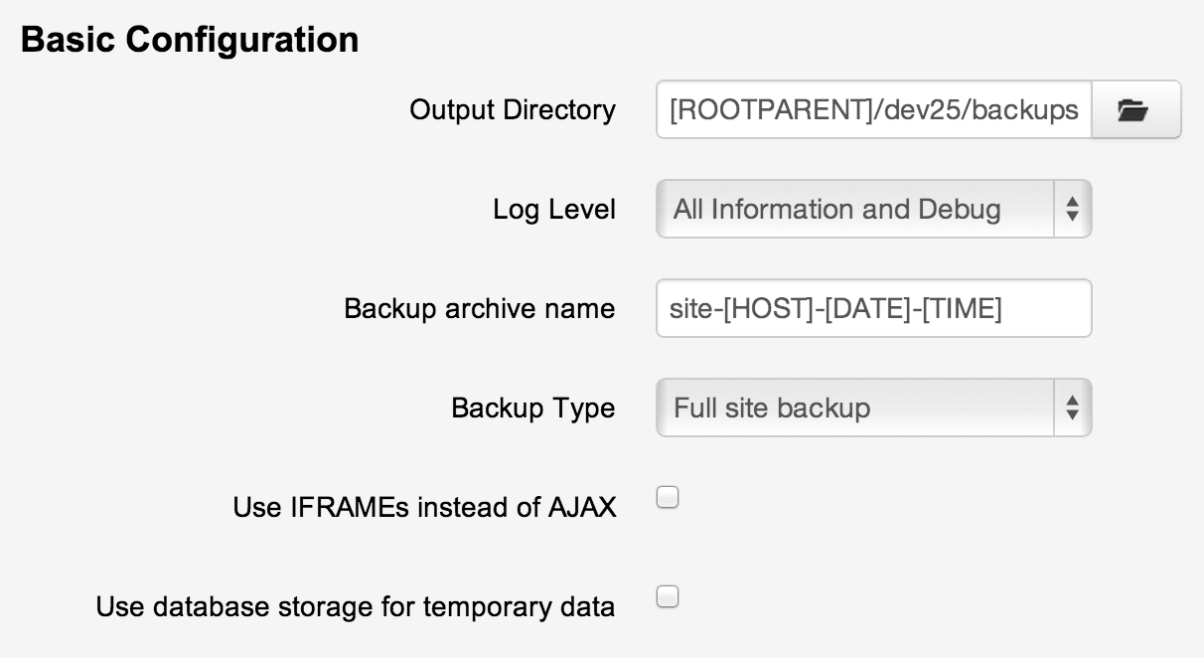
Below the toolbar you will find the Profile Description area. You can view and change the backup profile's description here, without having to go through to the backup profiles page.

The rest of this document is separated into sub-sections. The first sub-section describes the settings of each of the main configuration panes, whereas the rest of the sections discuss the settings made available to you through sub-panes.

3.3.1. The main settings

3.3.1.1. Basic Configuration

Basic configuration



Output Directory

This is the directory where the result of the backup process goes. The result of the backup - depending on other configuration options - might be an archive file or an SQL file. This is also where your *backup log file* will be stored. The output directory must be accessible and writable by PHP.

> **Important**
>
> Providing a directory with adequate permissions might not be enough! There are other PHP security mechanisms which might prevent using a directory, for example the `open_basedir` restriction which only allows certain paths to be used for writing files from within PHP. Akeeba Backup will try to detect and report such anomalies in the Control Panel page before you attempt a backup.

You can use the following variables to make your setting both human readable and portable across different servers - or even different platforms:

- **[DEFAULT_OUTPUT]** is replaced by the absolute path to your site's `administrator/components/com_akeeba/backup` directory. This is assigned as the default location of output files unless you change its location. If you leave it as it is, you are supposed to make sure that the permissions to this directory are adequate for PHP to be able to write to it.
- **[SITEROOT]** is automatically replaced by the absolute path to your site's root
- **[ROOTPARENT]** is automatically replaced by the absolute path to the parent directory of your site's root (that is, one directory above your site's root)

Is this over your head? No problem! Just click on the Browse... button and a pop-up directory navigator will allow you to find the proper directory. Next to the folder's location there is the button labeled Use. Click on it to make the current directory the selected one and close the pop-up. To make it even easier for you, Akeeba Backup displays a small icon next to the Use

button. If it's a green check mark the directory is writable and you can use it. If it's a red X sign, the directory is not readable and you either have to select a different directory, or change this directory's permissions.

Warning

NEVER, EVER, UNDER ANY CIRCUMSTANCES SHOULD YOU USE YOUR SITE'S ROOT AS YOUR OUTPUT DIRECTORY! This will usually lead to corrupt backup or backup failure. The reason is that the output directory and all of their contents are automatically excluded from the backup set. However, even if your backup succeeds due to a bug (remember, it's supposed to fail!), using your public, web accessible site root as your output directory is like a party invitation to hackers worldwide. If you come to our forum with such a setup and a broken backup we can't help you.

Log Level

This option determines the verbosity of Akeeba Backup's log file:

- **Errors only**. Only fatal errors are reported. Use this on production boxes where you have already confirmed there are no unreadable files or directories.
- **Errors and warnings**. The minimum recommended setting, reports fatal errors as well as warnings. Akeeba Backup communicates unreadable files and directories which it wasn't able to backup through warnings. Read the warnings to make sure you don't end up with incomplete backups! Warnings are also reported in the Backup Now page GUI irrespective of the log verbosity setting as a convenience.
- **All information**. As "Error and Warnings" but also includes some informative messages on Akeeba Backup's backup process.
- **All Information and Debug**. This is the recommended setting for reporting bugs. It is the most verbose level, containing developer-friendly information on Akeeba Backup's operation. This is what we need to help you in case of a problem. This will also create a 2-5Mb log file on most sites, so you should only use this until you have achieved consistently valid backup archives creation.
- **None**. This log level *is not recommended*. You should only use this if you are paranoid and want no log files written on the server. However, if you are truly concerned about security, you should protect the backup output directory instead of using this log level!

Our servers usually run on Errors and Warnings or All Information levels. When we are testing new releases or change our server setups, we switch to All Information and Debug until we are sure everything is flowing smoothly.

Backup archive name

Here you can define the naming template of backup files. There are a few available variables. Variables are special pieces of text which will be expanded to something else at backup time. They can be used to make the names of the files harder to guess for potential attackers, as well as allow you to store multiple backup archives on the output directory at any given time. The available variables and their expansion at backup time are:

[HOST] The configured host name of your site.

Note

This doesn't work in the native command-line CRON mode, i.e. using akeeba-backup.php for producing automated backups. In such a case, it will be replaced with an empty string (no text).

[DATE] The current server date, in the format YYYYMMDD (year as four digits, month as two digits, day as two digits), for example 20080818 for August 18th 2008.

[YEAR]	The year of the current server date, as four digits
[MONTH]	The month of the current server date, as two digits (zero-padded)
[DAY]	The day of the current server date, as two digits (zero-padded)
[WEEK]	The current week number of the year. Week #1 is the first week with a Sunday in it.
[WEEKDAY]	Day of the week, i.e. Sunday, Monday, etc. The full name is returned in your current Joomla! language. Front-end, remote and CRON backups may return this in English or your default Joomla! language. This is not a bug, it is how Joomla!'s translation system is supposed to work.
[RANDOM]	A 64-character random string. Use sparingly, it can cause backup failure due to the file name being too long for your server
[TIME]	The current server time, in the format HHMMSS (hour as two digits, minutes as two digits and seconds as two digits), for example 221520 for 10:15:20 pm.
[VERSION]	The version of Akeeba Backup. Useful if you want to know which version of Akeeba Backup generated this archive file.
[PLATFORM_NAME]	The name of the platform Akeeba Backup is currently running under. This always returns "Joomla!".
[PLATFORM_VERSION]	The version of the platform Akeeba Backup is currently running under. This always returns the current Joomla! version, e.g. 2.5.4.
[SITENAME]	The name of the site, lowercased and transformed into a format which guarantees compatibility with all filesystem types commonly found in modern Operating Systems. Please note that the site name will be trimmed at 50 characters if it's longer.

Backup Type — It defines the kind of backup you'd like to take. The backup types for Akeeba Backup are:

- **Full site backup** which backs up the Joomla! database, any extra databases you might have defined and all of the site's files. This produces a backup archive with an embedded installer so that you can restore your site with ease. This is the option 90% of the users want; it is the only option which creates a full backup of your site, capable of producing a working site if everything is wiped out of your server.
- **Main site database only (SQL file)** which backs up only the Joomla! database. It results in a single SQL file which can be used with any database administration utility (e.g. phpMyAdmin for MySQL, pgAdmin3 for PostgreSQL etc) to restore only your database should disaster strike. This option is recommended for advanced users only.
- **Site files only** which backs up nothing but the site's files. It is complementary to the previous option.

Warning

Having one "main site database" backup and one "sites files only" backup is not equal to having a full site backup! The full site backup also includes an installation script which, just like Joomla!'s web installer, allows you to effortlessly recover your site even if everything is wiped out of your server. It acts as the glue between the two pieces (files and database).

- **All configured databases (archive file)** which creates an archive file containing SQL files with dumps of your main site's database and all of the defined multiple databases. The database dumps can be restored by any database administration tool (for example phpMyAdmin for MySQL). The difference to the second option is that it produces an uncompressed SQL file and doesn't include any extra databases which you might have defined.

  ## Note

  Extra - or "multiple" - database definitions are only available in the Professional edition of the component.

- **Incremental (files only)**. This is the same as the Site files only option, but instead of backing up all of your site's files, it only backs up the files which changed since the last time you performed a backup. The only comparison made is between the file's modification time and the last successful backup's time. The "last successful backup" refers to the last backup made using this backup Profile and which has a status of "OK", "Remote" or "Obsolete".

  Restoring an incremental backup set is a *manual process*. You have to manually extract the files from your "base" backup (an archive made with a Full Site Backup profile), then extract all incremental archives on top of it. Finally, used this collection of extracted files to restore your site. This process should only be used if you really know what you are doing. Do not trust that Akeeba Backup can sort out the collection of incremental backups and help you restore them. It won't.

Client-side implementation of minimum execution time

Akeeba Backup splits the backup process into smaller chunks, called backup steps, to prevent backup failure due to server time-out or server protection reasons. Each backup step has a minimum and maximum duration defined by the Minimum Execution Time, Maximum Execution Time and Execution Time Bias parameters in this Configuration page. If the step takes less time to complete than the minimum duration Akeeba Backup will have to wait.

When this box is unchecked (default) Akeeba Backup will have the server wait until the minimum execution time is reached. This may cause some very restrictive servers to kill your backup. Checking this box will implement the waiting period on the browser, working around this limitation.

Important

This option only applies to back-end backups. Front-end, JSON API (remote) and Command-Line (CLI) backups always implement the wait at the server side.

Use IFRAMEs instead of AJAX

Normally, Akeeba Backup is using AJAX postbacks to perform the backup process without timing out. Its ability to do so depends on how well your server plays along with your browser's Javascript engine. Sometimes, this is just not possible at all and you'll experience the backup stalling at random points through the backup process. If modifying the other options doesn't help, enable this feature. When enabled, instead of using AJAX calls, Akeeba Backup will create a hidden IFRAME in the page and perform all server communications through it. Since IFRAMEs load the backup URL as if it were a regular web page, it minimizes the probability of conflicts. The major drawback is that this method is about 50% slower than the AJAX one, so your backup will take substantially longer.

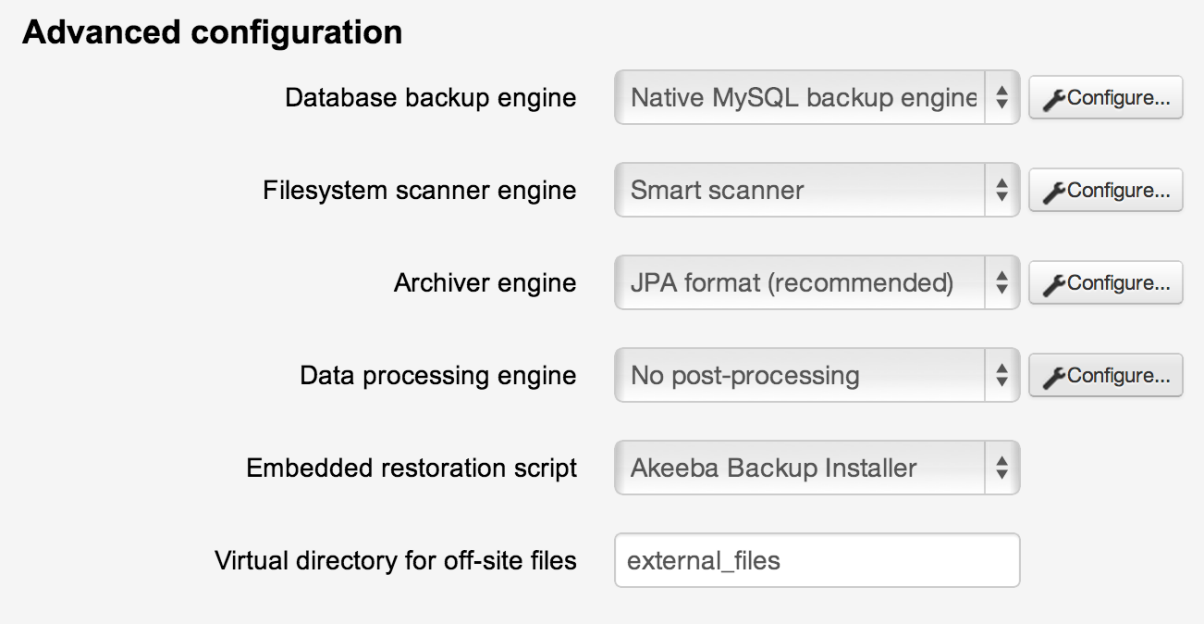
Use database storage for temporary data

Normally, Akeeba Backup stores temporary information required to process the backup in multiple steps inside files in your Output Directory. Sometimes, especially on low-end hosts with ancient versions of PHP, this causes backup issues such as the backup restarting all the time. In those cases, you can check this box and Akeeba Backup will use your site's database to store this temporary information.

Do note that on some hosts this will cause the backup to fail with a "MySQL server has gone away" error message. That is a problem with the host's configuration. In those cases, nothing can be done. Our suggestion: if you receive such an error, migrate your site to a new host as the one you are using is most likely very restricted and severely under-performant. Moving to a faster, more reliable host can benefit your site in many more ways than just being able to run a backup.

3.3.1.2. Advanced configuration

Advanced configuration



Database backup engine
: This option controls how Akeeba Backup will access your database and produce a dump of its contents to an SQL file. It is used with all backup types, except the files only type. The available options for this setting are discussed in the Database dump engines section of this document.

Filesystem scanner engine
: This option controls how Akeeba Backup will scan your site for files and directories to back up. The available options for this setting are discussed in the File and directories scanner engines section of this document.

Archiver engine
: This option controls which kind of archive will be produced by Akeeba Backup. The available options for this setting are discussed in the Archiver engines section of this document.

Data processing engine
: Akeeba Backup allows you to post-process the backup archives once the backup process is over. Post-processing generally means sending them somewhere off-server. This can be used, for example, to move your backup archives to cloud storage, increasing your data safety. The available options for this setting are discussed in the Data processing engines section of this document.

Upload Kickstart to remote storage
: By selecting this option you instruct Akeeba Backup to also upload kickstart.php on the remote storage alongside your backup archive. When used with the Upload to Remote FTP Server and Upload to Remote SFTP Server you can perform easy site transfers without leaving the comfort of your browser. Just enter the new site's (S)FTP information in the Data Processing Engine configuration and select the Upload Kickstart to Remote Storage option, then take a new backup. When the backup is complete just open the new site's kickstart.php URL (e.g. http://www.example.com/kickstart.php) in your browser to begin the restoration on the new site's server. This even works with mobile devices (we strongly recommend using

a tablet or phablet with a display size of at least 7"), allowing you to clone sites even you are on the go!

Archive integrity check

When enabled Akeeba Backup will go through the archive extraction process without writing anything to the disk. This makes sure that the archive is not corrupt. If the archive is found to be corrupt an error is raised and the backup process stops.

This feature will NOT work when the Process each part immediately option is enabled in the Post-processing Engine configuration. When you are processing each part immediately the backup archive parts are transferred away from your server before the end of the backup is reached. As a result it is not possible to do a test extraction (the archive file parts are no longer there, so there's nothing to try and extract). It WILL however work when you are simply using a post-processing engine (e.g. Upload to Amazon S3) without the process each part immediately option. Please bear in mind that the integrity check runs BEFORE post-processing (uploading) the backup archive parts to remote storage because there's no reason to put a broken archive for safe-keeping in remote storage.

This feature will only work if you are using an Archiver Engine which creates backup archives. This is typically the case with most Archiver Engines. Notable exceptions are, of course, the DirectFTP and DirectSFTP engines which do not produce backup archives. If you enable this feature on a backup profile using either of these Archiver Engines you'll get a warning.

Enabling this feature will increase the time required to complete the backup process and use substantially more memory and CPU resources. Akeeba Backup goes through the same archive extraction process as Kickstart with the only difference that it does not write anything to the disk.

Finally do keep in mind that this feature only makes sure the archive can be extracted, it does *not* test whether the database data can be restored or if the restored site works correctly. It's still up to you to do a periodic, complete test restoration of your site.

Embedded restoration script

Akeeba Backup will include a restoration script inside the backup archive in order to make restoration easy and the backup archive self-contained. You do not need anything else except the archive in order to restore a site. Restoration scripts honour the settings in your configuration.php, modifying only those necessary (for example, the database connection information), allowing you to create pristine copies ("clones") of your site to any host. You can find more information about restoration scripts in the next Chapter.

Virtual directory for off-site files

Using the off-site directories inclusion of Akeeba Backup Professional, the component will be instructed to look for files in arbitrary locations, even if they are outside the site's root (hence the name of that feature). All the directories included with this feature will be placed in the archive as subdirectories of another folder, in order to avoid directory name clashes. We call this folder the "virtual directory", because it doesn't physically exist on the server, it only exists inside the backup archive.

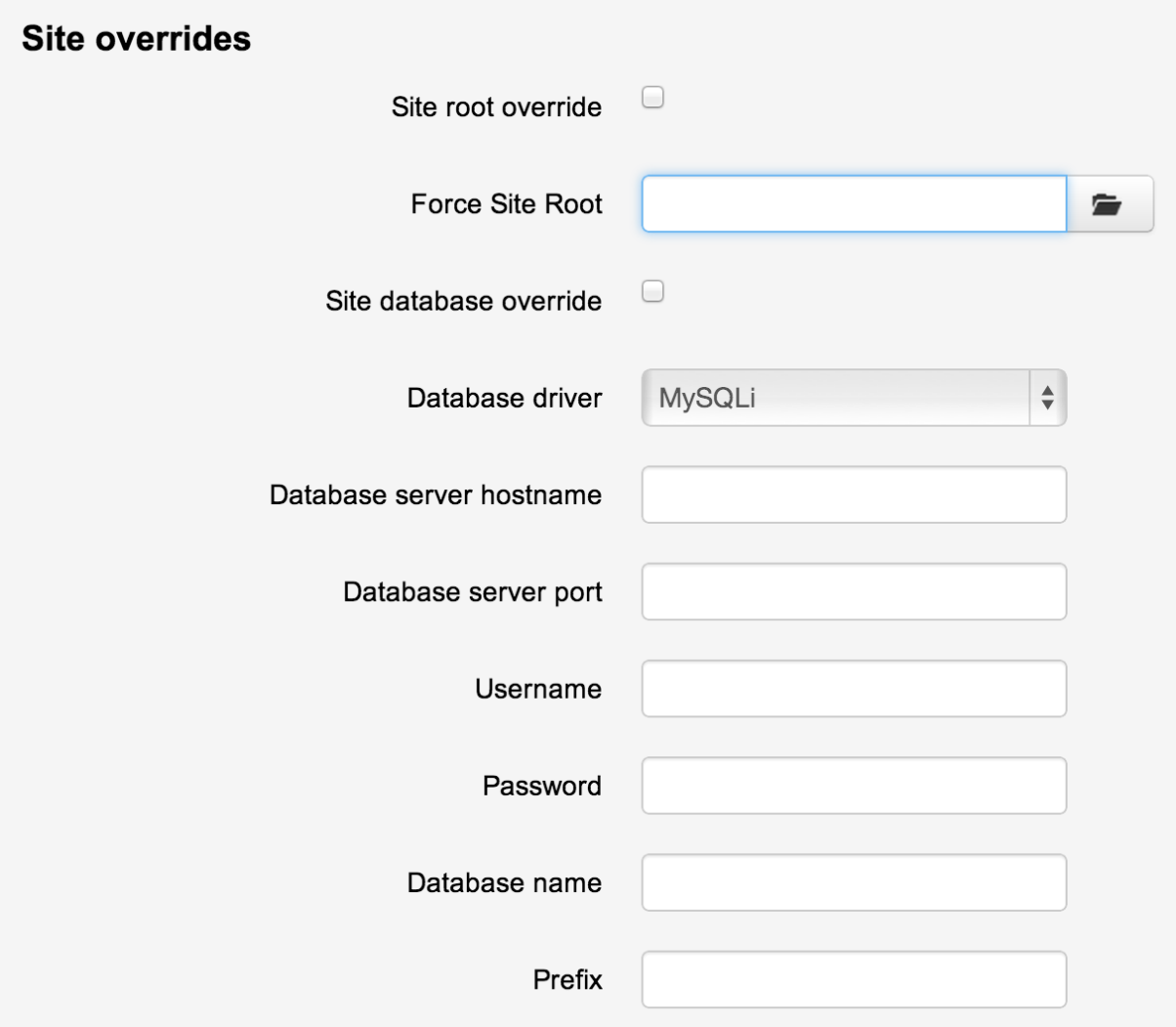
3.3.1.3. Site overrides

These settings are all optional and only available in Akeeba Backup Professional. They allow you to back up a different site than the one Akeeba Backup is currently installed. Essentially, you can install Akeeba Backup on one site and have it back up all sites on the server.

Note

You do not need to set anything up in this section if you only intend to backup or transfer your site. This is only required when you want Akeeba Backup to backup a different site than the one it is installed in.

Site overrides

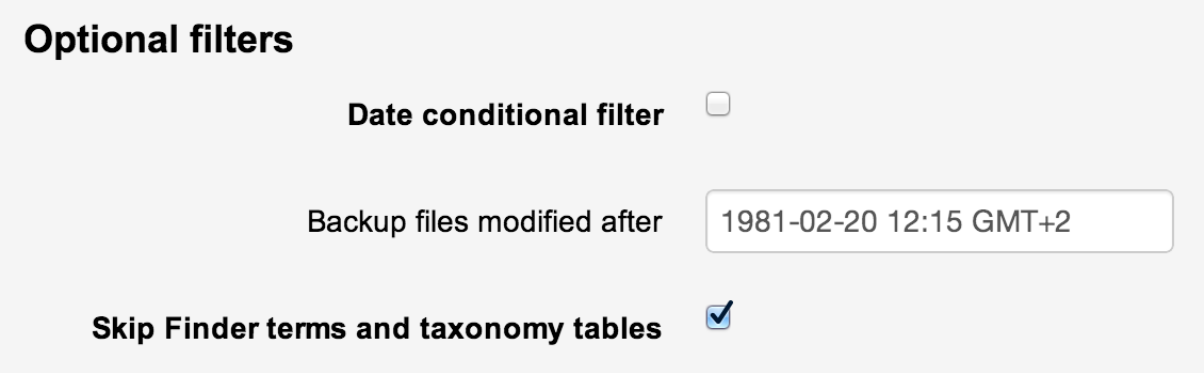


Site root override	When not checked (default), Akeeba Backup will back up the files and folders under the root of the site it is installed. When this option is checked, it will use the site root in the Force Site Root option below. Use this when you want to backup a different site than the one Akeeba Backup is installed in.
Force Site Root	The root of the site to back up. This is only necessary if you have checked the Site root override option above.
Site database override	When not checked (default), Akeeba Backup will back up the database tables inside the database to which the site Akeeba Backup is installed in connects to. In other words, when this option is not checked, Akeeba Backup will back up the current site's database. On the other hand, if this option is checked, Akeeba Backup will backup the database whose connection information you specify in the settings below. Use this when you want to backup a different site than the one Akeeba Backup is installed in.
Database driver	Choose between the database driver. For MySQL databases you can choose between the MySQL and MySQLi driver. If you do not know the difference between the two, MySQLi (with the trailing "i" which stands for "improved") is the best choice.
Database hostname	The hostname or IP address of the database server. Usually that's `localhost` or `127.0.0.1`. If unsure, ask your host.

Database server port	If your database server uses a non-standard port, enter it here. If you have no idea what this means, you most likely need to leave that field blank.
Username	The username to connect to your site's database.
Password	The password to connect to your site's database.
Database name	The nae of your database.
Prefix	The prefix of the tables of the site you're backing up. That's the common part of their names up to and including the first underscore.

3.3.1.4. Optional filters

Optional filters



Since Akeeba Backup 3.2 this section contains optional inclusion and exclusion filters which can be activated to customize your backup procedure. The available filters are:

Date conditional filter

It allows you to backup only files modified after a specific date and time. This is different than the incremental file only backup. It allows you to backup files newer than the specified date no matter which backup mode (full site backup, files only backup, incremental files only backup) you are using. The available options are:

Date conditional filter	Tick the checkbox to activate this filter
Backup files modified after	Files before this date and time will be skipped from the backup set. The format for the date and time parameter is YYYY-MM-DD HH:MM:SS TIMEZONE. This means that you have to specify the year as four digits, followed by a dash, then the month as two digits (e.g. 09 for September), followed by a dash, then the day as two digits (e.g. 01 for the 1st day of the month). For example, September 1st, 2010 is written as 2010-09-01. If you want to specify the time, leave a space after the date and write down the time as the hour using two digits (00-23, no a.m./p.m. is supported!), then a semicolon, then the minutes as two digits, followed by a semicolon, then the seconds as two digits. For example 59 seconds after 11:05 p.m. is written as 23:05:59. You can optionally leave a space after the time and specify the timezone as GMT+/-time. For example, GMT-6 is Dallas time which is six hours behind the GMT and GMT+2 is two hours ahead of GMT which is the Eastern Europe Time. If you do not specify a timezone the GMT timezone is assumed.

> **Important**
>
> You have to set your server's timezone in Joomla!'s Global Configuration for this feature to work reliably. If you get strange results, try editing your site's Global Configuration before asking us for support.

Skip Finder terms and taxonomy tables	Since Joomla! 2.5, the Joomla! CMS ships with a feature called "Smart Search", also known as "Finder". This is a mini search engine built into the CMS. It works by scanning your content and keeping a complex database structure linking potential search terms (words) with content items in compatible components. Due to its nature it stores an immense amount of information in the database. This information takes a very long time to back up. Moreover, this information doesn't need to be backed up as it can be regenerated by using the "Reindex" button in Smart Search's back-end interface. In the interest of speeding up your backups and not including redundant information in the backup Akeeba Backup by default has this option enabled. This instructs the database backup portion of our backup engine to skip backing up the contents of Finder's (Smart Search's) tables. If for some reason you want to back up this content please uncheck this box.

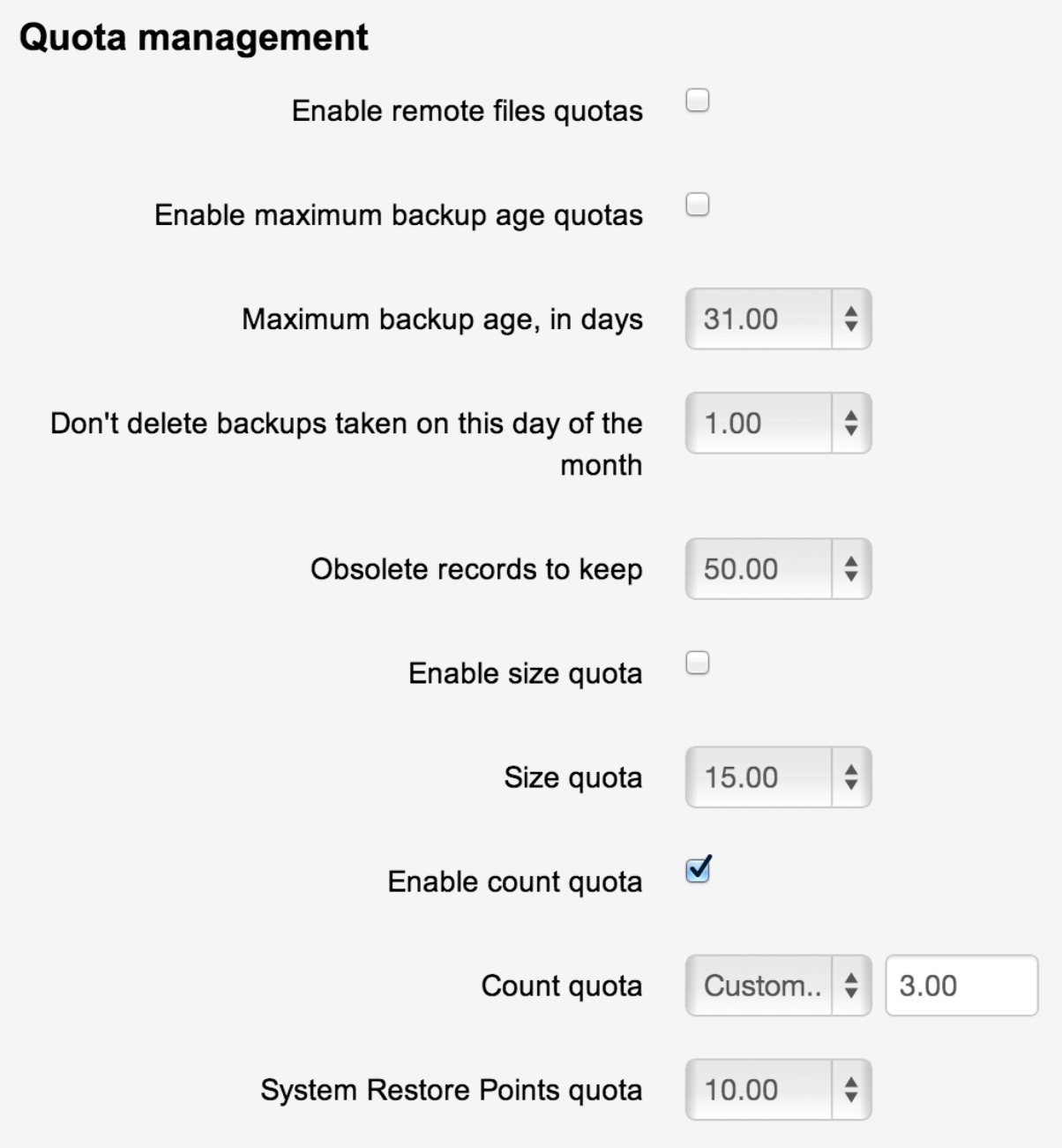
3.3.1.5. Quota management

Quotas let you automatically remove backup archives and / or backup records based on specific criteria. Quotas are always calculated against the **backup records**, not the backup archives on disk on or on remote storage. In other words, if you do not see a backup record in the Manage Backups page it is NOT taken into account when applying quotas.

Furthermore, quotas will take into account only the backup record, without checking if the file exists. If a backup is listed as OK or Remote in the Manage Backups page it participates in the quotas.

Finally, the quotas apply *per backup profile*. They will only take into account backup records in the same backup profile.

Quota management



Enable remote files quotas	When checked, the quota settings will also be applied to remotely stored files. This option only works with the cloud storage engines which support remote file deletion.
Enable maximum backup age quotas	When checked, Akeeba Backup will only apply quotas based on the date and time the backup was started. This allows you to easily do something like "keep daily backups for the last 15 days and always keep the backup taken on the first of each month".

Warning

Enabling this options DISABLES the size and count quotas.

Maximum back age, in days	Only applies when the Enable maximum backup age quotas option is enabled. Backups older than this number of days will be deleted. Newer backups will not be deleted.

Don't delete backups taken on this day of the month

Only applies when the Enable maximum backup age quotas option is enabled.

Even when a backup is older than the Maximum back age, in days setting, it won't be deleted if it was taken on this day of the month. For example, if you set this to 1, backups taken on the first day of each calendar month will not be deleted. Setting this option to 1, the backup age to 31 and enabling the maximum backup age quotas you end up keeping all backups taken the last month and keeping the backups taken on the first of each month.

Obsolete records to keep

When the locally stored files of a backup record are deleted (either manually or automatically after uploading it to a remote storage) the record is marked as Obsolete or Remote. Some users prefer to limit the number of the backup entries showing in the Manage Backups (formerly "Administer Backup Files") page. This option instructs Akeeba Backup to keep at most that many obsolete/remote records and automatically delete older obsolete/remote entries. This is different than the rest of the quotas because it doesn't remove files from your server, it removes the backup entry from Akeeba Backup's interface.

Warning

Backups marked as "Remote" are also considered obsolete records: the backup archive does not exist on your server, it only exists on the remote storage. Therefore this setting will also remove the backup records for the Remote backups. Since you are removing the backup records they WILL NOT participate in remote file quotas! Therefore the Obsolete records to keep setting MUST be higher than the total number of backups you will keep before the quotas kick in plus one.

For example, if you are taking 4 backups a day and you have enabled a maximum backup age quota of 30 days you need to set the Obsolete records to keep to at least 121 (4 backups / day x 30 days + 1 = 120 + 1 = 121). Otherwise the maximum backup age quotas will NOT work as expected.

Enable size quota

When checked, old backup archives will be erased when the total size of archives stored under this (and only this) profile exceed the Size quota setting.

Size quota

Defines the maximum aggregated size of backup archives *under the current profile* to keep. Only has an effect if the previous options is activated.

Enable count quota

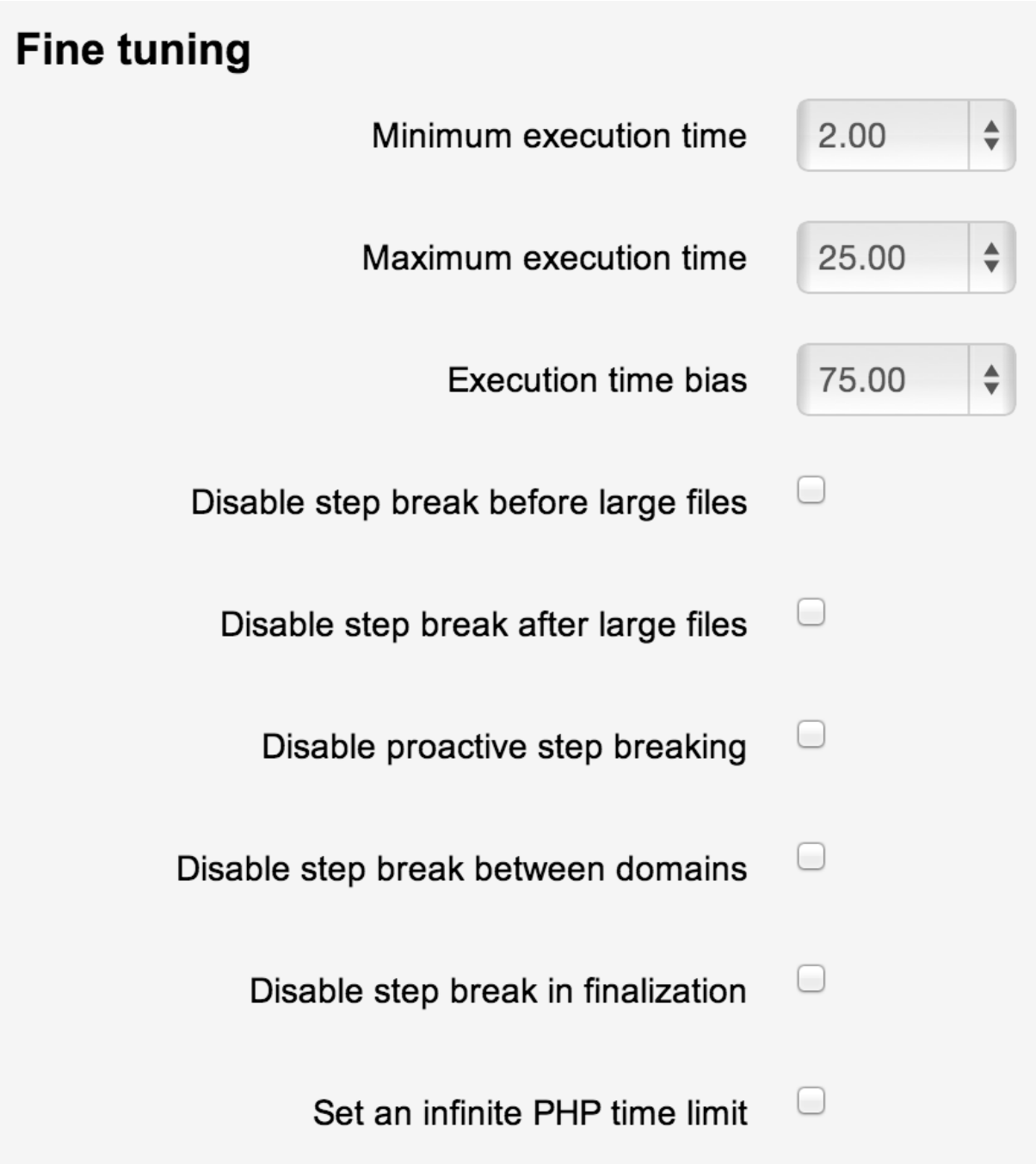
When checked, old backup archives will be erased when there are more backups stored under this (and only this) profile exceed the Count quota setting.

Count quota

Defines the maximum number of backups *under the current profile* to keep. Only has an effect if the previous options is activated.

3.3.1.6. Fine tuning

Fine tuning



Minimum execution time

Some servers deploy anti-hacker measures (such as mod_evasive or mod_security) which will deny connections to the server if the same URL is accessed multiple times in a limited amount of time. Akeeba Backup has to call its backup URL multiple times, so it runs the risk of being treated as a potential hacker and denied connection to your server, resulting to backup failure.

In order to work around this issue, Akeeba Backup can throttle the rate of server requests using this setting. A minimum execution time of 2 seconds means that calls to the backup URL will happen *at most* once every two seconds. You are suggested to keep the default value.

Maximum execution time

Akeeba Backup has to divide the backup process in individual small steps in order to avoid server timeouts. However, it has to know how small they have to be; that's why this setting

exists. Akeeba Backup will try to avoid consuming more time per step than this setting. You have to use a number lower than the `maximum_execution_time` setting in your host's php.ini file. In fact, we suggest using 50% of that value here: if your host allows up to 30 seconds in the php.ini, you have to enter no more than 15-17 seconds here. If unsure, 7 seconds is a very safe value under most configurations.

Execution time bias

When Akeeba Backup calculates the available time left for performing operations within the current backup step a number of external settings may skew this result and lead to timeout errors. This setting defines how conservative the backup engine will be when performing those calculations and is expressed as a percentage of the Maximum execution time parameter. The less this setting is, the more conservative Akeeba Backup gets. It is suggested not to use a value over 75%, unless you have a very fast server. If you experience timeouts, you may want to lower this setting to a value around 50%.

Resume backup after an AJAX error has occurred

When this option is unchecked Akeeba Backup will completely stop the backup when the server responds with an error or the communication with the server is cut short. When this option is enabled (default), Akeeba Backup will try to resume the backup by repeating the last backup step. This will not let you successfully resume all backups which result in an error: only backup attempts temporarily blocked by server CPU usage restrictions or network outage issues can be resumed. If the backup fails due to a timeout error, memory outage, incompatible server software etc the backup resumption will result in the same error until it leads to a permanent backup failure.

Important

This feature only applies to back-end backups. This feature will not be taken into account when you have enabled the Process each part immediately option in the configuration of the Data processing engine since it's impossible to retry backing up to a backup archive which may have already been transferred to remote storage and removed from the server.

Wait period before retrying the backup step

How many seconds to wait before resuming the backup. It is advisable to set this to 30 seconds or more (120 seconds is recommended in most cases) to give your server / network the necessary time to recover from the error condition which caused your backup to fail.

Maximum retries of a backup step after an AJAX error

How many consecutive times should we retry resuming the backup before finally giving up and throwing a permanent error (backup failure). 3 to 5 retries work best on most servers.

Disable step break before large files

When the application detects a large file (see the filesystem scanner engine configuration) it will try to break the execution of the current backup step and start backing up the large file in its own backup step. This is a conservative behaviour that increases the likelihood of being able to backup large files but makes the backup slower. If you check this box the backup will become faster, but it might fail backing up larger files.

Disable step break after large files

When the application finishes backing up a large file (see the filesystem scanner engine configuration) it will try to break the execution of the current backup step and continue the backup process in a step. This is a conservative behaviour that decreases the likelihood of the backup engine timing out after backing up a large file but makes the backup slower. If you check this box the backup will become faster, but it might fail after backing up larger files.

Disable proactive step breaking

The application tries to guess how much time it will take it to backup each file. If it believes that backing up the next file in its queue will take too long it will break the backup step and continue the backup in a new step. This decreases the likelihood of server timeouts, at the expense of making the backup a little slower, especially if you have lots of tiny files. If you check this box the backup will become faster, but it might fail in some cases.

Disable step break between domains	Do not check this box unless you are instructed by our support staff. The possibility of needing this option has been found to be less than 0.1%.
Disable step break in finalization	Do not check this box unless you are instructed by our support staff. The possibility of needing this option has been found to be less than 0.1%.
Set an infinite PHP time limit	If your server is using the CGI or FastCGI interface to PHP, checking this option will make it less likely that the backup dies due to a PHP timeout issue. We consider it generally safe checking this box as we have never observed or got reports of any side-effects.

3.3.2. Database dump engines

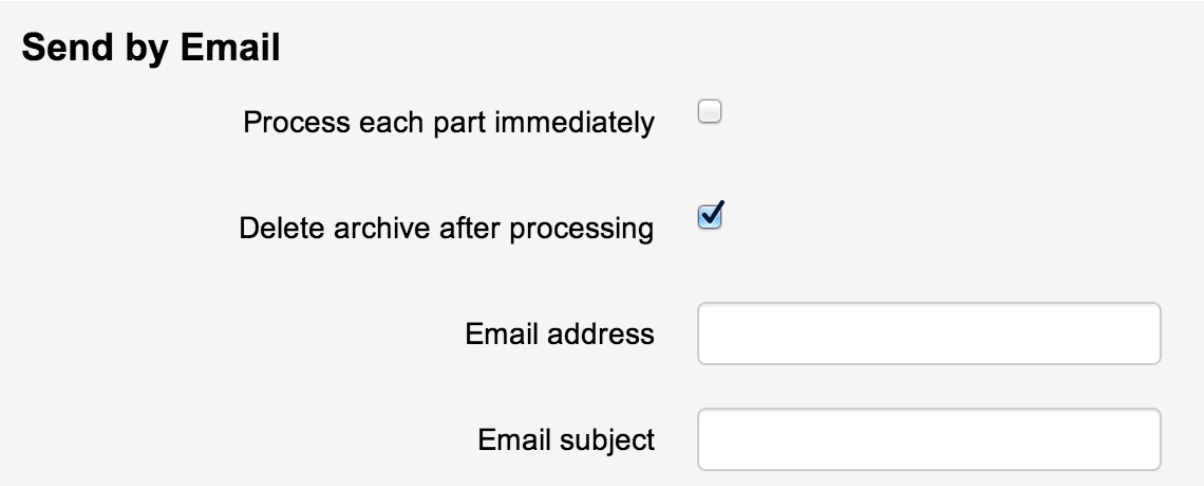
3.3.2.1. Native MySQL Backup Engine

This engine will take a backup of your MySQL database using nothing but PHP functions in order to accomplish that. This database dump engine supports all of the ground-breaking features available in MySQL 5, such as views, stored procedures and functions, triggers, merge tables, temporary/memory tables, even federated tables.

Important

Restoring views, triggers, stored procedures and functions requires adequate privileges for the database user during the restoration process. Most hosts do not assign this kind of privileges. If your restoration fails with a MySQL error when restoring such database entities you may have to ask your host to assign those privileges to your database user.

Native MySQL Backup Engine



MySQL Compatibility	This option controls the MySQL version compatibility when creating the database SQL dump file. In fact, it forces Akeeba Backup to request the appropriate CREATE TABLE commands from your database server. It is useful when migrating your site to another host with a different MySQL version. The available options are: • **Default**. This is the recommended option. The full feature set of your database server will be used when generating the CREATE command. Your target database server must run MySQL of a matching major version, i.e. MySQL 5 if the host you're backing up runs on MySQL 5. • **MySQL 4.1**. Akeeba Backup will request from your database server to provide definitions (CREATE commands) in a MySQL 4.1 friendly format.

Important

This option will take effect in MySQL 4.1 or greater database hosts. If you use it on older MySQL version the backup might fail!

Warning

Do not use this option if your site is already running on MySQL 4.x or if both your site and the target host run on MySQL 5.x. Otherwise, crucial information about the database's encoding might be lost in the process, causing broken text on sites using non-ASCII character sets.

Blank out username/ password

When enabled, Akeeba Backup will not include the username and password of database connections in the backup. Please note that this option only removes the database username and password from the installation/sql/databases.ini file which is included in the backup. It does not remove the database connection information from the configuration.php file of Joomla!. If you want to remove the database connection information for security reasons you should exclude configuration.php from your backup using the Files and Directories Exclusion filter feature of Akeeba Backup.

Generate extended INSERTs

When this is not checked, Akeeba Backup will create one INSERT statement for each data row of each table. When you have lots of rows with insignificant amount of data, such as banner and click tracking logs, the overhead of the INSERT statement is much higher than the actual data, causing a massively bloated database dump file. When this option is enabled, the dump engine will create a single INSERT statement for multiple rows of data, reducing the overhead and resulting into significantly smaller backup archives. Moreover, this will lead to much less SQL commands being run during restoration, which is of paramount importance on many restrictive shared hosting environments. It is suggested to turn this setting on, unless you are going to restore to a MySQL 4.1 host.

Max packet size for extended INSERTs

If the previous setting is enabled, this setting defines the maximum length of a single INSERT statement. Most MySQL servers have a configured limit of maximum statement length and will not accept an INSERT statement over 1Mb. It is suggested to leave the default conservative setting (128Kb) unless you know what you're doing. If you get restoration failures indicating that you exceeded the maximum query length, please lower this setting.

Dump PROCEDUREs, FUNCTIONs and TRIGGERs

By default, Akeeba Backup will only back up database tables and VIEWs. If your host supports this, you can also back up and restore advanced aspects of your MySQL database: stored procedures, stored functions and triggers. If your site makes use of any of those features you will have to tick the box. If the backup operation crashes or you the database tables filter page is blank you must turn this option off for Akeeba Backup to work properly.

Warning

Using this feature requires that your host allows you to execute privileged SQL commands against the MySQL database:

- **SHOW PROCEDURE STATUS**
- **SHOW FUNCTION STATUS**
- **SHOW TRIGGERS**

Most shared hosting providers do not allow you to execute these commands. Trying to do so will usually cause the script execution to abruptly halt, most often without indicating the source of error. If you are in doubt, **disable this option** and retry backup. This shouldn't be an issue with dedicated hosting, as long as you grant the **SUPER** privilege to the database user you use to connect to your site's database.

Size for split SQL dump files

Akeeba Backup is able to split your MySQL database dump to smaller files. This allows for an improved compression ratio and also helps avoid several problems with certain cheap hosts which put a restriction on the maximum size a file generated by PHP code can have.

Ideally, you should specify a setting which is about half as much as your Big file threshold setting in the archiver engine's configuration options pane. The reason to do that is that the archiver engines will not compress files with sizes over the value this threshold. Since it's impossible to have absolute control of the size of the database dump, using half the value of this setting allows for the expected size fluctuation.

If you want to disable this feature and create a single big SQL dump file instead, just set this option to 0 Mb.

Important

This setting has no effect on "Main site database only" backup profiles. This is because the nature of this backup type does not allow splitting the database archive dump. If you want something equivalent, please use the "All configured databases" backup type instead, as it creates an archive file which contains your (split) database dump and takes up MUCH less space on your web server.

Number of rows per batch

Dumping table data happens in "batches", i.e. a few rows at a time. This parameter defines how many rows will be fetched from the table at any given time. If you are backing up tables with large chunks of binary data (e.g. files stored in BLOB fields) or if you have very large chunks of text stored in the database, the default value - 1000 rows - may cause a PHP memory or MySQL buffer exhaustion. If you get memory outage errors during the table backup, it is advisable to lower this setting. This is especially true if your MySQL and PHP combination does not allow a cursor to be effectively created and all data has to be transferred in PHP's memory. A value of 20 is a very safe value, at the expense of making your backup process slower and run more queries against your database server. Most servers work fine with the default value of 1000 rows per batch.

No dependency tracking

When this option is enabled, Akeeba Backup's database dump engine will no longer try to figure out table and VIEW dependencies. This will speed up the database dump initialization step. This is recommended if and only if you have too many tables (over 200) in your database, you get timeout errors during the database dump initialization step and you do not use foreign keys, VIEWs, FUNCTIONs, PROCEDUREs, TRIGGERs or any tables using the MERGE database engine. If you do use any of those MySQL features in your tables there is a high probability that your SQL dump will be unable to be restored.

3.3.2.2. Reverse Engineering Database Dump Engine

Warning

Due to its nature we consider this method a beta feature.

This engine will take a backup of your database by reverse engineering its structure. This is the only possible method for non-MySQL databases (PostgreSQL, SQL Server, Windows Azure SQL). This database dump engine only supports a rudimentary feature set of your database server: tables and views only with their constraints and foreign key relations. It doesn't support advanced entities such as triggers, procedures and functions. The supported feature set should be adequate for backing up a Joomla! site.

Important

Reverse engineering the database structure usually requires adequate privileges for the database user during the backup process. The same goes for restoration of VIEWs during the restoration process. Most hosts do not assign this kind of privileges. If your backup or restoration fails with a database error when backing up or restoring your site you may have to ask your host to assign those privileges to your database user.

Blank out username/ password

When enabled, Akeeba Backup will not include the username and password of database connections in the backup. Please note that this option only removes the database username and password from the installation/sql/databases.ini file which is included in the backup. It does not remove the database connection information from the configuration.php file of Joomla!. If you want to remove the database connection information for security reasons you should exclude configuration.php from your backup using the Files and Directories Exclusion filter feature of Akeeba Backup.

Generate extended INSERTs

When this is not checked, Akeeba Backup will create one INSERT statement for each data row of each table. When you have lots of rows with insignificant amount of data, such as banner and click tracking logs, the overhead of the INSERT statement is much higher than the actual data, causing a massively bloated database dump file. When this option is enabled, the dump engine will create a single INSERT statement for multiple rows of data, reducing the overhead and resulting into significantly smaller backup archives. Moreover, this will lead to much less SQL commands being run during restoration, which is of paramount importance on many restrictive shared hosting environments. It is suggested to turn this setting on, unless you are going to restore to a MySQL 4.1 host.

Max packet size for extended INSERTs

If the previous setting is enabled, this setting defines the maximum length of a single INSERT statement. Most MySQL servers have a configured limit of maximum stement length and will not accept an INSERT statement over 1Mb. It is suggested to leave the default conservative setting (128Kb) unless you know what you're doing. If you get restoration failures indicating that you exceeded the maximum query length, please lower this setting.

Size for split SQL dump files

Akeeba Backup is able to split your MySQL database dump to smaller files. This allows for an improved compression ratio and also helps avoid several problems with certain cheap hosts which put a restriction on the maximum size a file generated by PHP code can have.

Ideally, you should specify a setting which is about half as much as your Big file threshold setting in the archiver engine's configuration options pane. The reason to do that is that the archiver engines will not compress files with sizes over the value this threshold. Since it's impossible to have absolute control of the size of the database dump, using half the value of this setting allows for the expected size fluctuation.

If you want to disable this feature and create a single big SQL dump file instead, just set this option to 0 Mb.

