Akeeba Backup for Joomla

Nicholas K. Dionysopoulos

Akeeba Backup for Joomla

by Nicholas K. Dionysopoulos

Abstract

This is the user manual to Akeeba Backup for Joomla!™ version 9 or later, available on Joomla 4.0 and later.

If you are looking for a quick start to using the component please watch our video tutorials [https://www.akeeba.com/videos].

Table of Contents

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

Table of Contents

Chapter 1. Introduction

1. What is Akeeba Backup?

Akeeba Backup is a complete site backup solution for your Joomla!™ powered website. It will take a copy of your entire site – files and database data – and put it in a backup archive file. You can restore the backup archive file on the same or a different site, even a completely different server. The restoration uses a web installer script. The installer script is included in the backup archive itself.

You do not need separate files and database backups. You do not need third party tools to restore your backups. You do not need to edit Joomla configuration files or database data manually. This is all handled automatically for you.

More than that, Akeeba Backup is putting you in control of your data. If you so with you can fine-tuning your backup choosing which directories, files or database tables to exclude.

Tip

If you are looking for a quick start to using the component please watch our video tutorials [https://www.akeeba.com/videos].

2. What can I use Akeeba Backup for?

Akeeba Backup can be used for much more than just backing up your site. 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 (or on a development server), upload the finished site easily**. Web professionals can build a complete site off-line on a local server, or on-line on a development server. When done they can take a backup with Akeeba Backup and restore it on the production site. This minimises the downtime of the production site.
    - **Testing upgrades locally, without risking breaking the on-line site**. Joomla!™ updates and updates to its extensions have the potential of breaking things, especially the more and more complex extensions you are using. Site owners 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.
- **Troubleshooting locally**. Web professionals can use Akeeba Backup to take a backup of a client's Joomla!™ site and restore it locally to reproduce, identify and resolve an issue with the site. If resolving the issue required a more substantial change than changing a file or two they can use Akeeba Backup once again to transfer the fixed site back to the production server.
- **Moving a site to a new host**. Site owners who want to take their site to a new host have found Akeeba Backup to be a major time saver. Just backup the original site and restore on the new host; your site is relocated with virtually no effort at all.

Our users tell us that Akeeba Backup has saved them hours of frustrating work, according to our users. This is what makes it unique. It doesn't make site backups and site transfers possible, it makes them *so much easier*.

Akeeba Backup comes in two editions, Core and Professional. Akeeba Backup Core is free of charge and contains all the features a typical hobby or semi-professional single site owner would like to have in order to easily complete backup and restoration jobs. Even web professionals find it very useful to transfer sites between servers. On top of that, the video tutorials and the full documentation are free of charge as well.

Akeeba Backup Professional addresses the more particular needs of web professionals and web agencies, handling numerous and / or high-value sites. It has features such as inclusion of external directories and databases, powerful filters based on regular expressions, and support for sending your backups on cloud storage services (such as Amazon S3, Dropbox or any other of the 40+ supported providers).

Both versions are released under the same license GNU General Public Licence version 3 (GPLv3) or, at your option, any later version of the licence published by the Free Software Foundation. This means that you can use it on as many sites as you want, yours or your clients', without paying additional fees or expensive "developer licenses". All we ask of you is that you make it clear to your clients that they are not our clients, therefore you are responsible for providing updates and support for Akeeba Backup to them.

Disclaimer: Whether Akeeba Backup and its restoration works for you is highly dependent on conditions outside our control such as your server setup, your server configuration and your PHP configuration as well as the way you configure and use the software. We do provide free of charge versions of our software which let you evaluate it before purchasing a subscription and we do offer support which can potentially work around some server configuration issues. We consider it self-evident that despite our best efforts if you are unwilling to let us help you or if the problem is ultimately something that only your host can address we bear no responsibility whatsoever and accept absolutely no liability per the license our software is distributed under and our Terms of Service.

3. A typical backup/restoration work flow

Tip

If you are looking for a quick start to using the component please watch our video tutorials [https://www.akeeba.com/videos].

Akeeba Backup is designed to make backing up and restoring a site easier and more user-friendly. 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 exactly the same. In short, Akeeba Backup doesn't care if you are restoring, copying, cloning or migrating your site. The process is always the same; you only have to learn it once. The learning curve is quite smooth, too!

Please note that only MySQL-compatible database servers are supported (e.g. MySQL, MariaDB, Percona). You cannot take a backup of Joomla running on PostgreSQL or restore a backup of a site on a server that only has a PostgreSQL database. Also note that some MySQL-compatible servers may have slight incompatibilities between them. When transferring a backup across database servers we attempt to account for these incompatibilities. If something doesn't work, please let us knwo.

The typical work flow involves using two utilities from the Akeeba Backup suite: the Akeeba Backup component itself, and Akeeba Kickstart (our archive extraction helper). 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 most suitable 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 page. The only files in there are `kickstart.php` and the translation INI file. 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 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 restoration script called ANGIE. This was placed in your backup archive when you took the backup. It's used to restore the database and do any site reconfiguration actions necessary at the end of the restoration. Do not close the Kickstart window yet!

7. Follow the prompts on your screen, filling in the details of the new server. If you are restoring to a different URL or server than where the original site was backed up from you will need the database connection information. If unsure about what this information is please ask your host.

8. When ANGIE is done restoring your site it prompts you to close the window / tab it was running in. Go ahead and do it.

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. Click on the View the front-end button to visit your new site. You're done.

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 rename or edit your .htaccess file for your site to work properly. ANGIE does give you an option to do either of these actions for you, in the Site Setup page.

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

Akeeba Backup, like all web applications, needs a suitable web hosting environment and web browser to run. In a nutshell you need this to run Akeeba Backup:

- Joomla!™ and PHP version compatibilities are detailed in our Compatibility page [https://www.akeeba.com/compatibility.html]. Your server must have the PHP extensions Joomla recommends installed and enabled.
- MySQL 5.1 or later, or a compatible server e.g. MariaDB. Akeeba Backup won't work with PostgreSQL or any other database technology.
- 32MB of PHP memory at the very least. 64 to 128MB recommended. More is better.
- Your server must allow PHP to list directory contents and read files and dump the contents of your site's database.
- You need a folder writable by PHP to store your backups in and enough disk space to do so.
- If you are using any feature that transfers data from or into your site you need the PHP cURL extension or the PHP fopen() wrappers installed and enabled for the respective protocol (HTTP, HTTPS, FTP, FTPS or SFTP) and your server(s) must not block the connection. That's the same requirement as Joomla itself e.g. for its own updates.

- A fairly modern version of one of the major web browsers.
- Your computer and browser must not block the connection to your site and JavaScript must be allowed to run.
- If you want to automate the backup you need a server that can do real (CLI) CRON jobs or use a third party service compatible with Akeeba Backup. You'll have to set up scheduled backups yourself.
- Some common sense and a basic understanding of how sites work.

In much more detail, Akeeba Backup requires the following server software environment at the bare minimum:

- Joomla!™ and PHP version compatibilities are detailed in our Compatibility page [https://www.akeeba.com/compatibility.html]. Your server needs to be *additionally* fully compatible with the published minimum requirements of the version of Joomla you are using per the information published by the Joomla project.
- MySQL 5.1 or later. MySQL 5.7 or later strongly recommended. Versions of MariaDB compatible with the MySQL–flavored SQL language for MySQL 5.1 to 8.0 are also supported but only for the features supported by MySQL itself. We do not have enough information about Percona compatibility.

Important

AKEEBA BACKUP DOES **NOT** SUPPORT POSTGRESQL A.K.A. POSTGRES.

- The following PHP extensions must be installed and enabled: mysqli or pdo and pdo-mysql to connect to the database (Joomla requirement); json, mbstring and simplexml (Joomla requirement); gzip for handling compression and decompression of files.
- Your database server needs to have no limits on operations over a period of time (e.g. queries per minute) or, alternatively, a realistic and high enough limit to run enough SQL queries to dump the entirety of your site's database. The same applies for restoration. While Akeeba Backup can work around some of the more restrictive limits, your backup and/or restoration may be too slow to be practical.

The following common sense requirements are necessary to do anything useful with Akeeba Backup and should be understood as minimum server requirements:

- As a rule of thumb, we recommend a PHP memory_limit setting of 64MB to 128MB for most sites.

  More precisely you need a minimum of 16MB of free PHP memory. If you are uploading files or backup archives to remote storage / remote servers increase this limit by 2.5 times the maximum size of your backup part size (or upload chunk, for post-processing engines that have chunked upload enabled). If you are using the DirectFTP or DirectSFTP archiver engines increase that by 1.5 times the size of the largest file on your site.
- Enough available free space or quota limit to store your backup archives and any temporary files.

  If you are uploading backup archives to remote storage you need at least enough space on your server to store the first and last backup archive part on your server (assuming you are using the Upload Files Immediately option in the post-processing engine). In this latter case, that's typically 1.5 to 2 times the part size for archive splitting configured in Akeeba Backup.
- PHP must be able to list the contents (files and folders) of your site's root, read all files under your site regardless of how many folders deep and list the contents of folders under your site regardless of how many folders deep. This is typically controlled by file/folder ownership and permissions on most Operating Systems and Permissions on Windows. Please note that on some servers this may be additionally controlled via open_basedir restrictions.
- If the default backup directory cannot be written to by PHP it's your responsibility to provide a writable directory to be used as the backup output directory, either under your site's root or above it. In case of a folder above your site's

root it must additionally not be limited by open_basedir restrictions, chroot environments or similar server–level technologies which restrict access to directories above your site's web root.

- Your PHP timeout limit must be at least 2 seconds to make backups practical (however, transferring backups to remote servers / remote storage is unlikely to work at all). We recommend a timeout limit of 30 to 60 seconds for most efficient operation with a reduced possibility of a timeout.

Server requirements for optional features:

- The PHP FTP extension must be installed and enabled to use any feature which transfers files to remote servers via FTP or FTPS.
- The PHP cURL extension must be enabled and compiled against a libcurl library which supports FTP to use any feature which transfers files to remote servers via FTP or FTPS and is additionally marked as “over cURL”.
- The PHP SSH2 extension must be installed and enabled to use any feature which transfers files to remote servers via SSH.
- The PHP cURL extension must be enabled and compiled against a libcurl library which supports SFTP to use any feature which transfers files to remote servers via SFTP and is additionally marked as “over cURL”.
- The PHP cURL extension must be enabled and compiled against a libcurl library which supports HTTP and HTTPS; or the fopen() URL wrappers must be enabled on your site (but not allow_url_includes, we don't make use of that dangerous PHP feature). This is required for using any feature which sends data to or retrieves information from any remote server including but not limited to most post-processing engines and the Site Transfer Wizard. *This is the same requirement your server must meet for Joomla to find, download and install updates for itself and third party extensions*.
- Your web server **and** PHP must be configured with a timeout limit adequate for sending or retrieving files to/from remote storage providers. This depends on the size of the files and your server–to–remote–storage network throughput. We recommend a timeout limit of at least 10 seconds, ideally 60 seconds to cover most practical use cases.
- Your web server must not block outgoing connections to use any feature which sends information to remote servers including but not limited to most post-processing engines and the Site Transfer Wizard.
- Your new server must be on–line and accessible over the web with a domain name or subdomain name (not just an IP address) if you are using the Site Transfer Wizard.
- Your web server must not block inbound connections and your site must be on–line to use any frontend features including but not limited to the Akeeba Backup Remote JSON API and the Legacy Frontend Backup URL. This is necessary for some backup scheduling options.
- Your server must provide real (CLI) CRON job support to use the command line commands such as but not limited to the command for taking a backup. The Operating System user limits must allow the CRON job to run for at least as long as it is necessary for the backup to run to completion. This is necessary for one of the backup scheduling options.

As far as the browser is concerned, you can use any modern version (i.e. published within the last year) of Microsoft Edge, Safari, Opera, Firefox or Google Chrome. We no longer support Internet Explorer; our software will display incorrectly or not work at all on this old, buggy and obsolete browser.