Important

This setting has no effect on "Main site database only" backup profiles. This is because the nature of this backup type does not allow splitting the database archive dump. If you want something equivalent, please use the "All configured databases" backup type instead, as it creates an archive file which contains your (split) database dump and takes up MUCH less space on your web server.

Number of rows per batch

Dumping table data happens in "batches", i.e. a few rows at a time. This parameter defines how many rows will be fetched from the table at any given time. If you are backing up tables with large chunks of binary data (e.g. files stored in BLOB fields) or if you have very large chunks of text stored in the database, the default value - 1000 rows - may cause a PHP memory or MySQL buffer exhaustion. If you get memory outage errors during the table backup, it is advisable to lower this setting. This is especially true if your MySQL and PHP combination does not allow a cursor to be effectively created and all data has to be transferred in PHP's memory. A value of 20 is a very safe value, at the expense of making your backup process slower and run more queries against your database server. Most servers work fine with the default value of 1000 rows per batch.

Dump PROCEDUREs,

By default, Akeeba Backup will only back up database tables and VIEWs. If your host supports this, you can also back up and restore advanced aspects of your database: stored procedures, stored functions and triggers. If your site makes use of any of those features you

FUNCTIONs and TRIGGERs — will have to tick the box. If the backup operation crashes or you the database tables filter page is blank you must turn this option off for Akeeba Backup to work properly.

> **Warning**
>
> THIS OPTION CURRENTLY HAS NO EFFECT! We are working on providing a solution in a future version.

No dependency tracking — When this option is enabled, Akeeba Backup's database dump engine will no longer try to figure out table and VIEW dependencies. This will speed up the database dump initialization step. This is recommended if and only if you have too many tables (over 200) in your database, you get timeout errors during the database dump initialization step and you do not use foreign keys or VIEWs. If you do use any of those database features there is a high probability that your SQL dump will be unable to be restored.

3.3.3. File and directories scanner engines

3.3.3.1. Smart scanner

This engine is the culmination of years of research in optimizing file system scanning for PHP scripts. The Smart Scanner will browse your file system tree for directories and files to include in the backup set, automatically breaking the step upon detecting a very large directory which could lead to timeout errors.

Large directory threshold — This option tells Akeeba Backup which directories to consider "large" so that it can break the backup step. When it is encountered with a directory having at least this number of files and subdirectories, it will break the step. The default value is quite conservative and suitable for most sites. If you have a very fast server, e.g. a dedicated server, VPS or MVS, you may increase this value. If you get timeout errors, try decreasing this setting.

Directory listing method — Akeeba Backup can use two different methods for asking your server to list the contents of a directory. The Regular method is very fast and works on the vast majority of servers. However, some servers refuse to list files with permissions lower than 0755 (it's absurd, I know!) and require the slightly slower, Alternate method. If your backup archive is missing files and you do not get "Unreadable file" or "Unreadable directory" warnings during backup, please switch this option to Alternate (failsafe) and retry backing up.

Large file threshold — Normally, Akeeba Backup tries to backup multiple files in a single step in order to provide a fast backup. However, doing that for larger files may result in a timeout or memory outage error. Files bigger than the large file threshold will be backed up in a backup step of their own. If you set it too low you will have a big performance impact in your backup (it will be slower). If you set it too high you might end up with a timeout or memory outage. The default setting (10Mb) is fine for most sites. If you are not sure what you're doing you're better off not touching it at all. If you find that your backup consistently fails while backing up a larger file (over 1Mb) you might want to lower this setting, e.g. to 2Mb. If you have a rather big PHP memory limit (128Mb or more) and you can afford the increased memory usage set it to a higher value, e.g. 25Mb (values over that tend to cause issues on all but the higher end dedicated servers).

3.3.3.2. Large site scanner

This engine is specifically optimised for large sites, containing folders with thousands of files. This is usually the case when you have a huge media collection such as news sites, professional blogers, companies with a large downloadable reference library or very active business sites storing hundreds of invoices daily on the server. In these cases the "Smart scanner" tends to consume unwieldy amounts of memory and CPU time to compile the list of files to backup, usually leading to timeout or memory outage issues. The "Large site scanner", on the other hand, works just fine by using a specially designed chunked processing technique. The drawback is that it makes the backup approximately 11% slower than the "Smart scanner".

Important

If your backup fails while trying to backup a directory with over 100 files you MUST use the "Large site scanner". It's very likely that this will solve your backup issues.

Warning

The developers of Akeeba Backup DO NOT recommend storing several thousands of files in a single directory. Due to reasons that have to do with the way most filesystems work at the Operating System level, the time required to produce a listing of files in a directory or access the files in a directory grows exponentially with the number of files. At about 5000 files the performance impact for accessing the directory, even on a moderately busy server, is big enough to both slow down your site noticeably (adversely impacting your search engine rankings) and make the backup slower and more prone to timeout errors. We strongly recommend using a sane number of subdirectories to logically organise your files and reduce the number of files per directory.

For the technically inclined (we really mean "serious geeks who aspire to do Linux server management as a living"), here is a nice discussion on the subject: http://stackoverflow.com/questions/466521/how-many-files-in-a-directory-is-too-many The problem is that readdir() which is also internally used by PHP only ever reads 32Kb of directory entries at a time. Further down the thread you can see that with 88,000 files in a directory the access becomes ten times slower. Per image. Add that up and you have a dead slow frontpage which is banished to the far end of search indexes. And if you wonder where the 5000 number popped up, it's from http://serverfault.com/questions/129953/maximum-number-of-files-in-one-ext3-directory-while-still-getting-acceptable-per and applies to older Linux distributions without Ext3/4 directory index support or using filesystems without directory index support (e.g. Ext2) which is, of course, the worst case scenario.

Directory scanning batch size

The Large site scanner creates a listing of folders by scanning a small number of them at a time. This setting determines how much this small number is. The larger this number the faster the backup is, but with the increased possibility of a backup failure on large sites. The smaller this number gets, the slower the backup becomes but the less likely it is to fail. We recommend a setting of 50 for most sites.

If your backup fails on deep nested folders containing many subdirectories we recommend setting this to a lower number, e.g. 20 or even 10. If you have a large PHP maximum memory limit and plenty of memory on your server to spare you may want to increase it to 100 or more. If you are unsure, don't touch this setting.

FIles scanning batch size

The Large site scanner will create a listing of files by scanning a small number of them at a time and then back them up. It will repeat this process until all files in the directory are backed up, then proceed to the next available directory. This setting determines how much this small number of files is. The larger this number the faster the backup is, but with the increased possibility of a backup failure on large sites. The smaller this number gets, the slower the backup becomes but the less likely it is to fail. We recommend a setting of 100 for most sites.

If your backup fails on folders containing many files we recommend setting this to a lower number, e.g. 50 or even 20. If you have a large PHP maximum memory limit and plenty of memory on your server to spare you may want to increase it to 500 or more. If you are unsure, don't touch this setting.

Large file threshold

Normally, Akeeba Backup tries to backup multiple files in a single step in order to provide a fast backup. However, doing that for larger files may result in a timeout or memory outage error. Files bigger than the large file threshold will be backed up in a backup step of their own. If you set it too low you will have a big performance impact in your backup (it will be slower). If you set it too high you might end up with a timeout or memory outage. The default setting (10Mb) is fine for most sites. If you are not sure what you're doing you're better off not touching it at all. If you find that your backup consistently fails while backing up a larger file (over 1Mb) you might want to lower this setting, e.g. to 2Mb. If you have a rather big

PHP memory limit (128Mb or more) and you can afford the increased memory usage set it to a higher value, e.g. 25Mb (values over that tend to cause issues on all but the higher end dedicated servers).

3.3.4. Archiver engines

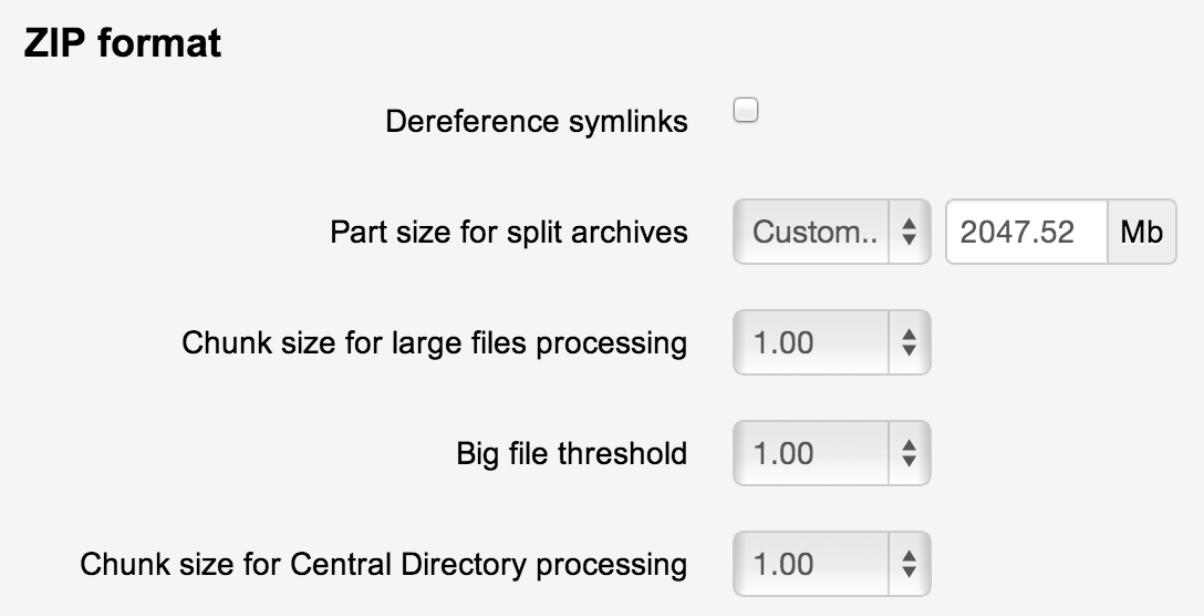
3.3.4.1. ZIP format

The ZIP format is the most well known archive format and is integrated in many operating systems and desktop environments, including Windows™, Mac OS X™, KDE and GNOME.

Warning

The ZIP format requires the calculation of CRC32 checksums for each file added in the archive. This is a resource intensive operation which will slow down your backup and may lead to timeouts when archiving big files on slow hosts. If this happens, your only choice is not to use the ZIP format; use JPA instead. Unfortunately, we can't do anything about it: it is a combined limitation of the ZIP specification, how PHP works and how your server is set up.

ZIP Format



Dereference symlinks

This setting is only valid on Linux and compatible *NIX hosts. Normally, when Akeeba Backup encounters symbolic links ("symlinks"), it follows them and treats them as regular files and directories, backing up their contents. Some site configurations may have symbolic links set up in such a way as to create an infinite loop, causing the backup to fail. When this option is set to `No`, Akeeba Backup will not follow symbolic links, but store their name and their target in the archive. Of course, if your symbolic links use absolute paths, restoring to a different server than the one you backed up from will result in broken symlinks.

Note

Even though Windows 7 supports symbolic links, it does so in a way that it's not possible for PHP to make use of this feature. As a result, this setting will only work on Linux, FreeBSD, Solaris and other compatible *NIX hosts.

Part size for split archives

Akeeba Backup supports the creation of Split Archives. In a nutshell, your backup archive is spanned among one or several files, so that each of these files ("part") is not bigger than the

value you specify here. This is a useful feature for hosts which impose a maximum file size quota. If you use a value of 0Mb, no archive splitting will take place and Akeeba Backup will produce a single backup archive (default).

Warning

If you want to post-process your archive files it is suggested that you use small, non-zero values here. The time it takes the post-processing engine to transfer an archive from your server to the remote server equals part size divided by available bandwidth. Since the available execution time is finite and the available bandwidth is constant, the only way to avoid a timeout is creating small parts.

Important

Split ZIP archives can not be opened with 7-zip, Linux unzip and other GUI clients. Only WinZIP and PKZIP understand them. If you want to extract them, you must use WinZIP, PKZIP, Akeeba Kickstart or Akeeba eXtract Wizard. This is not an Akeeba Backup "bug", it's a problem with most free archiver extraction tools.

Chunk size for large files processing	Each file is read in small increments, we call chunks, while being copied in the archive. Larger chunks will result in faster backup, at the price of taking longer to process each one of them and risking a timeout. Smaller chunks lead to slower but safer backups. On very slow hosts, this parameter should be set to a low value, for example 256Kb, or even lower - especially true if you constantly get timeout errors when backing up large files. On fast hosts you may want to increase this value in order to speed up your backup operation.
Big file threshold	Files over this size will be stored in the archive file uncompressed. Do note that in order for a file to be compressed, Akeeba Backup has to load it in its entirety to memory, compress it and then write it to disk. As a rule of thumb, you need to have free memory equal to 1.8 times the size of the file to compress, e.g. 18Mb for a 10Mb file. Joomla! with a lot of plug-ins might consume as much as 16Mb and Akeeba Backup's engine might consume another 5Mb, so plan this value carefully, or you will run into memory exhaustion errors. Compression is also resource intensive and will increase the time to produce a backup. If this value is too high, you might run into timeout errors.
Chunk size for Central Directory processing	At the end of the ZIP archive creation we have to attach a lookup table containing the names of all included files to the end of the archive file. This table is called the Central Directory. We have to do this in small chunks so as to avoid timeout or memory exhaustion errors. It is recommended that you leave the default value (1Mb) unless you know what you're doing.

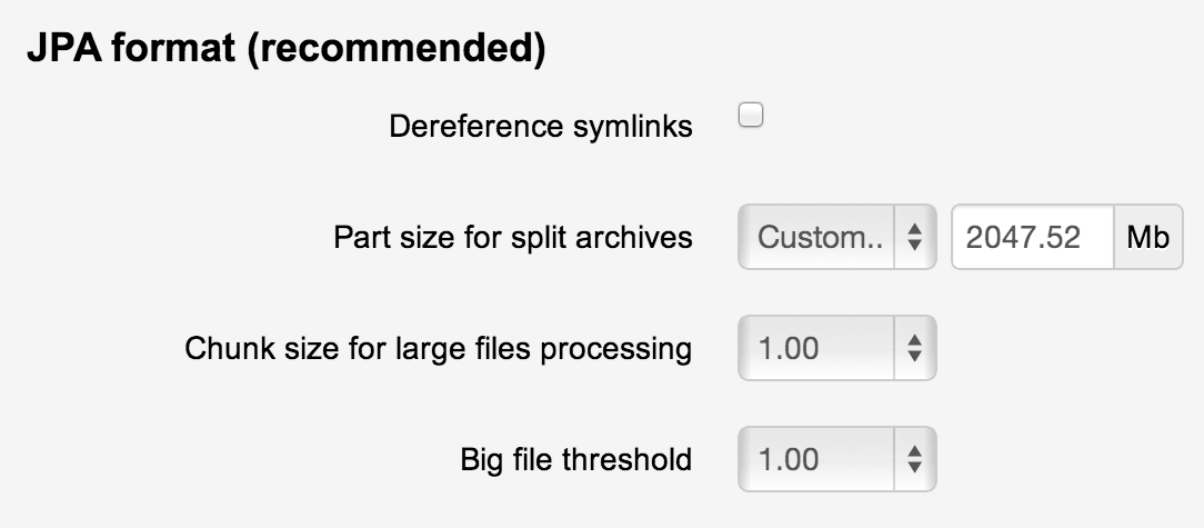
3.3.4.2. JPA format

The JPA format was conceived as an alternative to ZIP, designed to be extremely suitable for PHP scripts. The trick is that the JPA format doesn't store a checksum for each file - therefore it reduces the processing overhead during archiving - and it doesn't use a "lookup table" (central directory) as ZIP does. Both of these design decisions lead to extremely fast, low resource usage archiving processes.

Tip

It is recommended that you use the JPA format for all of your backups. You can extract JPA files either on your server using Kickstart, or on your desktop using Akeeba eXtract Wizard.

JPA Format



The settings for this engine are identical to those used in the ZIP engine.

3.3.4.3. Encrypted Archives (JPS format)

Note

This feature is only available in the Akeeba Backup Professional release.

The JPS is a further evolution of the JPA format, designed with the major goals of improving compression rations and enhancing the security of your data by encrypting the entire archive's contents with the industry standard AES-128 encryption format. The latter goal ensures that even in the unlikely event of your backup files ending up in the hands of hacker or another untrusted party, they would be useless. As per the strictest security standards, all information in the archive (including file names and file data) are encrypted. Without the password nobody can deduct any information about your site by examining a JPS archive. The contents of all files in the archive are compressed and encrypted in 64Kb blocks, allowing for better compression ratios over the JPA format.

Important

In order for JPS to work it requires that both the zlib and mcrypt or OpenSSL PHP extensions are installed and activated on your server. Please keep in mind that even if your site is using HTTPS this doesn't mean that you have the OpenSSL *PHP extension* installed. You usually have to ask your host to enable it for you. Moreover, the mcrypt or openssl library installed on the server must support AES-128 in CBC mode. If any of these conditions is not met, the backup process will halt with an error mentioning that encryption is not enabled on your server. In this case, please contact your host with the information in this paragraph so that they can perform the necessary server-side changes.

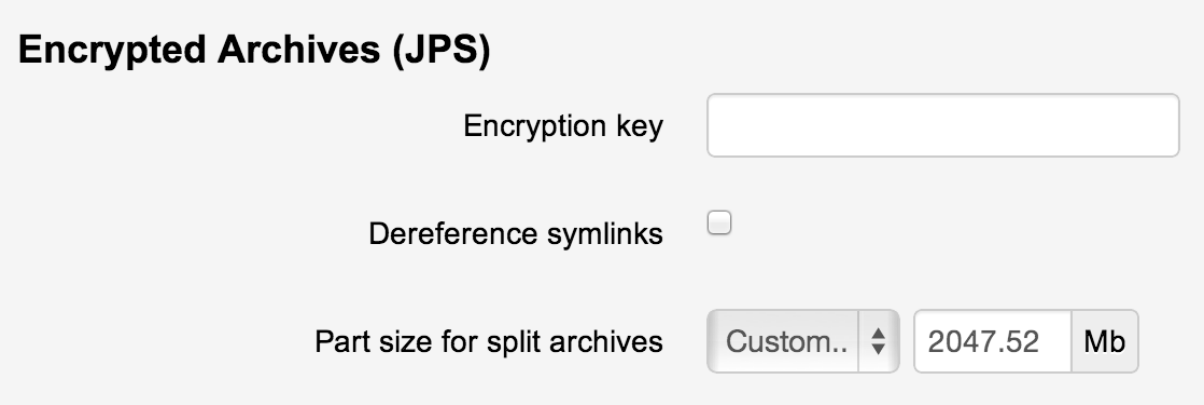
Important

JPS archives can only be extracted on hosts fulfilling the same per-requisites (zlib and mcrypt or OpenSSL PHP extensions installed and activated). They can also be extracted only by Kickstart 3.1.2 and Akeeba eXtract Wizard 3.0.4 or later. Earlier version can't read the JPS archives at all.

Important

We STRONGLY recommend using long (64 or more characters), completely random passwords which make use of lowercase and uppercase Latin letters, numbers and special characters (top row on US-format keyboards). Use a password manager to generate and store these passwords. Taking these precautions make password brute forcing with conventional technology highly impractical in the foreseeable future.

JPS Format



The settings for this engine are:

Encryption key
: This is the password to be used for encrypting the archive. For the sake of security, you are encouraged to enter a long passphrase which is hard to guess.

Warning

The key is case sensitive. This means that Abc, ABC and abc are three *completely different* keys!

Dereference symlinks
: This setting is only valid on Linux and compatible *NIX hosts. Normally, when Akeeba Backup encounters symbolic links ("symlinks"), it follows them and treats them as regular files and directories, backing up their contents. Some site configurations may have symbolic links set up in such a way as to create an infinite loop, causing the backup to fail. When this option is set to `No`, Akeeba Backup will not follow symbolic links, but store their name and their target in the archive. Of course, if your symbolic links use absolute paths, restoring to a different server than the one you backed up from will result in broken symlinks.

Note

Even though Windows 7 supports symbolic links, it does so in a way that it's not possible for PHP to make use of this feature. As a result, this setting will only work on Linux, FreeBSD, Solaris and other compatible *NIX hosts.

Part size for split archives
: Akeeba Backup supports the creation of Split Archives. In a nutshell, your backup archive is spanned among one or several files, so that each of these files ("part") is not bigger than the value you specify here. This is a useful feature for hosts which impose a maximum file size quota. If you use a value of 0Mb, no archive splitting will take place and Akeeba Backup will produce a single backup archive (default).

Warning

If you want to post-process your archive files it is suggested that you use small, non-zero values here. The time it takes the post-processing engine to transfer an archive from your server to the remote server equals part size divided by available bandwidth. Since the available execution time is finite and the available bandwidth is constant, the only way to avoid a timeout is creating small parts.

3.3.4.4. DirectFTP

Important

This feature is not meant for everyday users. It is designed for web professionals. If you don't understand the rest of this section, please do not use it. Akeeba Backup is equally useful as a site migration tool without using DirectFTP.

Note

This engine uses PHP's native FTP functions. This may not work if your host has disabled PHP's native FTP functions or if your remote FTP server is incompatible with them. In this case you may want to use the DirectFTP over cURL engine instead.

The DirectFTP engine allows power users to directly export a website from one server to another, without the need to download the backup file to their PC, upload it and extract it on the other server. In order to do so, instead of backing up to an archive, it directly writes the backed up files to the remote server using FTP, hence the name.

Do note that when using the DirectFTP engine, the post-processing engine will not run, as there is no archive produced.

In a nutshell, when this option is activated, Akeeba Backup operates as usual, backing up your database and files. Instead of putting the site files, installer files and database dump inside a backup archive, it transfers them to a remote server using FTP. You can then visit the installation URL on the remote server to complete the site transfer progress.

Warning

This is considered an advanced feature. Since there are many things which might go wrong in the process and due to the fact that the success of the operation depends on the server configuration of both the originating and target servers, you are advised not to use it unless you know what you're doing.

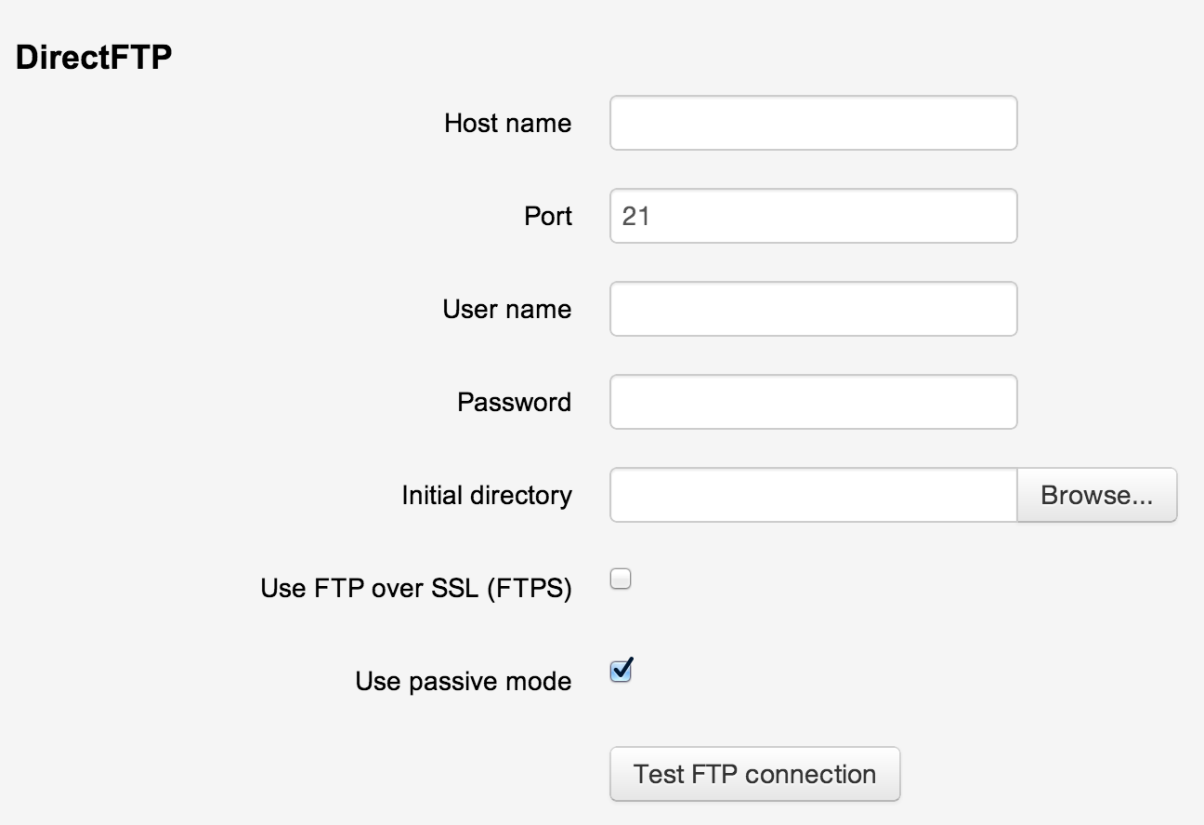
Moreover, bear in mind that the target server *must not* contain any files! If it does, it may not be possible to overwrite them, leading to an incomplete site transfer.

Finally note that due to the backup process being split in several steps (to avoid web server timeouts) a new FTP connection has to be created on each backup step, i.e. for every few files uploaded to your remote server. At best this makes the transfer extremely slow. At worst, your remote FTP server will decline further connections because it sees the same remote IP opening and closing FTP connections in rapid succession. We strongly recommend uploading entire backup archives using the post-processing engines instead of using this feature.

Your originating server must support PHP's FTP extensions and not have its FTP functions blocked. Your originating server must not block FTP communication to the remote (target) server. Some hosts apply a firewall policy which requires you to specify to which hosts your server can connect. In such a case you might need to allow communication to your remote host.

Normally, remote FTP connections consume a lot of time, therefore DirectFTP is very prone to time-outs. Theoretically, Akeeba Backup can automatically estimate the time required for transferring each file and avoid timing out. However, this is not always technically possible. In such a case you might want to lower the maximum execution time allowance and bias in the Configuration. Do note that large files have to be transferred in a single step, as most PHP and FTP configuration combinations disallow resuming uploads (chunked uploads). This means that a very large file, or a very large database dump may cause the process to fail with a timeout error.

DirectFTP



The available configuration options are:

- **Host name**. The hostname of your remote (target) server, e.g. `ftp.example.com`. You must NOT enter the ftp:// protocol prefix. If you do, Akeeba Backup will try to remove it automatically and issue a warning about it.
- **Port**. The TCP/IP port of your remote host's FTP server. It's usually 21.
- **User name**. The username you have to use to connect to the remote FTP server.
- **Password**. The password you have to use to connect to the remote FTP server.
- **Initial directory**. The absolute FTP directory to your remote site's location where your site will be cloned to. This is provided by your hosting company. Do not ask us to tell you what you should put in here because we can't possibly know. There is an easy way to find it, though. Connect to your target FTP server with FileZilla. Navigate to the web server's root (usually it's a subdirectory named httpdocs, htdocs, public_html, http_docs or www). Above the right-hand folder pane you will see a text box with a path. Copy this path and paste it to Akeeba Backup's setting.
- **Use FTP over SSL**. If your remote server supports secure FTP connections over SSL (they have to be explicit SSL; implicit SSL is not supported), you can enable this feature. In such a case you will most probably *have* to change the port. Please ask your hosting company to provide you with more information on whether they support this feature and what port you should use. You must note that this feature must also be supported by your originating server as well.
- **Use passive mode**. Normally you should enable it, as it is the most common and firewall-safe transfer mode supported by FTP servers. Sometimes, you remote server might require active FTP transfers. In such a case please disable this, but bear in mind that your originating server might not support active FTP transfers, which usually requires tweaking the firewall!

3.3.4.5. DirectFTP over cURL

Important

This feature is not meant for everyday users. It is designed for web professionals. If you don't understand the rest of this section, please do not use it. Akeeba Backup is equally useful as a site migration tool without using DirectFTP.

Note

This engine uses PHP's cURL functions. This may not work if your host has not installed or enabled the cURL functions. In this case you may want to use the DirectFTP engine instead.

The DirectFTP over cURL engine allows power users to directly export a website from one server to another, without the need to download the backup file to their PC, upload it and extract it on the other server. In order to do so, instead of backing up to an archive, it directly writes the backed up files to the remote server using FTP, hence the name.

Do note that when using the DirectFTP engine, the post-processing engine will not run, as there is no archive produced.

In a nutshell, when this option is activated, Akeeba Backup operates as usual, backing up your database and files. Instead of putting the site files, installer files and database dump inside a backup archive, it transfers them to a remote server using FTP. You can then visit the installation URL on the remote server to complete the site transfer progress.

Warning

This is considered an advanced feature. Since there are many things which might go wrong in the process and due to the fact that the success of the operation depends on the server configuration of both the originating and target servers, you are advised not to use it unless you know what you're doing.

Moreover, bear in mind that the target server *must not* contain any files! If it does, it may not be possible to overwrite them, leading to an incomplete site transfer.

Finally note that due to the nature of the cURL library a new FTP connection has to be created for each and every file uploaded to your remote server. At best this makes the transfer extremely slow. At worst, your remote FTP server will decline further connections because it sees the same remote IP opening and closing FTP connections in rapid succession. We strongly recommend uploading entire backup archives using the post-processing engines instead of using this feature.

Your originating server must support PHP's cURL extension. Your originating server must not block FTP communication to the remote (target) server. Some hosts apply a firewall policy which requires you to specify to which hosts your server can connect. In such a case you might need to allow communication to your remote host.

Normally, remote FTP connections consume a lot of time, therefore DirectFTP over cURL is very prone to time-outs. Theoretically, Akeeba Backup can automatically estimate the time required for transferring each file and avoid timing out. However, this is not always technically possible. In such a case you might want to lower the maximum execution time allowance and bias in the Configuration. Do note that large files have to be transferred in a single step, as most PHP and FTP configuration combinations disallow resuming uploads (chunked uploads). This means that a very large file, or a very large database dump may cause the process to fail with a timeout error.

The available configuration options are:

- **Host name**. The hostname of your remote (target) server, e.g. `ftp.example.com`. You must NOT enter the ftp:// protocol prefix. If you do, Akeeba Backup will try to remove it automatically and issue a warning about it.
- **Port**. The TCP/IP port of your remote host's FTP server. It's usually 21.

- **User name**. The username you have to use to connect to the remote FTP server.
- **Password**. The password you have to use to connect to the remote FTP server.
- **Initial directory**. The absolute FTP directory to your remote site's location where your site will be cloned to. This is provided by your hosting company. Do not ask us to tell you what you should put in here because we can't possibly know. There is an easy way to find it, though. Connect to your target FTP server with FileZilla. Navigate to the web server's root (usually it's a subdirectory named httpdocs, htdocs, public_html, http_docs or www). Above the right-hand folder pane you will see a text box with a path. Copy this path and paste it to Akeeba Backup's setting.
- **Use FTP over SSL**. If your remote server supports secure FTP connections over SSL (they have to be explicit SSL; implicit SSL is not supported), you can enable this feature. In such a case you will most probably *have* to change the port. Please ask your hosting company to provide you with more information on whether they support this feature and what port you should use. You must note that this feature must also be supported by your originating server as well.
- **Use passive mode**. Normally you should enable it, as it is the most common and firewall-safe transfer mode supported by FTP servers. Sometimes, you remote server might require active FTP transfers. In such a case please disable this, but bear in mind that your originating server might not support active FTP transfers, which usually requires tweaking the firewall!
- **Passive mode workaround**. Some badly configured / misbehaving servers report the wrong IP address when you enable the passive mode. Usually they report their internal network IP address (something like 127.0.0.1 or 192.168.1.123) instead of their public, Internet-accessible IP address. This erroneous information confuses FTP information, causing uploads to stall and eventually fail. Enabling this workaround option instructs cURL to ignore the IP address reported by the server and instead use the server's public IP address, as seen by your server. In most cases this works much better, therefore we recommend leaving this option turned on if you're not sure. You should only disable it in case of an exotic setup where the FTP server uses two different public IP addresses for the control and data channels.

3.3.4.6. DirectSFTP

Important

This feature is only available in the Akeeba Backup Professional edition.

Important

This feature is not meant for everyday users. It is designed for web professionals. If you don't understand the rest of this section, please do not use it. Akeeba Backup is equally useful as a site migration tool without using DirectSFTP.

Note

This engine uses the PHP extension called SSH2. The SSH2 extension is still marked as an alpha and is not enabled by default or even provided by many commercial hosts. In this case you may want to use the DirectSFTP over cURL engine instead which uses PHP's cURL extension, available on most hosts.

The DirectSFTP engine allows power users to directly export a website from one server to another, without the need to download the backup file to their PC, upload it and extract it on the other server. In order to do so, instead of backing up to an archive, it directly writes the backed up files to the remote server using SFTP (Secure File Transfer Protocol over SSH), hence the name.

Do note that when using the DirectSFTP engine, the post-processing engine will not run, as there is no archive produced.

In a nutshell, when this option is activated, Akeeba Backup operates as usual, backing up your database and files. Instead of putting the site files, installer files and database dump inside a backup archive, it transfers them to

a remote server using SFTP. You can then visit the installation URL on the remote server to complete the site transfer progress.

Warning

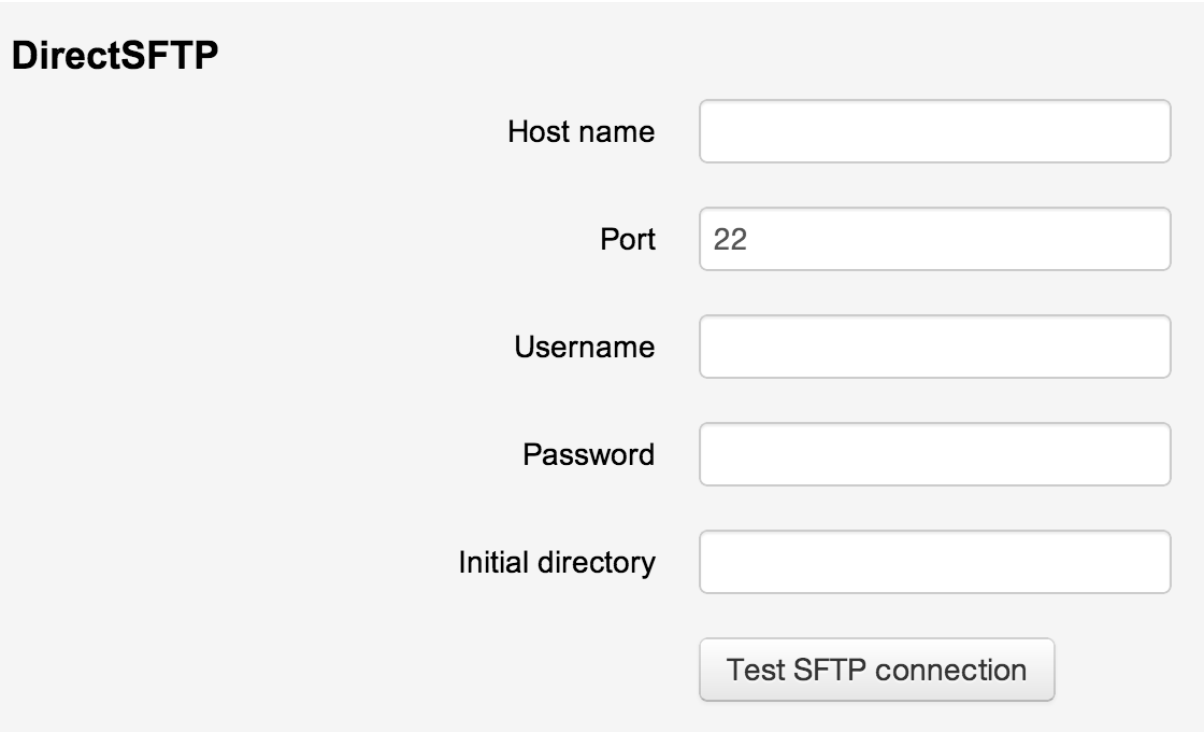
This is considered an advanced feature. Since there are many things which might go wrong in the process and due to the fact that the success of the operation depends on the server configuration of both the originating and target servers, you are advised not to use it unless you know what you're doing.

Moreover, bear in mind that the target server *must not* contain any files! If it does, it may not be possible to overwrite them, leading to an incomplete site transfer.

Your originating server (where you are backing up *from*) must a. support PHP's SSH2 extensions, b. allow outbound TCP/IP connections to your target host's SSH port and c. not have the SFTP functions of the SSH2 extension blocked. Please note that some hosts apply a firewall policy which requires you to specify to which hosts your server can connect. In such a case you might need to allow communication to your remote host.

Normally, remote SFTP connections consume a lot of time, therefore DirectSFTP is very prone to time-outs. Theoretically, Akeeba Backup can automatically estimate the time required for transferring each file and avoid timing out. However, this is not always technically possible. In such a case you might want to lower the maximum execution time allowance and bias in the Configuration. Do note that large files have to be transferred in a single step, as most PHP and SFTP configuration combinations disallow resuming uploads (chunked uploads). This means that a very large file, or a very large database dump may cause the process to fail with a timeout error.

DirectSFTP



The available configuration options are:

- **Host name**. The hostname of your remote (target) server, e.g. `sftp.example.com`. You must NOT enter the sftp:// or ssh:// protocol prefix. If you do, Akeeba Backup will try to remove it automatically and issue a warning about it.
- **Port**. The TCP/IP port of your remote host's FTP server. It's usually 22.
- **User name**. The username you have to use to connect to the remote SFTP server. This field must always be used, even when you're using certificate authentication.

- **Password**. The password you have to use to connect to the remote SFTP server. If you are using certificate authentication please enter the encryption key of your private key file. However, if you're using certificate authentication and your private key file is not encrypted please leave this field blank.
- **Private Key File (advanced)**. Only use this field when you want to perform certificate authentication. Enter the absolute path to your private SSH key file. If your private key file is encrypted with a password please provide the password in the Password field above.

  ## Important

  If the libssh2 library that the SSH2 extension of PHP is using is compiled against GnuTLS (instead of OpenSSL) you will NOT be able to use encrypted private key files. This has to do with bugs / missing features of GnuTLS, not our code. If you can't get certificate authentication to work please try providing an unencrypted private key file and leave the Password field blank.

- **Public Key File (advanced)**. Only use this field when you want to perform certificate authentication. Enter the absolute path to your public SSH key file.
- **Initial directory**. The absolute FTP directory to your remote site's location where your site will be cloned to. This is provided by your hosting company. Do not ask us to tell you what you should put in here because we can't possibly know. There is an easy way to find it, though. Connect to your target FTP server with FileZilla. Navigate to the web server's root (usually it's a subdirectory named httpdocs, htdocs, public_html, http_docs or www). Above the right-hand folder pane you will see a text box with a path. Copy this path and paste it to Akeeba Backup's setting.

3.3.4.7. DirectSFTP over cURL

Important

This feature is only available in the Akeeba Backup Professional edition.

Important

This feature is not meant for everyday users. It is designed for web professionals. If you don't understand the rest of this section, please do not use it. Akeeba Backup is equally useful as a site migration tool without using DirectSFTP.

Note

This engine uses the PHP cURL extension. If your host has disabled the cURL extension but has enabled the SSH2 PHP extension you may want to use the DirectSFTP engine instead which uses PHP's SSH2 extension.

The DirectSFTP engine allows power users to directly export a website from one server to another, without the need to download the backup file to their PC, upload it and extract it on the other server. In order to do so, instead of backing up to an archive, it directly writes the backed up files to the remote server using SFTP (Secure File Transfer Protocol over SSH), hence the name.

Do note that when using the DirectSFTP engine, the post-processing engine will not run, as there is no archive produced.

In a nutshell, when this option is activated, Akeeba Backup operates as usual, backing up your database and files. Instead of putting the site files, installer files and database dump inside a backup archive, it transfers them to a remote server using SFTP. You can then visit the installation URL on the remote server to complete the site transfer progress.

Warning

This is considered an advanced feature. Since there are many things which might go wrong in the process and due to the fact that the success of the operation depends on the server configuration of both the originating and target servers, you are advised not to use it unless you know what you're doing.

Moreover, bear in mind that the target server *must not* contain any files! If it does, it may not be possible to overwrite them, leading to an incomplete site transfer.

Your originating server (where you are backing up *from*) must a. have PHP's cURL extension installed and activated, b. have the cURL extension compiled with SFTP support and c. allow outbound TCP/IP connections to your target host's SSH port. Please note that some hosts provide the cURL extension without SFTP support. This feature will NOT work on these hosts. Moreover, some hosts apply a firewall policy which requires you to specify to which hosts your server can connect. In such a case you might need to allow communication to your remote host.

Normally, remote SFTP connections consume a lot of time, therefore DirectSFTP is very prone to time-outs. Theoretically, Akeeba Backup can automatically estimate the time required for transferring each file and avoid timing out. However, this is not always technically possible. In such a case you might want to lower the maximum execution time allowance and bias in the Configuration. Do note that large files have to be transferred in a single step, as most PHP and SFTP configuration combinations disallow resuming uploads (chunked uploads). This means that a very large file, or a very large database dump may cause the process to fail with a timeout error.

The available configuration options are:

- **Host name**. The hostname of your remote (target) server, e.g. `sftp.example.com`. You must NOT enter the sftp:// or ssh:// protocol prefix. If you do, Akeeba Backup will try to remove it automatically and issue a warning about it.

- **Port**. The TCP/IP port of your remote host's FTP server. It's usually 22.

- **User name**. The username you have to use to connect to the remote SFTP server. This field must always be used, even when you're using certificate authentication.

- **Password**. The password you have to use to connect to the remote SFTP server. If you are using certificate authentication please enter the encryption key of your private key file. However, if you're using certificate authentication and your private key file is not encrypted please leave this field blank.

- **Private Key File (advanced)**. Only use this field when you want to perform certificate authentication. Enter the absolute path to your private SSH key file. If your private key file is encrypted with a password please provide the password in the Password field above.

  ### Important

  If cURL is compiled against GnuTLS (instead of OpenSSL) you will NOT be able to use encrypted private key files. This has to do with bugs / missing features of GnuTLS, not our code. If you can't get certificate authentication to work please try providing an unencrypted private key file and leave the Password field blank.

- **Public Key File (advanced)**. Only use this field when you want to perform certificate authentication. Enter the absolute path to your public SSH key file. This is optional: some versions of the cURL library allow you to not provide a public key file, using the information of the private key file to derive this information. If in doubt, always provide both private and public key files to perform certificate authentication.

- **Initial directory**. The absolute FTP directory to your remote site's location where your site will be cloned to. This is provided by your hosting company. Do not ask us to tell you what you should put in here because we can't possibly know. There is an easy way to find it, though. Connect to your target FTP server with FileZilla. Navigate to the web server's root (usually it's a subdirectory named httpdocs, htdocs, public_html, http_docs or www). Above the right-hand folder pane you will see a text box with a path. Copy this path and paste it to Akeeba Backup's setting.

3.3.4.8. ZIP using ZIPArchive class

This engine produces ZIP archive using PHP's built-in ZIP archive class. It is only recommended for extremely small sites hosted on very slow hosts. If you have a larger site or quite big files you can expect that this engine will

time out, crash the backup or throw a memory outage error. Also note that this engine has absolutely no options and is bound to fail on hosts which impose limitations on the maximum size per file.

Frankly, this is the worst archiver engine. It was added because some users argued that it is faster (it is not) and this is why it is being used by competitive products. Well, try it out if you want. As soon as it causes backup errors do not ask for support, just switch to the classic ZIP engine or, even better, the JPA engine.

3.3.5. Data processing engines

3.3.5.1. No post-processing

This is the default setting and the only one one available to Akeeba Backup Core. It does no post-processing. It simply leaves the backup archives on your server.

3.3.5.2. Upload to CloudMe

Note

This feature is available only to Akeeba Backup Professional 3.10.1 and later.

Using this engine, you can upload your backup archives to the European cloud storage service CloudMe.

The required settings for this engine are:

Process each part immediately	If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.
Delete archive after processing	If enabled, the archive files will be removed from your server after they are uploaded to CloudMe.
Username	Your CloudMe username
Password	Your CloudMe password
Directory	The directory inside your CloudMe Blue Folder™ where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `directory/subdirectory/subsubdirectory`.

Tip

You can use Akeeba Backup's "variables" in the directory name in order to create it dynamically. These are the same variables as what you can use in the archive name, i.e. [DATE], [TIME], [HOST], [RANDOM].

3.3.5.3. Upload to Microsoft Windows Azure BLOB Storage service

Note

This feature is available only to Akeeba Backup Professional.

Using this engine, you can upload your backup archives to the Microsoft Windows Azure BLOB Storage [http://www.microsoft.com/windowsazure/windowsazure/] cloud storage service. This new cloud storage service from

Microsoft is reasonably priced (the cost is very close to CloudFiles) and quite fast, with lots of local endpoints around the globe.

Warning

Azure, unlike other cloud storage providers, doesn't support storing files over 64Mb without resorting to proprietary hacks. As a result you MUST use a part size for archive splitting lower than 64Mb at all times. Failure to do so might cause your backup uploads to fail.

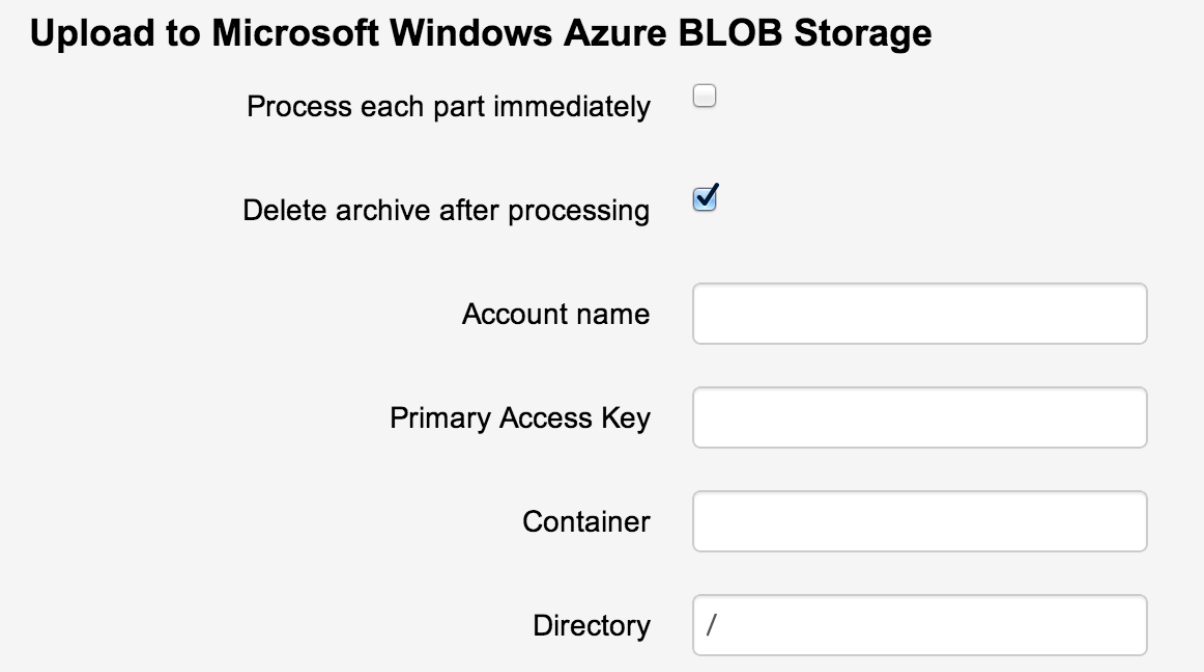
Before you begin, you should know the limitations. As most cloud storage providers, Azure does not allow appending to files, so the archive has to be transferred in a single step. PHP has a time limit restriction we can't overlook. The time required to upload a file to Azure equals the size of the file divided by the available bandwidth. We want to time to upload a file to be less than PHP's time limit restriction so as to avoid timing out. Since the available bandwidth is finite and constant, the only thing we can reduce in order to avoid timeouts is the file size. To this end, you have to produce split archives, by setting the part size for archive splitting in ZIP's or JPA's engine configuration pane. The suggested values are between 10Mb and 20Mb. Most servers have a bandwidth cap of 20Mbits, which equals to roughly 2Mb/sec (1 byte is 8 bits, plus there's some traffic overhead, lost packets, etc). With a time limit of 10 seconds, we can upload at most 2 Mb/sec * 10 sec = 20Mb without timing out. If you get timeouts during post-processing lower the part size.

Tip

If you use the native CRON mode (akeeba-backup.php), there is usually no time limit - or there is a very high time limit in the area of 3 minutes or so. Ask your host about it. Setting up a profile for use only with the native CRON mode allows you to increase the part size and reduce the number of parts a complete backup consists of.

Akeeba Backup is using the very stable official PHP bindings for Microsoft Windows Azure access, which is unlikely to stop working for the foreseeable future. As a result, we consider it a good candidate for backup archives storage.

Upload to Microsoft Windows Azure BLOB Storage



The required settings for this engine are:

Process each part immediately	If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after

processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.

Delete archive after processing	If enabled, the archive files will be removed from your server after they are uploaded to Azure.
Account name	The account name for your Microsoft Azure subscription. If your endpoint looks like `foobar.blobl.core.windows.net` then your account name is foobar.
Primary Access Key	You can find this Key in account page. It is lengthy and always ends in double equals marks.
Container	The name of the Azure container where you want to store your archives in.
Directory	The directory inside your Azure container where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `/directory/subdirectory/subsubdirectory`. Leave blank to store the files on the container's root.

3.3.5.4. Upload to RackSpace CloudFiles

Note

This feature is available only to Akeeba Backup Professional.

Using this engine, you can upload your backup archives to the RackSpace CloudFiles [www.rackspacecloud.com/cloud_hosting_products/files] cloud storage service. This service has been around for a long time, under the Mosso brand, and is considered one of the most dependable ones. Its cheap prices make it ideal for applications where storing large quantities of backup archives is more likely than downloading them.

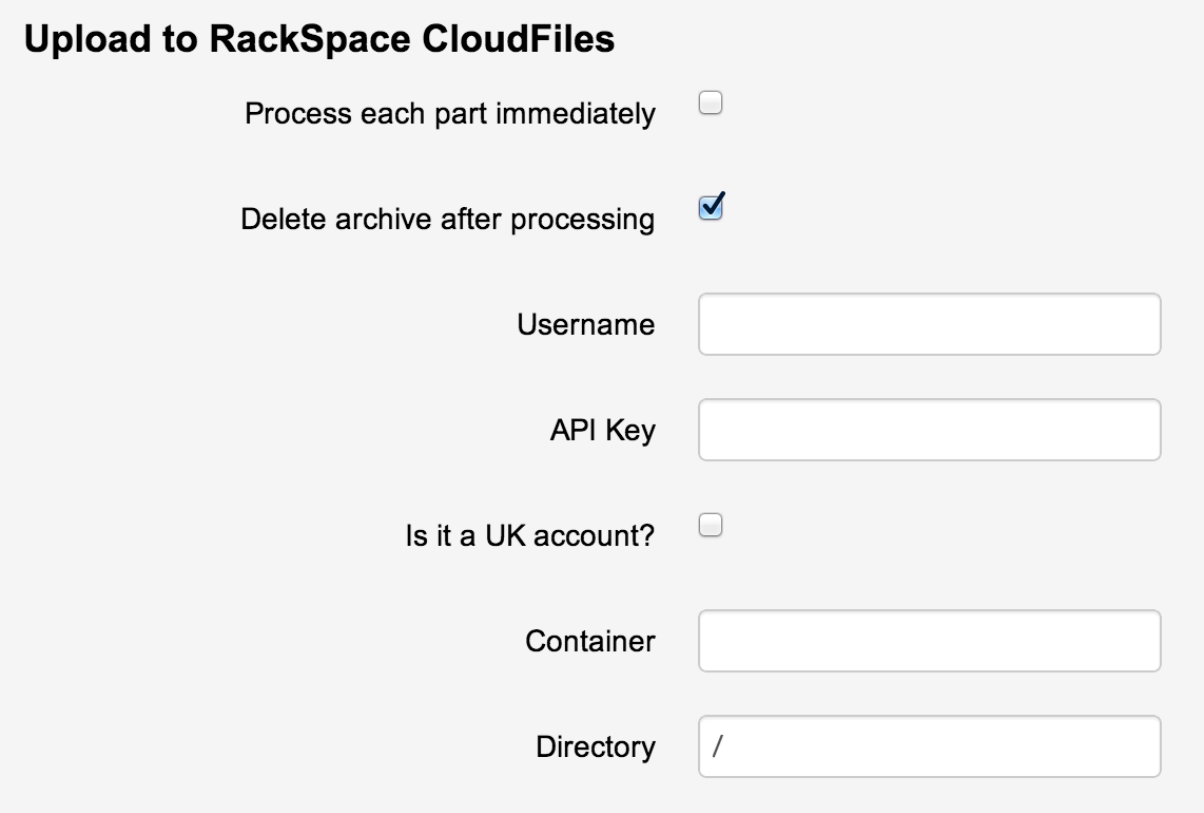
Before you begin, you should know the limitations. As most cloud storage providers, CloudFiles does not allow appending to files, so the archive has to be transferred in a single step. PHP has a time limit restriction we can't overlook. The time required to upload a file to CloudFiles equals the size of the file divided by the available bandwidth. We want to time to upload a file to be less than PHP's time limit restriction so as to avoid timing out. Since the available bandwidth is finite and constant, the only thing we can reduce in order to avoid timeouts is the file size. To this end, you have to produce split archives, by setting the part size for archive splitting in ZIP's or JPA's engine configuration pane. The suggested values are between 10Mb and 20Mb. Most servers have a bandwidth cap of 20Mbits, which equals to roughly 2Mb/sec (1 byte is 8 bits, plus there's some traffic overhead, lost packets, etc). With a time limit of 10 seconds, we can upload at most 2 Mb/sec * 10 sec = 20Mb without timing out. If you get timeouts during post-processing lower the part size.

Tip

If you use the native CRON mode (akeeba-backup.php), there is usually no time limit - or there is a very high time limit in the area of 3 minutes or so. Ask your host about it. Setting up a profile for use only with the native CRON mode allows you to increase the part size and reduce the number of parts a complete backup consists of.

Akeeba Backup is using an implementation of the version 2 API of CloudFiles access which is unlikely to stop working for the foreseeable future. As a result, we consider it a good candidate for cheap backup archives storage.

Upload to RackSpace CloudFiles



The required settings for this engine are:

Process each part immediately	If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.
Delete archive after processing	If enabled, the archive files will be removed from your server after they are uploaded to CloudFiles.
Username	The username assigned to you by the RackSpace CloudFiles service
API Key	The API Key found in your CloudFiles account
Container	The name of the CloudFiles container where you want to store your archives in.
Directory	The directory inside your CloudFiles container where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `/directory/subdirectory/subsubdirectory`. Leave blank to store the files on the container's root.

3.3.5.5. Upload to DreamObjects

Note

This feature is available only to Akeeba Backup Professional.

Using this engine, you can upload your backup archives to the DreamObjects cloud storage service by DreamHost.

The required settings for this engine are:

Process each part immediately
: If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.

Delete archive after processing
: If enabled, the archive files will be removed from your server after they are uploaded to DreamObjects.

Access Key
: Your DreamObjects Access Key

Secret Key
: Your DreamObjects Secret Key

Use SSL
: If enabled, an encrypted connection will be used to upload your archives to DreamObjects. In this case the upload will take slightly longer, as encryption - what SSL does - is more resource intensive than uploading unencrypted files. You may have to lower your part size.

Warning

Do not enable this option if your bucket name contains dots.

Bucket
: The name of your DreamObjects bucket where your files will be stored in. The bucket must be already created; Akeeba Backup can not create buckets.

Warning

DO NOT CREATE BUCKETS WITH NAMES CONTAINING UPPERCASE LETTERS OR DOTS. If you use a bucket with uppercase letters in its name it is very possible that Akeeba Backup will not be able to upload anything to it for reasons that have to do with the S3 API implemented by DreamObjects. It is not something we can "fix" in Akeeba Backup. Moreover, if you use a dot in your bucket name you will not be able to enable the "Use SSL" option since DreamObject's SSL certificate will be invalid for this bucket, making it impossible to upload backup archives. If this is the case with your site, please don't ask for support; simply create a new bucket whose name only consists of lowercase unaccented latin characters (a-z), numbers (0-9) and dashes.

Directory
: The directory inside your DreamObjects bucket where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `directory/subdirectory/subsubdirectory`.

Tip

You can use Akeeba Backup's "variables" in the directory name in order to create it dynamically. These are the same variables as what you can use in the archive name, i.e. [DATE], [TIME], [HOST], [RANDOM].

Regarding the naming of buckets and directories, you have to be aware of the S3 API rules used by DreamObjects:

- Folder names can not contain backward slashes (\). They are invalid characters.
- Bucket names can only contain lowercase letters, numbers, periods (.) and dashes (-). Accented characters, international characters, underscores and other punctuation marks are illegal characters.

Important

Even if you created a bucket using uppercase letters, **you must type its name with lowercase letters**. The S3 API implemented by DreamObjects automatically converts the bucket name to all-lowercase. Also note that, as stated above, you may NOT be able to use at all under some circumstances. Generally, your should avoid using uppercase letters.

- Bucket names must start with a number or a letter.
- Bucket names must be 3 to 63 characters long.
- Bucket names can't be in an IP format, e.g. 192.168.1.2
- Bucket names can't end with a dash.
- Bucket names can't have an adjacent dot and dash. For example, both `my.-bucket` and `my-.bucket` are invalid. It is preferable to NOT use a dot as it will cause issues.

If any - or all - of those rules are broken, you'll end up with error messages that Akeeba Backup couldn't connect to DreamObjects, that the calculated signature is wrong or that the bucket does not exist. This is normal and expected behaviour, as the S3 API of DreamObjects drops the connection when it encounters invalid bucket or directory names.

3.3.5.6. Upload to Dropbox (v2 API)

Important

This is the new method to connect to Dropbox. The v1 API may be removed by Dropbox at any time. We recommend that all users migrate to this method which uses the newer v2 API.

Using this engine, you can upload your backup archives to the low-cost Dropbox cloud storage service (http://www.dropbox.com). This is an ideal option for small websites with a low budget, as this service offers 2Gb of storage space for free, all the while retaining all the pros of storing your files on the cloud. Even if your host's data center is annihilated by a natural disaster and your local PC and storage media are wiped out by an unlikely event, you will still have a copy of your site readily accessible and easy to restore.

The required settings for this engine are:

Process each part immediately	If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.
Delete archive after processing	If enabled, the archive files will be removed from your server after they are uploaded to Dropbox.
Authorisation	Before you can use the application with Dropbox you have to "link" your Dropbox account with your Akeeba Solo / Akeeba Backup installation. This allows the application to access your Dropbox account without you storing the username (email) and password to the application. The authentication is a simple process. First click on the Authentication - Step 1 button. A popup window opens, allowing you to log in to your Dropbox account. Once you log in successfully, click the blue button to transfer the access token back to your Akeeba Solo / Akeeba Backup installation. Unlike the v1 API, you can perform the same procedure on every single site you want to link to Dropbox.

Directory — The directory inside your Dropbox account where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `/directory/subdirectory/subsubdirectory`.

Enabled chunked upload — The application will always try to upload your backup archives / backup archive parts in small chunks and then ask Dropbox to assemble them back into one file. This allows you to transfer larger archives more reliably and works around the 150Mb limitation of Dropbox' API.

When you enable this option every step of the chunked upload process will take place in a separate page load, reducing the risk of timeouts if you are transferring large archive part files (over 10Mb). When you disable this option the entire upload process has to take place in a single page load.

> **Warning**
>
> When you select Process each part immediately this option has no effect! In this case the entire upload operation for each part will be attempted *in a single page load*. For this reason we recommend that you use a Part Size for Split Archives of 5Mb or less to avoid timeouts.

Chunk size — This option determines the size of the chunk which will be used by the chunked upload option above. You are recommended to use a relatively small value around 5 to 20 Mb to prevent backup timeouts. The exact maximum value you can use depends on the speed of your server and its connection speed to the Dropbox server. Try starting high and lower it if the backup fails during transfer to Dropbox.

Token — This is the connection token to Dropbox. Normally, it is automatically fetched from Dropbox when you click on the Authentication - Step 1 button above. If for any reason this method does not work for you you can copy the Token from the popup window or another Akeeba Backup / Akeeba Solo installation you have already connected to Dropbox.

3.3.5.7. Send by email

> **Note**
>
> This feature is available only to Akeeba Backup Professional 3.4.b1 and later.

Send by email

This handy feature is available only in Akeeba Backup Professional. It will send you the backup archive parts as file attachments to your email address. That's right! No need to worry about downloading your backup archives, they will be emailed to you. That said, beware of the restrictions:

> **Warning**
>
> You **MUST** set the Part size for split archives setting of the Archiver engine to a value between 1-10 Megabytes. If you choose a big value (or leave the default value of 0, which means that no split archives will be generated) you run the risks of the process timing out, a memory outage error to occur or, finally, your email servers not being able to cope with the attachment size, dropping the email.

The available configuration settings for this engine, accessed by pressing the Configure... button next to it, are:

Process each part immediately	If you enable this, each backup part will be emailed to you as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the email fails, the backup fails. If you don't enable this option, the email process will take place after the backup is complete and finalized. This ensures that if the email process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.
Delete archive after processing	If enabled, the archive files will be removed from your server after they are emailed to you. Very useful to conserve disk space and practice the good security measure of not leaving your backups on your server.
Email address	The email address where you want your backups sent to. When used with GMail or other webmail services it can provide a cheap alternative to proper cloud storage.
Email subject	A subject for the email you'll receive. You can leave it blank if you want to use the default. However, we suggest using something descriptive, i.e. your site's name and the description of the backup profile.

3.3.5.8. Upload to OneDrive

Note

This feature is available only to Akeeba Backup Professional.

Using this engine, you can upload your backup archives to the low-cost Microsoft Live OneDrive cloud storage service (https://onedrive.live.com). This is an ideal option for small websites with a low budget, as this service offers 15Gb of storage space for free, all the while retaining all the pros of storing your files on the cloud. Even if your host's data center is annihilated by a natural disaster and your local PC and storage media are wiped out by an unlikely event, you will still have a copy of your site readily accessible and easy to restore. Do note that if you are a subscriber to Office 365 you get up to 1Tb of storage in OneDrive.

Warning

This feature does NOT support the unrelated, but confusingly similarly named, OneDrive for Business product by Microsoft which you typically get access to as part of an organization-level Microsoft Office 365 *for Business* subscription. Please note that the regular (not "for Business") Microsoft Office 365 subscription gives you access to the regular OneDrive product which is compatible with our software as explained above.

Important security and privacy information

OneDrive uses the OAuth 2 authentication method. This requires a fixed endpoint (URL) for each application which uses it, such as Akeeba Backup. Since Akeeba Backup is installed on your site, therefore has a different endpoint URL for each installation, you could not normally use OneDrive's API to upload files. We have solved it by creating a small intermediary script which lives on our own server and acts as an intermediary between your site and OneDrive. When you are linking Akeeba Backup to OneDrive you are going through the script on our site. Moreover, whenever the request token (a time-limited key given by OneDrive to your Akeeba Backup installation to access the service) expires your Akeeba Backup installation has to exchange it with a new token. This process also takes place through the script on our site. Please note that even though you are going through our site we DO NOT store this information and we DO NOT have access to your OneDrive account.

WE DO NOT STORE THE ACCESS CREDENTIALS TO YOUR ONEDRIVE ACCOUNT. WE DO NOT HAVE ACCESS TO YOUR ONEDRIVE ACCOUNT. SINCE CONNECTIONS TO OUR SITE ARE PROTECTED BY STRONG ENRYPTION (HTTPS) NOBODY ELSE CAN SEE THE INFORMATION EXCHANGED BETWEEN YOUR SITE AND OUR SITE AND BETWEEN OUR SITE AND ONEDRIVE. HOWEVER, AT THE FINAL STEP OF THE AUTHENTICATION PROCESS, YOUR BROWSER IS SENDING THE ACCESS TOKENS TO YOUR SITE. SOMEONE CAN STEAL THEM IN TRANSIT IF AND ONLY IF YOU ARE NOT USING HTTPS ON YOUR SITE'S ADMINISTRATOR.

For this reason we DO NOT accept any responsibility whatsoever for any use, abuse or misuse of your connection information to OneDrive. If you do not accept this condition you are FORBIDDEN from using the intermediary script on our site which, simply put, means that you cannot use the OneDrive integration.

Moreover, the above means that there are additional requirements for using OneDrive integration on your Akeeba Backup installation:

- You need the PHP cURL extension to be loaded and enabled on your server. Most servers do that by default. If your server doesn't have it enabled the upload will fail and warn you that cURL is not enabled.
- Your server's firewall must allow outbound HTTPS connections to www.akeebabackup.com over port 443 (standard HTTPS port) to get new tokens every time the current access token expires.
- Your server's firewall must allow outbound HTTPS connections to OneDrive's domains over port 443 to allow the integration to work. These domain names are, unfortunately, not predefined. Most likely your server administrator will have to allow outbound HTTPS connections to any domain name to allow this integration to work. This is a restriction of how the OneDrive service is designed, not something we can modify (obviously, we're not Microsoft).

Settings

The required settings for this engine are:

Process each part immediately
: If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.

Delete archive after processing
: If enabled, the archive files will be removed from your server after they are uploaded to OneDrive.

Authorisation – Step 1
: Before you can use Akeeba Backup with OneDrive you have to "link" your OneDrive account with your Akeeba Backup installation. This allows Akeeba Backup to access your OneDrive account without you storing the username (email) and password to. The authentication is a simple process. First click on the Authentication - Step 1 button. A popup window opens, allowing you to log in to your OneDrive account. Once you log in successfully, you are shown a page with the access and refresh tokens (the "keys" returned by OneDrive to be used for connecting to the service) and the URL to your site. Double check that the URL to your site is correct and click on the big blue "Finalize authentication" button. The popup window closes automatically.

  Alternatively, instead of clicking that big blue button you can copy the Access Token and Refresh Token from the popup window to Akeeba Backup's configuration page at the same-named fields. Afterwards you can close the popup.

  #### Important

  As described above, this process routes you through our own site (akeebabackup.com) due to OneDrive's API restrictions. We do NOT store your login information or tokens and we do NOT have access to your OneDrive account. If, however, you do not agree being routed through our site you are FORBIDDEN from using this intermediary service on our site and you cannot use the OneDrive integration feature. We repeat for a third time that this is a restriction imposed by the OneDrive API, not us. We CANNOT work around this restriction, so we created a very secure solution which works within the restrictions imposed by the OneDrive API.

Directory — The directory inside your OneDrive account where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `/directory/subdirectory/subsubdirectory`.

Enabled chunked upload — When enabled Akeeba Backup will try to upload your backup archives / backup archive parts in small chunks and then ask OneDrive to assemble them back into one file. If your backup archive parts are over 10Mb you are strongly encouraged to check this option.

Chunk size — This option determines the size of the chunk which will be used by the chunked upload option above. We recommend a relatively small value around 4 to 20 Mb to prevent backup timeouts. The exact maximum value you can use depends on the speed of your server and its connection speed to OneDrive's server. Try starting high and lower it if the backup fails during transfer to OneDrive. You cannot set a chunk size lower than 1Mb or higher than 60Mb because of OneDrive's API restrictions. We recommend using 4, 10 or 20Mb (tested and found to be properly working).

Access Token — This is the connection token to OneDrive. Normally, it is automatically sent to your site when clicking the blue button from the Authentication Step 1 popup described above. If you do not wish to click that button copy the (very, VERY long!) Access Token from that popup window into this box.

Warning

Unlike other engines, such as Dropbox, you CANNOT share OneDrive tokens between multiple site. Each site MUST go through the authentication process described above and use a different set of Access and Refresh tokens!

Refresh Token — This is the refresh token to OneDrive, used to get a fresh Access Token when the previous one expires. Normally, it is automatically sent to your site when clicking the blue button from the Authentication Step 1 popup described above. If you do not wish to click that button copy the (very, VERY long!) Refresh Token from that popup window into this box.

Warning

Unlike other engines, such as Dropbox, you CANNOT share OneDrive tokens between multiple site. Each site MUST go through the authentication process described above and use a different set of Access and Refresh tokens!

3.3.5.9. Upload to Remote FTP server

Note

This feature is available only to Akeeba Backup Professional.

Note

This engine uses PHP's native FTP functions. This may not work if your host has disabled PHP's native FTP functions or if your remote FTP server is incompatible with them. In this case you may want to use the Upload to Remote FTP server over cURL engine instead.

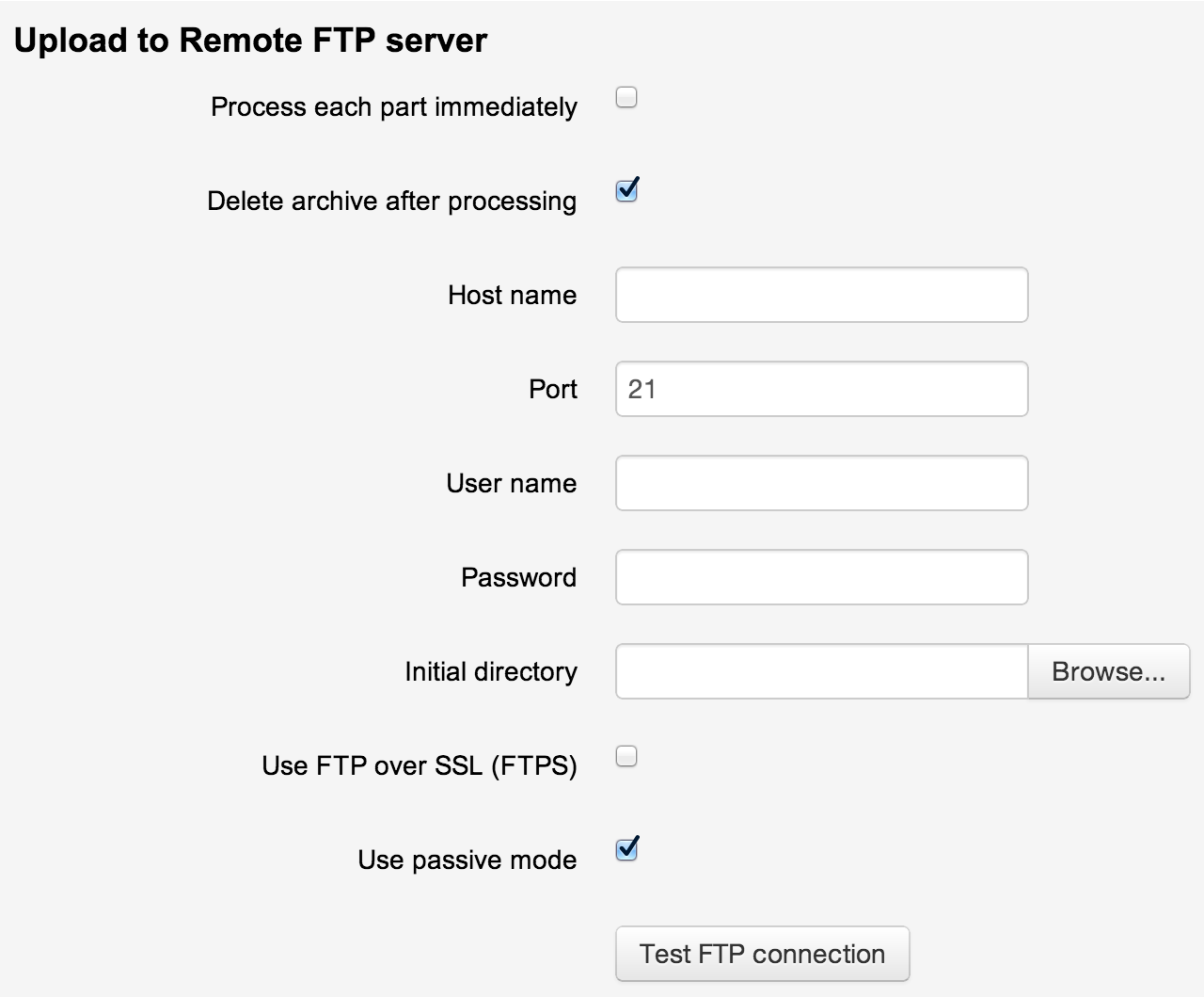
Using this engine, you can upload your backup archives to any FTP or FTPS (FTP over Implicit SSL) server. There are some "FTP" protocols and other file storage protocols which are not supported, such as SFTP, SCP, Secure FTP, FTP over Explicit SSL and SSH variants. The difference of this engine to the DirectFTP archiver engine is that this engine uploads backup archives to the server, whereas DirectFTP uploads the uncompressed files of your site. DirectFTP is designed for rapid migration, this engine is designed for easy moving of your backup archives to an off-server location.

Your originating server must support PHP's FTP extensions and not have its FTP functions blocked. Your originating server must not block FTP communication to the remote (target) server. Some hosts apply a firewall

policy which requires you to specify to which hosts your server can connect. In such a case you might need to allow communication to your remote host.

Before you begin, you should know the limitations. Most servers do not allow resuming of uploads (or even if they do, PHP doesn't quite support this feature), so the archive has to be transferred in a single step. PHP has a time limit restriction we can't overlook. The time required to upload a file to FTP equals the size of the file divided by the available bandwidth. We want to time to upload a file to be less than PHP's time limit restriction so as to avoid timing out. Since the available bandwidth is finite and constant, the only thing we can reduce in order to avoid timeouts is the file size. To this end, you have to produce split archives, by setting the part size for archive splitting in ZIP's or JPA's engine configuration pane. The suggested values are between 10Mb and 20Mb. Most servers have a bandwidth cap of 20Mbits, which equals to roughly 2Mb/sec (1 byte is 8 bits, plus there's some traffic overhead, lost packets, etc). With a time limit of 10 seconds, we can upload at most 2 Mb/sec * 10 sec = 20Mb without timing out. If you get timeouts during post-processing lower the part size.

Upload to Remote FTP Server



The available configuration options are:

Process each part immediately
: If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.

Delete archive after processing	If enabled, the archive files will be removed from your server after they are uploaded to the FTP server.
Host name	The hostname of your remote (target) server, e.g. `ftp.example.com`. You must NOT enter the ftp:// protocol prefix. If you do, Akeeba Backup will try to remove it automatically and issue a warning about it.
Port	The TCP/IP port of your remote host's FTP server. It's usually 21.
User name	The username you have to use to connect to the remote FTP server.
Password	The password you have to use to connect to the remote FTP server.
Initial directory	The absolute FTP directory to your remote site's location where your archives will be stored. This is provided by your hosting company. Do not ask us to tell you what you should put in here because we can't possibly know. There is an easy way to find it, though. Connect to your target FTP server with FileZilla. Navigate to the intended directory. Above the right-hand folder pane you will see a text box with a path. Copy this path and paste it to Akeeba Backup's setting.
Use FTP over SSL	If your remote server supports secure FTP connections over SSL (they have to be Implicit SSL; explicit SSL is not supported), you can enable this feature. In such a case you will most probably *have* to change the port. Please ask your hosting company to provide you with more information on whether they support this feature and what port you should use. You must note that this feature must also be supported by your originating server as well.
Use passive mode	Normally you should enable it, as it is the most common and firewall-safe transfer mode supported by FTP servers. Sometimes, you remote server might require active FTP transfers. In such a case please disable this, but bear in mind that your originating server might not support active FTP transfers, which usually requires tweaking the firewall!

3.3.5.10. Upload to Remote FTP server over cURL

Note

This feature is available only to Akeeba Backup Professional.

Note

This engine uses PHP's cURL functions. This may not work if your host has not installed or enabled the cURL functions. In this case you may want to use the Upload to Remote FTP server engine instead.

Using this engine, you can upload your backup archives to any FTP or FTPS (FTP over Implicit SSL) server. There are some "FTP" protocols and other file storage protocols which are not supported, such as SFTP, SCP, Secure FTP, FTP over Explicit SSL and SSH variants. The difference of this engine to the DirectFTP over cURL archiver engine is that this engine uploads backup archives to the server, whereas DirectFTP over cURL uploads the uncompressed files of your site. DirectFTP over cURL is designed for rapid migration, this engine is designed for easy moving of your backup archives to an off-server location.

Your originating server must support PHP's cURL extension and not have its FTP functions blocked. Your originating server must not block FTP communication to the remote (target) server. Some hosts apply a firewall policy which requires you to specify to which hosts your server can connect. In such a case you might need to allow communication to your remote host.

Before you begin, you should know the limitations. Most servers do not allow resuming of uploads (or even if they do, PHP doesn't quite support this feature), so the archive has to be transferred in a single step. PHP has a time limit restriction we can't overlook. The time required to upload a file to FTP equals the size of the file divided by the available bandwidth. We want to time to upload a file to be less than PHP's time limit restriction so as to avoid timing out. Since the available bandwidth is finite and constant, the only thing we can reduce in order to avoid timeouts is the file size. To this end, you have to produce split archives, by setting the part size for archive splitting in ZIP's or JPA's engine configuration pane. The suggested values are between 10Mb and 20Mb. Most

servers have a bandwidth cap of 20Mbits, which equals to roughly 2Mb/sec (1 byte is 8 bits, plus there's some traffic overhead, lost packets, etc). With a time limit of 10 seconds, we can upload at most 2 Mb/sec * 10 sec = 20Mb without timing out. If you get timeouts during post-processing lower the part size.

The available configuration options are:

Process each part immediately	If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.
Delete archive after processing	If enabled, the archive files will be removed from your server after they are uploaded to the FTP server.
Host name	The hostname of your remote (target) server, e.g. `ftp.example.com`. You must NOT enter the ftp:// protocol prefix. If you do, Akeeba Backup will try to remove it automatically and issue a warning about it.
Port	The TCP/IP port of your remote host's FTP server. It's usually 21.
User name	The username you have to use to connect to the remote FTP server.
Password	The password you have to use to connect to the remote FTP server.
Initial directory	The absolute FTP directory to your remote site's location where your archives will be stored. This is provided by your hosting company. Do not ask us to tell you what you should put in here because we can't possibly know. There is an easy way to find it, though. Connect to your target FTP server with FileZilla. Navigate to the intended directory. Above the right-hand folder pane you will see a text box with a path. Copy this path and paste it to Akeeba Backup's setting.
Use FTP over SSL	If your remote server supports secure FTP connections over SSL (they have to be Implicit SSL; explicit SSL is not supported), you can enable this feature. In such a case you will most probably *have* to change the port. Please ask your hosting company to provide you with more information on whether they support this feature and what port you should use. You must note that this feature must also be supported by your originating server as well.
Use passive mode	Normally you should enable it, as it is the most common and firewall-safe transfer mode supported by FTP servers. Sometimes, you remote server might require active FTP transfers. In such a case please disable this, but bear in mind that your originating server might not support active FTP transfers, which usually requires tweaking the firewall!
Passive mode workaround	Some badly configured / misbehaving servers report the wrong IP address when you enable the passive mode. Usually they report their internal network IP address (something like 127.0.0.1 or 192.168.1.123) instead of their public, Internet-accessible IP address. This erroneous information confuses FTP information, causing uploads to stall and eventually fail. Enabling this workaround option instructs cURL to ignore the IP address reported by the server and instead use the server's public IP address, as seen by your server. In most cases this works much better, therefore we recommend leaving this option turned on if you're not sure. You should only disable it in case of an exotic setup where the FTP server uses two different public IP addresses for the control and data channels.

3.3.5.11. Upload to Google Storage (Legacy S3 API)

Note

This feature is available only to Akeeba Backup Professional 3.5 and later.

Using this engine, you can upload your backup archives to the Google Storage cloud storage service using the interoperable API (Google Storage simulates the API of Amazon S3)

Warning

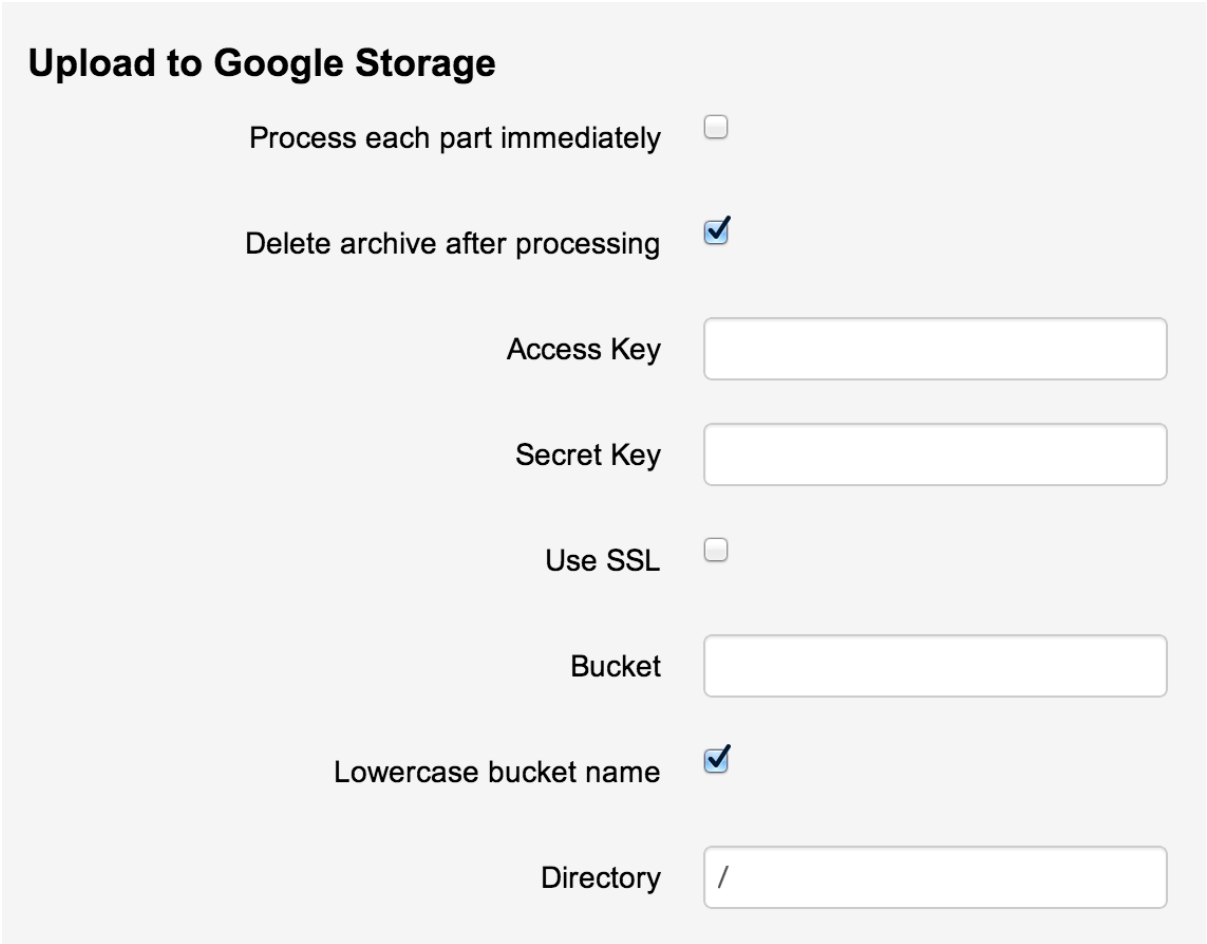
Google Storage is NOT the same thing as Google Drive. These are two separate products. If you want to upload files to Google Drive please look at the documentation for Upload to Google Drive.

Before you begin you have to go to the Google Developer's Console. After creating a storage bucket, in the left hand menu, go to Storage, Cloud Storage, Settings. Then go to the tab/option Interoperability. There you can go and enable interoperability and create the Access and Secret keys you need for Akeeba Backup.

You should also know the limitations. Google Storage's interoperable API does not allow appending to files, so the archive has to be transferred in a single step. PHP has a time limit restriction we can't overlook. The time required to upload a file to Google Storage equals the size of the file divided by the available bandwidth. We want to time to upload a file to be less than PHP's time limit restriction so as to avoid timing out. Since the available bandwidth is finite and constant, the only thing we can reduce in order to avoid timeouts is the file size. To this end, you have to produce split archives, by setting the part size for archive splitting in ZIP's or JPA's engine configuration pane. The suggested values are between 10Mb and 20Mb. Most servers have a bandwidth cap of 20Mbits, which equals to roughly 2Mb/sec (1 byte is 8 bits, plus there's some traffic overhead, lost packets, etc). With a time limit of 10 seconds, we can upload at most 2 Mb/sec * 10 sec = 20Mb without timing out. If you get timeouts during post-processing lower the part size.

Tip

If you use the native CRON mode (akeeba-backup.php), there is usually no time limit - or there is a very high time limit in the area of 3 minutes or so. Ask your host about it. Setting up a profile for use only with the native CRON mode allows you to increase the part size and reduce the number of parts a complete backup consists of.



The required settings for this engine are:

Process each part immediately
: If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.

Delete archive after processing
: If enabled, the archive files will be removed from your server after they are uploaded to Google Storage.

Access Key
: Your Google Storage Access Key, available from the Google Cloud Storage key management tool [https://code.google.com/apis/console#:storage:legacy].

Secret Key
: Your Google Storage Secret Key, available from the Google Cloud Storage key management tool [https://code.google.com/apis/console#:storage:legacy].

Use SSL
: If enabled, an encrypted connection will be used to upload your archives to Google Storage. In this case the upload will take longer, as encryption - what SSL does - is a resource intensive operation. You may have to lower your part size. We strongly recommend enabling this option for enhanced security.

  ## Warning

  Do not enable this option if your bucket name contains dots.

Bucket
: The name of your Google Storage bucket where your files will be stored in. The bucket must be already created; Akeeba Backup can not create buckets.

  ## Warning

  DO NOT CREATE BUCKETS WITH NAMES CONTAINING UPPERCASE LETTERS. If you use a bucket with uppercase letters in its name it is very possible that Akeeba Backup will not be able to upload anything to it. Moreover you should not use dots in your bucket names as they are incompatible with the Use SSL option due to an Amazon S3 limitation.

  Please note that this is a limitation of the API. It is not something we can "fix" in Akeeba Backup. If this is the case with your site, please **DO NOT** ask for support; simply create a new bucket whose name only consists of lowercase unaccented latin characters (a-z), numbers (0-9) and dashes.

Directory
: The directory inside your Google Storage bucket where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `directory/subdirectory/subsubdirectory`.

  ## Tip

  You can use Akeeba Backup's "variables" in the directory name in order to create it dynamically. These are the same variables as what you can use in the archive name, i.e. [DATE], [TIME], [HOST], [RANDOM].

Regarding the naming of buckets and directories, you have to be aware of the Google Storage rules:

- Folder names can not contain backward slashes (\). They are invalid characters.
- Bucket names can only contain lowercase letters, numbers, periods (.) and dashes (-). Accented characters, international characters, underscores and other punctuation marks are illegal characters.

- Bucket names must start with a number or a letter.
- Bucket names must be 3 to 63 characters long.
- Bucket names can't be in an IP format, e.g. 192.168.1.2
- Bucket names can't end with a dash.
- Bucket names can't have an adjacent dot and dash. For example, both `my.-bucket` and `my-.bucket` are invalid. It's best not to use dots at all as they are incompatible with the Use SSL option.

If any - or all - of those rules are broken, you'll end up with error messages that Akeeba Backup couldn't connect to Google Storage, that the calculated signature is wrong or that the bucket does not exist. This is normal and expected behaviour, as Google Storage drops the connection when it encounters invalid bucket or directory names.

3.3.5.12. Upload to Google Storage (JSON API)

Using this engine, you can upload your backup archives to the Google Storage cloud storage service using the official Google Cloud JSON API. This is the preferred method for using Google Storage.

Foreward and requirements

Setting up Google Storage is admittedly complicated. We did ask Google for permission to use the much simpler end-user OAuth2 authentication, a method which is more suitable for people who are not backend developers or IT managers. Unfortuantely, their response on July 14th, 2017 was that we were not allowed to. They said in no uncertain terms that we MUST have our clients use Google Cloud Service Accounts. Unfortunately this comes with increased server requirements and more complicated setup instructions.

First the requirements. Google Storage support requires the openssl_sign() function to be available on your server and support the "sha256WithRSAEncryption" method (it must be compiled against the OpenSSL library version 0.9.8l or later). If you are not sure please ask your host. Please note that the versions of the software required for Google Storage integration have been around since early 2012 so they shouldn't be a problem for any decently up-to-date host.

Moreover, we are only allowed to give you the following quick start instructions as an indicative way to set up Google Storage. If you need support for creating a service account or granting Akeeba Backup the appropriate permissions via the IAM Policies, Google requested that we direct you to their Google Cloud Support page [https://cloud.google.com/support/]. We are afraid this means that we will not be able to provide you with support about any issues concerning the Google Cloud side of the setup at the request of Google.

We apologize for any inconvenience. We have no option but to abide by Google's terms. It's their service, their API and their rules.

Initial Setup

Before you begin you will need to create a JSON authorization file for Akeeba Backup / Akeeba Solo. Please follow the instructions below, step by step, to do this. Kindly note that you can reuse the same JSON authorization file on multiple sites and / or backup profiles.

1. Go to https://console.developers.google.com/permissions/serviceaccounts?pli=1
2. Select the API Project where your Google Storage bucket is already located in.
3. Click on Create Service Account
4. Set the Service Account Name to `Akeeba Backup Service Account`
5. Click on Role and select Storage, Storage Object Admin

6. Check the Furnish a new private key checkbox.

7. The Key Type section appears. Make sure `JSON` is selected.

8. Click on the CREATE link at the bottom right.

9. Your server prompts you to download a file. Save it as `googlestorage.json` You will need to paste the contents of this file in the Contents of googlestorage.json (read the documentation) field in the Configuration page of Akeeba Backup / Akeeba Solo.

Important

If you lose the `googlestorage.json` file you will have to delete the Service Account and create it afresh. If you had any sites already set up with this `googlestorage.json` you will need to reconfigure them with the *new* file you created for the *new* Service Account. In short: don't lose that file, you *will* need it to (re)connect your sites with Google Storage.

Post-processing engine options

Upload to Google Storage (JSON API)

The settings for this engine are:

Process each part immediately
: If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.

Delete archive after processing
: If enabled, the archive files will be removed from your server after they are uploaded to Google Storage.

Enabled chunk upload
: When enabled, Akeeba Backup / Akeeba Solo will upload your backup archives in 5Mb chunks. This is the recommended methods for larger (over 10Mb) archives and/or archive parts.

Bucket
: The name of your Google Storage bucket where your files will be stored in. The bucket must be already created; the application can not create buckets.

  ## Warning

  DO NOT CREATE BUCKETS WITH NAMES CONTAINING UPPERCASE LETTERS. If you use a bucket with uppercase letters in its name it is very possible that the application will not be able to upload anything to it.

  Please note that this is a limitation of the API. It is not something we can "fix" in the application. If this is the case with your site, please simply create a new bucket whose name only consists of lowercase unaccented latin characters (a-z), numbers (0-9), dashes and dots.

Directory
: The directory inside your Google Storage bucket where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `directory/subdirectory/subsubdirectory`.

Tip

You can use the application's "variables" in the directory name in order to create it dynamically. These are the same variables as what you can use in the archive name, i.e. [DATE], [TIME], [HOST], [RANDOM].

Contents of googlestorage.json (read the documentation)	Open the JSON file you created in the Initial Setup stage outlined above. Copy all of its contents. Paste them in this field. Make sure you have included the curly braces, { and }, at the beginning and end of the file respectively. Don't worry about line breaks being "eaten up", they are NOT important.

Regarding the naming of buckets and directories, you have to be aware of the Google Storage rules:

- Folder names can not contain backward slashes (\). They are invalid characters.
- Bucket names can only contain lowercase letters, numbers, periods (.) and dashes (-). Accented characters, international characters, underscores and other punctuation marks are illegal characters.
- Bucket names must start with a number or a letter.
- Bucket names must be 3 to 63 characters long.
- Bucket names can't be in an IP format, e.g. 192.168.1.2
- Bucket names can't end with a dash.
- Bucket names can't have an adjacent dot and dash. For example, both `my.-bucket` and `my-.bucket` are invalid.

If any - or all - of those rules are broken, you'll end up with error messages that the application couldn't connect to Google Storage, that the calculated signature is wrong or that the bucket does not exist. This is normal and expected behaviour, as Google Storage drops the connection when it encounters invalid bucket or directory names.

3.3.5.13. Upload to Google Drive

Note

This feature is available only to Akeeba Backup Professional.

Using this engine you can upload your backup archives to Google Drive.

Important security and privacy information

Google Drive uses the OAuth 2 authentication method. This requires a fixed endpoint (URL) for each application which uses it, such as Akeeba Backup. Since Akeeba Backup is installed on your site it has a different endpoint URL for each installation, meaning you could not normally use Google Drive's API to upload files. We have solved it by creating a small script which lives on our own server and acts as an intermediary between your site and Google Drive. When you are linking Akeeba Backup to Google Drive you are going through the script on our site. Moreover, whenever the request token (a time-limited key given by Google Drive to your Akeeba Backup installation to access the service) expires your Akeeba Backup installation has to exchange it with a new token. This process also takes place through the script on our site. Please note that even though you are going through our site we DO NOT store this information and we DO NOT have access to your Google Drive account.

WE DO NOT STORE THE ACCESS CREDENTIALS TO YOUR GOOGLE DRIVE ACCOUNT. WE DO NOT HAVE ACCESS TO YOUR GOOGLE DRIVE ACCOUNT. SINCE CONNECTIONS TO OUR SITE ARE PROTECTED BY STRONG ENCRYPTION (HTTPS) NOBODY ELSE CAN SEE THE INFORMATION EXCHANGED BETWEEN YOUR SITE AND OUR SITE AND BETWEEN OUR SITE AND GOOGLE DRIVE. HOWEVER, AT THE FINAL STEP OF THE AUTHENTICATION PROCESS, YOUR BROWSER IS

SENDING THE ACCESS TOKENS TO YOUR SITE. SOMEONE CAN STEAL THEM IN TRANSIT IF AND ONLY IF YOU ARE NOT USING HTTPS ON YOUR SITE'S ADMINISTRATOR.

For this reason we DO NOT accept any responsibility whatsoever for any use, abuse or misuse of your connection information to Google Drive. If you do not accept this condition you are FORBIDDEN from using the intermediary script on our site which, simply put, means that you cannot use the Google Drive integration.

Moreover, the above means that there are additional requirements for using Google Drive integration on your Akeeba Backup installation:

- You need the PHP cURL extension to be loaded and enabled on your server. Most servers do that by default. If your server doesn't have it enabled the upload will fail and warn you that cURL is not enabled.
- Your server's firewall must allow outbound HTTPS connections to www.akeebabackup.com over port 443 (standard HTTPS port) to get new tokens every time the current access token expires.
- Your server's firewall must allow outbound HTTPS connections to Google Drive's domains over port 443 to allow the integration to work. These domain names are, unfortunately, not predefined. Most likely your server administrator will have to allow outbound HTTPS connections to any domain name matching *.googleapis.com to allow this integration to work. This is a restriction of how the Google Drive service is designed, not something we can modify (obviously, we're not Google).

Settings

The settings for this engine are:

Process each part immediately
: If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.

Delete archive after processing
: If enabled, the archive files will be removed from your server after they are uploaded to Google Drive

Enabled chunked upload
: The application will always try to upload your backup archives / backup archive parts in small chunks and then ask Google Drive to assemble them back into one file. This allows you to transfer larger archives more reliably.

  When you enable this option every step of the chunked upload process will take place in a separate page load, reducing the risk of timeouts if you are transferring large archive part files (over 5Mb). When you disable this option the entire upload process has to take place in a single page load.

  #### Warning

  When you select Process each part immediately this option has no effect! In this case the entire upload operation for each part will be attempted *in a single page load*. For this reason we recommend that you use a Part Size for Split Archives of 5Mb or less to avoid timeouts.

Chunk size
: This option determines the size of the chunk which will be used by the chunked upload option above. You are recommended to use a relatively small value around 5 to 20 Mb to prevent backup timeouts. The exact maximum value you can use depends on the speed of your server and its connection speed to the Google Drive server. Try starting high and lower it if the backup fails during transfer to Google Drive.

Authentication – Step 1

If this is the **FIRST** site you connect to Akeeba Backup click on this button and follow the instructions.

On **EVERY SUBSEQUENT SITE** do NOT click on this button! Instead copy the Refresh Token from the first site into this new site's Refresh Token edit box further below the page.

Warning

Google imposes a limitation of 20 authorizations for a single application –like Akeeba Backup– with Google Drive. Simply put, every time you click on the Authentication – Step 1 button a new Refresh Token is generated. The 21st time you generate a new Refresh Token the one you had created the very first time becomes automatically invalid without warning. This is how Google Drive is designed to operate. For this reason we strongly recommend AGAINST using this button on subsequent sites. Instead, copy the Refresh Token.

Directory

The directory inside your Google Drive where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `directory/subdirectory/subsubdirectory`.

Tip

You can use Akeeba Backup's "variables" in the directory name in order to create it dynamically. These are the same variables as what you can use in the archive name, i.e. [DATE], [TIME], [HOST], [RANDOM].

Warning

Object (file and folder) naming in Google Drive is ambiguous by design. This means that two or more files / folders with the same name can exist inside the same folder at the same time. In other words, a folder called My Files may contain ten *different* files all called "File 1"! Obviously this is problematic when you want to store backups which need to be uniquely named (otherwise you'd have no idea which backup is the one you want to use!). We work around this issue using the following conventions:

- If there are multiple folders by the same name we choose the first one returned by the Google Drive API. There are no guarantees which one it will be! **Please do NOT store backup archives in folders with ambiguous names** or the remote file operations (quota management, download to server, download to browser, delete) will most likely fail.
- If a folder in the path you specified does not exist we create it
- If a file by the same name exists in the folder you specified we delete it before uploading the new one.

Access Token

This is the temporary Access Token generated by Google Drive. It has a lifetime of one hour (3600 seconds). After that Akeeba Backup will use the Refresh Token automatically to generate a new Access Token. Please do not touch that field and do NOT copy it to other sites.

Refresh Token

This is essentially what connects your Akeeba Backup installation with your Google Drive. When you want to connect more sites to Google Drive please copy the Refresh Token from another site linked to the same Google Drive account to your site's Refresh Token field.

Warning

Since all of your sites are using the same Refresh Token to connect to Google Drive you must NOT run backups on multiple sites simultaneously. That would cause all backups to fail since one active instance of Akeeba Backup would be invalidating the

Access Token generated by the other active instance of Akeeba Backup also trying to upload to Google Drive. This is an architectural limitation of Google Drive.

3.3.5.14. Upload to iDriveSync

Using this engine, you can upload your backup archives to the iDriveSync low-cost, encrypted, cloud storage service.

The required settings for this engine are:

Process each part immediately
: If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.

Delete archive after processing
: If enabled, the archive files will be removed from your server after they are uploaded to iDriveSync

Username or e-email
: Your iDriveSync username or email address

Password
: Your iDriveSync password

Private key (optional)
: If you have locked your account with a private key (which means that all your data is stored encrypted in iDriveSync) please enter your Private Key here. If you are not making use of this feature please leave this field blank.

Directory
: The directory inside your iDriveSync where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `directory/subdirectory/subsubdirectory`.

  #### Tip

  You can use Akeeba Backup's "variables" in the directory name in order to create it dynamically. These are the same variables as what you can use in the archive name, i.e. [DATE], [TIME], [HOST], [RANDOM].

Use the new endpoint
: This is required for iDriveSync accounts created after 2014. If you have entered your username/e-mail and password correctly but Akeeba Backup can't connect to iDriveSync please try checking this box.

  Lengthier explanation. Sometime after 2014 iDriveSync started signing up new users through iDrive.com instead of iDriveSync.com. The new accounts need to access a new service endpoint (URL) to upload new files, delete existing files and so on. Meanwhile, accounts created before this change still need to access the old service endpoint (URL). The same service, two different interface implementations, making it impossible for us to automatically detect which method will work with your iDriveSync account. Therefore the only thing we could do was add this confusing checkbox. We're sorry about that.

3.3.5.15. Upload to Amazon S3 (Legacy API)

Note

This feature has been discontinued. If you were using it please upgrade your backup profiles to the Upload to Amazon S3 post-processing engine.

3.3.5.16. Upload to Amazon S3

Note

This feature is available only to Akeeba Backup Professional. Older versions of Akeeba Backup may not have all of the options discussed here.

Using this engine, you can upload your backup archives to the Amazon S3 cloud storage service and other storage services providing an S3-compatible API. With dirt cheap prices per Gigabyte, it is an ideal option for securing your backups. Even if your host's data center is annihilated by a natural disaster and your local PC and storage media are wiped out by an unlikely event, you will still have a copy of your site readily accessible and easy to restore.

This engine supports multi-part uploads to Amazon S3. This means that, unlike the other post-processing engines, even if you do not use split archives, Akeeba Backup will still be able to upload your files to Amazon S3! This new feature allows Akeeba Backup to upload your backup archive in 5Mb chunks so that it doesn't time out when uploading a very big archive file. That said, we STRONGLY suggest using a part size for archive splitting of 2000Mb. This is required to work around a PHP limitation which causes extraction to fail if the file size is over roughly 2Gb.

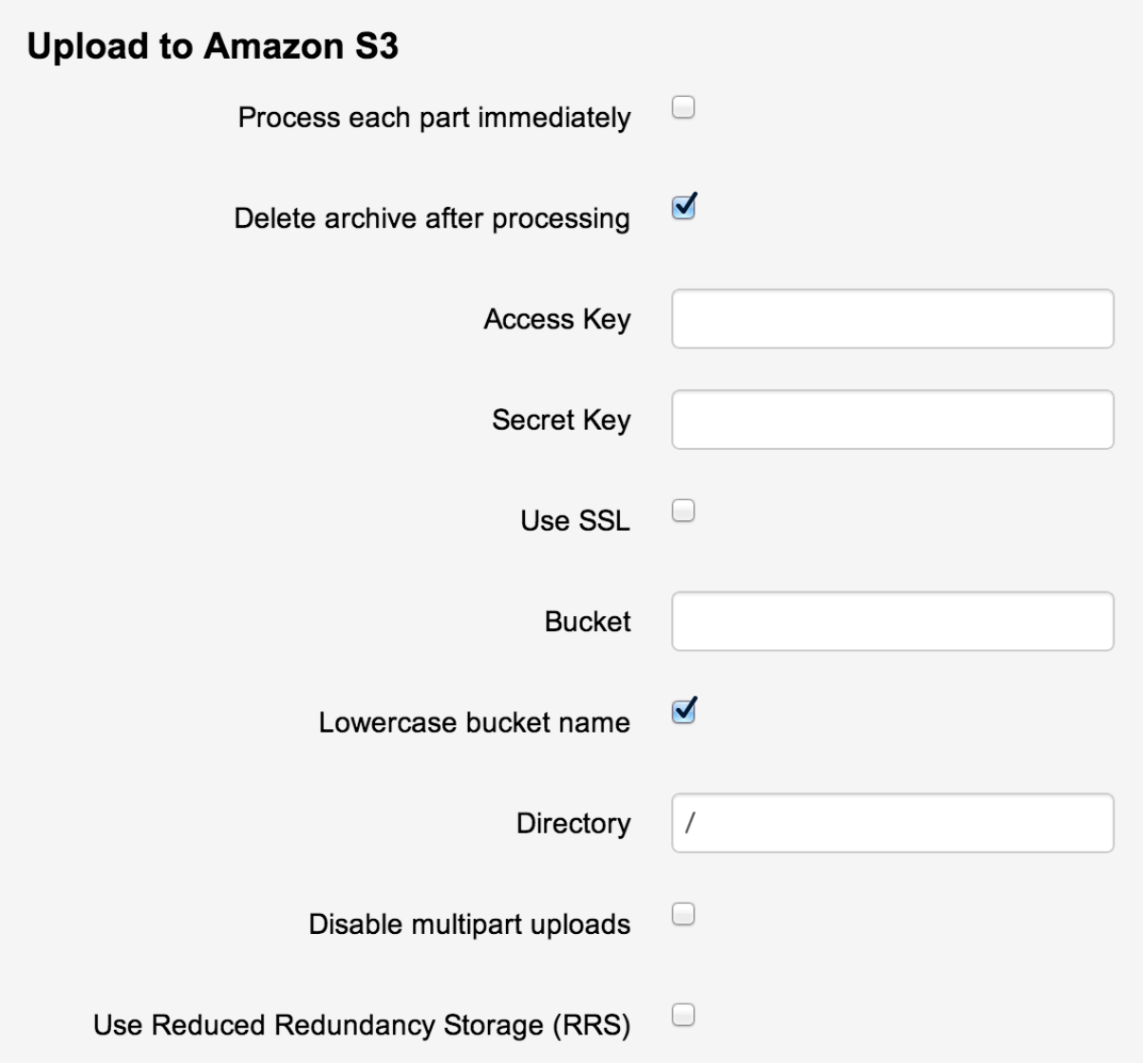
You can also specify a custom endpoint URL. This allows you to use this feature with third party cloud storage services offering an API compatible with Amazon S3 such as Cloudian, Riak CS, Ceph, Connectria, HostEurope, Dunkel, S3For.me, Nimbus, Walrus, GreenQloud, Scality Ring, CloudStack and so on. If a cloud solution (public or private) claims that it is compatible with S3 then you can use it with Akeeba Backup.