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.

You are very strongly advised to disable Internet firewalls, antivirus applications and browser extensions which interfere with the site's loading such as script blockers (such as NoScript) and ad blockers (such as AdBlockPlus) *only for*

the domains you are backing up from and restoring to. Remember that these applications and browser extensions are designed to protect you against third party sites. As a result they are very aggressive and WILL break your own sites. We can't do anything about it: your computer and your browser are under your control alone.

If you are using a CDN, such as CloudFlare, please remember to set it up so that all /administrator URLs on your site are pass-through (not cached). Otherwise the Joomla backend, including Akeeba Backup, will not work properly.

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 [https://docs.joomla.org/Installing_an_extension]. Throughout this chapter we assume that you are familiar with these instructions and we will try not to duplicate them.

1.1. Installing or manually updating the extension

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.

You cannot and MUST NOT use the Discover method in Joomla to install any modern Joomla extension shipped as a package, consisting of many related extensions. This will make it impossible to find and install updates automatically and will also cause problems when updating Joomla itself.

Keep in mind that installing and updating Akeeba Backup (and almost all Joomla! extensions) is actually the same thing. If you want to update Akeeba Backup 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 code 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 port 443 (HTTPS) to `www.akeeba.com` and `cdn.akeeba.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.akeeba.com/downloads/akeeba-backup.html]. Make sure you are logged in; if not, please log in now. These instructions won't work if you are not logged in. Click on the All Files button of the version you want to install. 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 Direct Install Link link. Right click on it and select Copy link address or whatever your browser calls this.

Now go to your site's administrator page and click on System, Extensions, Manage. 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! can't download the package, please use one of the methods described in this section of the documentation.

1.1.2. Upload and install.

You can download the latest installation packages our site's download page for Akeeba Backup [https://www.akeeba.com/downloads/akeeba-backup.html]. If you have an older version of Joomla! or PHP please consult our Compatibility page [https://www.akeeba.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 logged 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. Please click on the Upload Package File tab. Drag and drop the installation ZIP file you had previously downloaded to start the upload and the installation. After a short while, Joomla!™ will tell you that the component has been installed.

Warning

Akeeba Backup is a big extension (about 2Mb for the Professional release). Some servers do not allow you to upload files that big. If this is the case you can follow our installation troubleshooting instructions [https://www.akeeba.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.akeeba.com/documentation/troubleshooter/abinstallation.html] or try the manual installation described below.

1.1.3. Manual installation

Joomla does not support manually installing modern extensions which are distributed as a “package” extension, consisting of multiple related extensions. DO NOT try to extract the installation ZIP file to install each extension manually. DO NOT try to copy files manually and use the Discover feature in Joomla.

If you ignore these instructions please note that Joomla will NOT know that all of the included extensions are related to each other. As a result it will not be able to find and install updates to Akeeba Backup. Moreover, every time you try to update Joomla it will complain that our extensions are not compatible with the new version even though this is nto actualyl the case; this is a consequence of Joomla not being able to find update information.

In short, these are technical limitations in Joomla itself. Installing disparate extensions and using the Discover method are good tools for extensions developers and advanced site integrators, however they are NOT meant to be used on real world, live sites.

1.1.4. Troubleshooting the installation

Please note that extensions installation is performed by Joomla itself, not code that we have written ourselves. If you have a problem installing a Joomla extension of ours the root cause is in Joomla! and the way some of its functions work. While we don't offer support for generic installation issues, this page is meant to serve as a collection of the troubleshooting steps we'd follow on any site when any extension doesn't install correctly.

You get an error about the package not being uploaded to the server

The installation packages of our extensions are rather big, approaching 2Mb for the Professional versions. Many server have a maximum file upload size or a maximum POST request size which is too small – typically around 2MB – for our software to install. The best solution is to ask your host to set the following in the server's php.ini:

```
upload_max_filesize = 10M
post_max_size = 10M
```

On most hosts you can place these lines in a file called `.user.ini` (not the leading dot) or `php.ini` in your site's `administrator` directory.

If this is not possible, there's a good chance that the following lines in your `.htaccess` file may work on most servers:

```
php_value upload_max_filesize 10M
php_value post_max_size 10M
```

There is also another alternative, but it won't work on all hosts: installation by URL. Please go to the Download section of our site and select the software and version you want to download. Click on View All files. Next to the "Download now" button for the installation package you will see a link called "DirectLink". Right click on it and select "Copy link address" (the exact phrase depends on the browser, but it should have to do about copying the link or its address / URL / location).

Now go to your site's back-end, Extensions, Extensions Manager and find the Install from URL tab. Replace the contents of the "Install URL" field with the contents of your clipboard (the DirectLink URL you copied above) and click on the "Install" button next to this textbox. As long as your server supports installing extensions from URL the installation should go through.

If this still doesn't help please read on for further troubleshooting tips

"Install path does not exist"

Joomla! requires the PHP Gzip and ZIP extensions to be installed. If either is not installed or if it's blocked then Joomla! will be unable to install extensions. Unfortunately, a cascade of unhandled errors inside Joomla! itself will cause it to come up with the unhelpful and disorienting "Install path does not exist" error message.

Solution: ask your host to enabled the GZip and ZIP extensions in PHP. Furthermore, ask them to make sure that they are not blocking the functionality of these extensions e.g. by using `disable_functions` or `disable_classes` in their `php.ini` file.

Please note that we routinely see hosts disabling functions `zip_open`, `gzuncompress`, `gzdeflate` and `gzdecode` for ostensible "security reasons". First of all Joomla! WILL NOT work properly when any of these functions is unavailable. Moreover and despite what your host tells you, disabling this functions does not increase your site's security in any conceivable way. If your host denies to unblock these functions please take your site to a different host that understands how server security really works.

"Unable to write entry" or "Unable to create destination" error

This error message comes from Joomla! and it means that there is a file or directory permissions issue. Unfortunately this message is very non-specific and provides no useful information for troubleshooting. This is something we reported to Joomla in September 2017 and was fixed for the most part but internal issues in the way the extensions installer work still prevent the correct path from being shown.