Note

Akeeba Backup 5.1.2 and later support the Beijing Amazon S3 region, i.e. storage buckets hosted in China. These buckets are only accessible from inside China and have a few caveats:

- You can only access buckets in the Beijing region from inside China.
- Download to browser is not supported unless you have a license by the Chinese government to share content from your Amazon S3 bucket. That's because downloading to browser requires a pre-signed URL which could, in theory, be used to disseminate material from your Amazon S3 bucket to others. So even though you see the Download button it will most likely result in an error.
- Sometimes deleting and trying to re-upload an object or trying to overwrite fails silently (without an error message). WE strongly recommend using unique names for your backup archives and testing them frequently.

Upload to Amazon S3



The required settings for this engine are:

Process each part immediately	If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.
Delete archive after processing	If enabled, the archive files will be removed from your server after they are uploaded to Amazon S3.
Access Key	Your Amazon S3 Access Key
Secret Key	Your Amazon S3 Secret Key
Use SSL	If enabled, an encrypted connection will be used to upload your archives to Amazon S3.

Warning

Do not use this option if your bucket name contains dots.

Bucket

The name of your Amazon S3 bucket where your files will be stored in. The bucket must be already created; Akeeba Backup can not create buckets.

Warning

DO NOT CREATE BUCKETS WITH NAMES CONTAINING UPPERCASE LETTERS. AMAZON CLEARLY WARNS AGAINST DOING THAT. If you use a bucket with uppercase letters in its name it is very possible that Akeeba Backup will not be able to upload anything to it. More specifically, it seems that if your web server is located in Europe, you will be unable to use a bucket with uppercase letters in its name. If your server is in the US, you will most likely be able to use such a bucket. Your mileage may vary. The same applies if your bucket name contains dots and you try using the Use SSL option, for reasons that have to do with Amazon S3's setup.

Please note that this is a limitation imposed by Amazon itself. It is not something we can "fix" in Akeeba Backup. If this is the case with your site, please **DO NOT** ask for support; simply create a new bucket whose name only consists of lowercase unaccented Latin characters (a-z), numbers (0-9) and dashes.

Amazon S3 Region

Please select which S3 Region you have created your bucket in. This is MANDATORY for using the newer, more secure, v4 signature method. You can see the region of your bucket in your Amazon S3 management console. Right click on a bucket and click on Properties. A new pane opens to the left. The second row is labelled Region. This is where your bucket was created in. Go back to Akeeba Backup and select the corresponding option from the drop-down.

Important

If you choose the wrong region the connection WILL fail.

Please note that there are some reserved regions which have not been launched by Amazon at the time we wrote this engine. They are included for forward compatibility should and when Amazon launches those regions.

Signature method

This option determines the authentication API which will be used to "log in" the backup engine to your Amazon S3 bucket. You have two options:

- **v4 (preferred for Amazon S3)**. If you are using Amazon S3 (not a compatible third party storage service) and you are not sure, you need to choose this option. Moreover, you MUST specify the Amazon S3 Region in the option above. This option implements the newer AWS4 (v4) authentication API. Buckets created in Amazon S3 regions brought online after January 2014 (e.g. Frankfurt) will only accept this option. Older buckets will work with either option.

- **v2 (legacy mode, third party storage providers)**. If you are using an S3-compatible third party storage service (NOT Amazon S3) you MUST use this option. We do not recommend using this option with Amazon S3 as this authentication method is going to be phased out by Amazon itself in the future.

Directory

The directory inside your Amazon S3 bucket where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `directory/subdirectory/subsubdirectory`.

Tip

You can use Akeeba Backup's "variables" in the directory name in order to create it dynamically. These are the same variables as what you can use in the archive name, i.e. [DATE], [TIME], [HOST], [RANDOM].

Disable multipart uploads
: Since Akeeba Backup 3.2, uploads to Amazon S3 of parts over 5Mb use Amazon's new multi-part upload feature. This allows Akeeba Backup to upload the backup archive in 5Mb chunks and then ask Amazon S3 to glue them together in one big file. However, some hosts time out while uploading archives using this method. In that case it's preferable to use a relatively small Part Size for Split Archive setting (around 10-20Mb, your mileage may vary) and upload the entire archive part in one go. Enabling this option ensures that, no matter how big or small your Part Size for Split Archives setting is, the upload of the backup archive happens in one go. You MUST use it if you get RequestTimeout warnings while Akeeba Backup is trying to upload the backup archives to Amazon S3.

Custom endpoint
: Enter the custom endpoint (connection URL) of a third party service which supports an Amazon S3 compatible API. Please remember to set the Signature method to v2 when using this option.

Regarding the naming of buckets and directories, you have to be aware of the Amazon S3 rules (these rules are a simplified form of the list S3Fox presents you with when you try to create a new bucket):

- Folder names can not contain backward slashes (\). They are invalid characters.
- Bucket names can only contain lowercase letters, numbers, periods (.) and dashes (-). Accented characters, international characters, underscores and other punctuation marks are illegal characters.

Important

Even if you created a bucket using uppercase letters, **you must type its name with lowercase letters**. Amazon S3 automatically converts the bucket name to all-lowercase. Also note that, as stated above, you may NOT be able to use at all under some circumstances. Generally, your should avoid using uppercase letters.

- Bucket names must start with a number or a letter.
- Bucket names must be 3 to 63 characters long.
- Bucket names can't be in an IP format, e.g. 192.168.1.2
- Bucket names can't end with a dash.
- Bucket names can't have an adjacent dot and dash. For example, both `my.-bucket` and `my-.bucket` are invalid. It's best to avoid dots at all as they are incompatible with the Use SSL option.

If any - or all - of those rules are broken, you'll end up with error messages that Akeeba Backup couldn't connect to S3, that the calculated signature is wrong or that the bucket does not exist. This is normal and expected behaviour, as Amazon S3 drops the connection when it encounters invalid bucket or directory names.

3.3.5.17. Upload to Remote SFTP server

Note

This feature is available only to Akeeba Backup Professional.

Note

This engine uses the PHP extension called SSH2. The SSH2 extension is still marked as an alpha and is not enabled by default or even provided by many commercial hosts. In this case you may want to use the

Upload to Remote SFTP server over cURL engine instead which uses PHP's cURL extension, available on most hosts.

Using this engine, you can upload your backup archives to any SFTP (Secure File Transfer Protocol) server. Please note that SFTP is the *encrypted* file transfer protocol provided by SSH servers. Even though the name is close, it has nothing to do with plain old FTP or FTP over SSL. Not all servers support this but for those which do this is the most secure file transfer option.

The difference of this engine to the DirectSFTP archiver engine is that this engine uploads backup archives to the server, whereas DirectSFTP uploads the uncompressed files of your site. DirectSFTP is designed for rapid migration, this engine is designed for easy moving of your backup archives to an off-server location. Moreover, this engine also supports connecting to your SFTP server using cryptographic key files instead of passwords, a much safer (and much harder and geekier) user authentication method.

Your originating server must have PHP's SSH2 module installed and activated and its functions unblocked. Your originating server must also not block SFTP communication to the remote (target) server. Some hosts apply a firewall policy which requires you to specify to which hosts your server can connect. In such a case you might need to allow communication to your remote host over TCP port 22 (or whatever port you are using).

Before you begin, you should know the limitations. SFTP does not allow resuming of uploads so the archive has to be transferred in a single step. PHP has a time limit restriction we can't overlook. The time required to upload a file to SFTP equals the size of the file divided by the available bandwidth. We want to time to upload a file to be less than PHP's time limit restriction to avoid timing out. Since the available bandwidth is finite and constant, the only thing we can reduce in order to avoid timeouts is the file size. To this end, you have to produce split archives, by setting the part size for archive splitting in ZIP's or JPA's engine configuration pane. The suggested values are between 10Mb and 20Mb. Most servers have a bandwidth cap of 20Mbits, which equals to roughly 2Mb/sec (1 byte is 8 bits, plus there's some traffic overhead, lost packets, etc). With a time limit of 10 seconds, we can upload at most 2 Mb/sec * 10 sec = 20Mb without timing out. If you get timeouts during post-processing lower the part size.

The available configuration options are:

Process each part immediately	If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.
Delete archive after processing	If enabled, the archive files will be removed from your server after they are uploaded to the SFTP server.
Host name	The hostname of your remote (target) server, e.g. `secure.example.com`. You must NOT enter the sftp:// or ssh:// protocol prefix. If you do, Akeeba Backup will try to remove it automatically and issue a warning about it.
Port	The TCP/IP port of your remote host's SFTP (SSH) server. It's usually 22. If unsure, please ask your host.
User name	The username you have to use to connect to the remote SFTP server. This must be always provided
Password	The password you have to use to connect to the remote SFTP server.
Private key file (advanced)	Many (but not all) SSH/SFTP servers allow you to connect to them using cryptographic key files for user authentication. This method is far more secure than using a password. Passwords can be guessed within some degree of feasibility because of their relatively short length and complexity. Cryptographic keys are night impossible to guess with the current technology due to their complexity (on average, more than 100 times as complex as a typical password).

If you want to use this kind of authentication you will need to provide a set of two files, your public and private key files. In this field you have to enter the full filesystem path to your private key file. The private key file must be in RSA or DSA format and has to be configured to be accepted by your remote host. The exact configuration depends on your SSH/SFTP server and is beyond the scope of this documentation. If you are a curious geek we strongly advise you to search for "ssh certificate authentication" in your favourite search engine for more information.

If you are using encrypted private key files enter the passphrase in the Password field above. If it is not encrypted, which is a bad security practice, leave the Password field blank.

Important

If the libssh2 library that the SSH2 extension of PHP is using is compiled against GnuTLS (instead of OpenSSL) you will NOT be able to use encrypted private key files. This has to do with bugs / missing features of GnuTLS, not our code. If you can't get certificate authentication to work please try providing an unencrypted private key file and leave the Password field blank.

Public Key File (advanced)
: If you are using the key file authentication method described above you will also have to supply the public key file. Enter here the full filesystem path to the public key file. The public key file must be in RSA or DSA format and, of course, unencrypted (as it's a public key).

Initial directory
: The absolute filesystem path to your remote site's location where your archives will be stored. This is provided by your hosting company. Do not ask us to tell you what you should put in here because we can't possibly know. There is an easy way to find it, though. Connect to your target SFTP server with FileZilla. Navigate to the intended directory. Above the right-hand folder pane you will see a text box with a path. Copy this path and paste it to Akeeba Backup's setting.

3.3.5.18. Upload to Remote SFTP server over cURL

Note

This feature is available only to Akeeba Backup Professional.

Note

This engine uses the PHP cURL extension. If your host has disabled the cURL extension but has enabled the SSH2 PHP extension you may want to use the Upload to Remote SFTP server engine instead which uses PHP's SSH2 extension.

Using this engine, you can upload your backup archives to any SFTP (Secure File Transfer Protocol) server. Please note that SFTP is the *encrypted* file transfer protocol provided by SSH servers. Even though the name is close, it has nothing to do with plain old FTP or FTP over SSL. Not all servers support this but for those which do this is the most secure file transfer option.

The difference of this engine to the DirectSFTP over cURL archiver engine is that this engine uploads backup archives to the server, whereas DirectSFTP over cURL uploads the uncompressed files of your site. DirectSFTP over cURL is designed for rapid migration, this engine is designed for easy moving of your backup archives to an off-server location.

Your originating server (where you are backing up *from*) must a. have PHP's cURL extension installed and activated, b. have the cURL extension compiled with SFTP support and c. allow outbound TCP/IP connections to your target host's SSH port. Please note that some hosts provide the cURL extension without SFTP support. This feature will NOT work on these hosts. Moreover, some hosts apply a firewall policy which requires you to specify to which hosts your server can connect. In such a case you might need to allow communication to your remote host.

Before you begin, you should know the limitations. SFTP does not allow resuming of uploads so the archive has to be transferred in a single step. PHP has a time limit restriction we can't overlook. The time required to upload a file to SFTP equals the size of the file divided by the available bandwidth. We want to time to upload a file to be less than PHP's time limit restriction to avoid timing out. Since the available bandwidth is finite and constant, the only thing we can reduce in order to avoid timeouts is the file size. To this end, you have to produce split archives, by setting the part size for archive splitting in ZIP's or JPA's engine configuration pane. The suggested values are between 10Mb and 20Mb. Most servers have a bandwidth cap of 20Mbits, which equals to roughly 2Mb/sec (1 byte is 8 bits, plus there's some traffic overhead, lost packets, etc). With a time limit of 10 seconds, we can upload at most 2 Mb/sec * 10 sec = 20Mb without timing out. If you get timeouts during post-processing lower the part size.

The available configuration options are:

Process each part immediately
: If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.

Delete archive after processing
: If enabled, the archive files will be removed from your server after they are uploaded to the SFTP server.

Host name
: The hostname of your remote (target) server, e.g. `secure.example.com`. You must NOT enter the sftp:// or ssh:// protocol prefix. If you do, Akeeba Backup will try to remove it automatically and issue a warning about it.

Port
: The TCP/IP port of your remote host's SFTP (SSH) server. It's usually 22. If unsure, please ask your host.

User name
: The username you have to use to connect to the remote SFTP server. This must be always provided

Password
: The password you have to use to connect to the remote SFTP server.

Private key file (advanced)
: Many (but not all) SSH/SFTP servers allow you to connect to them using cryptographic key files for user authentication. This method is far more secure than using a password. Passwords can be guessed within some degree of feasibility because of their relatively short length and complexity. Cryptographic keys are night impossible to guess with the current technology due to their complexity (on average, more than 100 times as complex as a typical password).

  If you want to use this kind of authentication you will need to provide a set of two files, your public and private key files. In this field you have to enter the full filesystem path to your private key file. The private key file must be in RSA or DSA format and has to be configured to be accepted by your remote host. The exact configuration depends on your SSH/SFTP server and is beyond the scope of this documentation. If you are a curious geek we strongly advise you to search for "ssh certificate authentication" in your favourite search engine for more information.

  If you are using encrypted private key files enter the passphrase in the Password field above. If it is not encrypted, which is a bad security practice, leave the Password field blank.

Important

If cURL is compiled against GnuTLS (instead of OpenSSL) you will NOT be able to use encrypted private key files. This has to do with bugs / missing features of GnuTLS, not our code. If you can't get certificate authentication to work please try providing an unencrypted private key file and leave the Password field blank.

Public Key File (advanced)	If you are using the key file authentication method described above you will also have to supply the public key file. Enter here the full filesystem path to the public key file. The public key file must be in RSA or DSA format and, of course, unencrypted (as it's a public key). Some newer versions of cURL allow you to leave this blank, in which case they will derive the public key information from the private key file. We do not recommend this approach.
Initial directory	The absolute filesystem path to your remote site's location where your archives will be stored. This is provided by your hosting company. Do not ask us to tell you what you should put in here because we can't possibly know. There is an easy way to find it, though. Connect to your target SFTP server with FileZilla. Navigate to the intended directory. Above the right-hand folder pane you will see a text box with a path. Copy this path and paste it to Akeeba Backup's setting.

3.3.5.19. Upload to SugarSync

Note

This feature is available only to Akeeba Backup Professional 3.5.a1 and later.

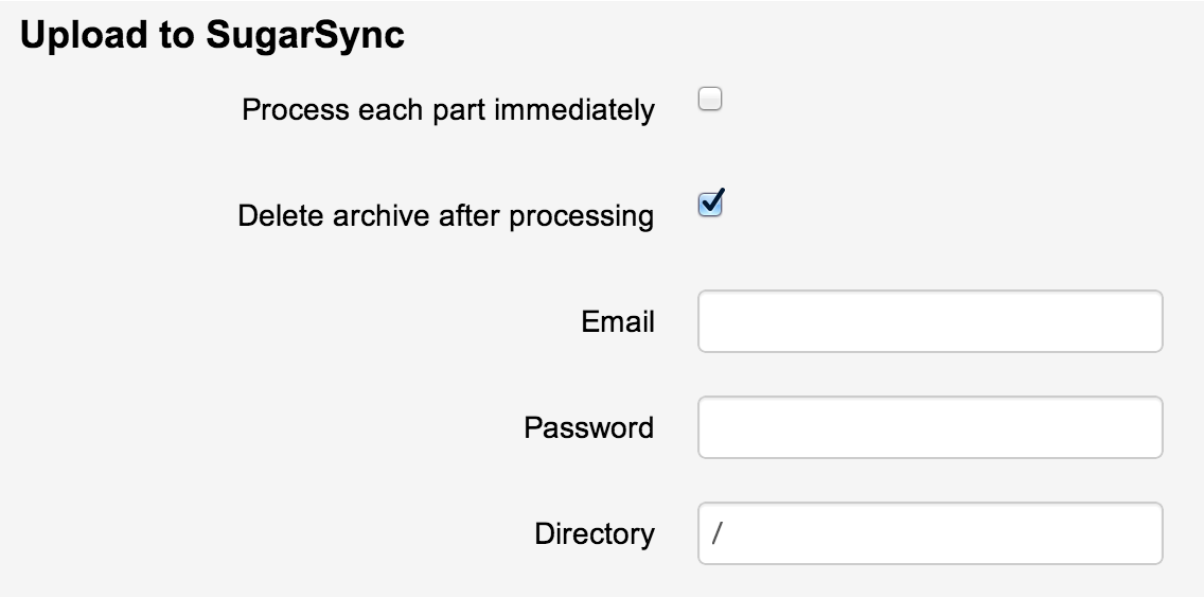
Using this engine, you can upload your backup archives to the SugarSync [http://www.sugarsync.com] cloud storage service. SugarSync has a free tier (with 5Gb of free space) and a paid tier. Akeeba Backup can work with either one.

Please note that Akeeba Backup can only upload files to Sync Folders, it can not upload files directly to a Workspace (a single device). You have to set up your Sync Folders in SugarSync before using Akeeba Backup. If you have not created or specified any Sync Folder, Akeeba Backup will upload the backup archives to your Magic Briefcase, the default Sync Folder which syncs between all of your devices, including your mobile devices (iPhone, iPad, Android phones, ...).

Before you begin, you should know the limitations. As most cloud storage providers, SugarSync does not allow appending to files, so the archive has to be transferred in a single step. PHP has a time limit restriction we can't overlook. The time required to upload a file to SugarSync equals the size of the file divided by the available bandwidth. We want to time to upload a file to be less than PHP's time limit restriction so as to avoid timing out. Since the available bandwidth is finite and constant, the only thing we can reduce in order to avoid timeouts is the file size. To this end, you have to produce split archives, by setting the part size for archive splitting in ZIP's or JPA's engine configuration pane. The suggested values are between 10Mb and 20Mb. Most servers have a bandwidth cap of 20Mbits, which equals to roughly 2Mb/sec (1 byte is 8 bits, plus there's some traffic overhead, lost packets, etc). With a time limit of 10 seconds, we can upload at most 2 Mb/sec * 10 sec = 20Mb without timing out. If you get timeouts during post-processing lower the part size.

Tip

If you use the native CRON mode (akeeba-backup.php), there is usually no time limit - or there is a very high time limit in the area of 3 minutes or so. Ask your host about it. Setting up a profile for use only with the native CRON mode allows you to increase the part size and reduce the number of parts a complete backup consists of.



The required settings for this engine are:

Process each part immediately	If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.
Delete archive after processing	If enabled, the archive files will be removed from your server after they are uploaded to SugarSync.
Email	The email used by your SugarSync account.
Password	The password used by your SugarSync account.
Directory	The directory inside SugarSync where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `/directory/subdirectory/subsubdirectory`. You may use the same variables used in archive naming, e.g. [HOST] for the site's host name or [DATE] for the current date. Please note that the first part of your directory should be the name of your shared folder. For example, if you have a shared folder named `backups` and you want to create a subdirectory inside it based on the site's name, you need to enter `backups/[HOST]` in the directory box. If a Sync Folder by the name "backups" is not found, a directory named "backups" will be created inside your Magic Briefcase folder. Yes, it's more complicated than, say, DropBox – but that's also why SugarSync is more powerful.

3.3.5.20. Upload to WebDAV

Note

This feature is available only to Akeeba Backup Professional 3.10.1 and later.

Using this engine, you can upload your backup archives to any server which supports the WebDAV (Web Distributed Authoring and Versioning) protocol. Examples of storage services supporting WebDAV:

- OwnCloud [http://doc.owncloud.org/server/5.0/user_manual/files/files.html] is a software solution that you can install on your own servers to provide a private cloud.

- CloudDAV [http://storagemadeeasy.com/CloudDav/] is a service which gives you WebDAV access to a plethora of cloud storage providers: Amazon S3, GMail, RackSpace CloudFiles, Microsoft OneDrive (formerly: SkyDrive), Windows Azure BLOB Storage, iCloud, LiveMesh, Box.com, FTP servers, Email (which, unlike the Send by email engine in Akeeba Backup, does support large files), Google Docs, Mezeo, Zimbra, FilesAnywhere, Dropbox, Google Storage, CloudMe, Microsoft SharePoint, Trend Micro, OpenStack Swift (supported by several providers), Google sites, HP cloud, Alfresco cloud, Open S3, Eucalyptus Walrus, Microsoft Office 365, EMC Atmos, iKoula - iKeepinCloud, PogoPlug, Ubuntu One, SugarSync, Hosting Solutions, BaseCamp, Huddle, IBM Files Cloud, Scality, Google Drive, Memset Memstore, DumpTruck, ThinkOn, Evernote, Cloudian, Copy.com, Salesforce. [TESTED with Amazon S3 as the storage provider]
- Apache web server (when the optional WebDAV support is enabled – recommended for advanced users only).
- 4Shared [http://www.4shared.com/].
- ADrive [http://www.adrive.com/].
- Amazon Cloud Drive [http://www.amazon.com/gp/feature.html/ref=cd_def?ie=UTF8&*Version*=1&*entries*=0&docId=1000828861].
- Box.com [https://www.box.com/].
- CloudSafe [https://secure.cloudsafe.com/login/].
- DriveHQ [https://www.drivehq.com/].
- DumpTruck [http://www.goldenfrog.com/].
- FilesAnywhere [https://www.filesanywhere.com/].
- MyDrive [http://www.mydrive.net/].
- MyDisk.se. [https://mydisk.com/web/main.php?show=home]
- PowerFolder [https://www.powerfolder.com/].
- OVH.net [http://ovh.net/]
- Safecopy Backup [http://safecopybackup.com/].
- Strato HiDrive [https://www.free-hidrive.com/index.html].
- Telekom Mediencenter [http://mediencenter.telekom.de/].
- Pretty much every storage provider which claims to support WebDAV

Tip

You can find more information for WebDAV access of each of these providers in http://www.free-online-backup-services.com/features/webdav.html

Note

We have not thoroughly tested and do not guarantee that any of the above providers will work smoothly with Akeeba Backup unless you see the notive [TESTED] next to it.

Before you begin, you should know the limitations. As most remote storage technologies, WebDAV does not allow appending to files, so the archive has to be transferred in a single step. PHP has a time limit restriction we can't overlook. The time required to upload a file to WebDAV equals the size of the file divided by the available bandwidth. We want to time to upload a file to be less than PHP's time limit restriction so as to avoid timing out. Since the available bandwidth is finite and constant, the only thing we can reduce in order to avoid timeouts is the file size. To this end, you have to produce split archives, by setting the part size for archive splitting in ZIP's or JPA's engine configuration pane. The suggested values are between 10Mb and 20Mb. Most servers have

a bandwidth cap of 20Mbits, which equals to roughly 2Mb/sec (1 byte is 8 bits, plus there's some traffic overhead, lost packets, etc). With a time limit of 10 seconds, we can upload at most 2 Mb/sec * 10 sec = 20Mb without timing out. If you get timeouts during post-processing lower the part size.

Tip

If you use the native CRON mode (akeeba-backup.php), there is usually no time limit - or there is a very high time limit in the area of 3 minutes or so. Ask your host about it. Setting up a profile for use only with the native CRON mode allows you to increase the part size and reduce the number of parts a complete backup consists of.

The required settings for this engine are:

Process each part immediately	If you enable this, each backup part will be uploaded as soon as it's ready. This is useful if you are low on disk space (disk quota) when used in conjunction with Delete archive after processing. When using this feature we suggest having 10Mb plus the size of your part for split archives free in your account. The drawback with enabling this option is that if the upload fails, the backup fails. If you don't enable this option, the upload process will take place after the backup is complete and finalized. This ensures that if the upload process fails a valid backup will still be stored on your server. The drawback is that it requires more available disk space.
Delete archive after processing	If enabled, the archive files will be removed from your server after they are uploaded to SugarSync.
Username	The username you use to connect to your WebDAV server
Password	The password you use to connect to your WebDAV server
WebDAV base URL	The base URL of your WebDAV server's endpoint. It might be a directory such as `http://www.example.com/mydav/` or even a script endpoint such as `http://www.example.com/webdav.php`. If unsure please ask your WebDAV provider for more information.

Warning

If the base URL of your WebDAV server's endpoint is a directory (almost always) you MUST use a trailing slash, e.g. `http://www.example.com/mydav/` (correct) but not `http://www.example.com/mydav` (WRONG!)

Directory	The directory inside the WebDAV folder where your files will be stored in. If you want to use subdirectories, you have to use a forward slash, e.g. `/directory/subdirectory/subsubdirectory`. You may use the same variables used in archive naming, e.g. [HOST] for the site's host name or [DATE] for the current date.

Warning

You MUST always use a directory. Most WebDAV servers, e.g. Box.com, allow you to use the root directory which is denoted by `/` (a single forward slash). Other WebDAV servers, such as CloudDAV, DO NOT allow you to use the root directory. In this case you MUST use a non-empty directory, e.g. `/backups` for the upload to WebDAV to work at all.

3.3.5.21. Upload to Box.net / Box.com

As of Akeeba Backup 3.10.1 you can use the Upload to WebDAV option to upload your backup archives to Box.com. You will need to use the following parameters:

Username	Your box.com email address

Password	Your box.com password
WebDAV base URL	`https://dav.box.com/dav`

For more information please check the official Box.com page explaining the Box.com over WebDAV feature: https://support.box.com/hc/en-us/articles/200519748-Does-Box-support-WebDAV-

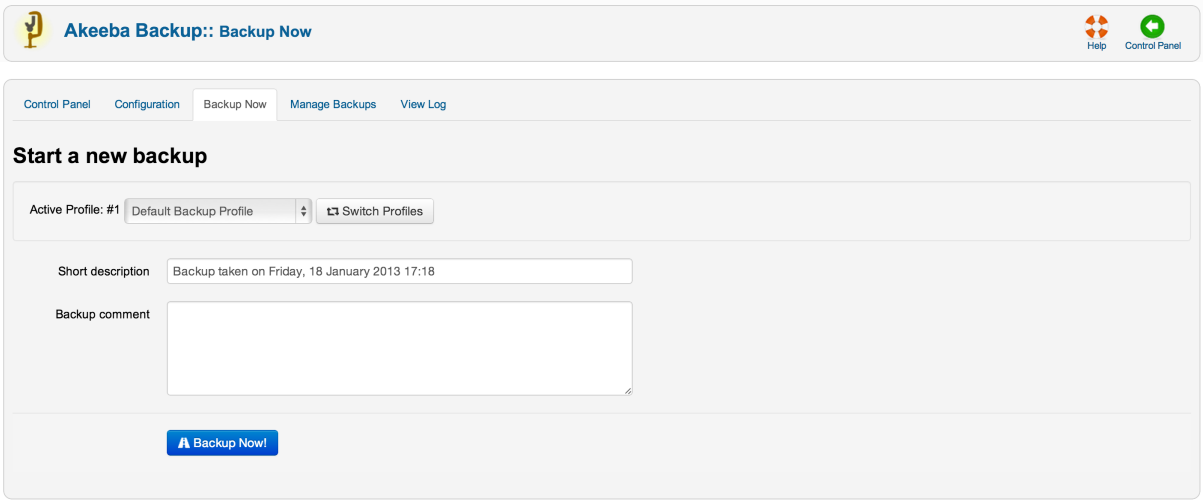
Important

Due to limitations in the Box.com implementation of WebDAV we strongly recommend using a Part Size for Split Archives smaller than 50Mb at all times.

3.4. Backup now

Before we go on describing the Backup Now page, we have to discuss something important pertaining to the overall backup and restoration process. In order for the restoration to work properly, the original site must have a readable and valid configuration.php on its root. This means that a 'trick' many webmasters use, that is providing a configuration.php which includes an off-server-root PHP file, is incompatible with the restoration procedure. If the 'trick' has been effective on the original site, the installer will have blanks in its options and if the user proceeds with the restoration/installation procedure the site will not work as expected, as crucial options will have the default or no value at all!

Backup start



That being said, the initial backup page lets you define a short description (required) and an optional lengthy comment for this backup attempt. This information will be presented to you in the backup administration page to help you identify different backups. The default description contains the date and time of backup. Both the description and comment will be stored in a file named `README.html` inside your archive's `installation` directory, but only if the backup mode is full backup.

Since Akeeba Backup 3.1.b1 both the description and the comment support Akeeba Backup's file naming "variables", e.g. `[SITE]`, `[DATE]` and `[TIME]`. These variables are documented in the Output Directory configuration option's description. It goes without saying, but these variables can also be used in the case of automated backups, e.g. CRON-mode backups.

There are two more fields which may be displayed on this page:

- JPS Password. When you are using the JPS (encrypted archive) format the contents of the archive are encrypted using the AES-256 algorithm and this password. In order to extract the archive you will need to enter this password. If you had entered a default password for JPS files in the Configuration page this field is pre-filled with that password.

Important

The password is case-sensitive. ABC, abc and Abc are three different passwords! Also note that the password is non-recoverable. If you lose or forget your password you will not be able to extract your JPS archive.

- ANGIE Password. As of Akeeba Backup 3.7.5 the ANGIE installer (embedded in the backup archive) allows you to password protect it. This means that you will have to enter this password before you can restore your site. This feature is designed to prevent unauthorised users from "stumbling" on your site while it's still undergoing restoration and copy your database passwords or obtain other information about your site.

Important

The password is case-sensitive. ABC, abc and Abc are three different passwords! Unlike the JPS password, setting an ANGIE password will not prevent anyone from extracting the archive and looking at its contents.

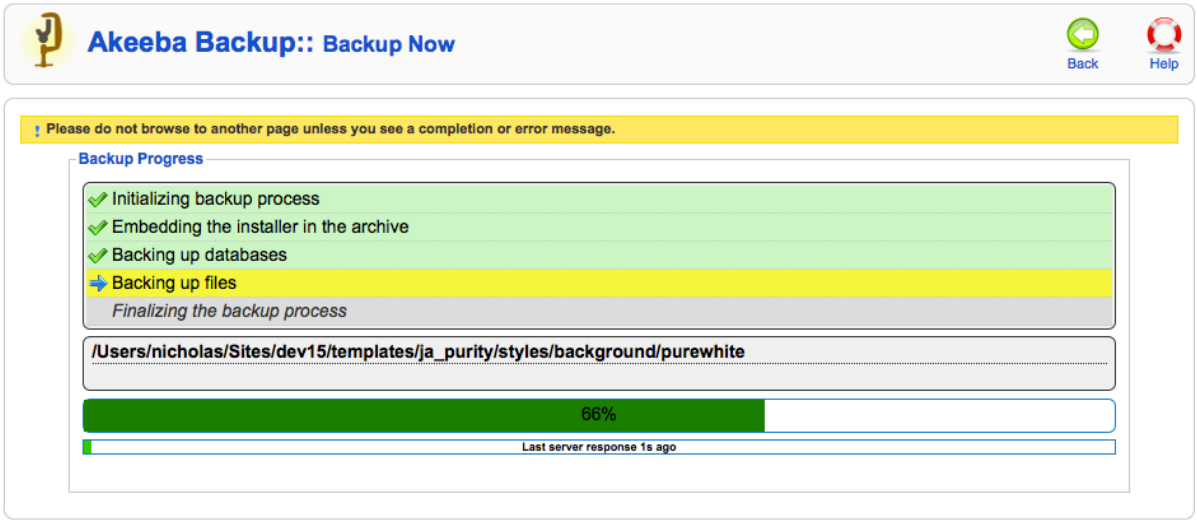
Whenever you are ready to start the backup, just click the Backup Now button. Do note that above the description field, there might be one or more warnings. These are the same warnings appearing in the Control Panel's right-hand pane and act as a reminder.

Important

Default output directory is in use *is not an error message*! It's just a reminder that the default output directory is a well known location on your site. In theory, a malicious user could figure out the name of the backup archive and download it directly over the web. In order to deter that, Akeeba Backup places a .htaccess file (compatible with virtually all Apache installations) and a web.config file (compatible only with IIS 7) to deter that. If you are using a host which doesn't support the directives of those two files, the contents of that directory may be inadvertently available over the web to malicious users. If in doubt, ask your host. Do not ask us. We can't know this information; we haven't set up your host's server.

Our recommendation: consult your host about the proper way to create a backup output directory above your site's root and make it writable by PHP. Then, use that directory as the Output Directory in all of your backup profiles. This method offers the greatest degree of protection.

Backup progress page



Once you click on the Backup Now button, the backup progress page appears. You must not navigate away from this page or close your browser window until the backup is complete. Otherwise, the backup process will be

interrupted and no backup file will be created (or you'll end up with an incomplete backup file). Akeeba Backup disables the Joomla! menu during backup to prevent accidentally switching to a different page.

The backup progress page consists of a large pane. The top section of the pane lists the steps Akeeba Backup has to take in order to complete your backup. Steps in gray background have not been dealt with yet. Steps in green background, featuring a green check mark on the left-hand side, have been successfully completed. The step in blue background is the one being currently processed.

Below that, you will find two lines. The first line will show you which table or directory has **been backed up until now**. This is very important. When the backup crashes, it hasn't crashed on the table or directory you see on the screen. In fact, you can be sure that this table/directory has been *successfully* backed up. The real problem appears in the log file and this is why we are adamant in asking for a backup log to be posted with your support request. The Substep line below is normally used for messages of lesser importance, such as noting the percentage of a table already completed (especially useful when backing up huge tables) and the name of the archive part which was processed by a data processing engine.

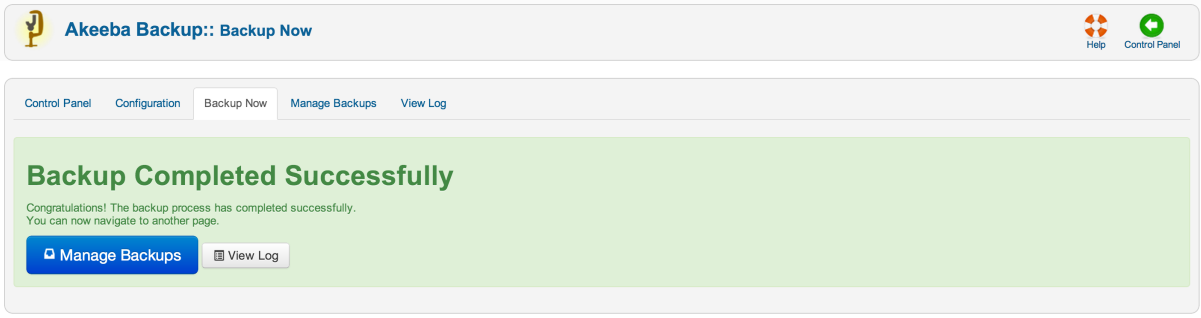
The big bar is the overall progress bar and displays an *approximation* of the backup progress. Do note that during file backup you may see this bar jump back and forth. This is normal and, please, do not report it as a bug. It is exactly how it is supposed to behave. The reason is rather simple. Before your site is backed up, Akeeba Backup doesn't know how many files and directories it contains. As a result, it tries to do an educated guess and display an approximate backup progress. Guesswork is never accurate, which causes some jumping back and forth. Nothing to worry about, your backup is working without a problem.

The next thing you see is time elapsed since the last server response. This resets to 0 when a new backup step is started. If you see a last server response over 300 seconds –except when the application is uploading your backup archives– you can safely assume that your backup has crashed. Only in this case you should navigate away from the backup page and take a look at the log file for any error messages. Always try different configuration options, especially changing the minimum and maximum execution time, before filing a support request.

Should a minor (non fatal) error occur, Akeeba Backup displays a new Warnings pane with yellow background. This box holds the warnings which have occurred during the backup process, in chronological order. These are also logged in the Akeeba Backup Debug Log and marked with the WARNING label, that is if your log level is at least Errors and Warnings. Usual causes of warnings are unreadable files and directories. Akeeba Backup regards them as minor errors because, even though the backup process can go through, what you get might be a partial backup which doesn't meet your expectations. In case warnings appear on your screen you are advised to review them and assess their importance.

Sometimes your backup may halt with an AJAX error. This means that there was a communications error between the browser and your server. In most cases this is a temporary server or network issue. Depending on your configuration preferences, Akeeba Backup may try to resume the backup after a while. By default, Akeeba Backup will retry resuming the backup at most three consecutive times and after waiting 10 seconds after each error. If the backup cannot be resumed you will receive an error page, at which point your backup has positively failed.

Backup completion page



After the whole process is complete, Akeeba Backup will clean up any temporary files it has created. Akeeba Backup will also clean temporary files and delete incomplete archive files upon detecting a backup failure. Please

note that log files are not removed by default. You will have to go to the Manage Backups page, select the failed backup attempt(s) and then click on Delete Files or Delete to have it remove the log files of failed backups.

By that point, your site backup file has been created. You can now navigate out of the backup page and possibly into the backup administration page, clicking on the handy button which appears below the backup completion message.

Frequently asked questions

Where are my backup files? [https://www.akeebabackup.com/documentation/troubleshooter/abwherearemyfiles.html]

How can I download my backup files? [https://www.akeebabackup.com/documentation/troubleshooter/abwherearemyfiles.html]

I got an "AJAX loading error" when backing up. What should I do? [???]

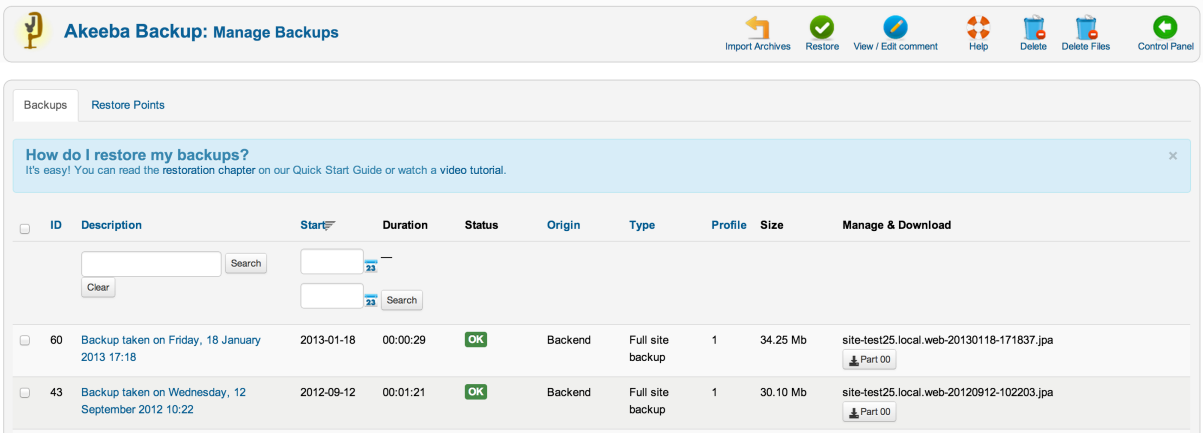
How do I know that my backup archive works? [https://www.akeebabackup.com/documentation/troubleshooter/abtestsupport.html]

What happens if I have a backup problem? [https://www.akeebabackup.com/documentation/troubleshooter/abtestsupport.html]

How do I get support? [https://www.akeebabackup.com/documentation/troubleshooter/abtestsupport.html]

3.5. Manage Backups

Manage Backups



This page is the single place you can review all your Akeeba Backup backup history, as well as administer the backup files. The bulk of the page consists of a standard Joomla!™ list table. Each row represents a backup attempt and displays a whole lot of information:

The check box column
: Clicking the check box on the leftmost cell of a row selects this backup for an operation to be applied to it. Operations are activated by clicking on tool bar buttons. In case of an operation allowing a single row to be selected, the topmost selected row is considered as the sole selection.

Description
: Displays the description you have set when you started the backup. If your backup has a comment attached to it, an info icon will also appear. Hovering your mouse over the info icon will show you a preview of that comment.

  To the left of the description there's an icon indicating the backup origin, e.g. Backend, Frontend, JSON API, CLI and so on. Hover over it to see what each icon means.

Below the description you will see the date and time of the backup. The date is expressed in the user's preferred time zone, as it is set in the User Managment page of Joomla!™ itself.

Note

Backups taken without a logged in user, i.e. remote, front-end and native CRON backups, express the time in the UTC time zone. We can't "fix" that; without a user, Joomla!™ can't reliably report the time zone.

Profile — Displays the numeric identifier (and description, if available) of the backup profile used during the backup. It is possible that since the time of the backup the profile may have been modified or even deleted!

Below it you will see the backup type. It indicates the backup type. A backup type may not be provided if the backup profile has been deleted in the meantime.

Duration — The duration of the backup in hours : minutes : seconds format. This information is not available for failed backups!

Status — Indicates the status of the backup. Hover over the icon to see what it means. It will be one of:

OK — A complete backup whose backup archive is available for download.

Obsolete — A complete backup whose backup archive is either deleted, or was overwritten by another backup attempt.

Note

If you move your backup output directory's location, all your previous backups will appear as "Obsolete", even though you might have moved these backup files as well. This is not a bug. Akeeba Backup internally stores the absolute path to the backup files. When you move the output directory its absolute path changes, so Akeeba Backup is unable to locate the old backup files.

Important

If your host uses MySQL 4.0 the status will always appear as Obsolete and you will be unable to download the backup archive through your browser, as the result of limitations of this *ancient, obsolete and unsupported* MySQL version. You can still use your favorite FTP client to download the backup archives, though.

Remote — Indicates a complete backup which has been uploaded to remote storage (e.g. Dropbox, Amazon S3, CloudFiles and so on), but it is no longer stored on your server. You can fetch the backup archive backup to your server any time (as long as you haven't manually removed the file from the remote storage) in order to restore it, clicking the Manage Remote Files link on the right-hand column.

Note

Not all remote storage engines support fetching back backup archives. Currently, only FTP, Amazon S3, CloudFiles and Dropbox support this feature.

Pending — A backup attempt which is still running. You should not see any such record, unless a backup attempt started while you were loading this page. In this case, you should not navigate to the Control Panel page! Doing so would invalidate

		the backup and wreck havoc. You have been warned! Another reason to see such an entry is a backup attempt which failed with a PHP fatal error, or which was abruptly interrupted (by the user or a PHP error). In this case, you can safely delete the entry and get rid of the backup file as well.
	Failed	A backup attempt which failed with a catchable error condition.
Size	The total size of the backup archive in Mb. If the files are not available on your server, i.e. the record is marked as "obsolete" or "remote", the size appears inside parentheses to let you know that the files are not available for download.	
Manage and Download	Depending on the status of the backup it will show two or more buttons: • Download. Opens a popup which allows you to download the backup archive file(s) directly from your browser. However, this is NOT recommended. The only guaranteed method of downloading your backup archives error-free is using FTP or SFTP in BINARY transfer mode. Anything else has the potential to CORRUPT your backup archives for reasons beyond our control! • Manage remotely stored files. If the file is stored on a remote storage location, e.g. Amazon S3 or a remote FTP server, you will also see this button. Clicking on it will allow you to transfer the files back to your server, download them directly from the remote location or remove them from the remote storage. • Upload to <remote storage name>. If Akeeba Backup failed to upload your backup archive to remote storage you will be shown this button. Clicking it will have Akeeba Backup retry the upload to remote storage. • View Log. If your backup archive has a backup ID you will also see this button. Clicking it takes you to the View Log page to see the backup log file. If the backup status is anything other than OK this button will be grayed over as Akeeba Backup can't guarantee that the log file is present. Hover your mouse over the button to get the Log file ID which you'll need in the View Log page to look for this log file. • Info. Clicking this button tells you if the backup archive is currently present on your server, where to find it (relative to your site's root directory) and what is the name of the backup archive file. This allows you to download the backup archive over FTP/SFTP as discussed above.	

Clicking on the label of each column allows you to sort the backup entries by the contents of that column. By default, Akeeba Backup sorts the records by the time of backup descending, so that the newest backup attempts will appear on top. Below the header there are four filter boxes. The first one allows you to filter by the backup description. The other two allow you to select a date range so that only backups attempted within this date range will be displayed. You can leave either of these boxes empty to allow an open start or end date respectively. The final box allows you to filter by backup profile.

On the top of the page you can find a tool bar with operations buttons. The Delete button will remove the selected backup attempt entries along with their backup archives (if applicable), whereas the Delete Files button will only remove the files (if found on your server). The Restore button (Akeeba Backup Professional only) will run the integrated restoration feature for the selected archive file. This feature can be used to restore your backup archive on the same server you backed up from or even a different server (live transfer of your site to another host!). The Discover and Import Archives (available since Akeeba Backup Professional 3.2) allows you to import any ZIP, JPA or JPS file, located anywhere in your server or Amazon S3, in the Manage Backups (formerly "Administer Backup Files") page in order to restore it on this or any other site.

Note

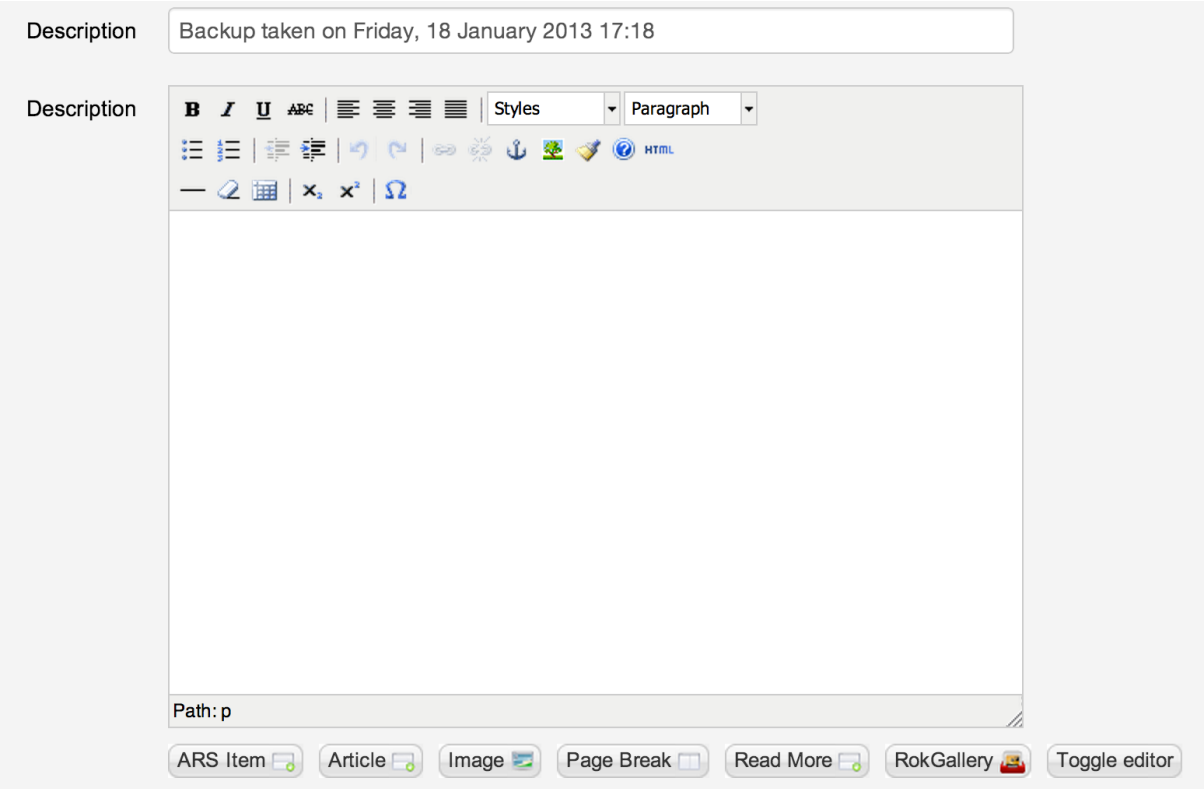
If you are interested in restoring your backup archives and your site is inaccessible or you're using the free Akeeba Backup Core edition, you can use Akeeba Kickstart or Akeeba eXtract Wizard to extract

the archive and restore it on their server. The procedure is detailed in our Quick Start Guide and our Video Tutorials.

Important

Integrated restoration is only supported for Full Site and Files Only backup archives. Trying to use it with any other type of backup files will ultimately result in an error. This feature is available only to Akeeba Backup Professional - the paid version. Users of the Akeeba Backup Core version can follow our video tutorials or Quick Start Guide instructions to easily restore their backups using Kickstart or eXtract Wizard.

Backup description / comment editor



The View / Edit Comment button will open a page showing the description and comment of the currently selected backup row. You can freely edit both the description and the comment on that page and save your changes using the Save & Close button. The same page will open if you click on a backup record's description (appearing as a link).

3.5.1. Integrated restoration

Note

This feature is only available in the Akeeba Backup Professional edition; users of Akeeba Backup Core - and users of the Professional edition when their site is completely inaccessible- can use Akeeba Kickstart or Akeeba eXtract Wizard to extract the archive and restore it on their server. The procedure is detailed in our Quick Start Guide and our Video Tutorials (both found under the Documentation menu item on our site).

Warning

THE INTEGRATED RESTORATION FEATURE MAY DESTROY YOUR SITE IF YOU ARE NOT CAREFUL.

Remember that you are OVERWRITING your site with the one contained in the backup archive. Do not do that on a live site unless it is absolutely necessary, i.e. you have already destroyed something vital in your site and want to revert to a "last known good" state.

As with any backup restoration method, practise on a local testing server first. Don't push your luck by trying a potentially dangerous procedure you are unfamiliar with on a live server. Many sites have been destroyed by human error, augmented by the "bliss of ignorance" effect. Never, ever, under any circumstances, attempt a restoration on a live site unless you are familiar with the procedure and confident of all the steps you take.

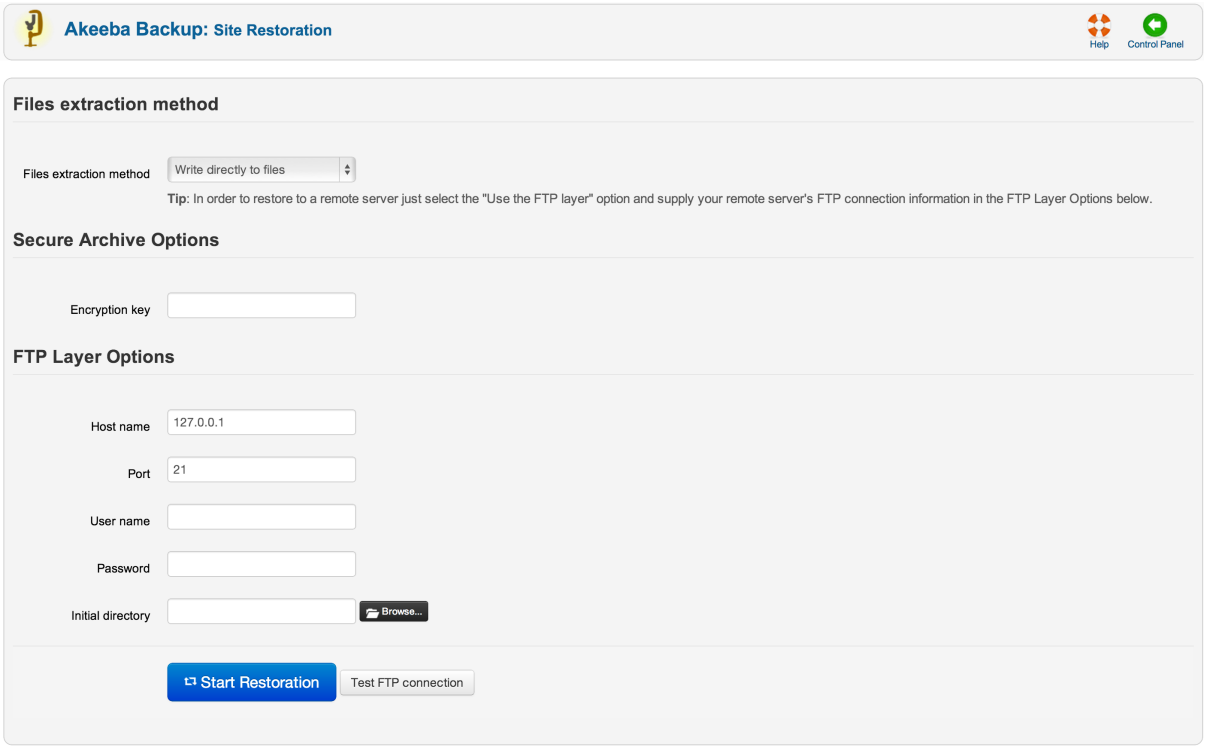
That said, we trust our own software and use it on our sites. Do note that we are extremely familiar with the procedure and extremely careful when doing restorations. This message tries to excessively - if that's ever possible - stress the point that you *must* be careful and that the best method to achieve that is practising on a local testing server first.

The integrated restoration feature allows you to easily restore a previous backup directly on your server, as long as your backup archive still exists on your server of course. The whole idea behind this feature is that it is not necessary to manually download Kickstart, place it in your site's root and move the backup archive from the output directory to the site's root in order to perform the restoration. Instead, the integrated restoration feature takes care of extracting your backup archive directly from the backup output folder into your site's root and then allow you to run the embedded installer (Akeeba Backup Installer) to complete the restoration procedure.

The communication between your browser and the archive extraction script is encrypted with the AES128 (Rijndael) encryption method, using a random key produced as soon as you initiate the restoration of a backup archive. This ensures that a malicious user can't exploit the restoration script to mischievously extract your backup archive in your site's root with the intent to steal your database password. The encryption/decryption algorithm is implemented with standard PHP and Javascript code, eliminating the need for third party cryptography libraries and ensuring that under no circumstances unencrypted data will be exchanged between the browser and the server.

In order to start an integrated restoration begin by going to the Manage Backups page of the component. In that page check the checkbox next to the backup you want to restore and click the Restore button in the toolbar to will run the integrated restoration feature for the selected archive file.

The integrated restoration setup page



When you first start the integrated restoration feature, you are presented with a few settings. The first setting, appearing above the Start Restoration button, determines how the file extraction will be performed. The two available options are:

Write directly to files
: All files will be extracted directly to their final location using direct PHP file writes. If your permissions settings do not allow some files or directories to be created/overwritten the process will fail and your site will be left in a half-restored state.

Use FTP uploads
: Using this method, each file is first extracted to the temporary directory specified by the current profile and then moved to its final location using FTP. This is a "best effort" approach and can work with most servers. Do note that only unencrypted FTP (plain FTP) is supported. If you choose this option, you'll also have to specify the FTP connection settings.

> ## Tip
>
> You can use this option to restore a backup on a different site. Just select this option and provide the FTP connection details to the other site before clicking on Start Restoration.

Hybrid
: This mode combines the previous two in an intelligent manner. When selected, Akeeba Backup will first attempt to write to the files directly. If this is not possible, i.e. due to permissions or ownership of the file or folder being exracted, it will automatically make use of the FTP mode to overcome the permissions / ownership problem. It effectively works around a situation commonly called "permissions hell", where different files and folders are owned by different users, making it extremely difficult to overwrite them. This is a situation which happens very commonly on shared hosting. Therefore **we strongly advise clients on shared hosting environments to use the Hybrid option**.

> ## Note
>
> You MUST supply your FTP information for this mode to have any effect. If you do not do that the Hybrid mode will function exactly as the "Write directly to files" mode.

The default mode is writing directly to files, unless your site's Global Configuration indicates that the FTP layer should be used in which case the Hybrid mode is selected by default.

In the event that a partial restoration happens, your site will be left in a semi-restored state. Trying to access it will pop up the restoration script (ANGIE or ABI). If you want to retry the restoration using different settings, please remove the `installation` directory from your site's root manually, for example using FTP, before trying to access your site's administrator back-end.

If you chose to use the FTP mode, there are some connection settings you have to take care of. Do note that they are filled in with Joomla!'s FTP layer settings by default. Unless you chose not to store your FTP password in Joomla!'s configuration or if you have not configured the FTP layer yet, there is no need to change them. The settings are:

Host name
: The host name of your site's FTP server, without the protocol. For example, `ftp.example.com` is valid, `ftp://ftp.example.com` is *invalid*.

Port
: The TCP/IP port of your site's FTP server. The default and standard value is 21. Please only use a different setting if your host explicitly specifies a non-standard port.

User name
: The username used to connect to the FTP server.

Password
: The password used to connect to the FTP server.

Initial directory
: The FTP directory to your web site's root. This *is not the same as the filesystem directory* and can't be determined automatically. The easiest way to determine it is to connect to your site using your favourite FTP client, such as FileZilla. Navigate inside your web site's root directory. You'll know you are there when you see the file `configuration.php` and directories such as `administrator`, `component`, `language`, `includes`, `cache` and

	`xmlrpc` in that directory. Copy (in FileZilla it appears on the right hand column, above the directory tree) and paste that path in Akeeba Backup's setting.
Test FTP connection	Clicking on this button will tell you if the FTP connection could be established or not. If the connection is not successful you should not proceed with a restoration in FTP mode as it will fail immediately.

The whole process is fully automated, so there is not much to tell you about it. However, you must not that in order for the restoration procedure to work properly you must take care of the following:

1. This feature is directly calling the `administrator/components/com_akeeba/restore.php` script. If you have a server-side protection, i.e. .htaccess rules, or permissions settings which prevent this file from being called directly the process will fail.

   Security note: The restore.php file is of no use to potential hackers. In order for it to work at all, it requires the `restoration.php` file (more on that on the next point of this list) to load. Even then, it expects encrypted data with a key which is not predefined and is only known to the `restore.php` script and the integrated restoration page of Akeeba Backup. As a result, it can't be used as a potential attack vector.

2. Before the restoration begins, Akeeba Backup needs to create the `administrator/components/com_akeeba/restoration.php` file with all the archive extraction setup parameters. It is intelligent enough to use Joomla!'s FTP mode if it is enabled so as to overcome any permission problems, but you are ultimately responsible for ensuring that the permission settings are adequate for Akeeba Backup to create this file.

   If you have disabled Joomla!'s FTP layer, the permissions of the `administrator/components/com_akeeba` directory should be 0777 for the integrated restoration to work, or 0755 on hosts which use suPHP.

   If you are using Joomla!'s FTP layer and it was active when you were installing Akeeba Backup, you'll need to give this directory at least 0744 permissions, but you may have to manually remove `restoration.php` (**but NOT** `restore.php`!!!) after the site restoration is over.

3. When the extraction of the backup archive finishes, you will be automatically forwarded to the Akeeba Backup Installer page on a new tab or window. DO NOT CLOSE THE INTEGRATED RESTORATION PAGE'S TAB/WINDOW! After you have competed the Akeeba Backup Installer process you are supposed to return to the Integrated Restoration page and click on the Finalize button to:

   - remove the `installation` directory from your site's root, and
   - remove the `administrator/components/com_akeeba/restoration.php` setup file to nullify the, already non-existent, potential risk of a malicious user abusing this script.

4. If you are restoring to a remote server, the previous step will result in a 404 page. Just point your browser to `http://`*`www.yoursite.com`*`/installation/index.php` (where www.yoursite.com is the domain name of the site you are restoring to) to access the restoration script. After finishing the restoration procedure, do NOT click the Finalize button. Instead, use your favorite FTP client to remove the `installation` directory from the site you were restoring to and rename any `htaccess.bak` file back to `.htaccess`.

3.5.2. Manage remotely stored files

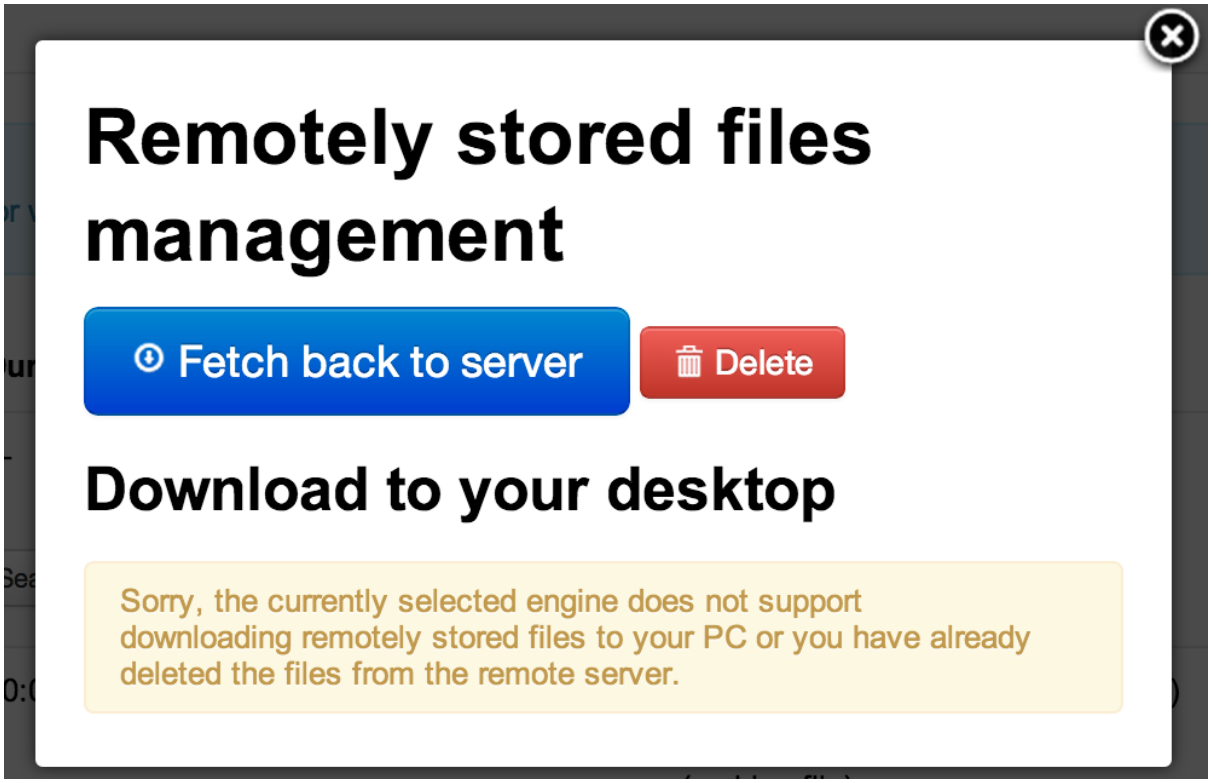
Note

This feature is only available in the Akeeba Backup Professional edition

Since Akeeba Backup 3.2 you have the option to manage backup archives stored in a remote storage location, for example Amazon S3 or a remote FTP server. You can do that by clicking on the Manage remotely stored files link on the far right of supported backup records in the Manage Backups (formerly "Administer Backup Files") page. Do note that, if you have upgraded from Akeeba Backup 3.0.x or 3.1.x, backup records created by older versions of the software do not support this feature. Clicking on that link opens a lightbox (modal dialog) with the options compatible with your backup archive.

Please note that not all of the following features may appear in the dialog. It depends on the remote storage engine used for the backup record. All options currently appear only for files stored on Amazon S3 and remote FTP.

The "Manage Remotely Stored Files" page



The Fetch back to server button will automatically download the backup archive from the remote location and store it again on your server. This allows you to easily import backup archives stored on a remote location back to your server's storage so that you can easily restore them on the same or a different site. If you are using S3, please make sure that the user credentials you have supplied have enough privileges for the files to be downloaded (i.e. they don't grant write-only access to the bucket). Also make sure that you have adequate free disk space on your server for the operation to complete.

The Delete button will permanently delete the archives from the remote storage. There is no confirmation. Once you click this button, your remotely stored files will be removed.

Finally, there are links under the Download to your desktop header. Clicking on them will instruct your browser to download the respective backup archive's part directly to your PC. Currently, only Amazon S3, CloudFiles, Dropbox and remote FTP support this feature. Do note that the backup archives are transferred directly from the remote storage to your PC. They are not stored to your site's server. If you want to store them to your server, use the Fetch back to server button instead.

If none of the above options are available, Akeeba Backup will display an error message. In that case, just close the modal dialog.

After finishing your remote files administration, please close the modal dialog by clicking on the X button on its top-right corner and *reload the Manage Backups (formerly "Administer Backup Files") page*. Until you reload the page the changes you made WILL NOT be visible. This is not a bug, it is the way it is meant to be.

3.5.3. Discover and import archives

Note

This feature is only available in the Akeeba Backup Professional edition

Sometimes you may have accidentally deleted a backup record from the Manage Backups (formerly "Administer Backup Files") page, or simply want to restore a backup file taken from another site. Normally, the only way to do that is to upload the archive file and Kickstart to your site and launch the restoration process from there. However, some users insisted that they are better off doing that from inside Akeeba Backup itself. In order to accommodate for their needs, we introduced the Discover and Import Archives features in Akeeba Backup 3.2.

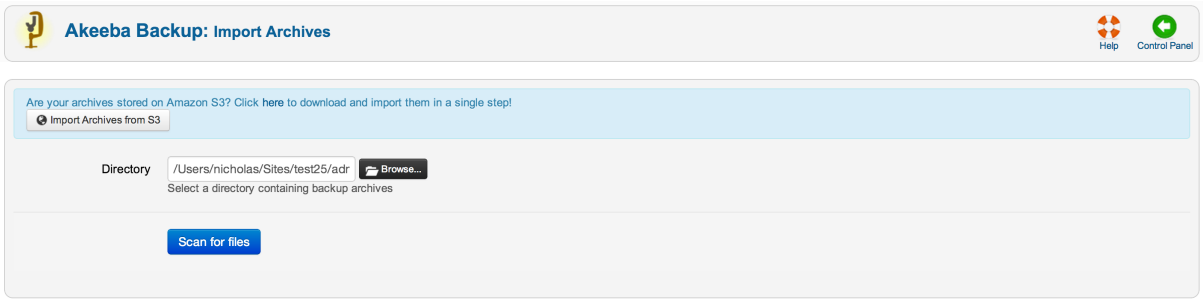
This feature allows you to automatically find and import archives stored anywhere on your account. This means that you can upload backup archives anywhere in your site's folder structure, or even on a private off-site directory and Akeeba Backup will be able to import them. All backup archives are imported as backup records of the default backup profile (profile with ID #1) and can be restored just like any other backup archive.

In order to launch this feature, go to the Manage Backups (formerly "Administer Backup Files") page and click on the Discover and import archives button on the toolbar. A new page appears which lets you select a directory.

Tip

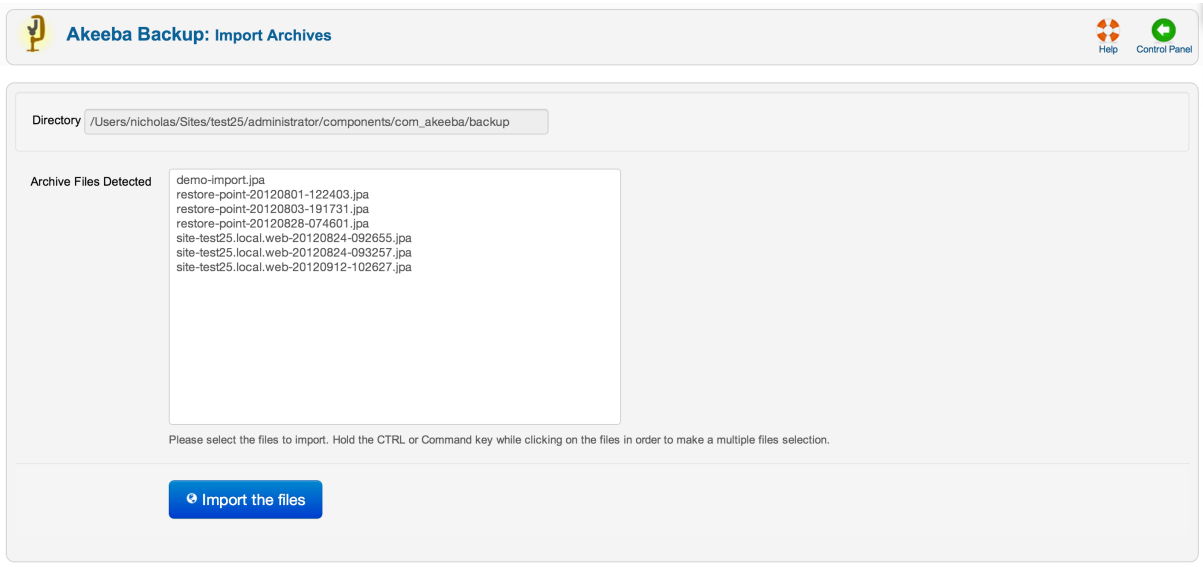
Since Akeeba Backup 3.4.a1 you have the option to import archives from Amazon S3. Click the link directly above the directory selection box. It will take you to a slightly different page where you can enter the connection credentials to your S3 account and allow you to browse for ZIP and JPA files to import.

The "Discover and Import archives" page



Use the Browse... button to open an interactive folder browser in a modal dialog. Navigate to the directory which contains the uploaded backup archives and click on the Use button. The dialog closes and you can now click on the Scan for files button to let Akeeba Backup search for backup archives inside that directory. You are presented with a new page, listing the discovered backup archives.

Importing discovered archives



Select the backup archive you want to import by clicking on them. If you want to select multiple files, Control-click (Windows, Linux) or Command-click (Mac OS X) the archive you want to import. After that, click on the Import

the files button. After a short while Akeeba Backup takes you back to the Control Panel page with a message that the import operation completed successfully. You can now click on the Manage Backups (formerly "Administer Backup Files") button to view the newly imported backup archives. You can now download or restore the imported backup archives.

3.6. View Log

The View Log option allows you to download or view the output from the most recent backup operation attempted on each origin. This information may be useful in diagnosing problems if you are having a problem completing a backup.

Selecting an origin

The first page allows you to select an origin. Backups attempted using the Joomla! administrative back-end belong to the Backend origin. The Frontend origin applies to backup archives taken with the front-end backup method (also referred to as legacy CRON in our documentation) or using the `akeeba-altbackup.php` script. The Command Line origin applies only to backups taken with the `akeeba-backup.php` script file of the Professional release. The XML-RPC origin applies to backups taken with Akeeba Remote Control up to version 3.x (this is now obsolete and your should never see it on a site powered by Joomla! 1.6 or later – yes, it's so obsolete!). Finally, the JSON API origin applies to backups taken with a remote client such as Akeeba Remote Control 4.x (obsolete since May 2011), Akeeba Remote CLI and compatible third party products and services.

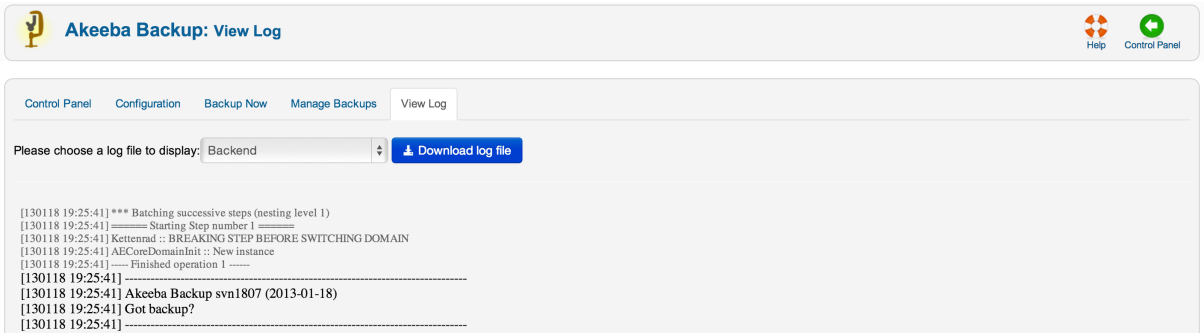
Since Akeeba Backup 4.0 some backups may have a Backup ID which allows them to have a separate log file. In this case you will see something like "Backend (id1234)" in this drop-down. This means that the log file in question is for the back-end backup with Backup ID "id1234". If unsure, go to the Manage Backups page and click the View Log button from there to open the correct log file for the backup attempt you want to examine.

Tip

If you just tried taking a backup using Akeeba Backup's interface, please select the Backend option from the drop-down.

This takes you to the View Log visualization page.

View Log



If you wish to ask for support, you must download the raw log (a text file). Just click on the download button above the log viewer. Do not copy and paste the text appearing in the log viewer. If you do that, you will lose a day as we're going to tell you to download the raw log, ZIP it and attach it to your next post. Once again, please DO NOT copy and paste text. We absolutely and beyond any doubt need the raw log file in order to support you. Help us help you so that we can solve your issues as soon as possible.

Warning

When asking for support, make sure that the Log Level was set to "All Information and Debug" in the Basic section of the Configuration page *before* backing up. Otherwise the log will be useless in supporting you.

The bulk of this page is the log visualization box. Each line is preceded by a time stamp, in the format YYMMDD hh:mm:ss (that's year, month, date with two digits, a space and time in 24-hour format). Each line is colour coded, for your convenience. Debug information is in smaller, grey type. Normal information is in black type. Warnings appear in bold yellow letters. It is important to read them as they convey information about skipped directories or other things that will be missing from the backup archive. If any errors occurred, these appear in bold red type.

Whenever you report bugs, all of this information is absolutely necessary. In order to reveal as little sensitive information as possible, whenever a file path has to be logged, your site's root folder is replaced with the string '<root>'. Keep this in mind when reading warnings and errors.

4. Include data to the backup

Note

This feature is available only in Akeeba Backup Professional, the paid version of our component

By default, Akeeba Backup automatically includes the whole database of your Joomla!™ installation as well as all the files under your site's root in the backup set. Sometimes you want to include a different database - for example, one used by your non-Joomla!™ newsletter software - or files you have placed above your site's root for increased security. Akeeba Backup Professional can cope with that need by providing you with handy data inclusion filters.

4.1. Multiple Databases Definitions

Note

This feature is available only in Akeeba Backup Professional, the paid version of our component

Sometimes your site grows beyond Joomla!. A forum, a torrent tracker, a custom script... Some of them get to be installed in a database of their own, not as tables in the same database as the one Joomla! is using. If you really want to take a full site backup, you really need these databases backed up as well. The solution to this is the Multiple databases definitions option of Akeeba Backup. You can define an unlimited number of additional MySQL, PostgreSQL, SQL Server and Windows Azure SQL databases which will get to be backed up (and restored!) along with your regular Joomla! database.

Warning

Do not use this feature to add your site's database. It is automatically added anyway. Doing so **will cause errors during the restoration of your site**! You have been warned. Do not seek support for this kind of issues.

Warning

Do not confuse the term "database" with your Joomla!™ tables. It is possible that a single *database* contains tables for the current Joomla!™ site, tables from a standalone photo gallery script, tables from another Joomla!™ site on the same server (e.g. a subdomain), tables from a standalone PHPList installation and so forth. As far as Akeeba Backup is concerned, all of those tables exist **in the same database**. Unless you tell it otherwise, it will backup ALL tables of the database.

A common misconception is that if you want to also backup a subdomain running on Joomla!™ and having its tables inside the same database as the main site, you should add its database a multiple database definition. **DO NOT DO THAT, IT WILL MAKE THE RESTORATION FAIL**! After all, Akeeba Backup already backs up those tables. Why should you have to back them up a second time?

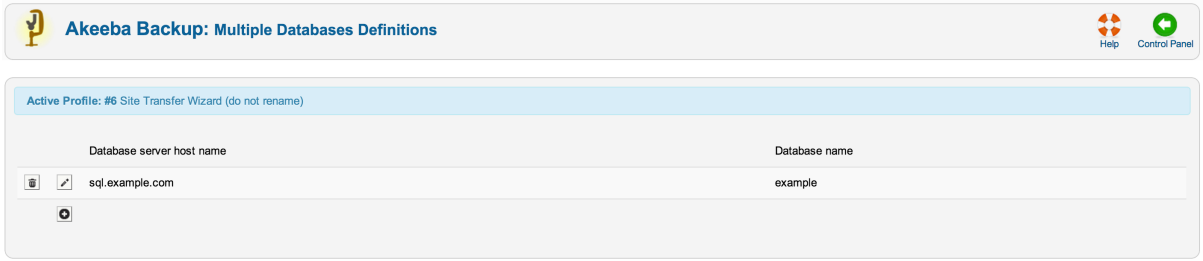
Warning

If you add an empty database (one which has no tables) it will result in backup errors!

Note

The settings on this page are defined *per profile* . Make sure you have selected the desired profile in the Control Panel page.

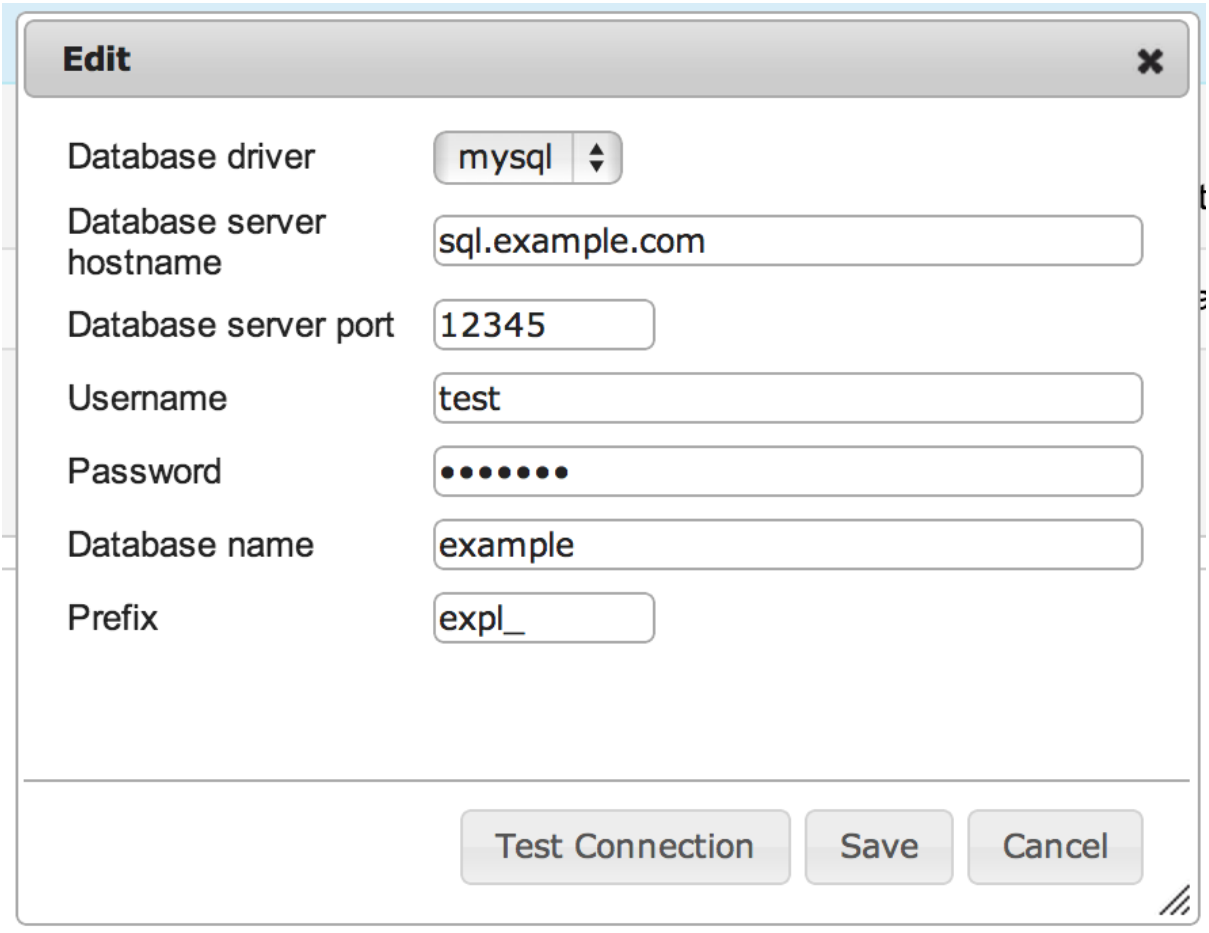
Multiple Databases Definitions



At first, you are presented with a grid view, listing all database definitions. On the left of each entry, there are two icons:

- **The trashcan**. Clicking on this icon will remove the current database definition from the backup set.
- **Pencil** or **Add**. Both will open the database definition editor: the former to edit the database definition, the latter to create a new one.

Multiple Databases Definitions - The editor



The database definition editor opens as a dialog box inside the multiple databases definitions page. The options you can select for each database are:

- **Database driver**. You can select which database driver Akeeba Backup will use to connect to the database. Your options are:
  - **MySQLi**. This is an improved MySQL 5 connection driver. We recommend using it for MySQL databases.
  - **MySQL**. This is the regular MySQL connection driver for PHP. It has the widest compatibility, but the lowest performance.
  - **PostgreSQL**. Connection to a PostgreSQL database.
  - **SQL Server**. Connection to a Microsoft SQL Server database. You must be running your site on a Windows server and have the Microsoft-supplied "sqlserver" PHP extension installed.
  - **Windows Azure SQL**. The same thing as "SQL Server". Windows Azure SQL databases are, in fact, Microsoft SQL Server databases running in a remote machine. You must be running your site on a Windows server and have the Microsoft-supplied "sqlserver" PHP extension installed.
- **Database server hostname**. The host of your database server. Usually it's `localhost`, but many hosts use something different. If in doubt, ask your host.
- **Database server port**. Leave it blank, unless your host has told you to use a non standard port for connecting to his database server.
- **Username**. The username of the database user needed to connect to the database.
- **Password**. The password of the database user needed to connect to the database.
- **Database name**. The name of the database you are connecting to.
- **Prefix**. The prefix used in the table name's prefixes.

Important

MAJOR PITFALL: Please do not leave the Prefix field blank if you intend to use the Database Table Exclusion feature to exclude tables or table data of this extra database from the backup. If you don't want to use a real prefix, please use a "fake" prefix, e.g. `thisIsAFakePrefix_` , to keep the Database Table Exclusion feature happy and functional.

Warning

Some hosts use your account name as a prefix for the database and username. **This is not the same as the Prefix setting above!** In fact, you have to incorporate that account prefix in your database and username values. For example, you're hosted under the account name `foobar` and you create a database `mydata` and a user `myuser`. Your host displays a prefix `foobar_` on the left of the edit boxes where you entered the database and user names. This means that your REAL database name is `foobar_mydata` and your real username is `foobar_myuser`. This is especially true for accounts hosted in cPanel and Plesk powered hosts. It goes without saying that your password doesn't take a prefix!!! Don't laugh, this question has been already asked in the forum.

If in doubt, contact your host. We can't guess the right values for you because we are neither your host nor your host's client (that is, you). If you ask your host to give you the connection information to your database, they must be able to do so.

When you think you have all the connection information ready, click on Test Connection. This will check all settings except the Prefix. If the connection test succeeds, it will inform you:

Same goes if it fails:

If your connection works properly, it's time to save your changes by clicking the Save & Close button. The top panel will briefly display a "loading" message and the dialog box will go away. That was it, your extra database definition is now saved.

4.2. Off-site Directories Inclusion

Note

This feature is available only in Akeeba Backup Professional, the paid version of our component

More often than not, seasoned web masters prefer to place file repositories outside the site's root (usually, outside the web server's root as well!) in order to deter potential crackers and "leechers" from having direct access to those files. Such repositories can include downloads, image galleries, media (audio and video) or controlled access documents files. As you know, Akeeba Backup Core will only backup file under the site's root, which made these files impossible to backup. Well, it's possible with Akeeba Backup Professional.

Using the off-site directories inclusion, Akeeba Backup can be instructed to look for files in arbitrary locations, even if they are outside the site's root (hence the name). All the directories included with this filter will be placed in the archive as subdirectories of another folder, in order to avoid directory name clashes. We call this folder the "virtual folder", because it doesn't physically exist on the server, it only exists inside the backup archive.

For example, if you want to backup an off-site directory named `images` , if we weren't using the virtual folder it's contents would end up being backed up (and subsequently restored!) inside the Joomla! `images` directory. This is something you'd like to happen. If your virtual folder is called `my_offsite_includes` , this directory would end up being backed up as something like `my_offsite_includes\1-images` . Notice the number and the dash before the actual directory name? This is a smart feature which allows you to backup many directories of the same name. You could, for instance, backup two directories named `images` , confident that there would be no name clash inside the archive.

Since keeping track of these folders is a pain, Akeeba Backup includes a `readme.txt` text file inside the virtual folder which tells you which backed up folder corresponds to which physical folder, making it easy for you to restore these directories to their rightful place.

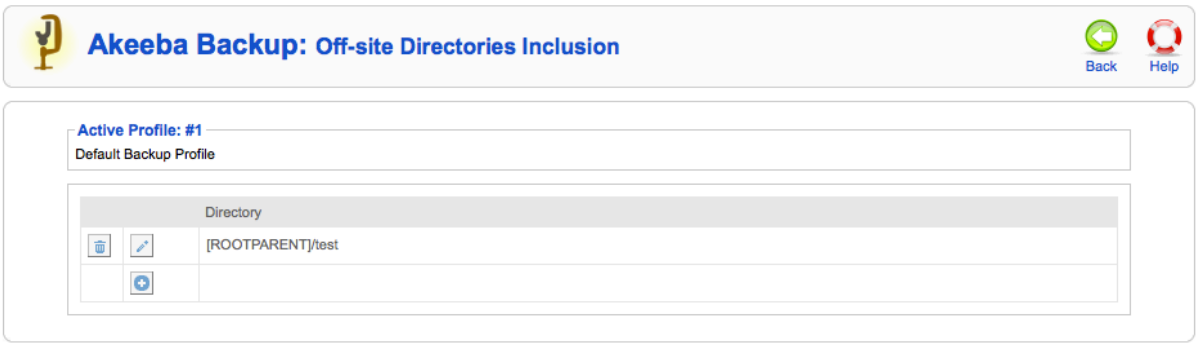
Important

Akeeba Backup *will not* automatically restore the off-site directories to their original location. Since Akeeba Backup is meant for backing up, restoring and *migrating* sites to another host we chose not to automatically restore off-site directories, as this would break the migration process. A future version of Akeeba Backup might address this issue more elegantly. We are open to suggestions!

Warning

Under no circumstances should you add your site's root as an off-site directory inclusion! Akeeba Backup already adds the contents of your site's root to the backup set without any manual intervention. If you manually add this directory you will be backing up the same files twice, bloating your backup size - which could in turn lead to backup problems, such as running out of disk space.

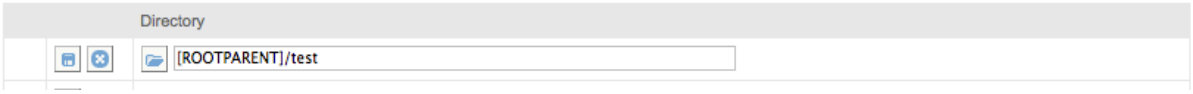
Off-site Directories Inclusion



At first you are presented with a grid view, listing all the off-site inclusions you may have already added. Next to each row and on the left hand side of it you will find two icons:

- **The trashcan**. Clicking on this icon will remove the current directory definition from the backup set.
- **Pencil** or **Add**. Both will toggle the row to edit mode: the former to edit the directory definition, the latter to create a new one.

Off-site Directories Inclusion - Edit mode



When a row enters the edit mode, the pencil icon changes to two different icons:

- **The diskette**. Clicking on this icon will save any changes you have made.
- **Cancel**. Clicking it will abort any changes you have made.

You will also observe that the path to the external directory has also turned to an edit box with a folder icon on its left. You can type in the absolute path to the external directory using the edit box, or click on the folder icon to launch a visual folder browser, much like the one you use to select an output directory in the component's Configuration page. If you choose to use the edit box, you can use the following variables:

- **[SITEROOT]** is the absolute path to your site's root
- **[ROOTPARENT]** is the absolute path to your site root's parent directory, i.e. one level above your site's root.

To the right of the directory you will see another field called Virtual Directory. This is the name of the subdirectory where Akeeba Backup stores the files and folders of these off-site directory's files. Normally, the subdirectory is placed inside the virtual directory for external files, as defined in your backup profile's configuration. If you do not enter a directory name Akeeba Backup will use a predetermined name. This name is a random value followed by a dash and the name of the off-site directory you are defining.

Sometimes you want to include off-site files directly inside the archive's root. Two very useful cases are overriding your regular configuration.php file with another one –presumably one tuned for use on your dev site– as well as overriding files in the installation directory, for example in order to customise the appearance of the installer.

In those cases you don't want the off-site files to be included inside the virtual directory for off-site files. With Akeeba Backup 3.7.5 and later this is very easy to accomplish. Just set the Virtual Directory to a single forward slash (it's this character: `/`) and Akeeba Backup will copy the off-site files inside the archive's root.

5. Exclude data from the backup

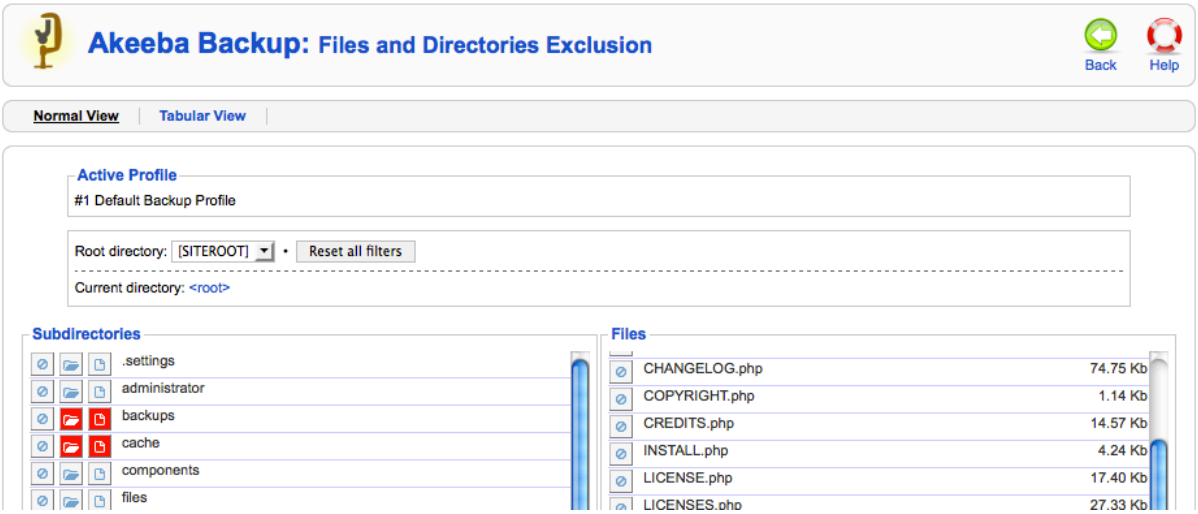
More often than not you have data on your site you don't want to include in the backup set. This can be host-specific directories (e.g. `cgi-bin`, `stats`, etc), log files, temporary data, an huge but immutable collection of large media files, click tracking tables, download log database records and so forth. The exclusion filters allow you to fine tune what should be left out of the backup set.

5.1. Files and Directories Exclusion

Ever had a file in your site's root put there by your host? Or how about that 200Mb video file in the media directory you don't want to backup? If you need to exclude just a few files here and there but let the other files in the directory be backed up, you can use this filter. Or, let's say you have a downloads folder with a size of 10Gb you don't want to backup every time. Or, maybe, your host saves Apache logs in your site's root so that they can be accessible by the provided analyser script. Possibly, you have another script (for example, a forum, a torrent tracker, you name it) in a subdirectory of your site's root - or even buried deeper in the directory structure - that you don't want to backup. Anyway, you need to exclude the contents of a directory from your backup. The Files and Directories Exclusion filters are just right for you.

Before we begin our discussion regarding the operation mode of this filter, you have to know some automatic filters put in place by Akeeba Backup. It will automatically exclude your site's temp-folder, the "cache" directory on your site's root as well as all files and directories inside the Akeeba Backup's output directory. This means that you should **never, ever use a folder as your backup output directory if you intend to backup the contents of that folder**.

Files and Directories Exclusion - Normal view



The normal view of this page consists of three discrete areas.

The top area contains the component and page names and two links to switch between the normal and the tabular view modes.

The middle area contains two interface elements:

- The Root Directory drop-down menu. Akeeba Backup can define filters for the site's files or for each of the off-site directories separately. The default selection, [SITEROOT], contains all filters pertaining to the main site's files. If you have defined off-site directories, you can select the appropriate directory from the drop-down list in order to define filters for that directory.

- The Current directory bread crumb list. It shows the current path relative to the Root directory above. Clicking on a subdirectory allows you to quickly navigate to it.

Below that, there is a button to Reset all filters. Clicking it will remove all Files and Directories Filters, for all of the current root's subdirectories. This is useful in case you have messed up with the filters a lot and you need a quick way to revert to the factory default settings.

The lower area consists of two panes. Each pane contains rows with icons and text. The icons represent an exclusion type and can have three states: on (yellow background), off (white background), or force enabled (red background). You can toggle between the on and off states by clicking on the icon. The force enabled state means that this exclusion type is active (on) and forcibly enabled by another feature of Akeeba Backup, such as the automatic exclusions discussed above, the regular expressions filters or a programmatic filter (plug-in) by a third-party developer.

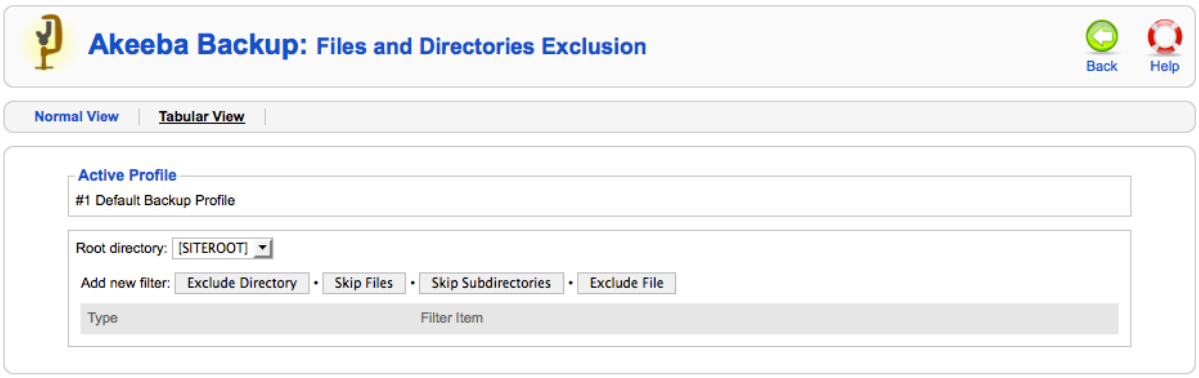
The left hand pane is a list of subdirectories of the Current directory. Each row consists of:

- **Exclusion**. When enabled, the entire directory will be skipped from the backup set. It will be as if this directory never existed on your server.
- **Skip subdirectories**. When enabled, the subdirectories of this directory will be skipped from the backup set. It will be as if this directory's subdirectories never existed on your server.
- **Skip files**. When enabled, the files inside this directory will be skipped from the backup set. It will be as if the files inside this directory never existed on your server.
- The directory name. Clicking on it will load the contents of this directory in both panes and will make this directory current.

The right hand pane is a list of files contained inside Current directory. Each row consists of:

- **Exclusion**. When enabled, the file will be skipped from the backup set. It will be as if this file never existed on your server.
- The file name.
- The file size. It will be expressed in the unit which is more convenient, i.e. bytes, KB, MB or GB. This enables you to quickly pick very large files within your site, which are usually the ones you'd like to exclude from the backup set.

Files and Directories Exclusion - Tabular view



When you click on the Tabular View link, the page radically changes format. Instead of browser panes, you now have a grid.

On the top side of the grid you have the Add new filter buttons:

- **Exclude directory**. Completely skips backing up the given subdirectory.
- **Exclude file**. Completely skips backing up the given file.

- **Skip subdirectories**. Skips backing up all the subdirectories inside the given directory.
- **Skip files**. Skips backing up all the files inside the given directory.

Each line of the grid displays the following information:

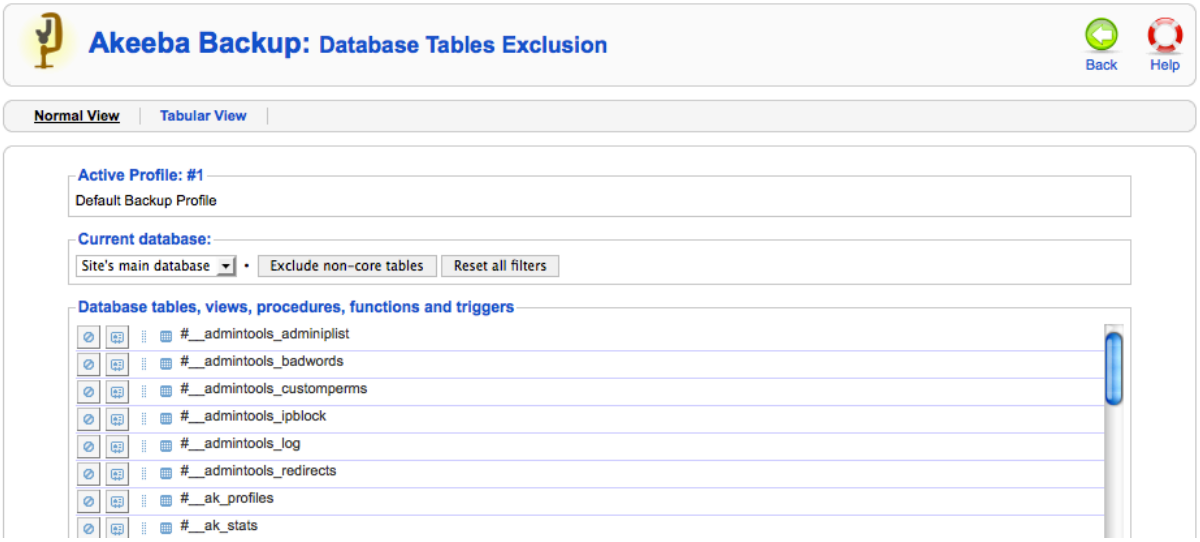
- **The filter type**. It can be one of:
  - **Exclude directory**. Completely skips backing up the given subdirectory.
  - **Exclude file**. Completely skips backing up the given file.
  - **Skip subdirectories**. Skips backing up all the subdirectories inside the given directory.
  - **Skip files**. Skips backing up all the files inside the given directory.
- **Trashcan**. When you click it, the filter row will be removed.
- **Pencil**. When you click it, the row switches to edit mode
- The **filter item** itself. It is the relative path to the directory or file which the filter row applies to. The path is relative to the Root directory displayed on the selection box on top.

When you click on the pencil icon, the filter item becomes an edit box. You can type in the new relative path and then click outside the edit box, or press Tab on your keyboard, to immediately save the changes. There is no way to undo your changes.

5.2. Database Tables Exclusion

Sometimes you can have multiple sites installed in the same database, a common situation with sub-domains on cheap hosts who allow only one database per account. Some other times you have installed a forum, a torrent tracker or whatever on a subdirectory of your site and it has created tables in your site's database. Now it is possible to exclude these tables using the Database tables exclusion feature.

Database Tables Exclusion - Normal View



The normal view of this page consists of three discrete areas.

The top area contains the component and page names and two links to switch between the normal and the tabular view modes.

The middle area contains the Current Database drop-down list. Akeeba Backup can define filters for the site's main database or for each of the extra database definitions separately. The default selection, Site's main database,

contains all filters pertaining to the main site's database, i.e. the one your Joomla!™ site runs on. If you have defined extra databases, you can select the appropriate database from the drop-down list in order to define filters for that database.

The middle area also contains two quick buttons:

- **Exclude non-core tables**. This option automatically filters out the tables whose name doesn't begin with your site's prefix. These are usually tables which do not belong to the current Joomla! installation. However, be warned of the major pitfall! If you host many Joomla! installations on the same database you'll have to use this option *every time* you add a new extension on any of the other Joomla! sites. Alternatively, you can use the Regular Expressions Database Tables feature of the Professional edition which can be set up to automatically deal with such installations.
- **Reset all filters**. Clicking this button will delete all database table filters.

The lower area consists of a single pane, showing the contents of the database: tables, views, triggers, stored procedures and functions. Each row represents one database entity and consists of icons and text. The two leftmost icons represent an exclusion type and can have three states: on (yellow background), off (white background), or force enabled (red background). You can toggle between the on and off states by clicking on the icon. The force enabled state means that this exclusion type is active (on) and forcibly enabled by another feature of Akeeba Backup, such as regular expressions filters or simply denote that a specific filter is not applicable to this entity. For example, there is no point the data dump of a view, or a stored procedure, as they have no data in the sense a table does. The third icon, next to the database entity's name, represents the type of the entity, e.g. table, view, etc. You can hover your mouse over the icon to get a tooltip describing the kind of this entity.

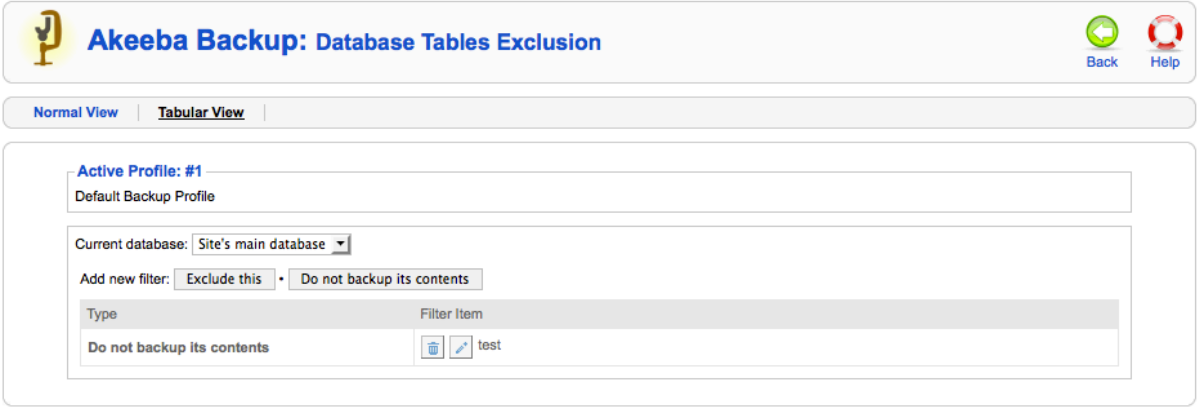
Important

The prefixes of the entities' names appear abstracted. If your site's prefix is `jos_` (the default Joomla!™ setting), the table `jos_users` will appear as `#__users`. This is done to help you quickly identify the tables your site runs on.