In the meantime, all you can do is ask your host to make sure that all folders and files on your site are writeable by the user under which your site runs. This is not something you or us can do. Please do ask your host.

If this doesn't help it might mean that you have reached the file system capacity of your server. Please note that your account on the server might have several limits:

- Maximum total size of files and database data. This is the most common limit, e.g. your host telling you that you can use 10G of space in total. Please remember that this includes your database data. Moreover, keep in mind that "unlimited" is a marketing term, not reality. Usually you get up to a certain size limit and you have to ask for more, explaining why.

- Maximum number of files. This is usually NOT advertised, documented or visible anywhere. Many hosts will only allow you up to a maximum number of files, e.g. 100,000. If you try to exceed that count the file is not created / replaced, as if the permissions were not adequate to write to it. Please note that most times the host engineers will call it "inode count" because that's technically what they are limiting on your hosting user account. Each inode is a filesystem index entry and each file and folder on your site consumes one inode (that's not very accurate but it's a well enough description to understand what an inode is).

- The physical disk size. All the aforementioned limits are great, but you cannot create files beyond the physical capacity of the disks on your server. Most modern hosts use virtualized, network attached storage to provide ever-expanding capacity on demand. However, some cheaper hosts and dedicated servers still have regular disks attached with finite storage limits.

- The block count. Please remember that computer permanent storage cannot be allocated with a single byte granularity. It is allocated in blocks, typically 4KB to 32KB each. Think of it as an apartment building where everyone gets the same size , four bedroom apartment regardless of whether they are single or a family of seven with a dog and two cats. This means that small files, whose size is nominally under one block, will occupy at least one block. Your host counts the disk space usage by the number of blocks you occupy but hosting control panels report the aggregated nominal sizes of the files. It is possible for the hosting control panel to report free space when you have reached the block limit of your hosting account.

- Also remember that your hosting control panel does not report the limit information in real time. You may have already exceeded your limits but your control panel not having been updated with this information.

If you are not sure about these limits please ask your host.

Upgrading from Core to Professional

In some cases we have seen that Joomla failed to copy all of the necessary files when upgrading from a Core to a Professional release or when installing a major update that spans major versions (e.g. 1.x to 2.y). If you believe this has happened to you please install our software twice in a row, without uninstalling it before or in between the subsequent installations.

Check your Joomla! and PHP version

We publish the compatibility of our software with Joomla! and PHP versions in the Compatibility page on our site. You can find a link on this page at the bottom of every page of our site.

Please remember that the PHP version your site is using may be different than the PHP version your host reports in their hosting control panel. If unsure, please refer to Joomla's System Information page. If you need to upgrade your PHP version please consult your host. The exact method to do that varies by host.

Checking your temporary directory

First, we will have to make sure that you are using a valid temporary directory. Many sites are configured to use the system-wide (/tmp) directory or an invalid directory, causing installation problems.

You can change your temporary directory from your site's Global Configuration page. You need to enter the full filesystem path to Joomla's `tmp` folder. This is typically something like `/home/mysite/public_html/tmp`. If unsure please ask your host. This information is not visible from within your site's administrator using any Joomla-provided feature and there is no way for us to know it.

File ownership (Previously: Enable FTP)

Some shared servers run PHP as an Apache module. As a result PHP runs under the same user as Apache which is a different user than the one your user account is owned and the one used when uploading files via FTP / SFTP. As a result you end up with mixed ownership of your files, making it often impossible to install or update extensions.

Old Joomla versions (1.x, 2.x and 3.x) included a Global Configuration feature called “Enable FTP”. This feature would make Joomla go through FTP to write to any file, therefore all your files would be owned by the same user, your hosting account's user. However, this was a security issue since Joomla needed to store your FTP password in the clear. In most cases that was also the hosting control panel password. In other words, any security mishap which would divulge Joomla's configuration to an unauthorised user — even a simple misconfiguration of your site's permissions — could cause major problems to your site. This feature has, therefore, been removed from Joomla 4.0 onwards.

If you find yourself on such a host it's a good time to move to a different, better set up host.

To give you a better idea, the PHP developers themselves have said that the PHP Apache module should not be used for shared hosting scenarios. It's something only really meant to be used for single site servers with the assumption that Apache is correctly configured for this use case. Solutions for PHP running as a different user for each site of a shared hosting server have existed since the early 00's. If your host has still not gotten wind of these problems and their respective solutions over a course of at least two decades they are probably unqualified to host any site, hence the recommendation to move to a different host.

Still problems?

If you still can't install our software and you are receiving messages regarding unwritable directories, inability to move files or other similar file system related error messages you can ask for our support but you already read what we're

going to try. At best you can expect us to find out the most likely root cause and tell you what you have to ask your host to do to fix it.

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 sometimes 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

Akeeba Backup can be updated just like any other Joomla! extension, using the Joomla! extensions update feature. Please note that Joomla! is fully responsible for discovering available updates and installing them on your site. Akeeba Ltd does not have any control of the update process.

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 port 443 (HTTPS) to `www.akeeba.com` and `cdn.akeeba.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.

Warning

Akeeba Backup Professional needs you to set up the Download ID before you can install the updates. Please consult the Entering your Download ID documentation section for more information.

You can access the extensions update feature in different ways:

- From the icon your Joomla! administrator control panel page. By default you will find the icon in the right-hand modules area, under the Update Checks header. 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 sidebar of your Joomla! Administrator click on System. On the new page find the Update area towards the bottom of the middle column and click the Extensions link. This takes 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 of the Joomla! Extensions: Update page. 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.

If Joomla can't download the package, please use one of the manual update methods described below.

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 do 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.

If the error occurs again after a while, without you updating our software, please contact your host. Some hosts will delete or rename files automatically and without any confirmation as part of a (broken and unfit for purpose) "malware scanner / antivirus". Unfortunately, these scanners return a lot of false positives -innocent files mistakenly marked as malicious- but rename / delete them nonetheless, breaking software installed on the server. If you are on such a host we very strongly recommend that you move to a decent host, run by people who actually know what they are doing. It will be far less headache for you and would actually improve your site's security.

3.1. Troubleshooting the update

Like most Joomla extensions, our software relies on Joomla's built-in extensions updater. In simple terms, code written by the Joomla project, shipped with Joomla itself and running on your site is responsible for retrieving information about the latest available versions, determining whether an update is available, downloading the update package and installing it on your site. Akeeba Ltd has no control over that code.

Despite this not being our code, we do understand that our clients do come across problems with updates and need our help. The way the Joomla built-in extensions updater is written makes it prone to some easily preventable, common errors. Its error reporting ranges from unhelpful to non-existent. In an effort to help you, we've compiled and condensed all the troubleshooting we've done for years on our sites and our clients' sites.

Please note that we do not have a choice on whether to use Joomla's built-in extensions updater or our own code. The Joomla Extensions Direction requires us to use Joomla's own extensions updater as a requirement for our software being listed there.

3.1.1. Addressing server issues

In some cases you will see that Joomla cannot retrieve the latest version information or update package for our software, reporting it cannot connect to cdn.akeeba.com. Related to that, Joomla may report that it's unable to download the Professional edition's update package, saying it's unable to connect to our site www.akeeba.com. This can mean a few different things which all have to do with how your host is set up.

Our CDN and our site are accessible over HTTPS and use a valid, signed TLS certificate. At the time of this writing the TLS certificates are issued by Let's Encrypt and Amazon Web Services. The TLS certificates used for HTTPS on our CDN and site use the recommended SHA-256 hashing algorithm and the servers only support modern versions

of the TLS protocol (at the time of this writing it's TLS 1.2 and later). If your host has an out of date Certification Authority cache or compiled PHP against an old TLS library which does not support modern versions of TLS your site will be unable to connect to our servers.

If this is not the case, please be aware that some hosts run a proxy server or a firewall which can either prevent or cache *outgoing* connections in front of their servers. Depending on how this is implemented it can cause two distinct types of problems.

The first problem is that your site might be unable to connect to our CDN and our server to retrieve the latest version information and the update package itself respectively. If this happens you need to ask your host to allow connections to TCP/IP port 443 (HTTPS) for www.akeeba.com and cdn.akeeba.com. If they ask you for an IP address please ask them to resolve these domain names from their server. The latter is a Content Delivery network (CDN) with hundreds of servers, powered by Amazon CloudFront, meaning that its IP address depends on where you are accessing it from.

The second problem is that when Joomla tries to retrieve the latest version information or an update file from our servers your host's proxy gets in the way and returns information it has cached. We explicitly ask for that information not to be cached, using standard HTTP headers, but some hosts choose to ignore web standards and do their own thing. Also worth noting is that your host should not interfering with HTTPS (encrypted) traffic, so all the more reason to be worried about their implementation in this case. Unfortunately, we have caught a few hosts doing that over the years.

None of these issues can be addressed by you or us. You will need to contact your host about them. Before you assume any of these issues are in play and if you are using the Professional edition of our software please do check that your Download ID is valid first.

3.1.2. Check the validity of your Download ID

Note

The information in this section only applies to the Professional edition. If you are using the Core edition you can skip over it.

If you are using the Professional version of our software we need to verify that you have an active subscription that gives you access to downloads of the software you are trying to update. We do that by means of a Download ID which has the format 0123456789abcdef0123456789abcdef (Main Download ID) or 12345:0123456789abcdef0123456789abcdef (Add-on Download ID). In and by itself the Download ID does not carry any information about your subscription status. It is an identifier linked to your account on our site.

First, you need to check that you are using a valid Download ID. **Do not assume that your Download ID** is entered at all, or that it is valid. This kind of false assumption accounts for half of the update issues we are asked to help our clients with. Always check on our site. Log into our site and go to Add-on Download IDs from the top menu. Copy the Download ID and paste it to our extension's Options page in the Download ID box, under the Update tab.

If you had to enter or change the Download ID but Joomla was already reporting an updated version you will need to wait for 1-2 days OR install the update manually. This has to do with how Joomla update information caching works — even if you tell it to clear the update cache.

3.1.2.1. Check your subscription status

Note

The information in this section only applies to the Professional edition. If you are using the Core edition you can skip over it.

As noted above, the Download ID itself does not carry any information about whether you are allowed to download an update. This check is done on our server when it receives the Download ID along with Joomla's request to download

an update. The check performed is simple: do you have an *active* subscription which gives you access to the software you are trying to download?

Do not assume that your subscription is active. It is possible that you missed an email warning you about the subscription expiring and a manual action to renew it being required on your part. Do note that we send two emails before your subscription reaches it validity limit and you lose access to downloads, 30 and 15 days in advance.

Always log into our site and go to the My Subscriptions page to check your subscription status. If your subscription has expired you can renew it. Once the payment is complete and accepted by our reseller you will be able to download the updates within the next 20' or less (typically: within seconds).

3.1.2.2. Multiple Professional edition Akeeba extensions with different Download IDs

It's possible that you have more than one of our Professional extensions but you want to use different Download IDs for each one of them. For example, if each extension was bought by a different company working on your site or if you are trying to migrate to a new user account on our site.

That is to say, one extension might use a Download ID that is valid and corresponds to an active subscription for that software, another extension might have an invalid Download ID or use a Download ID which refers to a user account on our site which does NOT have a valid subscription for that software.

Do remember to check the Download IDs for each and every of our extensions. Do remember to check that you have active subscriptions for all the products you are trying to update.

3.1.2.3. Entering or changing your Download ID after an update is available

On March 9th, 2021 we publicly and officially notified Joomla about known issues with their extensions updater when the Download Key is changed after an updated version has been detected. We had previously notified Joomla of these issues repeatedly, but privately, over the course of eight years. These issues are still not addressed.

If you enter or change your Download ID after Joomla has already determined an update is available the new Download ID will NOT take effect on many sites. This has to do with the way Joomla is caching not only the update availability information but also the raw information about Update Sites and Download Keys it uses to determine if updates are available and how to download them.