Each row of this pane consists of the following elements:

- **Exclusion** icon. If enabled, this database entity will not be backed up at all, i.e. it will be missing from the database dump.
- **Data exclusion** icon. If enabled, only the structure of a table will be backed up, but not its contents. This is useful e.g. for banner tracking or log tables. You need to keep their structure so that your site works, but you don't need to back up tens of thousands of historical data rows you can certainly live without.
- **Entity type** icon. Depends on the entity type, e.g. if it's a view, table, procedure, etc.
- **Entity name**. The name of the entity, as described above.

Database Tables Exclusion - Tabular View



When you click on the Tabular View link, the page radically changes format. Instead of a database browser pane, you now have a grid.

Above the grid you have the Add new filter buttons:

- **Exclude** this. Completely skips backing up the given database entity.
- **Do not backup its contents**. Backs up only the structure but not the contents of the given table.

Each line of the grid displays the following information:

- **The filter type**. It can be one of:
  - **Exclude** this. Completely skips backing up the given database entity.
  - **Do not backup its contents**. Backs up only the structure but not the contents of the given table.
- **Trashcan**. When you click it, the filter row will be removed.
- **Pencil**. When you click it, the row switches to edit mode
- The **filter item** itself. It is the abstracted database entity name which the filter row applies to. When we say "abstracted" we mean that the site's prefix has to be replaced by #__.

When you click on the pencil icon, the filter item becomes an edit box. You can type in the new abstracted database entity name and then click outside the edit box, or press Tab on your keyboard, to immediately save the changes. There is no way to undo your changes.

5.3. RegEx Files and Directories Exclusion

Note

This feature is available only in Akeeba Backup Professional, the paid version of our component

Sometimes you know that you have to exclude files or directories following a specific naming pattern, but they are so many that it's completely impractical going to the normal exclusion filters page and click them one by one. Or they are scattered around the file system tree, making it extremely complex to track them down and exclude them. Wouldn't it be nice to have an automated way to say, for example, "exclude all SVN directories from the backup"? Enter regular expressions. What are those regular expressions? Let's see what Wikipedia has to say on the subject:

> In computing, regular expressions, also referred to as regex or regexp, provide a concise and flexible means for matching strings of text, such as particular characters, words, or patterns of characters. A regular expression is written in a formal language that can be interpreted by a regular expression processor, a program that either serves as a parser generator or examines text and identifies parts that match the provided specification.
>
> —"Regular expression" article [http://en.wikipedia.org/wiki/Regular_expression] from Wikipedia

In a nutshell, regular expressions allow you to quickly define filters which span multiple subdirectories and match file or directory names based on a number of criteria. If you want a quick cheatsheet you can use, I suggest the Regular Expressions Cheat Sheet (V2) [http://www.addedbytes.com/cheat-sheets/regular-expressions-cheat-sheet/] from AddedBytes.com. Some practical examples will be presented at the end of this section.

There are some special considerations experienced regular expressions users must have in mind:

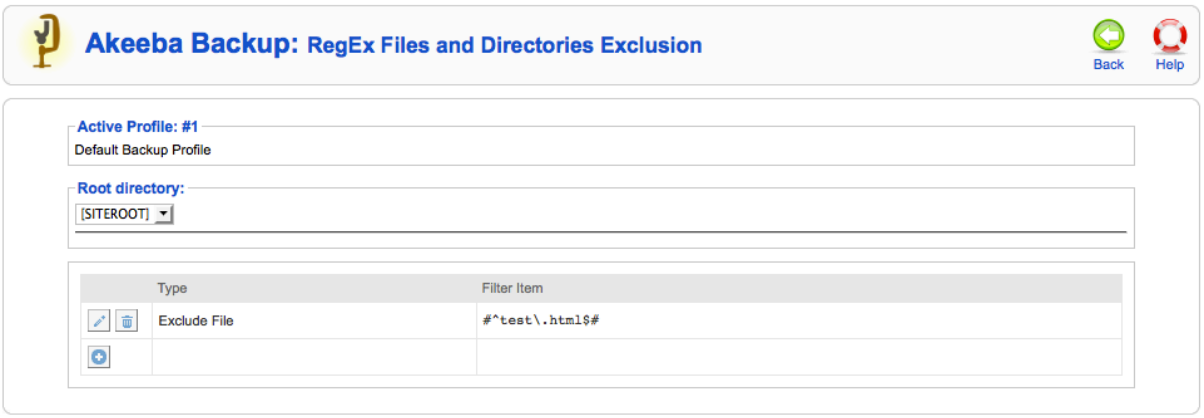
- You are supposed to specify a full regular expression, including its opening and ending separators. So "^foo" is invalid, but "/^foo/" and "#^foo#" are valid.
- Akeeba Backup supports an extension to the PCRE syntax. If you prefix the regex with an exclamation mark you negate its meaning. So "/^foo/" will match all entities starting with "foo", whereas "!/^foo/" will match all entities NOT starting with "foo".

- Akeeba Backup stores and parses your data as raw Unicode (UTF-8), provided that your database meets the minimum requirement of site database server version. This eliminates the need to use the u suffix of regular expressions in order to reference Unicode characters.

When it comes to files and directories exclusion filters in particular, you have to bear in mind:

- The path separator is always the forward slash, even on Windows. This means that c:\wamp\www\index.php is internally represented as c:/wamp/www/index.php. Therefore, all regular expressions must use the forward slash whenever referencing a path separator.
- The filenames are always relative to the root. That's why you have to select a root before entering a regex filter. For instance, the images/stories directory on the root of your Joomla!™ site is internally referenced as "images/stories". You have to take this into account when writing regular expressions.

RegEx Files and Directories Exclusion



This page primarily consists of a grid view. Above the grid, you can find the Root Directory drop-down menu. Akeeba Backup can define filters for the site's files or for each of the off-site directories separately. The default selection, [SITEROOT], contains all filters pertaining to the main site's files. If you have defined off-site directories, you can select the appropriate directory from the drop-down list in order to define filters for that directory.

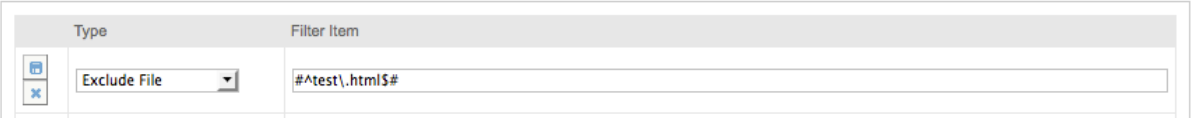
The grid contains three columns:

Icons column
: You can perform the basic operation by clicking on this column's icons:
  - **Trashcan**. When you click it, the filter row will be removed.
  - **Pencil**. When you click it, the row switches to edit mode
  - **Add** (only on the last row). Clicking this icon adds a new row at the end of the list and switches it to edit mode. You can select the type of the newly added filter.

Type
: The filter type defines what will happen when a directory or file matches the regex filter and can be one of:
  - **Exclude directory**. Completely skips backing up the given subdirectory.
  - **Exclude file**. Completely skips backing up the given file.
  - **Skip subdirectories**. Skips backing up all the subdirectories inside the given directory.
  - **Skip files**. Skips backing up all the files inside the given directory.

Filter Item
: This is the actual regular expression you have to write.

RegEx Files and Directories Exclusion - Edit Mode



When you click on the pencil or add icons, the respective row enters the edit mode. In this mode, the filter type becomes a drop-down list where you can select the type of this filter row. The filter item column also turns into an edit box so that you can enter your filter definition. The icon column now contains two different icons:

- **Diskette**. When you click it, the changes will be saved.
- **Cancel**. When you click it, any changes will be cancelled and the row will resume its previous state.

In order to make sure that your filters match the directories and/or files you meant to, you can do so very easily. Just go back to the Control Panel and click on the Files and Directory Exclusion button. The items filtered out by the regular expressions filters will be automatically highlighted in red. You can browse through the file system structure to make sure that only the items you really meant are being excluded.

5.3.1. Regular Expressions recipes for files and directories

No matter how good you are on writing regular expressions, it's always a good idea to have some recipes which serve as a starting point for cooking your own.

1. Exclude AVI files in all directories (note: the i at the end causes the regex to match .avi, .Avi, .AVI, etc without discriminating lower or upper case):

   ```
   #\.avi$#i
   ```

2. Exclude AVI files in your site's `images` directory and all of its subdirectories:

   ```
   #^images/(.*).avi$#i
   ```

3. Exclude AVI files in your site's `images` directory but *not* its subdirectories

   ```
   #^images/[^/]*.avi$#i
   ```

4. Exclude AVI files in your site's `images/video` subdirectory but *not* its subdirectories

   ```
   #^images/video/[^/]*.avi$#i
   ```

5. Exclude all files *except* for files ending in .php (note: the exclamation mark in the beginning is a custom Akeeba Backup notation which negates the meaning of the following regular expression)

   ```
   !#(?>\.php$)#
   ```

6. Exclude all `.svn` subdirectories anywhere and everywhere in your site. The idea is to match everything which ends in a slash (directory separator) and `.svn`, therefore it's a .svn subdirectory.

   ```
   #/\.svn$#
   ```

   However, this won't match the `.svn` directory in your site's root, so you will have to add yet another filter:

   ```
   #^\.svn$#
   ```

   This second filter matches only the `.svn` directory in your site's root.

5.4. RegEx Database Tables Exclusion

Note

This feature is available only in Akeeba Backup Professional, the paid version of our component

Sometimes you know that you have to exclude database tables which follow a specific naming pattern, but they are so many that it's completely impractical going to the normal exclusion filters page and click them one by one. Or you want to exclude everything which doesn't match a specific pattern (e.g. it's not part of the site's main database), but the matching set dynamically and constantly changes over time, making it impossible to create an accurate filter without lots of maintenance. Enter regular expressions. What are those regular expressions? Let's see what Wikipedia has to say on the subject:

> In computing, regular expressions, also referred to as regex or regexp, provide a concise and flexible means for matching strings of text, such as particular characters, words, or patterns of characters. A regular expression is written in a formal language that can be interpreted by a regular expression processor, a program that either serves as a parser generator or examines text and identifies parts that match the provided specification.
>
> —"Regular expression" article [http://en.wikipedia.org/wiki/Regular_expression] from Wikipedia

In a nutshell, regular expressions allow you to quickly define filters which match table names based on a number of criteria. If you want a quick cheatsheet you can use, I suggest the Regular Expressions Cheat Sheet (V2) [http://www.addedbytes.com/cheat-sheets/regular-expressions-cheat-sheet/] from AddedBytes.com. Some practical examples will be presented at the end of this section.

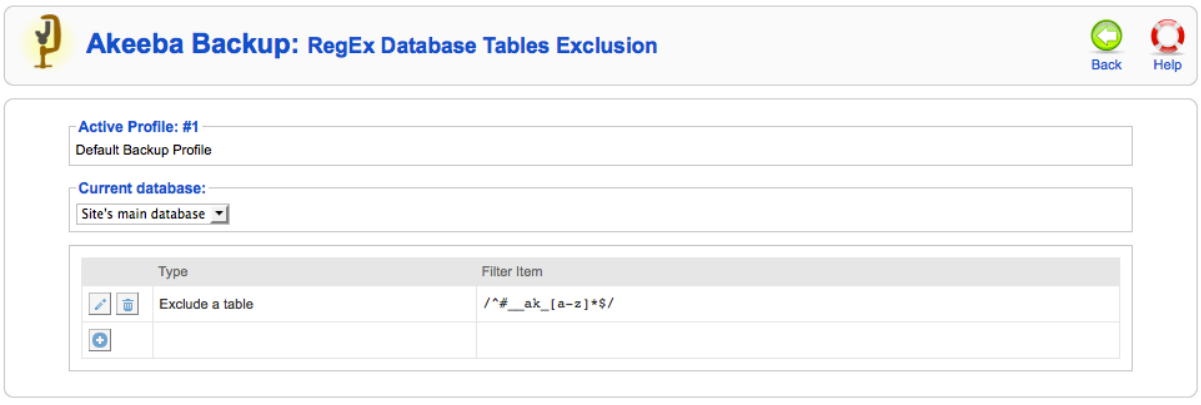
There are some special considerations experienced regular expressions users must have in mind:

- You are supposed to specify a full regular expression, including its opening and ending separators. So "^foo" is invalid, but "/^foo/" and "#^foo#" are valid.
- Akeeba Backup supports an extension to the PCRE syntax. If you prefix the regex with an exclamation mark you negate its meaning. So "/^foo/" will match all entities starting with "foo", whereas "!/^foo/" will match all entities NOT starting with "foo".
- Akeeba Backup stores and parses your data as raw Unicode (UTF-8), provided that your database meets the minimum requirement of site database version. This eliminates the need to use the u suffix of regular expressions in order to reference Unicode characters.

When it comes to database table filters in particular, you have to bear in mind:

- All Joomla!™ tables have their prefix stripped and replaced by the standard #__ placeholder. So, if your database prefix is `jos_`, `jos_users` is internally referenced as `#__users`. You must take this into account when writing regex filters, as this is the name you will have to match!
- The prefix replacement *is not* made in Database Only backup modes (either main site database, or all databases). As a result, you have to reference the tables by their full, normal name, e.g. `jos_users`.
- The examples at the end of this section apply to a full site backup scenario, where the replacement does take place.

RegEx Database Tables Exclusion



This page primarily consists of a grid view. Above the grid, you can find the Root Directory drop-down menu. Akeeba Backup can define filters for the site's main database or for each of the extra databases you may have defined. The default selection, Site's main database, contains all filters pertaining to the main site's database, of course. If you have defined extra databases, you can select the appropriate database from the drop-down list in order to define filters for that database.

The grid contains three columns:

Icons column — You can perform the basic operations by clicking on this column's icons:

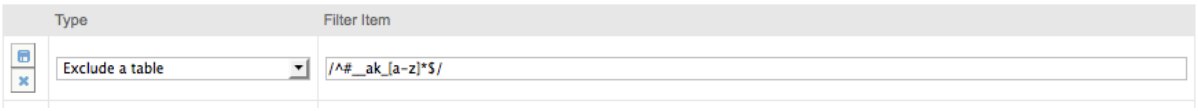
- **Trashcan**. When you click it, the filter row will be removed.
- **Pencil**. When you click it, the row switches to edit mode
- **Add** (only on the last row). Clicking this icon adds a new row at the end of the list and switches it to edit mode. You can select the type of the newly added filter.

Type — The filter type defines what will happen when a directory or file matches the regex filter and can be one of:

- **Exclude a table**. Completely skips backing up tables whose names match the regular expression.
- **Do not backup a table's contents**. Only backs up the structure of tables whose names match the regular expression, but not their contents.

Filter Item — This is the actual regular expression you have to write.

RegEx Database Tables Exclusion - Edit Mode



When you click on the pencil or add icons, the respective row enters the edit mode. In this mode, the filter type becomes a drop-down list where you can select the type of this filter row. The filter item column also turns into an edit box so that you can enter your filter definition. The icon column now contains two different icons:

- **Diskette**. When you click it, the changes will be saved.
- **Cancel**. When you click it, any changes will be cancelled and the row will resume its previous state.

In order to make sure that your filters match the directories and/or files you meant to, you can do so very easily. Just go back to the Control Panel and click on the Database Tables Exclusion button. The items filtered out by the regular expressions filters will be automatically highlighted in red. You can browse through the database structure to make sure that only the items you really meant are being excluded.

5.4.1. Regular Expressions recipes for database tables

No matter how good you are on writing regular expressions, it's always a good idea to have some recipes which serve as a starting point for cooking your own.

1. Exclude non-Joomla! database tables:

   ```
   /^(?>[^#]{1}|##|#_[^_]{1})/
   ```

2. Since nobody understood the previous filter, I have rewritten it in Akeeba Backup's compact proprietary notation which uses the non-standard negation operator (exclamation mark):

   ```
   !/^#__/
   ```

Much simpler, huh?

3. Exclude VirtueMart tables. We know that these tables have `vm_` in their name after the table prefix, e.g. `jos_vm_foobar` becomes `#__vm_foobar`, so you only need to filter `#__vm`.

   ```
   /^#__vm_/
   ```

6. Automating your backup

6.1. Taking backups automatically

Even though Akeeba Backup makes it very easy to take a backup of your Joomla!™ site, it still requires you to log in to the site's backend, click on the Backup Now button and wait for the backup to finish. If you do this daily, it is a drag. Our job is to automate your life, making repeated and time consuming procedures a breeze. To this end we offer not just one, but 3 (yes, THREE!) different backup automation possibilities for Akeeba Backup.

Important

Only one of those options is available in the free (as in "free beer") Akeeba Backup Core release

6.1.1. Front-end backup, for use with CRON

Tip

This option is available in both the Akeeba Backup Core and Akeeba Backup Professional releases. You don't need to subscribe to the Professional edition to use it.

The front-end backup feature is intended to provide the capability to perform an unattended, scheduled backup of your site.

The front-end backup URL performs a single backup step and sends a redirection (HTTP 302) header to force the client to advance to the next page, which performs the next step and so forth. You will only see a message upon completion, should it be successful or not. There are a few limitations, though:

- **It is not designed to be run from a normal web browser**, but from an unattended cron script, utilizing **wget** or **cron** as a means of accessing the function.
- The script is not capable of showing progress messages.
- Normal web browsers tend to be "impatient". If a web page returns a bunch of redirection headers, the web browser thinks that the web server has had some sort of malfunction and stop loading the page. It will also show some kind of "destination unreachable" message. Remember, these browsers are meant to be used on web pages which are supposed to show some content to a human. This behaviour is normal. Most browsers will quit after they encounter the twentieth page redirect response, which is bound to happen. Do not come to the Free Support Forum complaining that Firefox, Internet Explorer, Chrome, Safari, Opera or another browser doesn't work with the front-end backup feature. It was NOT meant to work by design.
- Command line utilities, by default, will also give up loading a page after it has been redirected a number of times. For example, **wget** gives up after 20 redirects, **curl** does so after 50 redirects. Since Akeeba Backup redirects once for every step, it is advisable to configure your command line utility with a large number of redirects; about 10000 should be more than enough for virtually all sites.

Tip

Do you want to automate your backups despite your host not supporting CRON? Webcron.org [http://webcron.org/] fully supports Akeeba Backup's front-end backup feature and is dirt cheap - you need to

spend about 1 Euro for 1000 backup runs. Just make sure you set up your Webcron CRON job time limit to be at least 10% more than the time it takes for Akeeba Backup to backup your site. Don't know how much is that? No problem! Just take a regular backup from your site's back-end, then go to Manage Backups (formerly "Administer Backup Files") page and take a look at the Duration column. That's what you're looking for!

Before beginning to use this feature, you must set up Akeeba Backup to support the front-end backup option. First, go to Akeeba Backup's main page and click on the Options button in the toolbar. Find the option titled Enable front-end and remote backup and set it to `Yes`. Below it, you will find the option named Secret key. In that box you have to enter a password which will allow your CRON job to convince Akeeba Backup that it has the right to request a backup to be taken. Think of it as the password required to enter the VIP area of a night club. After you're done, click the Save & Close button on top to save the settings and close the dialog.

Tip

Use only lower- and upper-case alphanumeric characters (0-9, a-z, A-Z) in your secret key. Other characters may need to be manually URL-encoded in the CRON job's command line. This is error prone and can cause the backup to never start even though you'll be quite sure that you have done everything correctly.

Most hosts offer a CPanel of some kind. There has to be a section for something like "CRON Jobs", "scheduled tasks" and the like. The help screen in there describes how to set up a scheduled job. One missing part for you would be the command to issue. Simply putting the URL in there is not going to work.

Warning

If your host only supports entering a URL in their "CRON" feature, this will most likely not work with Akeeba Backup. There is no workaround. It is a hard limitation imposed by your host. We would like to help you, but we can't. As always, the only barrier to the different ways we can help you is server configuration.

If you are on a UNIX-style OS host (usually, a Linux host) you most probably have access to a command line utility called **wget**. It's almost trivial to use:

wget --max-redirect=10000 "http://*`www.yoursite.com`*/index.php?option=com_akeeba&

view=backup&key=*`YourSecretKey`*"

Of course, the line breaks are included for formatting clarity only. You should not have a line break in your command line!

Important

Do not miss the **--max-redirect=10000** part of the **wget** command! If you fail to include it, the backup will not work with **wget** complaining that the maximum number of redirections has been reached. This is normal behavior, it is not a bug.

Important

`YourSecretKey` must be URL-encoded. You can use an online tool like http://www.url-encode-decode.com or simply consult the Schedule Automatic Backups page.

Warning

Do not forget to surround the URL in double quotes. If you don't the backup will fail and it will be your fault! The reason is that the ampersand is also used to separate multiple commands in a single command line. If you don't use the double quotes at the start and end of the backup URL, your host will think that you tried to run multiple commands and load your site's homepage instead of the front-end backup URL.

If you're unsure, check with your host. Sometimes you have to get from them the full path to wget in order for CRON to work, thus turning the above command line to something like:

/usr/bin/wget --max-redirect=10000 "http://*www.yoursite.com*/index.php?option=com_akeeba&

view=backup&key=*YourSecretKey*"

Contact your host; they usually have a nifty help page for all this stuff. Read also the section on CRON jobs below.

Optionaly, you can also include an extra parameter to the above URL, `&profile=`*`profile_id`*, where *`profile_id`* is the numeric ID of the profile you want to use for the backup. If you don't specify this parameter, the default backup profile (ID=1) will be used. In this sense, the aforementioned URL becomes:

/usr/bin/wget --max-redirect=10000 "http://*www.yoursite.com*/index.php?option=com_akeeba&

view=backup&key=*YourSecretKey*&profile=*profile_id*"

`wget` is multi-platform command line utility program which is not included with all operating systems. If your system does not include the `wget` command, it can be downloaded at this address: http://wget.addictivecode.org/FrequentlyAskedQuestions#download. The `wget` homepage is here: http://www.gnu.org/software/wget/wget.html. Please note that the option `--max-redirect` is available on `wget` version 1.11 and above.

Important

Using a web browser (Internet Explorer, Google Chrome, ...) or `wget` version 1.10 and earlier will most probably result into an error message concerning the maximum redirections limit being exceeded. This is *not* a bug. Most network software will stop dealing with a web site after it has redirected the request more than 20 times. This is a safety feature to avoid consuming network resources on misconfigured web sites which have entered an infinite redirection loop. Akeeba Backup uses redirections creatively, to force the continuation of the backup process without the need for client-side scripting. It is possible, depending on site size, Akeeba Backup configuration and server setup, that it will exceed the limit of 20 redirections while performing a backup operation.

Warning

The ampersands above should be written as a single ampersand, not as an HTML entity (&). Failure to do so will result in a 403: Forbidden error message and no backup will occur. This is not a bug, it's the way wget works.

Using webcron.org to automate your backups

Assuming that you have already bought some credits on webcron.org, here's how to automate your backup using their service.

First, go to Akeeba Backup's main page (Control Panel) and click on the Options button in the toolbar. Find the option titled Enable front-end and remote backup and set it to `Yes`. Below it, you will find the option named Secret key. Type in a secret key. We strongly recommend using only alphanumeric characters, i.e. 0-9, a-z and A-Z. For the sake of this example, we will assume that you have entered `ak33b4s3cRet` in that field. We will also assume that your site is accessible through the URL `http://www.example.com`.

Log in to webcron.org. In the CRON area, click on the New Cron button. Here's what you have to enter at webcron.org's interface:

- **Name of cronjob**: anything you like, e.g. "Backup www.example.com"
- **Timeout**: 180sec; if the backup doesn't complete, increase it. Most sites will work with a setting of 180 or 600 here. If you have a very big site which takes more than 5 minutes to back itself up, you might consider using

Akeeba Backup Professional and the native CRON script (akeeba-backup.php) instead, as it's much more cost-effective.

- **Url you want to execute**: `http://www.example.com/index.php?option=com_akeeba&view=backup&key=ak33b4s3cRet`
- **Login** and **Password**: Leave them blank
- **Execution time** (the grid below the other settings): Select when you want your CRON job to run
- **Alerts**: If you have already set up alert methods in webcron.org's interface, we recommend choosing an alert method here and not checking the "Only on error" so that you always get a notification when the backup CRON job runs.

Now click on Submit and you're all set up!

A PHP alternative to wget

As user DrChalta pointed out in a forum post, there is an alternative to **wget**, as long as your PHP installation has the cURL extension installed and enabled. For starters, you need to save the following PHP script as backup.php somewhere your host's **cron** feature can find it. Please note that this is a command-line script and needn't be located in your site's root; it should be preferably located above your site's root, in a non web-accessible directory.

The script below is a modification over DrChalta's original script, taking into account changes made in later versions of our software. In order to configure it for your server, you only have to change the first three lines.

```
<?php
define('SITEURL', 'http://www.example.com'); // Base URL of your site
define('SECRETKEY', 'MySecretKey'); // Your secret key
define('PROFILE',1); // The profile's ID

// ====================== DO NOT MODIFY BELOW THIS LINE ======================
$curl_handle=curl_init();
curl_setopt($curl_handle,CURLOPT_URL,
SITEURL.'/index.php?option=com_akeeba&view=backup&key='.
SECRETKEY.'&profile='.PROFILE);
curl_setopt($curl_handle,CURLOPT_FOLLOWLOCATION,TRUE);
curl_setopt($curl_handle,CURLOPT_MAXREDIRS,10000); # Fix by Nicholas
curl_setopt($curl_handle,CURLOPT_RETURNTRANSFER,1);
$buffer = curl_exec($curl_handle);
curl_close($curl_handle);
if (empty($buffer))
    echo "Sorry, the backup didn't work.";
else
    echo $buffer;
?>
```

Where *`www.yoursite.com`* and *`YourSecretKey`* should be set up as discussed in the previous section.

Warning

The ampersands above should be written as a single ampersand, not as an HTML entity (&). Failure to do so will result in a 403: Forbidden error message and no backup will occur. This is not a bug, it's the way wget and PHP work.

In order to call this script with a schedule, you need to put something like this to your crontab (or use your host's CRON feature to set it up):

```
0 3 * * 6 /usr/local/bin/php /home/USER/backups/backup.php
```

Where */usr/local/bin/php* is the absolute path to your PHP command-line executable and */home/USER/backups/backup.php* is the absolute path to the script above.

If you set up your **cron** schedule with a visual tool (for example, a web interface), the command to execute part is "*/usr/local/bin/php /home/USER/backups/backup.php*".

Thank you DrChalta for this wonderful tip!

Using the front-end backup in SiteGround and other hosts using cURL instead of wget

As one of our users pointed out in the support forum, finding the correct command to issue for the CRON job is tricky. What he writes applies not only to his host, SiteGround, but many other commercial hosts as well. We'll simply quote our user, bzcoder.

In the CPanel for SiteGround there is a cronjob option, you create a cronjob using that and use:

curl -b /tmp/cookies.txt -c /tmp/cookies.txt -L --max-redirs 1000 -v "*<url>*"

as your command.

Replace *<url>* with your backup URL. Make sure to use the initial url displayed on the backend NOT the final URL when you run the backup manually (been there, done that) - when you do that you end up with a url that doesn't work because of the extra parameter used in continuing the backup process.

6.1.2. Native CRON script

Tip

This option is only available in the Akeeba Backup Professional releases. You need to subscribe to the Professional edition to use it.

Tip

Our users report that they get no joy using this script on GoDaddy hosting, but our alternative script (detailed on the next chapter) works.

Note

This file was found in `administrator/components/com_akeeba/backup.php` in Akeeba Backup 3.0.x up to and including 3.4.x. Since Akeeba Backup 3.5 this file is now present under `cli/akeeba-backup.php`.

If you have access to the command-line version of PHP, Akeeba Backup Professional includes an even better - and faster - way of scheduling your backups. All Akeeba Backup Professional releases include the file `cli/akeeba-backup.php`, which can be run from the command-line PHP interface (PHP CLI). In contrast with previous releases, it doesn't require the front-end backup in order to work; it is self-contained, native backup for your Joomla!™ site, even if your web server is down!

In order to schedule a backup, you will have to use the following command line to your host's CRON interface:

/usr/local/bin/php /home/USER/webroot`/cli/akeeba-backup.php`

where */usr/local/bin/php* is the path to your PHP CLI executable and */home/USER/webroot* is the absolute path to your web site's root. You can get this information from your host.

The backup script accepts three optional parameters:

- **--profile *profile_id*** allows you to select which backup profile you would like to use for the scheduled backup. The *profile_id* is the numeric profile ID you can see in your Control Panel page.

- **--description "*`Your description`*"** allows you set a backup description different than the default. Do not forget to enclose your description in double quotes, or this parameter will not work! Since Akeeba Backup 3.1 the description supports Akeeba Backup's file naming "variables", e.g. `[SITE]`, `[DATE]` and `[TIME]`. These variables are documented in the Output Directory configuration option's description. This allows you to use them in conjunction with this parameter to provide flexible backup descriptions.

- **--override "keyname=value"** allows you to override profile configuration variables. This parameter can appear an unlimited number of times in the command line. It can be used, for example, to provide the username and password to your cloud storage service in the command line, without having to store it in the backup profile's configuration, therefore never storing it in database and hiding it from other administrators. Please take a look at the "Overriding configuration variables" subsection for more information.

- **--quiet** will suppress all output except warnings and error messages. If the backup runs successfully you get no output at all. Note: this option was added in Akeeba Backup Professional 3.3.4.

The `akeeba-backup.php` script will return a different exit code, depending on the backup status. When the backup is successful and without warnings, the exit code will be 0. When the backup completed but with warnings, the exit code will be 1. Finally, if the backup fails, the exit code will be 2. This allows you to check the backup status, for example inside a shell script, for automation purposes.

In order to give some examples, I will assume that your PHP CLI binary is located in `/usr/local/bin/php` - a common setting among hosts - and that your web site's root is located at `/home/johndoe/httpdocs`.

1. Backup with the default profile (ID = 1) and default description:

   ```
   usr/local/bin/php /home/johndoe/httpdocs/cli/akeeba-backup.php
   ```

2. Backup with profile number 2 and default description:

   ```
   usr/local/bin/php /home/johndoe/httpdocs/cli/akeeba-backup.php --profile=2
   ```

3. Backup with the default profile (ID = 1) and a description reading "My automated backup":

   ```
   usr/local/bin/php /home/johndoe/httpdocs/cli/akeeba-backup.php --description="My auto
   ```

4. Backup with profile number 2 and a description reading "My automated backup":

   ```
   usr/local/bin/php /home/johndoe/httpdocs/cli/akeeba-backup.php --profile=2
     --description="My automated backup"
   ```

It goes without saying that the line breaks are for readability only. You should not include line breaks in your command line.

Special considerations:

- All parameters must start with a double dash. If you use a single dash, they will be ignored. This is a limitation of Joomla!'s JApplicationCli iterface –used by our script– which follows the UNIX conventions of command line parameters.

- Most hosts do not impose a time limit on scripts running from the command-line. If your host does and the limit is less than the required time to backup your site, the backup will fail. We are working on a workaround to allow operation even within such time constraints.

- This script is not meant to run from a web interface. If your host only provides access to the CGI or FastCGI PHP binaries, `backup.php` will not work with them. The solution to this issue is tied to the time constraint above. The workaround we're planning will solve both issues.

- Some servers do not fully support this backup method. The usual symptoms will be a backup which starts but is intermittently or consistently aborted in mid-process without any further error messages and no indication of something going wrong. In such a case, trying running the backup from the back-end of your site will work properly. If you witness similar symptoms please use the Alternative CRON Script, outlined in the next section.

Setting up a CRON job on cPanel

Go to your cPanel main page and choose the CRON Jobs icon from the Advanced pane. In the Add New CRON Job box on the page which loads, enter the following information:

Common Settings
: Choose the frequency of your backup, for example once per day.

Command
: Enter your backup command. Usually, you have to use something like:

  ```
  /usr/bin/php5-cli /home/myusername/public_html/cli/akeeba-backup.php --pro
  ```

  where *myusername* is your account's user name (most probably the same you use to login to cPanel) and *YourProfileID* is the numeric profile number you want to use for your backup job. Do note the path for the PHP command line executable: `/usr/bin/php5-cli`. This is the default location of the correct executable file for cPanel 11 and later. Your host may use a different path to the executable. If the command never runs, ask them. We can't help you with that; only those who have set up the server know the changes they have made to the default setup.

Finally, click the Add New Cron Job button to activate the CRON job.

Special considerations for HostGator

The location of the PHP CLI binary is `/usr/bin/php-cli`. This means that your CRON command line should look like:

```
/opt/php53/bin/php /home/myusername/public_html/cli/akeeba-backup.php --profile=YourPro
```

Overriding configuration variables

Since Akeeba Backup 3.1 the Native CRON Script allows you to override or supply missing configuration variables in the command line. This is especially useful for security reasons. One security issue with the cloud storage service integration is that other administrators can peek at Akeeba Backup's configuration and read the username, password or API keys used to access the cloud storage service. You can, however, leave these fields blank in the configuration and supply their values in the command line.

Overriding a configuration variable requires knowing its key name. The key names are represented in dot-format, i.e. `engine.postproc.s3.accesskey` for Amazon S3's access key. Determining the key name is quite easy, as they are stored in INI files throughout the component's back-end. The first location you should look at is `administrator/components/com_akeeba/engine/core`, where you will find four INI files with general settings. Inside the `administrator/components/com_akeeba/engine` subdirectories you will find one INI file per engine.

In order to save you from trouble, here are the most useful key names. The names are designed to be self-explanatory.

JPS archive password	• engine.archiver.jps.key
ANGIE password	• engine.installer.angie.key
Amazon S3	• engine.postproc.s3.accesskey
	• engine.postproc.s3.secretkey
Microsoft Windows Azure BLOB Storage	• engine.postproc.azure.account
	• engine.postproc.azure.key
RackSpace CloudFiles	• engine.postproc.cloudfiles.username

	• engine.postproc.cloudfiles.apikey
CloudMe	• engine.postproc.cloudme.username
	• engine.postproc.cloudme.password
DreamObjects	• engine.postproc.dreamobjects.accesskey
	• engine.postproc.dreamobjects.secretkey
Dropbox (v1 API, old)	• engine.postproc.dropbox.token
	• engine.postproc.dropbox.token_secret
Dropbox (v2 API, new)	• engine.postproc.dropbox2.access_token
Remote FTP server	• engine.postproc.ftp.user
	• engine.postproc.ftp.pass
Google Drive	• engine.postproc.googledrive.refresh_token
Google Storage	• engine.postproc.googlestorage.accesskey
	• engine.postproc.googlestorage.secretkey
iDriveSynce	• engine.postproc.idrivesync.username
	• engine.postproc.idrivesync.password
	• engine.postproc.idrivesync.pvtkey
OneDrive	• engine.postproc.onedrive.access_token
	• engine.postproc.onedrive.refresh_token
Remote SFTP server	• engine.postproc.sftp.user
	• engine.postproc.sftp.pass — Either the password for the username specified above, or the password to the private key file
	• engine.postproc.sftp.privkey — Absolute path to the private key file (optional, for certificate authentication)
	• engine.postproc.sftp.pubkey — Absolute path to the public key file (optional, for certificate authentication)
SugarSync	• engine.postproc.sugarsync.email
	• engine.postproc.sugarsync.password
WebDAV	• engine.postproc.webdav.username
	• engine.postproc.webdav.password

Applying them on the command line is easy. Take this command line as an example:

```
usr/local/bin/php /home/johndoe/httpdocs/cli/akeeba-backup.php
--profile=2 --description="My automated backup"
--override="engine.postproc.s3.accesskey=ABCDEF"
--override="engine.postproc.s3.secretkey=1234567890abcdefgh"
```

In this case, we are telling the backup script to use the backup Profile with ID=2, give the backup description of "My automated backup" and then supply the S3 access and secret keys.

Important

The values of the override parameters must be enclosed in double or single quotes (depends on your Operating System), otherwise the operating system will not pass them back to the backup.php script.

Important

Your script MUST NOT include the line breaks in the previous example. The line breaks are there only for typesetting purposes.

Finally, it should be noted that you can use the command-line override feature to do more tricky configuration overrides, for example turning off the archive splitting or using a different backup output directory to enhance your security. If it's something you can do in the Configuration page of the component, you can also do it using command line overrides.

6.1.3. Alternative CRON script

Tip

This option is only available in the Akeeba Backup Professional releases. You need to subscribe to the Professional edition to use it.

Note

This file was found in `administrator/components/com_akeeba/altbackup.php` in Akeeba Backup 3.0.x up to and including 3.4.x. Since Akeeba Backup 3.5 this file is now present under `cli/akeeba-altbackup.php`.

On some hosts it is impossible to use the native CRON script outlined in the previous section. On such hosts the CRON script will get aborted if it is using too much CPU time, or if the system load exceeds a value predefined by your host company. In order to accomodate for these hosts, Akeeba Backup Professional includes an alternative CRON script. The alternative CRON script performs the backup by using the front-end backup feature of Akeeba Backup. The alternative CRON script is located in `cli/akeeba-altbackup.php`, and must be run from the command-line PHP interface (PHP CLI).

In order to schedule a backup, you will have to use the following command line to your host's CRON interface:

`/usr/local/bin/php /home/USER/webroot``/cli/akeeba-altbackup.php`

where *`/usr/local/bin/php`* is the path to your PHP CLI executable and *`/home/USER/webroot`* is the absolute path to your web site's root. You can get this information from your host.

The backup script accepts the following optional parameters:

- **--profile *`profile_id`*** allows you to select which backup profile you would like to use for the scheduled backup. The *`profile_id`* is the numeric profile ID you can see in your Control Panel page.
- **--no-verify** Only applies to HTTPS connections. If you have a plain HTTP site you can ignore this setting. Since Akeeba Backup 5.5.2 this script will check that the SSL certificate presented by the server is issued by a known, trusted Certification Authority and that the domain name included in the certificate matches the domain name of your site. If you want to disable this verification, e.g. because you're using a self-signed certificate or a certification authority internal to your organization you need to pass the --no-verify option. This will disable the verification, emulating the way this script worked in Akeeba Backup 5.5.1 and earlier.

In order to give some examples, we will assume that your PHP CLI binary is located in `/usr/local/bin/php` - a common setting among hosts - and that your web site's root is located at `/home/johndoe/httpdocs`.

1. Backup with the default profile (ID = 1)

```
usr/local/bin/php /home/johndoe/httpdocs/cli/akeeba-altbackup.php
```

2. Backup with profile number 2

```
usr/local/bin/php /home/johndoe/httpdocs/cli/akeeba-altbackup.php --profile=2
```

It goes without saying that the line breaks are for readability only. You should not include line breaks in your command line.

Special considerations:

- Most hosts do not impose a time limit on scripts running from the command-line. If your host does and the limit is less than the required time to backup your site, the backup will fail.
- This script is not meant to run from a web interface. If your host only provides access to the CGI or FastCGI PHP binaries, `backup.php` will not work with them. The solution to this issue is tied to the time constraint above. The workaround we're planning will solve both issues.
- You must enable the front-end backup feature of your Akeeba Backup Professional instalaltion and assign a "secret key" for it. This is possible by going to the Akeeba Backup Professional's Control Panel page and clicking on the Parameters button on the top right corner of the toolbar. You will find the front-end backup options further down the Parameters page.
- Before using the alternative CRON script for the first time, you must visit the Akeeba Backup's Control Panel page at least once. Since the command-line version of PHP used to run the backup is oblivious to the domain name used by your site, we have to cache this information. Caching of this information occurs as soon as you visit the Control Panel page. The host name is absolutely required in order for the script to be able to access your Akeeba Backup installation's front-end backup feature.
- Your host must support one of the three methods used by the helper script to access your front-end backup URL:

  1. The PHP cURL module.
  2. The fsockopen() method
  3. The fopen() URL wrappers

  If none of these methods is available, the backup will fail.
- Your host may have a firewall setup which doesn't allow the CRON script to access the front-end backup URL. In such a case, the backup will consistently fail without a new log file being produced and without a backup entry being written to the database. You will have to contact your host so that they can allow the script to access the front-end backup URL. Do note that despite the alternative CRON script and your site running on the same server, the firewall restriction might still be in place. This is counter-intuitive, but we've seen this happening on man hosts.

If you are seeking assistance in our forums regarding a failed CRON job, please indicate if and which of these steps you have already tried. Not doing so will hinder our ability to help you in a timely manner.

Setting up a CRON job on cPanel

Go to your cPanel main page and choose the CRON Jobs icon from the Advanced pane. In the Add New CRON Job box on the page which loads, enter the following information:

Common Settings — Choose the frequency of your backup, for example once per day.

Command — Enter your backup command. Usually, you have to use something like:

```
/usr/bin/php5-cli /home/myusername/public_html/cli/akeeba-altbackup.php --
```

where *myusername* is your account's user name (most probably the same you use to login to cPanel) and *YourProfileID* is the numeric profile number you want to use for your backup job. Do note the path for the PHP command line executable: `/usr/bin/php5-cli`. This is the default location of the correct executable file for cPanel 11 and later. Your host may use a different path to the executable. If the command never runs, ask them. We can't help you with that; only those who have set up the server know the changes they have made to the default setup.

Finally, click the Add New Cron Job button to activate the CRON job.

Special notes for GoDaddy

According to our users who have tried this, this laternative script does work with GoDaddy. The command line you have to use is:

```
/usr/local/php5/bin/php "$HOME/html/cli/akeeba-altbackup.php" --profile=YourProfileID
```

where *YourProfileID* is the numeric profile number you want to use for your backup job.

The PHP executable we are using is the CLI rather than the default CGI. This is important; if you use the CGI executable then the script will not run. Don't forget to enable frontend backup and insert your secret word. To enable frontend backup go to Akeeba Backup under components, select configuration, select options from the navigation, then select the front-end backup tab to enable the settings.

If the backup completes successfully but the backup appears as "Failed" in the Manage Backups (formerly "Administer Backup Files") page, you'll have to apply a workaround. Go to Akeeba Backup and select your backup profile from the drop-down list. Then click on the Configuration button. In the configuration page check the Use database storage for temporary data option.

6.2. Checking for failed backups automatically

Tip

This option is only available in the Akeeba Backup Professional releases. You need to subscribe to the Professional edition to use it.

While you can automate backups with any of the three methods above, there is a small drawback. It is impossible to catch a failed backup if the backup failure was caused by a PHP error or the server killing the backup script for any reason (usually: time, file size and memory limits). This has the unwanted side effect of not knowing when your backup has failed unless you keep track of the backup records on your sites or the emails sent out by your CRON jobs (if any are sent at all – it depends on the server / service you are using).

Since Akeeba Backup 3.10.2 you can automate the check for failed backups and have it email you when it detects that the latest backup has failed.

Warning

This is an optional, advanced and DANGEROUS feature. If you check for failed backups while a backup is still running it is very possible that you will cause the backup to fail! We recommend scheduling backup checks a substantial amount of time (e.g. 1 hour) after the expected end time of your backups.

6.2.1. Front-end backup failure check, for use with CRON

Tip

This option is only available in the Akeeba Backup Professional releases. You need to subscribe to the Professional edition to use it.

Warning

This is an optional, advanced and DANGEROUS feature. If you check for failed backups while a backup is still running it is very possible that you will cause the backup to fail! We recommend scheduling backup checks a substantial amount of time (e.g. 1 hour) after the expected end time of your backups.

The front-end backup feature is intended to provide the capability to perform an unattended, scheduled failed backup check.

Before beginning to use this feature, you must set up Akeeba Backup to support the front-end backup option. First, go to Akeeba Backup's main page and click on the Options button in the toolbar. Find the option titled Enable front-end and remote backup and set it to `Yes`. Below it, you will find the option named Secret key. In that box you have to enter a password which will allow your CRON job to convince Akeeba Backup that it has the right to request a backup to be taken. Think of it as the password required to enter the VIP area of a night club. After you're done, click the Save & Close button on top to save the settings and close the dialog.

Tip

Use only lower- and upper-case alphanumeric characters (0-9, a-z, A-Z) in your secret key. Other characters may need to be manually URL-encoded in the CRON job's command line. This is error prone and can cause the backup to never start even though you'll be quite sure that you have done everything correctly.

Most hosts offer a CPanel of some kind. There has to be a section for something like "CRON Jobs", "scheduled tasks" and the like. The help screen in there describes how to set up a scheduled job. One missing part for you would be the command to issue. Simply putting the URL in there is not going to work.

If you are on a UNIX-style OS host (usually, a Linux host) you most probably have access to a command line utility called **wget**. It's almost trivial to use:

wget "http://*www.yoursite.com*/index.php?option=com_akeeba&view=check&key=*YourSecretKey*"

Warning

Do not forget to surround the URL in double quotes. If you don't, the check will fail to execute. The reason is that the ampersand is also used to separate multiple commands in a single command line. If you don't use the double quotes at the start and end of the backup URL, your host will think that you tried to run multiple commands and load your site's homepage instead of the front-end backup URL.

Important

YourSecretKey must be URL-encoded. You can use an online tool like http://www.url-encode-decode.com or simply consult the Schedule Automatic Backups page.

If you're unsure, check with your host. Sometimes you have to get from them the full path to wget in order for CRON to work, thus turning the above command line to something like:

/usr/bin/wget "http://*www.yoursite.com*/index.php?option=com_akeeba&view=check&key=*YourSecretKey*"

Contact your host; they usually have a nifty help page for all this stuff. Read also the section on CRON jobs below.

`wget` is multi-platform command line utility program which is not included with all operating systems. If your system does not include the `wget` command, it can be downloaded at this address: http://wget.addictivecode.org/FrequentlyAskedQuestions#download. The `wget` homepage is here: http://www.gnu.org/software/wget/wget.html.

Warning

The ampersands above should be written as a single ampersand, not as an HTML entity (&). Failure to do so will result in a 403: Forbidden error message and no backup will occur. This is not a bug, it's the way wget works.

Using webcron.org to automate your failed backup checks

Assuming that you have already bought some credits on webcron.org, here's how to automate your failed backup checks using their service.

First, go to Akeeba Backup's main page (Control Panel) and click on the Options button in the toolbar. Find the option titled Enable front-end and remote backup and set it to `Yes`. Below it, you will find the option named Secret key. Type in a secret key. We strongly recommend using only alphanumeric characters, i.e. 0-9, a-z and A-Z. For the sake of this example, we will assume that you have entered `ak33b4s3cRet` in that field. We will also assume that your site is accessible through the URL `http://www.example.com`.

Log in to webcron.org. In the CRON area, click on the New Cron button. Here's what you have to enter at webcron.org's interface:

- **Name of cronjob**: anything you like, e.g. "Backup www.example.com"
- **Timeout**: 30sec; if the failed backup check doesn't complete, increase it. Most sites will work with a setting of 60 or 90 here.
- **Url you want to execute**: `http://www.example.com/index.php?option=com_akeeba&view=check&key=ak33b4s3cRet`
- **Login** and **Password**: Leave them blank
- **Execution time** (the grid below the other settings): Select when you want your CRON job to run

Now click on Submit and you're all set up!

A PHP alternative to wget

As user DrChalta pointed out in a forum post, there is an alternative to **wget**, as long as your PHP installation has the cURL extension installed and enabled. For starters, you need to save the following PHP script as check.php somewhere your host's **cron** feature can find it. Please note that this is a command-line script and needn't be located in your site's root; it should be preferably located above your site's root, in a non web-accessible directory.

The script below is a modification over DrChalta's original script, taking into account changes made in later versions of our software. In order to configure it for your server, you only have to change the first three lines.

```
<?php
define('SITEURL', 'http://www.example.com'); // Base URL of your site
define('SECRETKEY', 'MySecretKey'); // Your secret key

// ====================== DO NOT MODIFY BELOW THIS LINE ======================
$curl_handle=curl_init();
curl_setopt($curl_handle,CURLOPT_URL,
SITEURL.'/index.php?option=com_akeeba&view=check&key='.
SECRETKEY);
curl_setopt($curl_handle,CURLOPT_FOLLOWLOCATION,TRUE);
curl_setopt($curl_handle,CURLOPT_RETURNTRANSFER,1);
$buffer = curl_exec($curl_handle);
curl_close($curl_handle);
if (empty($buffer))
    echo "Sorry, the failed backup check didn't work.";
else
    echo $buffer;
```

```
?>
```

Where *www.yoursite.com* and *YourSecretKey* should be set up as discussed in the previous section.

Warning

The ampersands above should be written as a single ampersand, not as an HTML entity (&). Failure to do so will result in a 403: Forbidden error message and no backup will occur. This is not a bug, it's the way wget and PHP work.

In order to call this script with a schedule, you need to put something like this to your crontab (or use your host's CRON feature to set it up):

```
0 3 * * 6 /usr/local/bin/php /home/USER/backups/check.php
```

Where */usr/local/bin/php* is the absolute path to your PHP command-line executable and */home/USER/backups/backup.php* is the absolute path to the script above.

If you set up your **cron** schedule with a visual tool (for example, a web interface), the command to execute part is "*/usr/local/bin/php /home/USER/backups/check.php*".

Thank you DrChalta for this wonderful tip!

Using the front-end backup in SiteGround and other hosts using cURL instead of wget

As one of our users pointed out in the support forum, finding the correct command to issue for the CRON job is tricky. What he writes applies not only to his host, SiteGround, but many other commercial hosts as well. We'll simply quote our user, bzcoder.

In the CPanel for SiteGround there is a cronjob option, you create a cronjob using that and use:

curl -b /tmp/cookies.txt -c /tmp/cookies.txt -L -v "*<url>*"

as your command.

Replace *<url>* with your failed backup check URL.

6.2.2. Native CRON script for failed backup checks

Tip

This option is only available in the Akeeba Backup Professional releases. You need to subscribe to the Professional edition to use it.

Tip

Our users report that they get no joy using this script on GoDaddy hosting, but our alternative script (detailed on the next chapter) works.

Warning

This is an optional, advanced and DANGEROUS feature. If you check for failed backups while a backup is still running it is very possible that you will cause the backup to fail! We recommend scheduling backup checks a substantial amount of time (e.g. 1 hour) after the expected end time of your backups.

If you have access to the command-line version of PHP, Akeeba Backup Professional includes an even better way of scheduling your failed backup checks. All Akeeba Backup Professional releases include the file `cli/akeeba-check-failed.php`, which can be run from the command-line PHP interface (PHP CLI). In contrast with previous releases, it doesn't require the front-end backup in order to work; it is self-contained, native backup for your Joomla!™ site, even if your web server is down!

In order to schedule a backup, you will have to use the following command line to your host's CRON interface:

```
/usr/local/bin/php /home/USER/webroot/cli/akeeba-check-failed.php
```

where */usr/local/bin/php* is the path to your PHP CLI executable and */home/USER/webroot* is the absolute path to your web site's root. You can get this information from your host.

In order to give an example, we will assume that your PHP CLI binary is located in `/usr/local/bin/php` - a common setting among hosts - and that your web site's root is located at `/home/johndoe/httpdocs`.

```
usr/local/bin/php /home/johndoe/httpdocs/cli/akeeba-check-failed.php
```

Special considerations:

- This script is not meant to run from a web interface. If your host only provides access to the CGI or FastCGI PHP binaries, `backup.php` will not work with them. The solution to this issue is tied to the time constraint above. The workaround we're planning will solve both issues.

Setting up a CRON job on cPanel

Go to your cPanel main page and choose the CRON Jobs icon from the Advanced pane. In the Add New CRON Job box on the page which loads, enter the following information:

Common Settings — Choose the frequency of your backup, for example once per day.