If you find yourself in this case we strongly recommend installing the update manually (download the new version and install it using Upload and Install over the old one) OR wait 1-2 days before retrying. Kindly note that retrying after 1-2 days MAY STILL NOT WORK.

We understand that this is especially disruptive if you are managing dozens or hundreds of sites. This is why we have been trying to convince Joomla to fix these issues since 2014. We have even gone as far as to point out exactly why these issues occur and how to fix them. That's the full extent of what we can do. It is ultimately Joomla's responsibility to take your needs seriously enough and include the *three lines of code to fix these issues*.

3.1.3. Updates are showing after installing the latest version

Sometimes you might see that Joomla reports that the version you have installed or even a previous version is available as an update. This can mean three things:

- Joomla's update cache is stuck. Please file a bug report to Joomla. We have already been reporting this privately since 2014 and made an official, public issue report on March 9th, 2021.
- You have a server issue connecting to our CDN. See the information on addressing server issues.

- You have found a bug in Joomla's built-in extensions updater. You need to contact the Joomla! forum [https://forum.joomla.org]. Unfortunately there is nothing we can do about Joomla core bugs.

3.1.4. Updates not showing despite having an older version

Sometimes you may see that Joomla refuses to report the availability of a new version of our software. This can mean four things:

- The update site for our software is disabled. See the information on checking the update site.
- Joomla's update cache is stuck. Please file a bug report to Joomla. We have already been reporting this privately since 2014 and made an official, public issue report on March 9th, 2021.
- You have a server issue connecting to our CDN. See the information on addressing server issues.
- You have found a bug in Joomla's built-in extensions updater. You need to contact the Joomla! forum [https://forum.joomla.org]. Unfortunately there is nothing we can do about Joomla core bugs.

3.1.4.1. Check the update site

First we are going to check if the Update Site is disabled. Go to the System menu item, find the Update area and click on the Update Sites link.

On that page you will see a list of the update sites for the extensions you have installed on your site. If you see our software in that list – you may have to search for it – make sure it's published, i.e. there's a green checkmark in the Status column. If it's not already published publish it now. If you had to publish the Update Site you will have to install the latest version manually, on top of the old one. This is due to Joomla's caching of the information required to retrieve update information. In other words, for a period of a few hours to a few days Joomla might be unable to automatically detect the new version of our software.

If our software does not appear on that list you will need to click on Rebuild. Watch out, though! Joomla has a bug. When using Rebuild **removes** all Download IDs and Download Keys from all installed extensions on your site and will cause extension installation to fail. You will also need to follow the instructions under Check the validity of your Download ID for your updates to work.

3.1.5. Miscellaneous troubleshooting and information

3.1.5.1. The update fails to download

If you are trying to update a Professional edition please check your Download ID. Typically you will get an error message telling you that an error 403 or 500 was received when trying to download the update package. Whether you see that message or a generic download failure message depends on the version of Joomla you have installed on your site.

If this doesn't help you need to check if you have a server issue.

3.1.5.2. Updating with a third party service fails

Typically, third party site management services ask Joomla to provide the update information and install update on your behalf. Therefore the troubleshooting information in this section would solve both in-site and remote (via a service) extension updates.

If you can install an update by logging into your site's backend but NOT through a service you need to contact the third party site management service and report this issue. Unfortunately we cannot help with it. Third party services DO NOT ask us for permission to implement an updater for our software.

3.1.5.3. Manual update

As noted earlier in the documentation, a manual update is the same as installing the extension. Download the latest version from our site and install it on your site **without** uninstalling our extension.

3.1.5.4. Update installation problems

If your update does download but fails to install try the manual update method (installing the new version on top of the old one). If that fails, too, you should follow the instructions on the installation troubleshooting section you can read earlier in this documentation.

3.2. Entering your Download ID

> **Note**
>
> If you are using Akeeba Backup Core, the free of charge edition of Akeeba Backup, you do not need to and must not enter a Download ID. The Download ID is only required for the for-a-fee Akeeba Backup Professional edition.

Akeeba Backup Professional is the for-a-fee edition of Akeeba Backup with additional features. Downloading it, either for installation from scratch or as an update to an already installed but older version on your site, requires confirming that you have an active subscription which gives you access to Akeeba Backup Professional downloads. When you download the installation ZIP file from our site this means that you need to log in to our site first. However, when downloading updates through Joomla you really don't want to and usually cannot be asked to log in to our site. The Download ID is used in this case to identify you to our download servers. Furthermore, a Download ID linked with an active subscription is also required to use some remote file storage services such as Dropbox and OneDrive (specifically: any remote storage service which uses OAuth2 or a variation thereof, therefore requiring a special “mediator” script which runs on our servers).

Using your Download IDs on your clients' sites

Our software license allows you to use your Download IDs on the sites of your clients. However, you must tell your clients that:

- Downloads and support for the software covered by the Download ID is provided by you, not Akeeba Ltd.
- If they want to receive support and / or downloads directly from Akeeba Ltd they need to purchase a qualifying subscription on our site. In this case they do not qualify for the renewal discount.
- They are not allowed to use the Download ID on any other site or use the Download ID to download the software for any reason other than updating or reinstalling the covered software on the same site the Download ID was entered in. In other words, they cannot use the Download ID to install or update our software on any other site.

If you are no longer administering a site where you have entered a Download ID you must revoke or regenerate that Download ID. You need to do the same if you believe that your Download ID is being used by third parties in an unauthorized manner. Please note that unauthorized use of Download IDs could have consequences with regards to your subscription with us.

Finding your Download ID

Download IDs come in two flavors, your main Download ID and Add-on Download IDs.

You can find your main Download ID in the My Subscriptions [https://www.akeeba.com/my-subscriptions.html] page of our site. We recommend using this Download ID only on your own site(s). This Download ID cannot be revoked,

it can only be regenerated. If it's regenerated you will need to enter the new Download ID on all of your sites which can be a significant hassle.

You can generate an unlimited number of Add-on Download IDs without additional charge in the Add-on Download IDs [https://www.akeeba.com/download/add-on-dlid.html] page. Unlike the main Download ID you can revoke (disable) any Add-on Download ID at any time. As long as you only use one Add-on Download ID per site revoking or regenerating it will not affect the other sites' ability to download and install updates.

Enter or view your Download ID

From the main administrator page of your site click on System on the sidebar.

Click on the Update Sites link towards the bottom of the middle column on the System page.

Find the Akeeba Backup for Joomla! package entry on the list and click on it to open the edit page.

Note

You might see an entry for "Akeeba Backup package" (without the "for Joomla!" part). These are older versions of Akeeba Backup, i.e. Akeeba Backup 8 or earlier. Please check out the "Migrating from older versions of Akeeba Backup" section in our documentation.

Enter your main or Add-on Download ID in the Download Key area. Click on the Save & Close button on the toolbar to apply the Download ID.

If Akeeba Backup or Joomla! was already showing you that an update for Akeeba Backup is available you will need to install the update manually.

Troubleshooting updates to the Professional release

If you still cannot install our software please check that the Download ID is entered correctly. If it's not entered correctly enter the correct Download ID and follow all of these instructions again.

If the Download ID is entered correctly but it's not active in the Add-on Download IDs page you will need to enable it. After enabling it you will be able to download and install the update *without* having to follow these instructions again.

If the Download ID is correct please make sure that you have an *active* qualifying subscription on our site. If your subscription has expired you need to purchase a renewal on our site. Once the renewal is active you will be able to download and install the update *without* having to follow these instructions again, as long as you have not changed your Download ID.

If you still cannot download updates despite having the correct Download ID and an active subscription try waiting for 24 to 48 hours. In very rare cases Joomla's update cache gets stuck despite following the instructions above and you just need to wait until Joomla decides it has to reload it.

If the updates are still not downloading please make sure that you are using a version of Joomla and PHP that is supported by the new version of our software. If you are not sure please consult our Compatibility page [https://www.akeeba.com/compatibility.html].

If you've followed all these troubleshooting steps and the update is not downloading at all you need to contact your host and ask them to allow traffic to `www.akeeba.com` and `cdn.akeeba.com` over port TCP 443 (HTTPS), make sure that the PHP cURL module is installed and activated on the version of PHP your site is using and that finally the libcurl and libssl system libraries the cURL module is compiled against are up-to-date versions. If your host cannot help you with any of these requests (despite this being literally what you are paying them to do) you can install updates manually. Kindly note that Akeeba Ltd is not responsible for your hosting environment and that the

requirements for downloading updates from our site are met by server software released roughly 5 years ago. If your host cannot provide 5 year old software and open ports in their firewall you should probably be migrating your site to a more up-to-date, competent host.

4. Uninstalling Akeeba Backup

Akeeba Backup can be uninstalled just like any other Joomla extension.

Warning

Uninstalling Akeeba Backup will delete your backup profiles, the list of your backup attempts (what you see in the Manage Backups page) and any backups stored in the default backup output folder which is inside Akeeba Backup's component folder. There is no further confirmation. This process is **IRREVERSIBLE**. If you lose your backups by uninstalling Akeeba Backup we cannot help you retrieve them, they are gone forever.

First, go the extensions manager page. From the sidebar of your Joomla! Administrator click on System. On the new page find the Manage area towards the top of the middle column and click the Extensions link.

In the Search box type Akeeba Backup package. It will show you a single item called "Akeeba Backup for Joomla package" whose Type is Package.

Note

You might see an entry for "Akeeba Backup package" (without the "for Joomla!" part). These are older versions of Akeeba Backup, i.e. version 8 or earlier.

Important

Only ever try to uninstall the Package type extension. DO NOT try to uninstall the component, its plugins or module individually. It will leave stuff behind.

Select the item's checkbox and click on the Uninstall button in the toolbar. The extension and all its dependencies will be automatically uninstalled.

5. Requesting support and reporting bugs

Support can be provided only to subscribers and only through our site's Support section. 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 for Joomla! and click on the New Ticket button. If you can't see the button please make sure you have an active subscription that gives you access to Akeeba Backup for Joomla! support. If you do and still don'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. An issue is not a bug unless it can be reliably reproduced *on multiple sites and servers*. Please make sure you include clear instructions on reproducing the issue. If the issue cannot be reproduced it's not a bug report, it's a support request.

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.

6. Migrating from old versions of Akeeba Backup

Akeeba Backup has been around since the early days of Joomla 1.0 all the way back in 2006. It has gone through several rewrites, the latest one being in mid-2021 with Akeeba Backup for Joomla version 9. If your site already had an older version of Akeeba Backup installed you will need to take some easy, manual steps to migrate to the new version.

First, make sure that the old version already installed is in the 7.x or 8.x version range. Go to your site's backend and click on System from the side menu. From the Manage pane select Extensions. Search for **`Akeeba Backup package`**. If the version number listed there begins with 6 or a lower number you will need to upgrade this to Akeeba Backup 8. Go to our site, download the latest version of Akeeba Backup 8 and install it. Then download and install Akeeba Backup 9 or later, *even if you had previously installed it*. **This last step is VERY important.**

From Joomla's sidebar menu click on Components, Akeeba Backup for Joomla!™, Control Panel. Towards the top of page, above the configuration and action controls, you will see an area titled "Migrate your settings from an older Akeeba Backup version". Click on the Migrate settings button and follow the instructions on your screen (essentially, click a button). This will migrate your settings from the old version of Akeeba Backup to the new one.

You can now uninstall the old version of Akeeba Backup. Go to your site's backend and click on System from the side menu. From the Manage pane select Extensions. Search for **`Akeeba Backup package`**. Select it, making sure its version number starts with 7 or 8. Then click on Uninstall from the toolbar.

Caveats

The migration will copy over your component settings, Akeeba Backup profiles and backup history. It will also copy the backup archives stored in the default backup output directory of Akeeba Backup 8 and earlier (administrator/components/com_akeeba/backup) to the new default backup output directory (administrator/components/com_akeebabackup/backup).