Command — Enter your backup command. Usually, you have to use something like:

```
/usr/bin/php5-cli /home/myusername/public_html/cli/akeeba-check-failed.php
```

where *myusername* is your account's user name (most probably the same you use to login to cPanel). Do note the path for the PHP command line executable: `/usr/bin/php5-cli`. This is the default location of the correct executable file for cPanel 11 and later. Your host may use a different path to the executable. If the command never runs, ask them. We can't help you with that; only those who have set up the server know the changes they have made to the default setup.

Finally, click the Add New Cron Job button to activate the CRON job.

Special considerations for HostGator

The location of the PHP CLI binary is `/usr/bin/php-cli`. This means that your CRON command line should look like:

```
/opt/php53/bin/php /home/myusername/public_html/cli/akeeba-check-failed.php
```

6.2.3. Alternative CRON script

Tip

This option is only available in the Akeeba Backup Professional releases. You need to subscribe to the Professional edition to use it.

Warning

This is an optional, advanced and DANGEROUS feature. If you check for failed backups while a backup is still running it is very possible that you will cause the backup to fail! We recommend scheduling backup checks a substantial amount of time (e.g. 1 hour) after the expected end time of your backups.

On some hosts it is impossible to use the native CRON script outlined in the previous section. In order to accomodate for these hosts, Akeeba Backup Professional includes an alternative CRON script. The alternative

CRON script performs the failed backup check by using the front-end bacup check URL of Akeeba Backup. The alternative CRON script is located in `cli/akeeba-altcheck-failed.php`, and must be run from the command-line PHP interface (PHP CLI).

In order to schedule a backup, you will have to use the following command line to your host's CRON interface:

`/usr/local/bin/php /home/USER/webroot``/cli/akeeba-altcheck-failed.php`

where *`/usr/local/bin/php`* is the path to your PHP CLI executable and *`/home/USER/webroot`* is the absolute path to your web site's root. You can get this information from your host.

In order to give an example, we will assume that your PHP CLI binary is located in `/usr/local/bin/php` - a common setting among hosts - and that your web site's root is located at `/home/johndoe/httpdocs`.

```
usr/local/bin/php /home/johndoe/httpdocs/cli/akeeba-altcheck-failed.php
```

Special considerations:

- This script is not meant to run from a web interface. If your host only provides access to the CGI or FastCGI PHP binaries, `akeeba-altcheck-failed.php` will not work with them.
- You must enable the front-end backup feature of your Akeeba Backup Professional instalaltion and assign a "secret key" for it. This is possible by going to the Akeeba Backup Professional's Control Panel page and clicking on the Parameters button on the top right corner of the toolbar. You will find the front-end backup options further down the Parameters page.
- Before using the alternative CRON script for the first time, you must visit the Akeeba Backup's Control Panel page at least once. Since the command-line version of PHP used to run the backup is oblivious to the domain name used by your site, we have to cache this information. Caching of this information occurs as soon as you visit the Control Panel page. The host name is absolutely required in order for the script to be able to access your Akeeba Backup installation's front-end backup feature.
- Your host must support one of the three methods used by the helper script to access your front-end backup URL:

  1. The PHP cURL module.
  2. The fsockopen() method
  3. The fopen() URL wrappers

  If none of these methods is available, the backup will fail.
- Your host may have a firewall setup which doesn't allow the CRON script to access the front-end backup check URL. In such a case, the backup check will consistently fail. You will have to contact your host so that they can allow the script to access the front-end backup check URL. Do note that despite the alternative CRON script and your site running on the same server, the firewall restriction might still be in place. This is counter-intuitive, but we've seen this happening on many hosts.

Setting up a CRON job on cPanel

Go to your cPanel main page and choose the CRON Jobs icon from the Advanced pane. In the Add New CRON Job box on the page which loads, enter the following information:

Common Settings
: Choose the frequency of your backup, for example once per day.

Command
: Enter your backup command. Usually, you have to use something like:

  `/usr/bin/php5-cli /home/`*`myusername`*`/public_html/cli/akeeba-altcheck-failed.`

  where *`myusername`* is your account's user name (most probably the same you use to login to cPanel). Do note the path for the PHP command line executable: `/usr/bin/php5-cli`. This is the default location of the correct executable file for cPanel 11 and later. Your host

may use a different path to the executable. If the command never runs, ask them. We can't help you with that; only those who have set up the server know the changes they have made to the default setup.

Finally, click the Add New Cron Job button to activate the CRON job.

Special notes for GoDaddy

According to our users who have tried this, this alternative script does work with GoDaddy. The command line you have to use is:

```
/usr/local/php5/bin/php "$HOME/html/cli/akeeba-altcheck-failed.php"
```

The PHP executable we are using is the CLI rather than the default CGI. This is important; if you use the CGI executable then the script will not run. Don't forget to enable frontend backup and insert your secret word. To enable frontend backup go to Akeeba Backup under components, select configuration, select options from the navigation, then select the front-end backup tab to enable the settings.

7. Site Transfer Wizard

What is the Site Transfer Wizard?

One of the most common uses of Akeeba Backup is transferring a site between different locations (folders, subdomains, domains and servers). Typically this involves taking a backup, downloading it to your computer, uploading it to the new location alongside Kickstart and launching Kickstart to extract the backup archive and proceed with the restoration. The download and upload part of this process takes a lot of time, especially when you have a slower connection. The Site Transfer Wizard will save you some precious time by eliminating the need to transfer the backup archive through your computer, instead performing a server to server transfer.

Tip

We recommend that you try using the Site Transfer Wizard *without* reading this documentation section. You only need to refer to this documentation in case a server issue or a mistake in the information you entered prevents you from using it. That's why this documentation section is brutally long; it's *troubleshooting*, not regular usage documentation. The Site Transfer Wizard is intuitive enough to use without reading its documentation.

Prerequisites

Before you begin you must have create a new database for the destination site. This is something that Akeeba Backup and its restoration script is not allowed to do due to the configuration of most servers. This has to do with your server's database security settings and cannot be "worked around" in any way. If you are not sure how to do it please contact your host - this is a server-specific task and they are the only people who can help you with it.

You also need to know how to connect to the target location. This requires knowing the FTP, FTPS or SFTP connection information to the target location. This is required even if you are transferring to a subdirectory, subdomain or domain on the same server your site is currently on. If you are not sure how to obtain this information please contact your host; they are the only people who can help you accurately figure out this information.

If you will be using FTP or FTPS to transfer your site your current server must either have the PHP cURL extension installed with FTP support or the PHP FTP functions enabled.It must not block outbound connection to the remote server's FTP port (typically port 21). The remote FTP server must allow connections from your site's current server and allow at least 7 connection attempts to be made within 1 second.

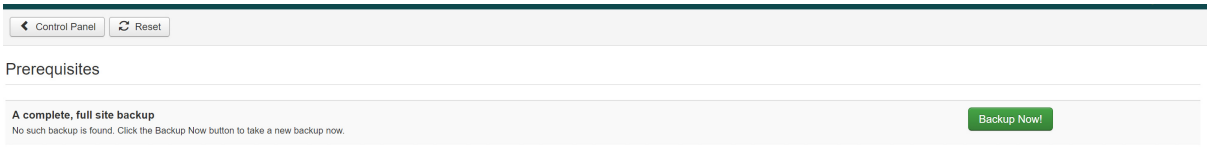
If you will be using SFTP to transfer your site your current server must either have the PHP cURL extension installed with SFTP support of the PHP SSH2 extension installed. It must not block outbound connection to the remote server's FTP port (typically port 22). The remote FTP server must allow connections from your site's current server and allow at least 7 connection attempts to be made within 1 second.

In every case the remote location MUST be accessible through HTTP/HTTPS over the Internet from your site's server and your computer. Akeeba Backup will be checking that and won't let you proceed with the transfer if it can't connect.

Backup age check

The Site Transfer Wizard requires a recent backup, taken within the last 24 hours using *the currently active backup profile*. If one is not detected you will be notified. If you want to use a backup taken with a different profile please remember to activate that profile from Akeeba Backup's main page before clicking on Site Transfer Wizard.

Backup age check

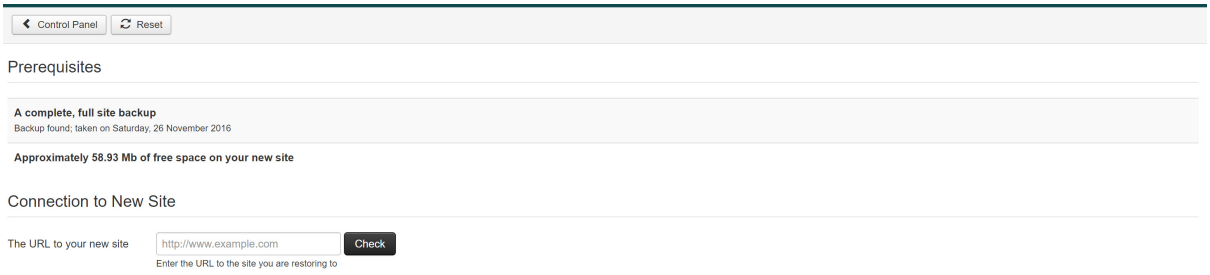


Click the Backup Now! button to take a new backup with the current backup profile.After the backup is complete you will need to go back to the main page and then click on Site Transfer Wizard again.

Setting up the transfer target URL

When a recent backup is detected the Site Transfer Wizard will let you know how much free space you will need (approximately!) on the target server. Please make sure that you have enough disk space before proceeding.

Setting up the transfer target URL



Afterwards please enter the URL of the target site and click the Check button. You must enter the full URL to the target site including the `http://` or `https://` prefix and any path to the site but without the index.php part. For example you need to enter something like `https://www.example.com`, `http://subdomain.example.net` or `http://localhost/mysite`.

The Site Transfer Wizard will check that the URL is accessible from your server.Please note that if the URL returns an error, including but not limited to 403 Forbidden and 500 Internal Server Error, you will receive a message telling you that the URL is inaccessible. In some *very rare* circumstances you may be receiving this message in error. In those cases you can click on the I want to ignore this warning and proceed at my own risk button and proceed anyway. Please note that you are doing so *at your own risk*. We will not be able to help you if something doesn't work or breaks!

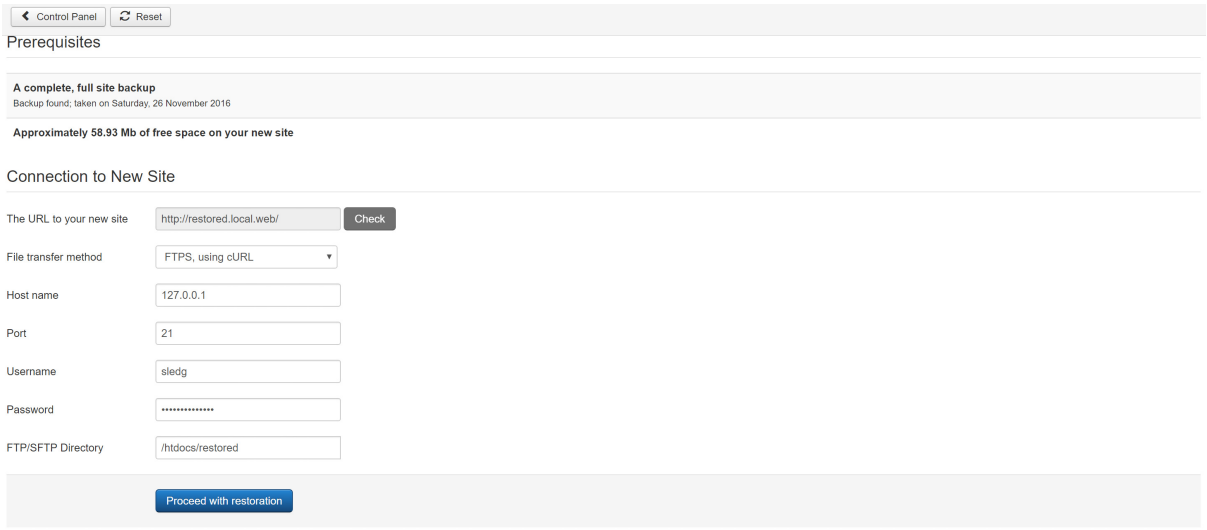
Tip

If at any point you realise you have entered the wrong URL you can click on the Reset button in the toolbar to clear all Site Transfer Wizard settings and start over.

Setting up the connection

The next step lets you tell the Site Transfer Wizard how to connect to your target site to transfer files.

Setting up the connection



Select one of the available transfer methods (not all of them may be available on your server):

FTP, using cURL	You will connect to your site using plain (insecure) FTP. This is the simplest file transfer protocol, supported by most hosts - however it's also the least secure. This method uses the PHP cURL extension which is compatible with most hosts.
FTPS, using cURL	You will connect to your site using plain FTP over a secure SSL/TLS connection. This is a simple file transfer protocol, supported by many hosts and it is quite secure. This method uses the PHP cURL extension which is compatible with most hosts.
SFTP, using cURL	You will connect to your site using file transfer over SSH (a.k.a. Secure File Transfer Protocol, or SFTP for short). This is an advanced file transfer protocol, supported by some hosts and it is the most secure. This method uses the PHP cURL extension which is compatible with most hosts.
FTP, native PHP functions	You will connect to your site using plain (insecure) FTP. This is the simplest file transfer protocol, supported by most hosts - however it's also the least secure. This method uses the PHP native FTP functions. You may experience some compatibility issues with badly configured FTP servers.
FTPS, native PHP functions	You will connect to your site using plain FTP over a secure SSL/TLS connection. This is a simple file transfer protocol, supported by many hosts and it is quite secure. This method uses the PHP native FTP functions. You may experience some compatibility issues with badly configured FTP servers.
SFTP, native PHP SSH2 extension	You will connect to your site using file transfer over SSH (a.k.a. Secure File Transfer Protocol, or SFTP for short). This is an advanced file transfer protocol, supported by some hosts and it is the most secure. This method uses the PHP SSH2 extension. Since this extension is currently marked as experimental it may not be available on your server or not work properly.
Manually	If all else fails (your servers just can't talk to each other) choose this option. It will give you instructions for performing a manual backup archive transfer, including a tutorial for restoring it after it's transferred. This is your failsafe method, one which has been used by thousands of site developers since 2006 to transfer their sites between different locations.

If your target site supports more than one transfer methods please try using the most secure ones first. The order of preference, from MOST to least secure is: SFTP, FTPS, FTP. Moreover, if you are given the choice between a method that uses cURL and one which doesn't please try using the cURL one first. If none of them works for you please check your connection information and retry. If nothing works despite the connection information being correct you have a case where the two servers cannot talk to each other due to networking, firewall or setup issues.

The easiest thing you can do is use the Manually option to transfer your site by manually uploading your backup archive.

Enter the connection information below and click on Proceed with restoration to get to the next step. Please note that if you chose Manually above the next step simply gives you instructions for performing the transfer and the rest of this documentation section does not apply.

Files transfer and restoration

At this point the Site Transfer Wizard is going to make some sanity checks and upload some files on your server.

If the connection fails for any reason you will be told so. Please double check the connection information and the FTP/SFTP directory. The latter must exist and be both readable and writeable. If you still get an error despite all the connection information being correct please try a different connection method. If all available methods fail please do contact both hosts (the one your site is currently on and the one you're trying to transfer to). One or both of the servers have a server protection which prevents the two servers from talking to each other. If you cannot get your hosts to resolve that issue your only choice will be using the Manually option above. Unfortunately there is nothing we can do about it since it's the server which doesn't allow the server-to-server transfer in any way.

If the target server and location is the same as the one where your current site exists the process will be aborted. You MUST NOT use the Site Transfer Wizard to restore a backup archive on your own site. Either use the Restore feature in the Manage Backups Page (Professional version only) or use Akeeba Kickstart or Akeeba eXtract Wizard to extract the backup archive and start the restoration.

If a .htaccess file is detected on the target location the process will be aborted. The .htaccess files can interfere in the way PHP script execute, corrupting the upload of the backup archive or simply blocking the upload, extraction or restoration process. As a precaution the Site Transfer Wizard will not proceed in this case unless you delete the .htaccess file.

After these basic checks the Site Transfer Wizard will try to upload the two Kickstart files (`kickstart.php` and `kickstart.transfer.php`) to your target location and create a new world-writable (0777 permissions) directory called `kicktemp`. Yes, we are aware that the world writable permissions are REALLY BAD for security - but only if you let them persist. We only create this directory temporarily and only use it for temporary data. After the process is done this directory is removed, therefore eliminating any possible security concern. If any of these operations fails you will receive an error message. If this happens please make sure that the target directory is writeable. If you are not sure please ask your host for assistance.

If the FTP/SFTP Directory you've entered does not correspond to the URL to the new site you have entered you will be told so. You CANNOT receive this message in error. If you get this message you MUST check that the directory corresponds to the URL you've entered. If you are not sure, or if you think that Akeeba Backup is wrong (it's not), do check with your host. **This is the most common mistake people make**. Trust us. This is exactly why we added this check.

Afterwards the Site Transfer Wizard will attempt to upload the backup archive file(s). This is done in small, 1Mb chunks. The file is NOT uploaded using FTP, FTPS or SFTP. Why? Because, as we explained previously in this documentation, transferring a big file can take too long which will cause PHP or your web server to halt with a timeout error. The Site Transfer Wizard is instead sending 1Mb of data at a time to Kickstart (which it uploaded in the previous step). Kickstart on the target location "assembles" the archive file(s) from these 1Mb chunks behind the scenes. This lets us transfer really big backup archives without timing out. The progress of the upload is displayed on the page.

However, this *may* lead to problems on some servers. Since the Site Transfer Wizard is making a lot of repeated requests to the kickstart.php URL on the target location some servers may mistakenly assume that this is an attack on the server. Other servers may not like that a lot of the CPU is being used by that site hosting account all of a sudden. If this kind of server protection is triggered you will receive an error message. Depending on the server and host they might also temporarily block the IP address of your site's current server, making it impossible to run the Site Transfer Wizard again for a period of a few minutes to a full day. If you get in this kind of situation you will have to use the Manually option and transfer the backup archives yourselves. Unfortunately there is nothing we can do about it since it's the server which doesn't allow the server-to-server transfer of large files.

When the backup archive files are fully transferred you will see a button called Run Kickstart. Click on it to launch Kickstart on the target URL. Kickstart allows you to extract the backup archive on the target server. This is required since the actual restoration script is stored inside the backup archive. If you are unsure how to proceed after this point please consult our video tutorials on transferring your site to a new server. Ignore the part where you upload Kickstart and the backup archive; this is already done for you by the Site Transfer Wizard.

Chapter 4. Miscellaneous Extensions (Modules, Plugins)

1. Akeeba Backup Notification plugin

This plugin is obsolete as of Akeeba Backup 3.11.0. Please use the Joomla! Extensions Manager to get informed for and install the updates of our components.

2. The CLI update notification and automatic update script

Note

This feature was removed in May 2017.

The automatic update script was removed in May 2017 due to massive bugs in Joomla! 3.7. These bugs broke all CLI scripts which make direct or indirect use of the JSession package, including simply checking if a user is logged in. This includes the update script. It cannot be fixed because the update script uses the core JUpdater and JInstaller APIs to fetch and install updates. Both of them use JSession which means they cannot be used from a CLI script.

The only thing we can do is remove a feature which can no longer work because Joomla! broke backwards compatibility with itself.

3. Backup on Update

Note

Displayed on the Plugin Manager as System - Backup on update

Joomla! 2.5 and later versions include the Joomla! Update component (originally developed as part of Admin Tools by our company, later donated to Joomla! and now maintained by the Joomla! team) which allows you to update Joomla! to its latest version. When you are updating between minor versions of Joomla! (e.g. 3.2 to 3.3) or between major versions of Joomla! (e.g. 2.5 to 3.x) some extensions on your site might experience problems or make your site inaccessible. It's always a good idea to take a backup of your site before upgrading Joomla!, but how many times did you forget to do it only to end up with an inaccessible site and a furious client? Fear no more, our plugin is here to automate this process for you.

When this plugin is enabled it will "see" your attempt to update Joomla! and automatically launch Akeeba Backup to take a backup of your site. Once the backup is successfully complete it will take you back to Joomla! Update, allowing it to install the new Joomla! version. All this happens automatically. You and your clients can no longer forget to take a backup before updating Joomla!: the backup will be taken automatically.

Editing the plugin you will find the sole option, Backup Profile, which lets you define which Akeeba Backup profile to use for these automated backups. If you don't specify anything the default backup profile (the one with ID=1) will be used.

Tip

We recommend using a backup profile which stores a copy of the backup archive in external storage (e.g. Amazon S3, Dropbox or Box.com) on top of leaving a copy of the backup archive on your server. This way you have maximum protection against any kind of accidents caused by a failed or problematic Joomla! update.

Chapter 5. Restoring backups

1. Restoration and troubleshooting instructions

How do I restore my backups?

We have replaced this chapter with the corresponding chapter [https://www.akeebabackup.com/documentation/quick-start-guide/restoring-backups.html] in our Quick Start Guide. Alternatively, you can try watching our Video Tutorials [https://www.akeebabackup.com/documentation/video-tutorials.html] for a quick (less than 10 minutes) overview of the whole process, from installing Akeeba Backup to restoring your backup archives.

Troubleshooting non-functional restored sites

Please refer to our Troubleshooting Wizard's section on solving post-restoration issues [https://www.akeebabackup.com/documentation/troubleshooter/post-restoration.html]. Please note that all of them have nothing to do with Akeeba Backup, but can be attributed either to some server configuration mismatch or a pesky setting in some component, plugin, module or template.

2. Unorthodox: the emergency restoration procedure

Warning

THIS IS NOT THE REGULAR RESTORATION PROCEDURE.

I will say it again.

THIS IS ****NOT**** HOW YOU ARE SUPPOSED TO RESTORE BACKUPS!!!!!

You must follow these instructions ONLY if the restoration script which is included inside the backup archive, under the installation directory (Akeeba Backup Installer) is not working on your host and you really, REALLY are in a BIG hurry to get your site up and running.

And I will say it once more.

THIS IS NOT HOW YOU ARE SUPPOSED TO RESTORE BACKUP ARCHIVES UNDER NORMAL CIRCUMSTANCES.

For normal restoration instructions, please take a look in our Quick Start Guide [https://www.akeebabackup.com/documentation/quick-start-guide/restoring-backups.html].

Note

These instructions are meant to be first read before disaster strikes. Therefore, a fair amount of humour has been used throughout. If you try to read it after disaster struck you will naturally find the humorous parts inappropriate, or even offensive. Rest assured that this is because you are under a huge amount of stress. As soon as you'll have finished following the instructions herein, you will be able to re-read this document with a light heart and enjoy the humorous puns as they were intended.

Inevitably, some people will end up with a backup file, a ruined site and a problem in the restoration procedure they can't work out. Almost always, the recipe includes a pressing deadline which requires that the site is on-line... yesterday. If you are in a situation like the one we just described, breathe. Do not panic. We've got you

covered, with this concise manual site restoration guide. So, here it goes... it's manual Joomla! Site restoration In 7 steps or even less.

Step 1. Making sure it won't get worse.

Assuming such a situation, it's only human to be in panic and despair. Panic is a bad counsellor. It will give you wrong advice. Despair will only make you careless. So, people, get it together! Make a backup of the only thing separating you from complete disaster: the backup file. Burn it on a CD. Write it on your USB key. Put it on a couple of locations on your file server. Just make sure you'll have an extra copy in case you screw up.

This exercise has been proven to lower the probability of anything going wrong. Furthermore, it's good for your psychology. It gives you a sense of security you didn't have five minutes ago.

Step 2. Extracting the archive.

Now, we have to extract the archive somewhere on your local hard drive.

If the archive is of the JPA type, you'll have to use Akeeba eXtract Wizard, available without charge from our website.

If you have a ZIP package, there are a couple of gotchas. If you are working on a Linux machine, unzip will work just fine. If you're on Windows and under certain configuration circumstances on the server you took the backup on, you might not be able to extract it with WinZIP, WinRAR, 7-Zip or other archiver software. So you'll have to use Akeeba eXtract Wizard available for free from our website. This is a GUI utility which allows direct extraction of backup archives on your Windows™ PC. It is possible to run it under other operating systems, such as Mac OS X™ and Linux™, using DarWINE and WINE respectively. Please refer to the Akeeba eXtract Wizard documentation, available on-line on our site, for more information on using it.

Step 3. Editing your database backup.

Take a look at the directory where you extracted your backup archive. Inside it there is a directory named `installation`. Inside this, there is a subdirectory named `sql`. Inside this there is a file, `site.sql` (older versions: `joomla.sql`), containing your database data. *COPY THIS TO ANOTHER LOCATION NOW!* We'll have to edit it, so please, don't tamper with the original, will you?

Open the copy of `site.sql` (older versions: `joomla.sql`). Use a text editor (we recommend gedit and Kate on Linux™, Notepad++ on Windows™; do not use Wordpad or Word!). If you were ever familiar with SQL, you'll recognize that each line consists of a single SQL command. But they have a problem: table names are mangled. You'll see that tables are in a form similar to #__banner instead of jos_banner. Ah, nice! We'll have to fix that.

Using your text editors Replace command, do the following changes:

- search for **CREATE TABLE `#__** replace with **CREATE TABLE `jos_**
- search for **DROP TABLE IF EXISTS `#__** replace with **DROP TABLE IF EXISTS `jos_**
- search for **INSERT INTO `#__** replace with **INSERT INTO `jos_**
- search for **CREATE VIEW `#__** replace with **CREATE VIEW `jos_**
- search for **CREATE PROCEDURE `#__** replace with **CREATE PROCEDURE `jos_**
- search for **CREATE FUNCTION `#__** replace with **CREATE FUNCTION `jos_**
- search for **CREATE TRIGGER `#__** replace with **CREATE TRIGGER `jos_**

The idea is to replace all instances of #__ (note that there are two underscores after the hash sign) with jos_ in the SQL command part (not the data part). DO NOT PERFORM A BLIND SEARCH AND REPLACE OF #__ WITH jos_ AS IT WILL CAUSE SEVERE PROBLEMS WITH SOME COMPONENTS. Easy, wasn't it? *NOW SAVE THAT FILE*!

Step 4. Restoring the database.

In order to restore the database on the server you'll have to use some appropriate tool. For small to moderately sized database dumps (up to 2Mb), we find that phpMyAdmin [http://www.phpmyadmin.net] does the trick pretty well, plus it's installed on virtually all PHP enabled commercial hosts. For larger dumps, we found that bigdump.php from Alexey Ozerov [http://www.ozerov.de/bigdump.php] works wonders. Another useful and very easy (or, should I say, easier) to use tool which also works with PostgreSQL and Microsoft SQL Server is Adminer [http://www.adminer.org/]. Use either of those tools - or any other of your liking - to restore your database.

Step 5. Upload your site's files.

First of all, delete the installation subdirectory from the directory you extracted the backup archive to. We won't be needing this any more. Then, using FTP - or any method you please - upload all of the files to the target server.

If you want to be thorough remember to set the directory and file permissions accordingly. If you just want to get the damn thing on-line ASAP, just skip this permissions thing; it will remind you of itself as soon as you try to do some website administration (like uploading a picture) after the site's back on-line.

Step 6. Edit configuration.php, if necessary.

If you were restoring to the same server location you took the backup on, nothing else is necessary. Your site should be back on-line now. If not, you'll have to edit the `configuration.php`.

You have Joomla! 1.5.x. Good news! Joomla! 1.5.x doesn't require you to specify some of the hard-to-obtain parameters. Your `configuration.php` consists of several lines. Each one is in the following form:

```
var $key = "value";
```

The key is the name of the configuration variable and value (inside double quotes!) is the value of the variable. Below we provide a list of the configuration variables which have to be modified to get up on-line.

dbtype — is the database driver Joomla! will use. It can be mysql, mysqli (notice the extra i in the end), postgresql, sqlsrv or sqlazure. This depends on the kind of database you are using. If unsure, your best bet is mysqli.

host — is the database host name, usually localhost

user — is the database user name, assigned from your host company

password — is - obviously - the database password, assigned from your host company

db — is the database's name, assigned from your host company

dbprefix — is the database prefix; if you followed our instructions, it is jos_

live_site — Normally this is an empty string. If, however, your Joomla! site's front page looks as if all images and CSS files are not loading, you have to modify it and enter your site's base URL. For example, if the new site is located in `http://www.example.com/mysite/`, you have to locate the line starting with `var $live_site` and change it to become:

```
var $live_site = "http://www.example.com/mysite";
```

That's all! You're good to go.

Step 7. Enjoy success.

Your mission is accomplished. You are exhausted. Go drink whatever is your favourite drink and enjoy sweet success!

Chapter 6. Step by step guides

Even though the previous chapters provide a good reference, they assume that you know what you're doing. Many times, especially when you are a novice user, just the number of options can be intimidating. We are perfectly aware of that, hence this section. It is designed to get you up to speed with performing complex operations or creating advanced setups for your backup operation needs. It is not meant to be a thorough reference; if you have questions about how each of the individual settings work, you should refer to the appropriate section of the other chapters in this User's Guide.

1. Backing up your site to a cloud storage service

1.1. Introduction

For most of us, our websites are a key element to our business. Either being the business itself, or acting as the storefront to the Internet, they provide a significant added value. The last thing any web site owner want is to see their site defaced, damaged or even lost. Dangers lurk everywhere. From a simple human error in site administration to malicious activity and from hardware failure to natural disasters, no web server is the bulletproof vault we'd like it to be.

While nobody expects a catastrophe to hit his site, a good deal of precaution is required. It's pretty much the same rationale as in wearing a seatbelt while driving; you don't expect to crash, but if you do you most certainly want to evade the incident unharmed. The web site equivalent to a safety belt is none other than backup.

Web site backup comes with its own set of limitations and pitfalls. If you trust your web host for backup you might find your expectations fall short. Most hosts take daily backups – if any at all –on a secondary hard disk on the same server or, even worse, on a secondary partition of the same hard disk. If the server goes down due to a hardware fault, so does your backup. A few enlightened hosts also take backups on remote storage, for example NAS arrays. Even they do so on rather sparse intervals, for example twice per week. This means that on a complete catastrophe you will most assuredly lose a fair amount of data.

The solution is simple in concept. Take your own backups and store them on a cloud storage service, like Amazon S3 or even Dropbox. Taking your own backups means that you get to decide which data and how often has to be backed up, making sure that the crucial, regularly updated information routinely ends up in a backup archive. Using a cloud storage device adds a strong data safety clause to your procedure, while keeping costs low. Cloud storage is designed to be redundant and reliable, boasting a negligible risk of data corruption or data loss. Combined with its incredibly low cost (or even no cost for very low storage requirements!), it is reasonably attractive to businesses of all sizes: from hobbyists and sole proprietorships up to large corporations and government agencies.

But how can you implement this seemingly Utopian data protection scheme on your Joomla!™ site today, with the lowest possible cost? Enter Akeeba Backup Professional. The Professional edition sports significant features added on top of those offered to our free of charge Akeeba Backup Core edition (formerly known as JoomlaPack). One of those features we are going to use to accomplish our objective: transferring backup archives to cloud storage.

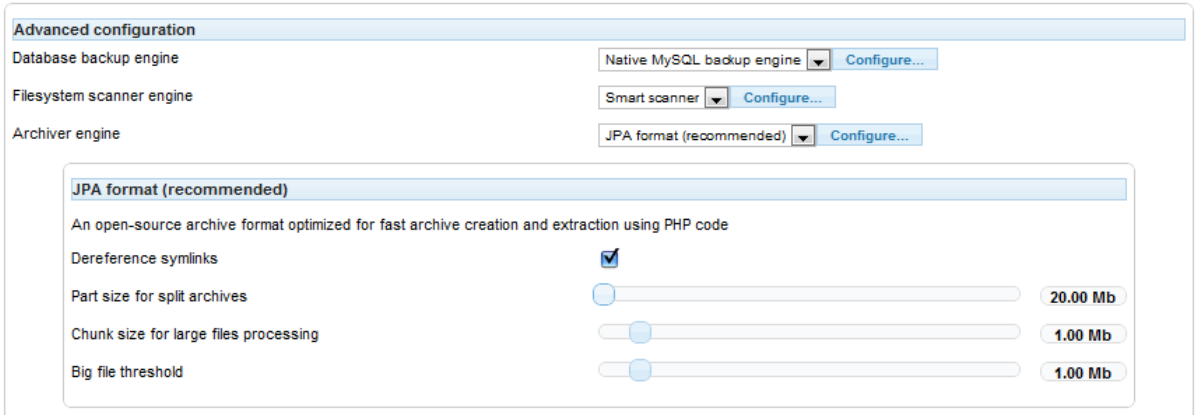
This section describes how to set up your site to store its backup archives to either Amazon S3 or Dropbox. More cloud storage providers will be added in the future. The setup always follows the same principle, no matter which cloud storage you want to use. Read along and you'll pick up the idea really fast.

1.2. Basic configuration

The most essential step is to download and install the Akeeba Backup Professional component to your site. In order to do that, you'll have to subscribe to the Professional download service first. After that, simply follow the step-by-step installation instructions. You can try to take your first, non-cloud backup to make sure that everything is in working order. If something goes wrong, just post as much information you can on our support forum. We will get back to you in 24-48 hours. Usually, we'll reply in much less time, even on weekends and bank holidays.

Provided that you are in your Joomla!™ administrator back-end, just click on the Components, Akeeba Backup menu item. In the Control Panel page which loads, click on the Configuration button. This will bring you to a quite lengthy configuration page. Locate the Archiver Engine setting in the pane titled Advanced Configuration. Click the button labeled Configure… next to it in order for the detailed settings to display. You should get something like this:

JPA Format settings for cloud backup



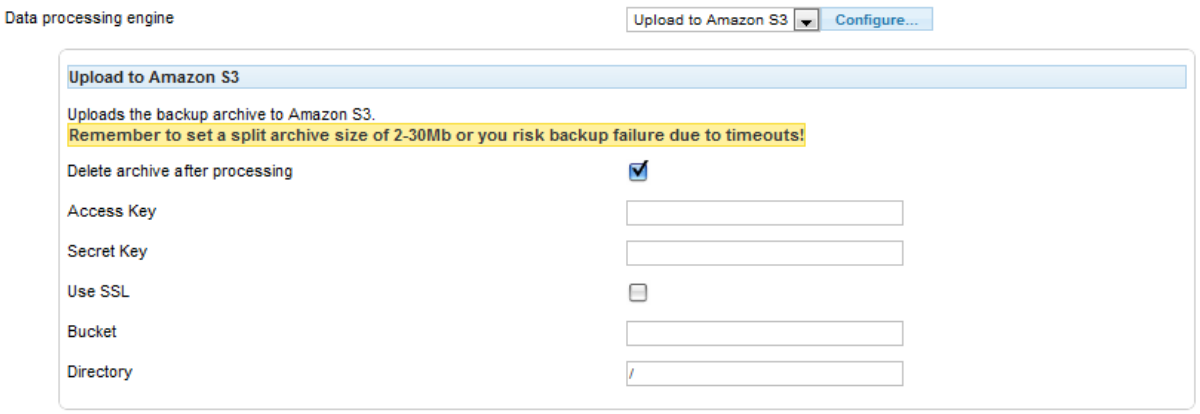
We will have to change just one option: Part size for archive splitting. Select the "Custom..." option and type in 20 in the text box that appears to the right of the drop-down. This setting will chunk our backup archive into multiple files, the maximum size of each one being the value of this setting.

You might wonder why we need to do that. PHP always has a strict time limit, i.e. the maximum time a PHP page may process data before the web server aborts it. Uploading the backup archives to cloud storage takes time, the exact amount of which depends on the size of the file and the network speed. The time limit and the bandwidth are beyond our control, so we can change the only parameter we can touch in order to avoid timeouts: the file size. Akeeba Backup Professional is smart enough to upload each part of the backup archive on a PHP page load of each own, so as to avoid timing out.

1.3. Using Amazon S3

If you've followed the instructions so far, it's Amazon S3 setup time! In the Configuration page, right below the Archiver Engine setting there's another setting called Data processing engine. Use the drop-down to select the Upload to Amazon S3 value and then click the button titled Configure… next to it. You should now see something like this:

Setting up the Amazon S3 engine



In this configuration details pane you have to enter your Amazon S3 Access key and Private key.

These are created by you in your Amazon S3 Console [https://console.aws.amazon.com/s3/]. If you have never done this, it's easy. Click on your name towards the top of the page and then click on Security Credentials. Follow the instructions on that page to create a new Access and Private key pair.

Back to our configuration page, checking the Use SSL setting will make your data transfer over a secure, encrypted connection at the price of taking a little longer to process. We recommend turning it on anyway. The Bucket setting defines the Amazon S3 bucket you are going to use to store your backup into. You also need to set the Amazon S3 Region to the region where your bucket was created in. If you are not sure, go to the Amazon S3 management console, right click on your bucket and click on Properties. The properties pane opens to the right and you will see the Region of your bucket. Back to our software, the Directory setting defines a directory inside the bucket where you want the backup files stored and must have been already created.

Warning

DO NOT CREATE BUCKETS WITH NAMES CONTAINING UPPERCASE LETTERS. AMAZON CLEARLY WARNS AGAINST DOING THAT. If you use a bucket with uppercase letters in its name it is very possible that Akeeba Backup will not be able to upload anything to it. More specifically, it seems that if your web server is located in Europe, you will be unable to use a bucket with uppercase letters in its name. If your server is in the US, you will most likely be able to use such a bucket. Your mileage may vary. The same applies with dots in the bucket name as they are incompatible with the Use SSL option due to limitations in Amazon S3.

Please note that this is a limitation imposed by Amazon itself. It is not something we can "fix" in Akeeba Backup (I did spent 5 hours on Christmas trying to find a workaround, with no success, because it's a limitation by Amazon). If this is the case with your site, please **DO NOT** ask for support; simply create a new bucket whose name only consists of lowercase unaccented latin characters (a-z), numbers (0-9) and dashes.

Do note that as per S3 standards the path separator for the directory is the forward slash. For example, writing `first_level\second_level` is wrong, whereas `first_level/second_level` is the correct form. I recommend using one bucket for nothing but site backups, with one directory per site or subdomain you intend to backup. If you want to use a first-level directory, just type in its name without a trailing or leading forward slash.

Tip

Should you need a visual interface for creating and managing Amazon S3 buckets, you can do so through your Amazon S3 console.

Enough said. Click on Save & Close to store the changed settings. Back to the Akeeba Backup Professional Control Panel, click on the Backup Now icon. It's backup time!

Ignore any warning about the Default output directory in use. We don't need to care about it; our backup archives will end up securely stored on Amazon S3 anyway. Just click on the big Backup Now! button and sit back. The upload to Amazon S3 takes place in the final step of the process, titled Finalizing the backup process. If during this stage you observe that the timeout bar – the bar which looks like a progress bar – fills all the way to the right, you have a timeout error. This means that you have to go back to the configuration and lower the Part size for archive splitting setting.

Important

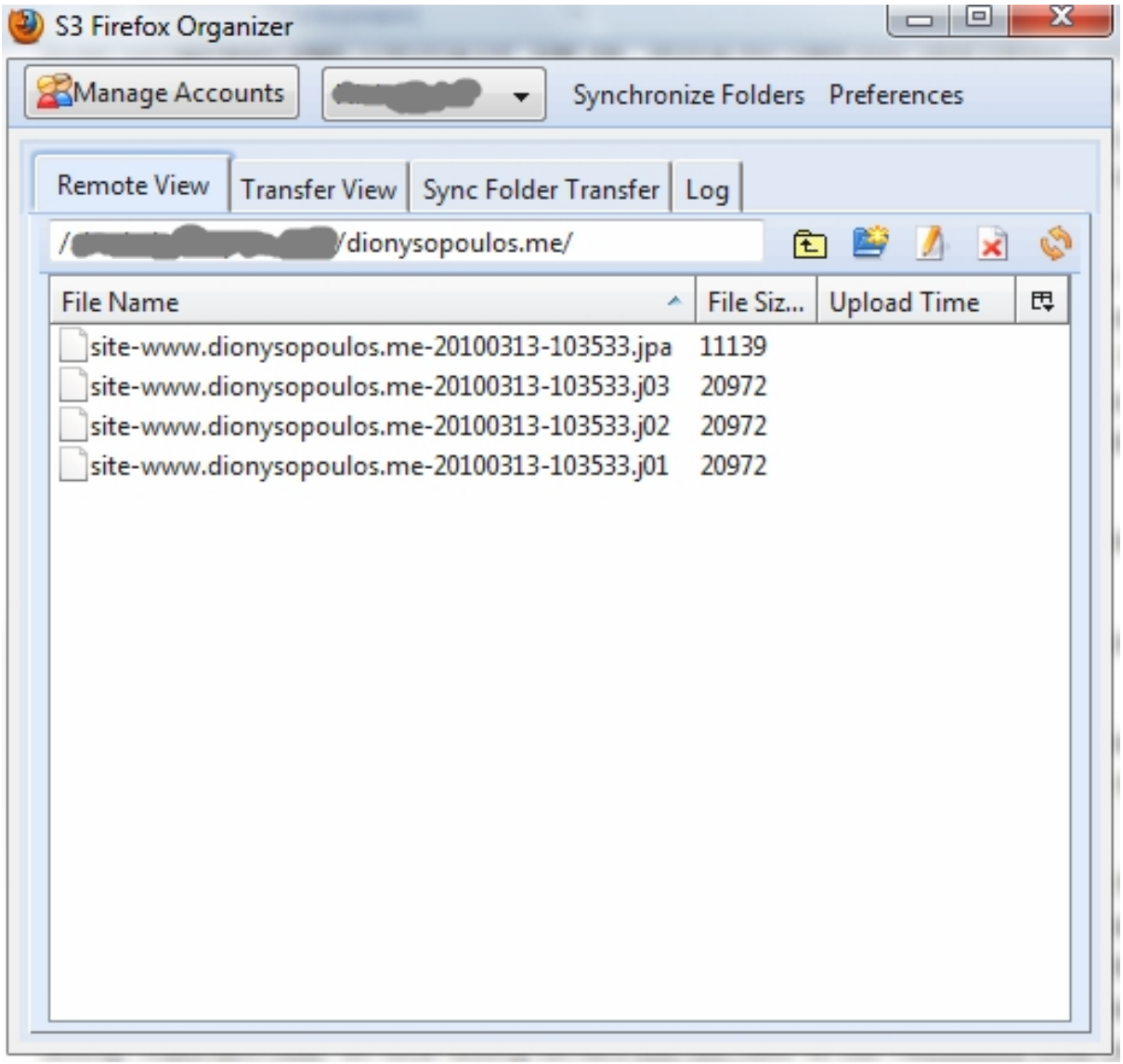
On local testing servers you will have to use ridiculously small part sizes, in the area of 1-5Mb, as the xDSL consumer Internet service has a much more limited bandwidth than your host.

Warning

If you get a RequestTimeout warning while Akeeba Backup is trying to upload your backup archive to the cloud, you MUST go to the Configuration engine and enable the "Disable multipart uploads" option of the S3 engine. If you don't do that, the upload will not work. You will also have to use a relatively small part size for archive splitting, around 10-20Mb (depends on the host, your mileage may vary).

As you can see, I just backed up my personal blog to Amazon S3:

A backup stored on Amazon S3



1.3.1. Making your backups accessible by other Amazon S3 accounts

Often, you may find yourself in need to have one write-only user to upload the backup archives to Amazon S3 for security reasons. In that case, you need to make the backups accessible for read/write by other accounts. You can do so with Amazon IAM Policies. There are two ways to do that, with the console or with a graphical environment.

The following methods were shared with us by members of our community. We have not tried them thoroughly, but they all seem to work without any known issues.

Using Amazon's graphical interface

1. Create bucket to store backups in.

2. Log into the AWS Management Console.

3. Click on the "IAM" tab.

4. Click Users under the Navigation section.
5. Click Create New Users.
6. Enter the name of the new user.
7. Copy & paste the credentials to a text editor.
8. Click on the newly created user.
9. Go to the user's Permissions tab.
10. Click Incline Policy and then Attach Policy
11. Choose `Custom Policy`.
12. Enter the following Policy Document:

```
{
    "Version": "2012-10-17",
    "Statement": [
  {
   "Sid": "AllowGroupToSeeBucketListInTheConsole",
   "Action":["s3:ListAllMyBuckets"],
   "Effect": "Allow",
   "Resource": ["arn:aws:s3:::*"]
  },
        {
            "Sid": "AllowListBucket",
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketLocation",
                "s3:ListBucketMultipartUploads"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::mybucket"
            ]
        },
  {
            "Sid": "Stmt1416670692010",
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:ListMultipartUploadParts"
            ],
            "Resource": [
                "arn:aws:s3:::mybucket/*"
            ]
        },
        {
            "Sid": "AllowPutObjectFullBackup",
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:DeleteObject",
                "s3:DeleteObjectVersion",
                "s3:PutObject",
                "s3:PutObjectAcl",
```

```
                "s3:PutObjectAclVersion"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::mybucket/site-www.example.com*"
            ]
        }
    ]
}
```

where *mybucket* is the name of your S3 bucket and *site-www.example.com* is the beginning of your backup archive's name.

If you are storing backups in a subdirectory substitute `"arn:aws:s3:::mybucket/site-www.example.com*"` with something like `"arn:aws:s3:::mybucket/my_directory/site-www.example.com*"` where *my_directory* is the path to the directory you store your backup archives in. If you are storing them in some/deeper/directory (three levels deep) the above would be `"arn:aws:s3:::mybucket/some/deeper/directory/site-www.example.com*"`. Likewise, if you use date-based directories, such as my_directory/2016-03-01, you could use something like `"arn:aws:s3:::mybucket/my_directory/*/site-www.example.com*"`.

13.Click Apply Policy

That's it! Now you can use those S3 credentials in your Akeeba Backup profile.

Using a third party application

In this example, we will use Cloudberry S3 Explorer Pro which has an "Access Manager" function that make this process very easy. There is a 15 day free trial you can use to set this all up. The full version costs about $39.

You can create as many groups as you want and easily create policies for each group. Just create a group and policy for each subdirectory in your S3 bucket. Each subdirectory is supposed to contain backups of a single website. Then create a user to put in each group. Once that is done you can right click on the user and select "Manage Access Keys". That will give you the S3 Access keys you need to enter into your Akeeba Backup profile.

The only gotcha is that when you create the policy and are browsing for the bucket and directory to apply the "PutObject" rights to, it doesn't add the "/*" at the end that is needed to make this work. You have to specify the resource and then go back and edit it so you can add the slash and star.

Further thoughts

Giving a user only the PutObject privilege allows the user to upload backups to the bucket, but any remote quotas will fail, as the user is not allowed to delete any objects (files) in your S3 bucket. You can opt for a compromise between tight security and ease of use. If you give your user both the PutObject and DeleteObject privileges he will be able to upload backups (PutObject) and run quotas (DeleteObject) but not list or download backup archives. In other words, even in the unlikely event that an attacker gains complete access to your site's database *and* filesystem, recovers the encrypted contents of Akeeba Backup's configuration table and encryption key file, writes a script to decode the configuration, gets access to your S3 credentials and tries to use them, he won't be able to download backup archives or even use those S3 credentials with any graphical S3 tool.

1.4. Using Dropbox

Under some circumstances using a for-a-fee cloud storage service may be beyond the budget of the client, as is usually the case with personal or very small business websites. Dropbox offers an inexpensive storage service, giving out the first 2 Gb of storage for free. Moreover, they offer a desktop client application which synchronizes the files stored in the cloud with those stored locally on a specified directory. Within the scope of backup, this is a very desirable feature, as it allows for automatic redundant storage of the backup archives on the local PC (actually, on any number of local PCs!), without any manual intervention.

To this end, we decided to include support for Dropbox storage. If you've followed the instructions so far, it's Dropbox setup time! In the Configuration page, right below the Archiver Engine setting there's another setting called Post-processing engine. Use the drop-down to select the Upload to DropBox (v2 API) option and then click the button titled Configure… next to it. You should now see a pane opening below this row.

In this configuration details pane you have to login to Dropbox. It is a two step authentication process. First click on the Authorisation - Step 1 button. It will open a popup box which will ask you to log in to Dropbox. This popup executes a page in Dropbox's servers, so that Akeeba Solo / Akeeba Backup never knows your email and password. After logging in, you are asked to press a button to transfer the access token to your Akeeba Backup / Akeeba Solo installation. Please do so. The popup closes automatically. Should you want to revoke this access, you can do that from your Dropbox account.

Tip

You can follow the above procedure to connect as many sites as you please.

The Directory defines a directory inside your Dropbox account where you want the backup files to be stored and must have been already created. Do note that as per Dropbox.com standards the path separator for the directory is the forward slash. For example, writing `first_level\second_level` is wrong, whereas `first_level/second_level` is the correct form. I recommend using one directory for nothing but site backups, with one subdirectory per site or subdomain you intend to backup. If you want to use a first-level directory, just type in its name without a trailing or leading forward slash.

Tip

If you have installed the Dropbox desktop client application on your PC you can simply create the directory on your local Dropbox directory. The desktop client application will automatically synchronize the folders to your on-line account.

Click on Save & Close to store the changed settings. Back to the Akeeba Backup Professional main page, click on the Backup Now icon.

Ignore any warning about the Default output directory in use. We don't need to care about it; our backup archives will end up securely stored on Dropbox anyway. Just click on the big Backup Now! button and sit back. The upload to Dropbox takes place in the final step of the process, titled Finalizing the backup process. If during this stage you get an error it means that you have to go back to the configuration and lower the Part size for archive splitting setting.

Important

On local testing servers you will have to use very small chunk sizes, in the area of 1-5Mb, as the xDSL consumer Internet service has a much more limited upload rate than your host.

1.5. Where to go from here?

Backing up your site to the cloud is the first step to backup autonomy and data safety. However, you still have to login to your site's back-end to take a cloud backup. This is suboptimal. What about when you are on the road for days, without reliable Internet connection? What about not wanting to go through this daily drill?

I am 100% behind you on this. I don't like routine either. You know, programmers are lazy and get bored easily.

With Akeeba Backup Professional you have three (that's not a typo, three) different options to automate your backup! Two of them are designed to utilize your host's CRON scheduling, i.e. your host's ability to run specific commands on his server, on a predefined schedule. This would mean that your backup is fully automated; you sleep at night, your site backs up itself.

You can read more about Akeeba Backup's scheduling options in the Automating your backup section of this User's Guide.

Overall, the Amazon S3 upload was our first, successful experiment in adding affordable, enterprise-grade qualities to full a site backup solution. Even though that's years ahead of the competition, we do not settle with it. Akeeba Backup has always been in very active development. Our desire to push the envelope is a core ingredient of the philosophy behind the software. As a result, S3 - and Dropbox - was just the beginning. Our roadmap includes support for several options of taking the backup off your server. We will try to integrate practically all major storage facilities, as long as they have a publicized integration API. If you have a specific need not covered by our base software, just contact us. We listen carefully to the community feedback and we make the impossible happen. All that for a very low subscription fee to the Professional downloads service.

1.6. Alternatives to cloud storage

Many people inquire us about the possibility of integrating with one of the myriads of free or low cost "cloud" storage providers that have started popping up since mid-2016. In most cases this is impossible either because the storage provider doesn't offer an API for developers or because most of them provide an API which only works for other services using this storage provider to store media files (images, audio, video) of their clients. In the latter case you can't use them for storing backups either due to restrictions in their terms of service or their API. In fact, we consider inquiries for using these services rather misguided for two reasons: cost vs risk and technical deficiencies.

Cost vs risk. We have found that the lowest cost per GB is actually offered by Amazon S3. Taking everything into account (including failure rate of hard disks, data redundancy and availability, cost of electricity and so on) they do offer a price nobody can beat. Not even other behemoths of the industry such as Microsoft. The only other company that can compete with their prices is the one other company with vast storage services, Google. In either case, the cost of storing your backups (assuming daily backups kept for 30 days and one monthly backup stored forever) are approximately 50% of your site hosting costs. When you have a mission critical site, e.g. an e-commerce site, backups is NOT where you want to cut costs. Backups are vital to your site's health, therefore the small cost of retaining them is disproportionately small to the cost of the risk of losing backup data when your site is in need of restoration. The operation of your site will compensate this amount extremely fast: in most cases the first day of the site's operation in a year will have made enough money to offset all hosting and backup storage costs for the entire year ahead. Even better for cost management, you can move older backups (over 6 months old) to Amazon Glacier where it is still available, but with a 4 hour delay. It's equivalent of storing backup tapes in a fire-proof safe in an off-site location, i.e. how corporations archive their older backups.

For personal and non-critical sites it's a different story. In this case the cost does matter and you do have two very low cost solutions:

A. Using Dropbox. Use a free Dropbox account to store your backup archives. When it gets close to filling up move your old backups from Dropbox onto external media such as external hard disks, flash drives, DVDs etc.

B. Using a cheap NAS (network attached storage) with FTP or SFTP support installed in your office / home. You just need a free DynDNS account to make it accessible over the Internet and use Akeeba Backup Professional's Upload to Remote FTP / SFTP to store the backups directly to your NAS. If you are more technically inclined you can easily build your own NAS solution using an inexpensive Raspberry Pi and a cheap external hard drive for less than $80. Or you could use that old PC that's collecting dust in your basement.

All and all, you have to weigh the risk factors involved in your choice. If you value data preservation you will need to use a proper cloud storage provider for a small fee. If you put cost and/or ownership of your data before data preservation you can use a cheap NAS solution to store your backups right at home / office. In any case, backups are best stored using one of these two solutions. Going for a cheap / free "cloud" storage is really only an option for consumers who want to store their photos, music and home videos – and don't care if anyone else gets a peek at them for marketing or more nefarious reasons.

Part II. Security information

Table of Contents

Chapter 7. Introduction

1. Foreword

Since you have chosen Akeeba Backup for backing your site up, it is obvious that you are using Joomla!™ as your web-based Content Management System. By using Joomla!™ you have embarked to the joyful adventure of managing a PHP powered website. Usually, this last part is gone unnoticed. The fact that you are using a PHP application is often taken for granted, but when it comes down to security and problem solving, this is the key concept of which you should have a strong grasp.

This part of the documentation deals with the basic concepts of PHP website management and their implications upon using Akeeba Backup. In this part, we will see the intricacies of access permissions, web site users and the impact of various PHP settings on your site's operability and security. This is not meant to be a concise manual on website administration. There are plenty of web and off-line resources with more in-depth information on the subject, but this introduction will quickly get you up to speed.

This document is no light reading; it is purposely sprinkled with a lot of tech-talk, albeit explained in layman's terms. Our objective was not to write a document which can be read and understood in a single reading. Some things you will understand by the first time you'll have read it. Most of it you will only get it after reading it again. A few shady areas will only become clear reading over again and referring to it every time you get stuck managing your site.

2. Why you need to care about ownership and permissions?

Most probably your server is running on Linux™, or another UNIX™-derivative operating system. You might have read, or heard, how these operating systems are safer and more secure than others. This is just half the story. The real security power of such operating systems stems from the way they manage files and directories, allowing or disabling access to them depending on who asks for it and what he's trying to do.

This management is pretty much like electricity in the Western world. It never gets in your way and you don't think about it, but you must have some basic understanding of it so as not to run the risk of getting toasted by it. That's how it goes with ownership and permissions. You might not think about them a lot, but potentials crackers do. If you don't manage permissions wisely, you might be creating a security hole on your server which can be exploited by a malicious cracker. Nobody wants his site cracked, right?

The following chapter will analyze how your web server works under the hood, so that you can grasp the third chapter, which analyzes all the ways you can secure your backup files so as not to fall prey on a cracker.

Chapter 8. How your web server works

1. Users and groups

The concept of users is the fundamental block of ownership separation on multiuser operating systems. All Windows™ versions based on the NT kernel are such; Windows™ NT, 2000, XP, Vista are all multiuser operating systems. Other UNIX variants are also inherently multiuser, including Linux™ , BSD™ flavours, MacOSX™ , etc. Since most web servers capable of running Joomla!™ are based on Linux™ , we will talk about the Linux™ user system, which is in fact the same as the UNIX user system; after all, GNU/Linux is nothing but an open-source UNIX variant which became very popular among geeks and recently among other people, too.

1.1. Users

As we mentioned, the fundamental block of ownership separation is a *user* . Each user has an entry in the system's password database and consists of a *user name* and a numeric *user ID* . A user is not necessarily linked to a physical person; in fact, most utilities and services create and operate under a user of their own.

The numeric user ID is an unsigned integer, therefore it can take a value between 0 and 65534. The user name and the numeric user ID are usually linked with an one to one relationship, meaning that if you know either one you can find the other one. The exception to this is most ISPs. In this case, because there are more users than the available number of user IDs, some numeric IDs will be reused, breaking the one to one relationship. However, on most - if not all - hosts, the one to one relationship exists.

Some user IDs are special. By convention, user IDs below 500 are reserved for system users. These are special users which are not assigned to some physical person. One of them, zero (0), has a very special meaning; it is assigned to the *super user* , commonly called *root* . This user is the God of the system. He has unlimited powers. He can override all access restrictions and make any kind of modification. For this reason, no sane system administrator logs in under that user. They will always log in under a normal user and only temporarily log in as root whenever they need to change system-wide settings.

1.2. Groups

Defining permissions per user is tiresome on systems which have more than a few users. In order to combat this inconvenience, all UNIX systems have the notion of *groups* . A group is nothing but a collection of users. The relationship to users is a many-to-many relationship, meaning that one user can belong to many groups and one group can contain many users. To keep things dead simple, groups have the same format as users. Each group has a *group name* and a numeric *group ID* . Again, not all groups are linked to a physical person; in fact there are a number of de facto group names used to control access to crucial system resources.

The numeric group ID is an unsigned integer, therefore it can take a value between 0 and 65534. The group name and group ID are linked with an one to one relationship, meaning that if you know either one you can find the other one. I am not aware of exceptions to this rule and I can't think a reason, either.

There are some special group ID's. By convention, zero (0), is assigned to the root's group. Its sole member should be root, or other users with a user ID of 0. It empowers its members to do anything they please on the system, almost like the user ID 0 does. Noticed the "almost" part? Belonging to the root group alone, without having a user ID of 0, does not give you infinite powers but it *does* grant you very broad access indeed!

Every user can belong to many different groups. To simplify things a little bit, every user has a so-called default group. This means that one of the groups he is a member of will be his effective group, unless otherwise specified, in all operations.

1.3. How users and groups are understood by UNIX-derived systems

This section is a bit ahead of the rest of this chapter, I know that. The information contained here, though, clarify a lot of what will follow, so it seemed only appropriate to include it here.

Every time the system has to store the owning user and group of a system item, it does so by storing the numeric user and group IDs, not the names! The names are only used as a convenience; you can't remember that John's user ID is 637, but it's easy to remember that his user name is john. Likewise, remembering that group ID 22 controls access to the CD-ROM drive is improbable, while remembering that the group named cdrom does that is self-understood.

Important

User IDs for a user with the same user name on different systems can be different. A user named example on system A and system B might have one user ID on system A and a completely different on system B. However, all UNIX-derived systems really know about are IDs, not names!

This is very (read: extremely) important when you transfer files from one system to another. All archive types which store owner information (for example GNU `tar`) store nothing but the numeric ID's. Moving these to another system and extracting them will screw up ownership and permissions. Just because you have the user ID 567 on Host A doesn't mean that you won't end up with user ID 678 on Host B; extracting such an archive would make all your files owned by someone else, effectively screwing up your site.

2. Ownership

The term *ownership* implies that system items belong to someone. In the context of web site management the items we are interested in are files and *processes* . Everybody understands what files are, but the term *processes* is rarely understood amongst webmasters. So, let's explain it.

2.1. Process ownership

Every time you run a program, be it interactive or a system service, you create a process. A process is a piece of code being executed by the operating system. A process can *spawn* child processes which can spawn new *threads* . In layman's terms, a program can start other instances of itself or another program and they, in turn, can start small pieces of executable code which can run in parallel with the main program.

Programs do not start spontaneously. Someone has either got to start them, or instruct the system to start them when some criteria are met. This sentence is the acknowledgement of the simplicity behind a computer system; it can't think on its own, humans have to tell it what to do one way or the other. Based on how a program starts, it process will be owned by some user.

In the first and simplest case, when you start a program, the ownership is almost self-understood. You are logged in as some user, so the process of the program you have executed is owned by your user. It's simple as that. This also implies that the process has the same permissions as the owning user, that's why we say that the process runs *under* this user; its access level is at most as much as the owning user, so the process is *under* the user.

The other case, instructing the system to start a process, is somewhat different. Usually, the utilities which are used to start programs automatically are the system initialisation scripts, time-based execution programs (for example, `cron` and `at`), etc. All of these programs are in most cases owned by root and are executed under root privileges. On top of that, most programs started this way are system services, running as long as the system is up and running. But do you remember what we said before? Root is the God of the system. Normally, these programs would get root's privileges, posing a huge security hole. If there is a bug in the program and some malicious user exploits it, he could wreck havoc on the system; root is above all restrictions.

In order to combat this possibility, UNIX systems employ a feature which allows processes to *drop privileges* and run under a different user than the one which started them. In fact, they change their ownership! To prevent

abuse of this feature, a process must run under root privileges to be able to switch to another user. This feature is extensively used by system services, including MySQL and Apache.

In the context of web site management, Apache is of special interest. Apache is the de facto web server for Linux systems and is being used by over 50% of Internet sites, according to NetCraft's August 2008 survey. Chances are you are using it on your site, too. Apache, like most UNIX services (affectionately called *daemons*) uses the feature to drop privileges. The user and group under which it runs are defined in its configuration files. These configuration files are usually out of the reach of regular users (like you!) on commercial hosts, for security reasons.

There is a **special case** which acts as the exception to the Apache rule. Many commercial hosts run **suPHP**-enabled Apache installations. This is an extension to the normal PHP's mode of operation which allows each PHP page to run in a process owned by the file's owner (more on file ownership in the next sub-section). This means that each of the PHP files under your account on such a host run as the user which has been assigned to your account. And, if this still isn't apparent to you, such hosts nullify the burden of ownership and permissions (more on permissions in the next section). To put it clearly: with suPHP the file owner, your own user and the Apache user are one and the same. If you are looking for a decent host, find one which is using suPHP. It's better for security and removes a lot of administrative burden from you. A win-win situation.

2.2. File ownership

Everybody knows what a file is, right? Well, we all know intuitively what a file *might* be, but we seldom know what *exactly* it is. A file is actually consisted of at least two parts. The first part is the file data, what we intuitively understand as the file contents. The second part is the file system entry, which makes the file data an identifiable entity. This is where the operating system stores all kinds of information, such as how the file is named, where it is located in the file system hierarchy, when it was modified, etc. It also contains information about who owns the files and what are the file's permissions. You might be surprised reading this, but only this latter, informative, part is required for a file. Really!

It seems absurd to have a file without file data, but it is anything but that. There are some special "files" (more correctly: file system entries) in the UNIX world. You have devices, whose "files" actually point to a serial input/output provided by this device, for example the serial port of your computer. There are directories, which obviously don't have any data contained; they are used for organising files only. There are soft links, which are pointers to other files in the file system, used to have standardised names and locations on files which might be moved around or have varying names. There are also these wired beasts called "hard links", some peculiar file system entries which point to the file data of another file, making virtually impossible to know which is the "original" file and which is its clone. Their usefulness is only apparent to the UNIX gurus, therefore out of the scope of this document. For the purpose of website management we are only concerned about regular files (hereby called "files"), directories and soft links (hereby called "links").

All files, directories and links are owned by a user and a group, be they files or links. In fact, they are owned by a user ID and a group ID. Normally, the ownership is inherited by the creating process's ownership. When you create a file directly from an interactive editor application the editor's process is owned by your user ID and your default group ID, therefore the file will be owned by your user ID and your default group ID.

Links are a special case on their own. They are not files, they are pointer to files. The ownership (and permissions) of links is irrelevant. Whenever a process tries to access a link, the underlying operating system "follows" the link, until it finds a regular file. Therefore, the ownership that matters is that of the file linked to, not the link itself. This feature of the operating system prevents unauthorised access to arbitrary files, normally accessible to specific users only, from users who just happen to know the path to those files.

What is especially interesting is the correlation between FTP, web server and file ownership. Whenever you access FTP, you log in as some user. This user is linked to a system user (often the same user assigned to you by host), so logging in FTP actually has the same effect as logging into the system as this user. Common sense implies that all file operations are performed under this user and all files created (read: uploaded) through FTP will be owned by this user.

Conversely, whenever you are using a web interface to perform file operations, you are using a web application - or any PHP script/application for that matter - running on the web server whose process is owned by a different

user. Therefore, whenever you create files from a web application, they will be owned by the user the web server runs under.

The distinction of file ownership in these two cases is of paramount importance when you get stuck with files which are accessible to FTP but inaccessible to the web server, or vice versa. This minute distinction is the cause of a lot of grief to many webmasters, so beware!

3. Permissions

So far you have learned about users, groups and ownerships. But how do they all stick together? Why these are necessary to have in the first place? The reason is simple: security. In multiuser operating systems you normally don't like users snooping around other people's files, especially when those files contain sensitive information, such as passwords. The most common method for overcoming this problem is to assign *permissions* on each system item, controlling who can do what. This simple concept works wonderfully; it's like putting doors on a building and giving people only the keys for the doors to areas they should have access to.

3.1. The three types of permissions

We already learned that each system item is owned by a user ID and a group ID. Whenever a process tries to access a system item, the operating system checks the permissions and decides if it will proceed with the operation or deny access. It seems reasonable to have control over what a process with the same owning user ID can do with it, what the a process with the same owning group ID can do with it and, finally, what the rest of the world can do with it. Indeed, this is the rationale behind the three types of permissions we can define on UNIX systems. In order of precedence they are:

User permissions	They are the access rights granted to the owning user of the item. Every process with the same owning user ID as the item's owning user ID has these access rights. These access rights have precedence over all other permissions.
Group permissions	These are the access rights granted to the owning group of the item. Every process with the same owning group ID as the item's owning group ID has these access rights. These access rights are applied only if the owning user ID's of the process and the item do not match, but their owning group ID's match.
Other permissions	These are the access rights granted to the rest of the world. If the owning user ID's of the process and the item do not match and the same happens for the owning group ID's as well, these access rights will be applied.

3.2. What permissions can control

We will be focused on permissions on files and directories, the building blocks of a web site. Permissions can control only three different actions:

Read	The ability to read a file, or get a directory listing.
Write	The ability to write to a file, or the ability to create, rename and delete files and subdirectories on a directory.
Execute (or Browse, for directories)	For files, it controls the ability to be directly executable from the command line. It is only meaningful for binary programs and executable scripts. For directories, it controls the ability to change to that directory. Note that if this is disabled you can't usually obtain a directory listing and file read operations might fail.

These three actions, combined with the three access request groups (owning user, owning group and the rest of the world) give us a total of nine distinct operations which can be controlled. Each action is an on/off switch. If a permission is set, it is turned on and the right to perform the action is granted. If the permission is not set, the switch is off and the right to perform the action is not granted.

3.3. Permissions notation

The two most common notations for permissions is the *textual notation* and the *octal notation*. Each one has its own virtues.

3.3.1. The textual notation

The textual notation is traditionally used in UNIX long directory listing format and in most FTP clients listings as well. It consists of ten characters. The first one displays the file type. It can be one of dash (regular file), "d" (a directory) or "l" (a link). The following nine characters display the permissions, consisting of three groups of three letters each. The groups are in order of appearance: owning user, owning group and others. The permissions on each group are in order of appearance: read (denoted with r), write (denoted with w) and execute/browse (denoted with x). If a permission is not set, a dash appears instead of the letter.

For example, the string `-rwxr-xr-x` means that it is a regular file, the owning user has read/write/execute permissions, the owning group has read and execute permissions and so does the rest of the world. On the other hand, the string `dr-x------` indicates that we have a directory whose owning user has read and browse permissions and everybody else (owning group and the rest of the world) have no right to access it.

3.3.2. The octal notation

This is the de facto standard geeks use to communicate permissions. The benefit of this approach is that you only need four characters to fully define them and they're easier to read (to the trained eye, at least).

Permissions are in fact a bit field. Each permission is a bit which can be turned on or off. If you put bits together they form bytes (by grouping eight bits together). Many bytes one next to the other form a computer-readable representation of a whole number (an integer). If you write this down in base 8, you've got the octal representation. If you didn't understand this, it's OK. We'll explain it the easy way.

The octal notation consists of four numbers. In the context of web site management you can consider the first to be always zero and sometimes omitted. The next three numbers describe each one the permissions. The second number describes owning user permissions. The third number describes owning group's permissions. The fourth number describes the permissions for the rest of the world. Each number is 0 to 7. The meaning of each number is simple:

0 No access

1 Execute/browse access only

2 Write access only

3 Write and execute/browse access

4 Read access only

5 Read and execute/browse access

6 Read and write access

7 Full access

It is almost apparent that "1" stands for execute only, "2" stands for write only and "4" stands for read only. Adding these values together gives you the rest of the combinations. You can't add together the same value (1+1 is forbidden as it is meaningless), so each of the composite values can be broken down to its components very easily. You don't even have to memorise the whole table!

A permission of 0777 means that the owning user, owning group and the rest of the world can read, write and execute the file (full permissions for everyone). A 0764 permission means that the owning user has full access, the owning group has read and write access and the rest of the world have read only access.

Chapter 9. Securing your Akeeba Backup installation

1. Access rights

As with every software which can access your site as a whole, Akeeba Backup needs to control who's got access to its backup functionality. Due to the lack of a thorough ACL mechanism in Joomla! 1.0 and 1.5, we have decided to make the administrator (back end) of this component available by default to the Super Administrators only. This group of people already has infinite access to the access, making them the ideal candidate for backup operators. You can change this default behavior from the component's Parameters button in the Control Panel page.

The front-end backup feature is a different story. Since it has to be available to unattended scripts which can't use cookies and interactive user authentication, a different approach was taken. Instead of requiring the user to have logged in with Joomla! it uses a simple "secret word" authentication model. Because this "secret word" is transmitted in clear text we strongly advise against using it over anything else than a local network (for example, an automated tool running on the same host as the web server). If you have to use it over the Internet we strongly advise using a secure protocol connection (HTTPS) with a valid commercially acquired certificate.

If you want to enhance the security of your site, we strongly advise you to use a commercial-grade ACL system, such as Dioscouri's JUGA or `CorePHP` Community ACL on top of Akeeba Backup's rudimentary access control and Joomla! 1.6's ACL system. Such ACL systems allow you to fine-tune the permission settings down to the user and component view level, if so required. Using such an ACL scheme you can create, for example, a backup operator user who has access to the Backup Now and configuration pages of Akeeba Backup, but not the Download function.

2. Securing the output directory

Securing the backup output directory

By default the component uses a non secure location to store its backup files and temporary files, within your site's file system hierarchy, namely `administrator/components/com_akeeba/backup`. This location is well known and can be - theoretically - accessed directly from a web browser. Since the backup output directory stores the results of your backup attempts, that is SQL files containing database backups and archive files containing all of your site, a malicious person with access to this location could steal sensitive information or compromise your site's integrity.

The first line of defense, is to use mangled, hard to guess, names for the SQL backup. However, in the era of multi-MbPS xDSL Internet connections and scripting, it wouldn't take an attacker that long to figure out the filename. Remember: security through obscurity is no security at all!

As a second line of defense, we include a secure `.htaccess` on the default backup output directory to disable direct web access. However, this is only possible on Apache-powered web servers which allow the use of `.htaccess` files. You should check with your host to ensure that this kind of protection is possible on your site.

However, this is not enough. Security experts argue that storing backups within the potentially vulnerable system itself might be a security risk. It is possible that a malicious person could gain access via other means. Think of a simple scenario. You have an Administrator with a weak password a hacker eventually guesses. Now the hacker can log in to your site, but doesn't have access to the component. Despite that, you have installed a file administration component, such as eXtplorer, which allows administrators to browse the site's file system and download files. How long would it take before your site got compromised? Right. Not very long indeed!

The best approach is to use a directory which is outside your web server's root. By definition, this is not directly exposed to the web and is usually unavailable to file administration utilities.

3. Securing file transfers

Whenever you download your backup files you can fall prey to a malicious user. Backup files are transferred unencrypted (unless you access your site's administrator section through the HTTPS protocol). It is possible for a resourceful hacker to launch a man-in-the-middle attack. In such a case, whatever you download from your site will be directed to the hacker's computer before reaching yours.

To avoid such insecure scenarios, we advise against using the Download button in the backup administration page. We suggest that you use Secure FTP (SFTP) instead. Avoid using the plain old FTP, because your password and data are transmitted in clear text (unencrypted) over the Internet. Also avoid FTPS and FTPES (FTP over SSL) as they have some security restrictions, like requiring your FTP server to have a commercially obtained SSL certificate in order to be really effective. Sometimes, your host will allow secure access to a web based control panel which has a file download feature. You could use this, it's as safe as it gets.

There is also another reason why not to use the Download button in the backup administration page. Your host neither discriminates the back end and front end pages of your Joomla! site, nor your IP from the rest of the world. As a result, every time you use the Joomla!™ back end, the data transferred counts towards your monthly bandwidth quota. Backup archives are large, sometimes in the hundreds of megabytes. Transferring them through the Download feature will incur a huge loss on your monthly bandwidth quota. Using Secure FTP or your host's control panel *usually* does not count through the bandwidth quota and should be used instead. It's better to ask your host, though; some include the FTP and SFTP traffic in your monthly bandwidth quota. Finally, the Download feature doesn't work with all possible configurations and has objective problems with the handling of very large archives; this is a technical limitation which can not be overcome in the PHP level the component operates. Most notably, many servers which use the FastCGI mode do not work at all with the Download button. They will simply throw an HTTP 500 error page, or a "file not found" message. We've tried all the tricks in the book and then some more, but there's really absolutely nothing we can do about it. Sorry.

Important

The preferred and suggested method for downloading your backup files - for several reasons - is using FTP in BINARY mode, preferably over an encrypted connection. Alternatively, you can use Remote CLI which allows you to use this approach when downloading backup archives.

Part III. Appendices

Table of Contents

Appendix A. The JPA archive format, v.1.2

Design goals

The JPA format strives to be a compressed archive format designed specifically for efficiency of creation by a PHP script. It is similar in design to the PKZIP format, with a few notable differences:

- CRC32 is not used; calculation of file checksums is time consuming and can lead to errors when attempted on large files from a script running under PHP4, or a script running on PHP5 without the hash extension.
- Only allowed compression methods are store and deflate.
- There is no Central Directory (simplifies management of the file).
- File permissions (UNIX style) are stored within the file.

Even though JPA is designed for use by PHP scripts, creating a command-line utility, a programming library or even a GUI program in any other language is still possible. JPA is not supposed to have high compression rations, or be secure and error-tolerant as other archive formats. It merely an attempt to provide the best compromise for creating archives of very large directory trees using nothing but PHP code to do it.

This is an open format. You may use it in any commercial or non-commercial application royalty-free. Even though the PHP implementation is GPL-licensed, we can provide it under commercial-friendly licenses, e.g. LGPL v3. Please ask us if you want to use it on your own software.

Structure of an archive

An archive consists of exactly one Standard Header and one or more Entity Blocks . Each Entity Block consists of exactly one Entity Description Block and at most one File Data Block . All values are stored in little-endian byte order, unless otherwise specified.

All textual data, e.g. file names and symlink targets, must be written as little-endian UTF-8, non null terminated strings, for the widest compatibility possible.

Standard Header

The function of the Standard Header is to allow identification of the archive format and supply the client with general information regarding the archive at hand. It is a binary block appearing at the beginning of the archive file and there alone. It consists of the following data (in order of appearance):

Signature, 3 bytes	The bytes 0x4A 0x50 0x41 (uppercase ASCII string “JPA”) used for identification purposes.
Header length, 2 bytes	Unsigned short integer represented as two bytes, holding the size of the header in bytes. This is now fixed to 19 bytes, but this variable is here to allow for forward compatibility. When extra header fields are present, this value will be 19 + the length of all extra fields.
Major version, 1 byte	Unsigned integer represented as single byte, holding the archive format major version, e.g. 0X01 for version 1.2.
Minor version, 1 byte	Unsigned integer represented as single byte, holding the archive format minor version, e.g. 0X02 for version 1.2.
File count, 4 bytes	Unsigned long integer represented as four bytes, holding the number of files present in the archive.

Uncompressed size, 4 bytes	Unsigned long integer represented as four bytes, holding the total size of the archive's files when uncompressed.
Compressed size, 4 bytes	Unsigned long integer represented as four bytes, holding the total size of the archive's files in their stored (compressed) form

Extra Header Field - Spanned Archive Marker

This is an optional field, written after the Standard Header but before the first Entity Block, denoting that the current archive spans multiple files. Its structure is:

Signature, 4 bytes	The bytes 0x4A, 0x50, 0x01, 0x01
Extra Field Length, 2 bytes	The length of the extra field, without counting the signature length. It's value is fixed and equals 4.
Number of parts, 2 bytes	The total number of parts this archive consists of.

When creating spanned archives, the first file (part) of the archive set has an extension of .j01, the next part has an extension of .j02 and so on. The last file of the archive set has the extension .jpa.

When creating spanned archives you must ensure that the Entity Description Block is within the limits of a single part, i.e. the contents of the Entity Description Block must not cross part boundaries. The File Data Block data can cross one or multiple part blocks.

Entity Block

An Entity Block is merely the aggregation of an Entity Description Block and at most one File Data Block. An Entity can be at present either a File or a Directory. If the entity is a File of zero length or if it is a Directory the File Data Block is omitted. In any other case, the File Data Block must exist.

Entity Description Block

The function of the Entity Description Block is to provide the client information about an Entity included in the archive. The client can then use this information in order to reconstruct a copy of the Entity on the client's file system. It is a binary block consisting of the following data (in order of appearance):

Signature, 3 bytes	The bytes 0x4A, 0x50, 0x46 (uppercase ASCII string “JPF”) used for identification purposes.
Block length, 2 bytes	Unsigned short integer, represented as 2 bytes, holding the total size of this Entity Description Block.
Length of entity path, 2 bytes.	Unsigned short integer, represented as 2 bytes, holding the size of the entity path data below.
Entity path data, variable length.	Holds the complete (relative) path of the Entity as a UTF16 encoded string, without trailing null. The path separator must be a forward slash (“/”), even on systems which use a different path separator, e.g. Windows.
Entity type, 1 byte.	• 0x00 for directories (instructs the client to recursively create the directory specified in Entity path data). • 0x01 for files (instructs the client to reconstruct the file specified in Entity path data) • 0x02 for symbolic links (instructs the client to create a symbolic link whose target is stored, uncompressed, as the entity's File Data Block). When the type is 0x02 the Compression Type MUST be 0x00 as well.
Compression type, 1 byte.	• 0x00 for no compression; the data contained in File Data Block should be written as-is to the file. Also used for directories, symbolic links and zero-sized files.

- 0x01 for deflate (Gzip) compression; the data contained in File Data Block must be deflated using Gzip before written to the file.
- 0x02 for Bzip2 compression; the data contained in File Data Block must be uncompressed using BZip2 before written to the file. This is generally discouraged, as both the archiving and unarchiving scripts must be ran in a PHP environment which supports the bzip2 library.

Compressed size, 4 bytes	An unsigned long integer representing the size of the File Data Block in bytes. For directories, symlinks and zero-sized files it is zero (0x00000000).
Uncompressed size, 4 bytes	An unsigned long integer representing the size of the resulting file in bytes. For directories, symlinks and zero-sized files it is zero (0x00000000).
Entity permissions, 4 bytes	UNIX-style permissions of the stored entity.
Extra fields data, variable length	The extra fields for each file are stored here. The total length of extra fields is included in the Block Length above

Each Extra Fields consists of:

Extra Field Identifier, 2 bytes	A signature denoting the data stored in the extra field
Extra Field Length, 2 bytes	The length (in bytes) of the Extra Field Data
Extra Field Data, variable length	The internal structure varies by the type of the Extra Field, as noted in the Extra Field Identifier

Timestamp Extra Field

Its purpose is to store the date and time the file was modified. This extra field should be ignored for directories and symlinks, or - if present - the Timestamp should be set to 0x00000000. Its format is:

Extra Field Identifier, 2 bytes	The bytes 0x00 0x01
Extra Field Length, 2 bytes	The value 0x08 stored in little-endian format
Timestamp, 4 bytes	A 4-byte UNIX timestamp of the file's modification time, as returned by filemtime().

File Date Block

The File Data Block is only present if the Entity is a file with a non-zero file size. It can consist of one and only one of the following, depending on the Compression Type:

- Binary dump of file contents or textual representation of the symlink's target, for CT=0x00
- Gzip compression output, without the trailing Adler32 checksum, for CT=0x01
- Bzip2 compression output, for CT=0x02

Change Log

Revision History

June 2009	NKD, Akeeba Developers	http://www.akeebabackup.com

Updated to format version 1.1, fixed incorrect descriptions of header signatures

Appendix B. The JPS archive format, v.2.0

Design goals

The JPS format strives to be a compressed archive format designed specifically for efficiency of creation by a PHP script, while providing secure AES-128 encryption of the file descriptor and file contents. It is similar in design to the JPA, with a few notable differences:

- Both the file descriptor and the file data are split to 64Kb blocks encrypted using Rijndael-128 in CBC mode (that's the same as AES-128)
- All files are compressed using Deflate (ZLib)

Even though JPS is designed for use by PHP scripts, creating a command-line utility, a programming library or even a GUI program in any other language is still possible. JPS is supposed to have low to medium compression rations, and be secure. However it is not as error-tolerant as other archive formats.

This is an open format. You may use it in any commercial or non-commercial application royalty-free. Even though the PHP implementation is GPL-licensed, we can provide it under commercial-friendly licenses, e.g. LGPL v3. Please ask us if you want to use it on your own software.

Important

When the password is blank, no encryption takes place. Archivers should take this into account when creating files. Unarchivers should also take this into account when the user passes an empty string as their password.

When a non-blank password is used, all files are encrypted using the same password. More specifically, all data blocks are encrypted using the same password.

Security

The security of the format largely hinges on the assumption that Rijndael-128 in CBC mode with randomized IVs in each encrypted stream is not susceptible to KPA (known plain-text attacks). Should a KPA be found against the encryption algorithm the obvious crib would be the encrypted file header of the first file in an Akeeba Backup / Akeeba Solo archive which is very predictable. Even if the order of files were randomized, there are well-known files (part of the installer) with known contents, making them relatively easy to identify by their relative size in the archive. However, as we said above, the encryption algorithm is not known to be susceptible to KPA, nullifying this threat.

Another defense you can use when creating the archive is the use of a non-static salt for PBKDF2 key expansion. This means that the cryptographic key which could theoretically be brute forced by means of a KPA would only apply to a specific encrypted block. It would then take another, more computationally expensive, brute force attack against the password to decrypt the entire archive. The downside is that this is a much slower encryption method since a key needs be derived for every encrypted block of data. Counter-intuitively this could lead to worse security since the practical considerations of the implementation lead to using a much smaller number of iterations with a weaker hashing algorithm which may end up being easier to brute-force, especially for the shorter passwords.

Our recommendation for v2.0 archives is using key expansion with a static salt, a high number of iterations (e.g. 64000) and a strong hashing algorithm (e.g. SHA512).

Key Expansion

JPS v.2.0 (PBKDF2)

JPS v.2.0 is using a different, more secure, key expansion scheme that JPS v.1.x. PBKDF2 is used on the user-supplied password to generate the encryption key. PBKDF2 was selected over memory-hard algorithms (like bcrypt, scrypt, Argon2 etc) for performance reasons, considering that encryption has to also take place on shared hosts with limited resources and old versions of PHP which don't even support these newer hashing algorithms. As processors get faster and old PHP versions become increasingly obsolete we might revise the key expansion algorithm in the future.

The supported PBKDF2 algorithms at this time are SHA-1 (used by default), SHA-256 and SHA-512. The algorithm used throughout the archive is specified in the archive header. Even though SHA-1 is not collision-resistant, the high number of iterations mitigates that risk.

The number of iterations used throughout the archive is also specified in the archive header. By default it's 100,000. This is a moderately high number of iterations while still being practical on resource-limited shared hosting.

There are two possibilities for the salt used for PBKDF2. One possibility is using a static salt, found in the archive's header. In this case you only perform key expansion once and use the expanded key for all encrypted blocks in the archive. The other possibility is having a different salt per encrypted block. In this case a key expansion is executed per encrypted block, therefore using a different encryption key for each block.

Important

We STRONGLY recommend using long (64 or more characters), completely random passwords which make use of lowercase and uppercase Latin letters, numbers and special characters (top row on US-format keyboards). Use a password manager to generate and store these passwords. Taking these precautions make password brute forcing with conventional technology highly impractical in the foreseeable future.

JPS v.1.x (Rijndael-128 CTR)

All JPS v.1.x format use a very naive key expansion, based on Rijndael-128 running in CTR (counter) mode. The implementation details can be found in the Encrypt class' expandKey method. The obvious downside is that only up to 16 bytes of the password (which may be as little as 5.3 characters in UTF-8 encoding) are taken into account. The other obvious downside is that the key is simply the password being encrypted with a version of itself in CTR mode which is not very cryptographically safe. The shortcomings of this approach were exacerbated in the first public version of the JPS format (1.9) which used the key as an IV for all encrypted blocks, weakening the security of the format.

This key expansion is not supported since JPS v.2.0.

Structure of an archive

An archive consists of exactly one Standard Header and one or more Entity Blocks . Each Entity Block consists of exactly one Entity Description Block and at most one File Data Block. Each FIle Data Block consist of one or several Data Chunk Blocks. All values are stored in little-endian byte order, unless otherwise specified.

All textual data, e.g. file names and symlink targets, must be written as little-endian UTF-8, non null terminated strings, for the widest compatibility possible.

Standard Header

The function of the Standard Header is to allow identification of the archive format and supply the client with general information regarding the archive at hand. It is a binary block appearing at the beginning of the archive file and there alone. It consists of the following data (in order of appearance):

Signature, 3 bytes	The bytes 0x4A 0x50 0x54 (uppercase ASCII string “JPS”) used for identification purposes.
Major version, 1 byte	Unsigned integer represented as single byte, holding the archive format major version, e.g. 0x02 for version 2.0.
Minor version, 1 byte	Unsigned integer represented as single byte, holding the archive format minor version, e.g. 0x00 for version 2.0.
Spanned archive, 1 byte	When set to 1, the archive spans multiple files
Extra header length, 2 bytes	The total length of extra headers. In version 2.0 of the format it is always 76.

The total size of this header is 8 bytes, plus the size of the extra headers (if any).

Key Expansion Extra Header

The function of the Key Expansion Extra Header is to let you know of the PBKDF2 key expansion algorithm's configuration parameters used throughout this backup archive. It consists of the following data:

Identification Header, 4 bytes	The bytes 0x4A 0x48 0x00 0x01 used for identification purposes
Extra Header Size, 2 bytes	Unsigned short integer, little endian, holding the total size of this extra header (including the 4 bytes of the identification header), i.e 76 for a version 2.0 header
Algorithm, 1 byte	Unsigned byte holding the ID of the hash algorithm used for PBKDF2. The valid algorithms are: • 0 = SHA-1 • 1 = SHA-256 • 2 = SHA-512 Values up to and including 127 are reserved for future use.
Iterations, 4 bytes	Unsigned long integer, little endian, with the number of iterations to use in PBKDF2
Use Static Salt, 1 byte	Unsigned byte. When it is 1 use the Static Salt below with PBKDF2 unless otherwise specified in the encryption block. This allows you to cache the expanded key for encryption / decryption purposes. This is only recommended if you are using SHA-256 or SHA-512 with a high number of iterations. If this is 0 we recommend setting the Static Salt to all null bytes.
Static Salt, 64 bytes	The rest of the extra header (64 bytes in v.2.0) is the Static Salt mentioned above.

Entity Block

An Entity Block is merely the aggregation of exactly one Entity Description Block, followed by the encrypted contents of exactly one Entity Description Block Data and zero or one instances of a File Data Block. An Entity can be at present a File, Symbolic Link or Directory. If the entity is a File of zero length or if it is a Directory the File Data Block is omitted. In any other case, the File Data Block must exist.

Entity Description Block Header

The function of the Entity Description Block Header is to allow a client to read the encrypted Entity Description Block Data. It is a binary block consisting of the following data (in order of appearance):

Signature, 3 bytes	The bytes 0x4A, 0x50, 0x46 (uppercase ASCII string “JPF”) used for identification purposes.

Encrypted size, 2 bytes	The encrypted size of the following Entity Description Block Data
Decrypted size, 2 bytes	The decrypted size of the following Entity Description Block Data

Entity Description Block Data

it purpose is to provide the client information about an Entity included in the archive. The client can then use this information in order to reconstruct a copy of the Entity on the client's file system. The data is written to the archive encrypted with Rijndael-128 in CBC mode. The Entity Description Block Data consists of the following information before it is encrypted:

Length of entity path, 2 bytes.	Unsigned short integer, represented as 2 bytes, holding the size of the entity path data below.
Entity path data, variable length.	Holds the complete (relative) path of the Entity as a UTF16 encoded string, without trailing null. The path separator must be a forward slash (“/”), even on systems which use a different path separator, e.g. Windows.
Entity type, 1 byte.	• 0x00 for directories (instructs the client to recursively create the directory specified in Entity path data). When the entity type is 0x00 the Compression Type MUST be 0x00 as well. • 0x01 for files (instructs the client to reconstruct the file specified in Entity path data) • 0x02 for symbolic links (instructs the client to create a symbolic link whose target is stored, uncompressed, as the entity's File Data Block). When the type is 0x00 the Compression Type MUST be 0x00 as well.
Compression type, 1 byte.	• 0x00 for no compression; the data contained in File Data Block should be written as-is to the file. Also used for directories, symbolic links and zero-sized files. • 0x01 for deflate (Gzip) compression; the data contained in File Data Block must be deflated using Gzip before written to the file. • 0x02 for Bzip2 compression; the data contained in File Data Block must be uncompressed using BZip2 before written to the file. This is generally discouraged, as both the archiving and unarchiving scripts must be ran in a PHP environment which supports the bzip2 library.
Uncompressed size, 4 bytes	An unsigned long integer representing the size of the resulting file in bytes. For directories, symlinks and zero-sized files it is zero (0x00000000).
Entity permissions, 4 bytes	UNIX-style permissions of the stored entity.
File Modification Time, 4 bytes	The UNIX timestamp of the file's last modification time. For directories and symlinks it must be ignored and set to 0x00000000.

File Data Block

The File Data Block is only present if the Entity is a file with a non-zero file size. It consists of one or more Data Chunk Blocks. Do note that the File Data Block has no header. The collection of one or several Data Chunk Blocks is called the "File Data Block".

Data Chunk Block

Each Data Chunk Block consists of the following information:

Encrypted size, 4 bytes	Unsigned long containing the size, in bytes, of the encrypted data.
Decrypted size, 4 bytes	Unsigned long containing the size, in bytes, of the decrypted data. If the decryption yields more bytes, the extraneous bytes must be trimmed off.
Encrypted data, variable length	The decrypted data is compressed, depending on the Compression Type, and then encrypted using AES-128 in CBC mode. The compression format used may be: • Binary dump of file contents or textual representation of the symlink's target, for CT=0x00 • Gzip compression output, without a trailing Adler32 checksum, for CT=0x01 • Bzip2 compression output, for CT=0x02

In split archives, the first 8 bytes must appear within the same part. They may or may not be in the same part as the Entity Description Block Data. The Encrypted Data can span multiple parts. Since the minimum part size is 64Kb and the maximum Decrypted Size can't be over 64Kb, the Encrypted Data will either be in the same part in its entirety, or span exactly two parts.

Encrypted data block format

The encrypted blocks have one of the following possible formats. You can detect the data format in two ways.

First, the legacy format is only used with JPS version 1.9 and below. If the file header claims that the archive is JPS 1.10 then the current format MUST be used.

If you do not or cannot trust the file header you can do a simple heuristics. Read the last 24 bytes of the encrypted block. If the first four bytes are JPIV you definitely have a current format block. Otherwise you most likely have a legacy format block (there's 1 in 4,228,250,625 chance of false detection).

JPS 2.0 (Current)

In this format the IV for each encryption block is always different, produced by a crypto safe PRNG (either OpenSSL or mcrypt). Therefore the encryption *is* safe against cryptanalysis (as far as an attack against Rijndael-128 itself is not discovered). Moreover, it allows for the inclusion of a per-block salt for PBKDF2 key expansion.

Encrypted data, variable length	This data is encrypted with Rijndael-128 using the IV described below.
Per-Block Salt, 68 bytes (OPTIONAL)	The literal string JPST followed by the 64 bytes of the per-block salt. Discard the JPST marker and use the rest as the salt for the PBKDF2 algorithm. This section MUST be present when the Use Static Salt flag in the archive header is 0. This section MAY be present when the Use Static Salt flag in the archive header is 1. This means that you shouldn't simply skip checking the existence of this section just because Use Static Salt is 1. If it's present, use it and derive a new, per-block encryption key.
Initialization Vector (IV) data block, 20 bytes	The literal string JPIV followed by the 16 bytes (128-bit) Initialization Vector data. Discard the JPIV marker and use the rest of the block as the IV for your Rijndael-128 decryption engine.
Decrypted data length, 4 bytes	The size of the decrypted data in bytes. Since Rijndael-128 in CBC mode encrypts data in 16-byte (128-bit) blocks it needs to pad data with length not exactly divisible by 16 with zero bytes. These zero bytes are not part of the input data and need to be discarded. For example, if your decrypted data length is 24 and the Rijndael-128 decryption result is 32 bytes you need to throw away the last 8 bytes which are just the zero (null) padding bytes.

JPS 1.10 (Previous)

Note

Only compatible with JPS 1.10 archive files. Not compatible with JPS 1.9 archive files. Obsolete since JPS 2.0.

In this format the IV for each encryption block is always different, produced by a crypto safe PRNG (either OpenSSL or mcrypt). Therefore the encryption *is* safe against cryptanalysis (as far as an attack against Rijndael-128 itself is not discovered).

Encrypted data, variable length	This data is encrypted with Rijndael-128 using the IV described below.
Initialization Vector (IV) data block, 20 bytes	The literal string JPIV followed by the 16 bytes (128-bit) Initialization Vector data. Discard the JPIV marker and use the rest of the block as the IV for your Rijndael-128 decryption engine.
Decrypted data length, 4 bytes	The size of the decrypted data in bytes. Since Rijndael-128 in CBC mode encrypts data in 16-byte (128-bit) blocks it needs to pad data with length not exactly divisible by 16 with zero bytes. These zero bytes are not part of the input data and need to be discarded. For example, if your decrypted data length is 24 and the Rijndael-128 decryption result is 32 bytes you need to throw away the last 8 bytes which are just the zero (null) padding bytes.

JPS 1.9 and below (Legacy)

Note

Only compatible with JPS 1.9 and 1.10 archive files. Obsolete since JPS 2.0.

In this format the IV is always the same and derived from the encryption key. For this reason the encryption is NOT safe against some methods of cryptanalysis which could compromise the encryption key.

Encrypted data, variable length	This data is encrypted with Rijndael-128.
Decrypted data length, 4 bytes	The size of the decrypted data in bytes. Since Rijndael-128 in CBC mode encrypts data in 16-byte (128-bit) blocks it needs to pad data with length not exactly divisible by 16 with zero bytes. These zero bytes are not part of the input data and need to be discarded. For example, if your decrypted data length is 24 and the Rijndael-128 decryption result is 32 bytes you need to throw away the last 8 bytes which are just the zero (null) padding bytes.

End-of-archive header

This header is written after the end of the archive data, at the end of the last part of the archive.

When creating spanned archives, the first file (part) of the archive set has an extension of .j01, the next part has an extension of .j02 and so on. The last file of the archive set has the extension .jps. You must also ensure that the Entity Description Block is within the limits of a single part, i.e. the contents of the Entity Description Block must not cross part boundaries. The File Data Block data can cross one or multiple part blocks, but the header of each Data Chunk Block must both be inside the same part.

This header is written after the end of the archive data, at the end of the last part of the archive. Its structure is:

Signature, 3 bytes	The bytes 0x4A, 0x50, 0x45 ("JPE")
Number of parts, 2 bytes	The total number of parts this archive consists of. Non-spanned archives should set this to 1.

File count, 4 bytes	Unsigned long integer represented as four bytes, holding the number of files present in the archive.
Uncompressed size, 4 bytes	Unsigned long integer represented as four bytes, holding the total size of the archive's files when uncompressed.
Compressed size, 4 bytes	Unsigned long integer represented as four bytes, holding the total size of the archive's files in their stored (compressed) form

The size of the EOA header is 17 bytes for version 1.9 of the format.

Change Log

Revision History		
	July 2010	NKD, Akeeba Ltdhttp://www.akeebabackup.com
Described version 1.9		
Revision History		
	January 2017	NKD, Akeeba Ltdhttps://www.akeebabackup.com
Described version 2.0		

Appendix C. Things which will (most likely) not be implemented

The following often requested features are likely to never be implemented due to several technical limitations. In order to save you and us some time, we are documenting what will not get implemented and why.

Thankfully, sometimes we're wrong

Below I'll be maintaining a list of things we told you were not possible, ever... only to eventually implement them when the conditions outside our control that prevented us from implementing them ceased to exist.

- **SkyDrive (OneDrive) support, since Akeeba Backup 4.2.0**. On February 25th, 2015 Microsoft launched a new RESTful API for OneDrive which made it possible to upload very large files of any type to their service. It took a bit less than 48 hours between the announcement of the new API and the implementation and testing of native OneDrive support for Akeeba Backup and Akeeba Solo.
- **Google Drive**. Google opened their API for generic data storage.

1. Automatic sync between sites

Site synchronisation is a completely different concept to backups. In fact, it is impossible to have reliable site sync in Joomla! (and all major PHP CMS, actually). For more information read my blog post on the subject [http://www.dionysopoulos.me/blog/myths-and-facts-on-site-sync]. The closest you can get to that concept is performing the procedure underlined in Advanced Site Transfers [https://www.akeebabackup.com/documentation/walkthroughs/advanced-site-transfers.html].

While I am at it, since Joomla! 1.6 it has become even more difficult to perform site sync or even an advanced site transfer. Extensions, categories and articles all come with ACL records in the #__assets table. This means that if you only want to copy articles between sites with, say, even one different extension, you can't. If you don't have the exact same user groups and view levels (down to the numeric ID level) in both sites, you can't transfer anything between them. In short, unless the sites are identical except the new articles, you can't merge them. Keep that in mind and be ready for a lot of hair pulling if you decide to go through the dev-staging-live site work flow.

2. Automatic backups without CRON

A full site backup is something which takes a heck of a lot of time, CPU and RAM to perform. It can't happen in a single page load. That's why you need Akeeba Backup, because it splits the process into smaller steps suitable for execution on the server. Asking for an automatic backup without CRON means that the backup would have to be taken as a visitor tries to load a page on your site. That would lead to a blank page, or -in the best, unrealistic, case scenario- your visitor would have to wait for several minutes or hours before the page loads. Come on, get real. There is a reason why there are CRON jobs. Long running operations, like backups, are one of them.

Before you argue that it is possible, please read our documentation regarding the known issues of the Lazy Scheduling plugin. That plugin was designed to solve that problem. Guess what? It turns out that the limitations on the servers where you can't use CRON are so big that even this solution can't work reliably. If you can't get a CRON job working on your server, try webcron.org. It rocks.

3. Automatic backups after saving/creating/whatever an article

It's the same thing as automatic backups without CRON. It's exrtemely easy for me to create a plugin which waits for an article to be saved, then take you to the backup page. But, honestly, would you want a full site backup

every time you hit Apply on the article editor? Hardly. What you really want is make a bunch of changes to a few articles, then take a backup. Well, that's possible ever since JoomlaPack 1.0, released back in 2006 :)

4. Put Akeeba Backup in Joomla!

First, it's the licensing issue. Akeeba Backup is licensed under GPL v3 or later, whereas Joomla! is licensed under GPL v2 or later. Including Akeeba Backup inside Joomla! would make Joomla! GPL v3 or later. This would make hundreds of extensions licensed as GPL v2 (without the "or later" part) to become illegal. I believe that's the primary reason why Joomla! has not adopted GPL v3 yet.

Anyway, you really don't want that to happen. The reason that Akeeba Backup is successful is because of its support, rapid releases and feedback from its users. If it were included in Joomla! itself:

- Support would suffer. Obviously, I couldn't offer support for free because it doesn't scale (that's why support became for-a-fee). I simply don't make enough money to hire an army of support personnel to do free support on the Joomla! forum. Due to the nature of backups and the (mostly host-dependent) issues you might encounter generic support on the Joomla! forum by volunteers doesn't really work. I mean, you can already witness that if you try to ask an Akeeba Backup question on the Joomla! forum today.

- Fixes would take too long to get published. Even if a major issue was discovered, we couldn't just fix it and release a new Joomla! version. Joomla! is a huge cruiseship. It takes a lot of effort to get new releases packaged, tested and released. This would degrade the quality of your backup experience. Or it would lead us to not offer any new features, ever, in fear of breaking what works. This is what happens with other core components.

- New features would suffer. Actually, no new features would be added mainly due to a. lack of feedback (by isolating the developer from the users there is no feedback, ergo no new ideas) and b. because no developer would assume the responsibility of introducing something potentially buggy in a new Joomla! release. At best, new features would be introduced once every 18 months. Put that in contrast with the current 3 month release cycle of Akeeba Backup and you'll understand how bad that would suck.

- I might have to abandon it altogether. Unfortunately, I need money for me and my family. Maintaining Akeeba Backup takes up the better part of my days, nights, weekends and vacation time. Having subscriptions is the only way for me to make enough money to justify living, breathing and not sleeping for or because of this software. If you take away the money from me, I would have to be a slave. Nope. I would just abandon the software and get another job to pay my bills. And, since you might wonder, no, I couldn't have a paid version even if just Akeeba Backup Core was included in Joomla!. The first reason would be API compatibility. I have a successful for-a-fee version because I am free to change the backup engine's API very often. Adding that to Joomla!, my hands would be tied. Moreover, the free version acts as an advertisement for the for-a-fee version. If the free version is in Joomla! itself and no longer called Akeeba Backup I've lost my major source of new customers. No new customers = no income = abandoned development.

So just watch what you wish for. Adding something to the core is not a positive thing. It is usually the end of the life of a component. In fact, I'd argue that the Joomla! core should be as lightweight as possible and now it's anything but. Having people outside of the Joomla! project maintain each extension is not a drawback, it's an asset. Organisations like the Joomla! project are by definition slow-moving. Small teams of 1-3 people per component are infinitely more flexible and can produce much better results in the long run.

Note

If the above sounds a bit far-fetched, it's what happened to FOF, the Rapid Application Development framework I created and which got included in Joomla!. I could no longer push bug fixes in a timely manner so I had to fork my own code. This created a bunch of other problems, e.g. confusion among the users who saw two FOF library installations on their site and tried to remove one, with catastrophic results. Just trust me on this: the best way to KILL innovation and development for a piece of software is including it in Joomla! itself.

This is not a problem unique to Joomla!. It happens wherever you have large collections of packaged software, including all Operating Systems for computers and mobile devices alike. If you release too often and break too much people hate you (because you broke their environment or forced them to lag

behind – see Android). If you release too infrequently and never break anything you stiffle innovation and become irrelevant (Windows Mobile beat iOS to market by FIVE years and now it doesn't even exist – Windows Phone is a completely different product). On the other hand if you have one specific piece of end user software you can release very often, rewrite large parts of it to make it better and people love you... as any of the million of iOS and Android app developers out there can tell you. So, yeah, watch what you're wishing for.

Appendix D. GNU Free Documentation License

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety

of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to

ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections

as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket

the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/ [http://www.gnu.org/copyleft/] .

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright (C) YEAR YOUR NAME.
>
> Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

> with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.