The migration will NOT copy backup archives from non-default backup output directories created inside the old Akeeba Backup component directory. That is to say, any backup archive inside a subdirectory of administrator/components/com_akeeba (other than administrator/components/com_akeeba/backup), components/com_akeeba or media/com_akeeba WILL NOT be copied over. You need to download these backup archives **PRIOR** to uninstalling the old Akeeba Backup version or they will be forever lost. We do not accept any responsibility for data loss resulting from your failure to heed this warning.

CRON scripts

If you were using the akeeba-backup.php, akeeba-altbackup.php, akeeba-check-failed.php or akeeba-altcheck-failed.php scripts please note that they have been removed and replaced by commands for the Joomla! CLI application. You will need to edit your CRON jobs and make the following replacements:

- **akeeba-backup.php** to **joomla.php akeeba:backup:take**
- **akeeba-altbackup.php** to **joomla.php akeeba:backup:alternate**
- **akeeba-check-failed.php** to **joomla.php akeeba:backup:check**

- **akeeba-altcheck-failed.php** to **joomla.php akeeba:backup:alternate_check**

For example, if your CRON command line was

```
/usr/local/bin/php-cli /home/mysite/cli/akeeba-backup.php --profile=2 1>/dev/null 2>/dev/n
```

you need to change it so that it now reads

```
/usr/local/bin/php-cli /home/mysite/cli/joomla.php akeeba:backup:take --profile=2 1>/dev/n
```

You also need to make sure that the plugin Command – Akeeba Backup is published on your site. This plugin is published by default when you install Akeeba Backup for Joomla version 9 or later for the first time.

Please note that the `joomla.php` application is part of Joomla itself. We do not have control over it. If you get an error before the Akeeba Backup command produces any output the problem lies with Joomla, not our software. Also note that unlike our old CLI scripts, Joomla's CLI application will NOT run with PHP-CGI binaries. It will simply throw an error. Please do not report this as a bug to us, report it to the Joomla project who's responsible for this code and can actually fix it.

Dark Mode

Unlike Akeeba Backup 7 and 8 we no longer include Dark Mode CSS. This is intentional as we are no longer using our own CSS files; we use the Bootstrap CSS which is included with Joomla itself. This means that if you use a dark themed administrator template or a dark mode plugin for the default administrator template our software will also display in the same dark theme as the rest of your site.

Why is a migration necessary?

Akeeba Backup 3.x to 8.x inclusive had the component name com_akeeba, therefore their files were stored in the administrator/components/com_akeeba and components/com_akeeba folders on your site. Its tables had the _ak_ prefix.

During the first two major versions (1.x and 2.x) Akeeba Backup was called JoomlaPack. Since it was compatible with Joomla 1.0 which did not have an extension framework per se they were simply a bunch of PHP scripts with some Joomla binding thrown together. JoomlaPack 2.2 was the first version using Joomla 1.5's MVC. In version 3.0 the component was renamed to Akeeba Backup. Akeeba Backup 3.0 to 3.4 were based on Joomla 1.5's core MVC API. Akeeba Backup 3.5.0 and later were based on our FOF framework (version 1.x, 2.x, 3.x or 4.x of the FOF framework, to make matters more complicated).

The fact that all of these versions of our component had the same component folder name but entirely different internal architectures meant that updates from one version to the next were complicated and error-prone. It was a necessary evil since Joomla 1.5, 1.6, 1.7, 2.5 and 3.x had a very old core MVC API which was unsuitable for writing *and efficiently maintaining* complex extensions. As a result we had to maintain our own framework and go through complicated extension updates. Even though that was time consuming and potentially error-prone on updates it was less problematic than using the outdated core MVC API.

Starting with Joomla 4.0 the core MVC API includes many of the features we had introduced in FOF. Moreover, bumping the minimum PHP version support to 7.2 in Joomla 4.0 means that the few features we are missing can easily be implemented with minimal amounts of reusable code, called Traits. As a result we decided to discontinue our own framework and use the core MVC API again.

However, this meant that trying to upgrade from Akeeba Backup 8.x and earlier would be very complicated if we kept the same component name due to the nature of the Akeeba Backup component. We needed to use a different component name. We decided to use com_akeebabackup since this addressed another long standing issue: com_akeeba was a misnomer. Akeeba Ltd is the name of our company, our extension is Akeeba Backup, not Akeeba. The original component name comes from a time when we only had one extension so it made sense to call it com_akeeba.

The other problem we had to address was that using the core Joomla MVC API makes it hard to update the database tables if they were created without using the core Joomla extensions installer to begin with, as was the case with Akeeba Backup 4.0 to 8.x. If you were to update from Akeeba Backup 7 or earlier to Akeeba Backup 9 the updated extension would fail to work properly because the database tables wouldn't be updated correctly. Therefore we had to use a different table prefix, _akeebabackup_ which is good practice as well: the table prefix should ideally follow the component name.

These two changes meant that a migration is required from older versions of Akeeba Backup to Akeeba Backup 9 and later. Ideally we'd like to do that when you install the new version of our extension. However, due to limitations of the Joomla extensions installer and updater we could not guarantee there would be enough time available to do that without risking a PHP timeout. Hence the need to do a manual migration. The migration is required exactly once.

Chapter 3. Using the Akeeba Backup component

This chapter documents all pages and features of the Akeeba Backup component. We decided to organise it by each core Joomla feature or component page you will see using our extension. When there is an interconnection between features we attempt to provide links between the documentation pages and adequate information to explain this interconnection.

1. Custom administrator menu items

Joomla allows you to create custom administrator menus. This is a very powerful feature, allowing you to customise the backend Joomla interface for your clients to provide better User Experience e.g. by creating task-based menu items instead of the generically-named default Joomla menu items. Akeeba Backup offers full support for this feature.

Most of Akeeba Backup's custom menu item types were created with site integrators and 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 easily 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

Note

This menu option type is only available and will only work with Akeeba Backup Professional.

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. Site Transfer Wizard

Note

This menu option type will only work with Akeeba Backup Professional.

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 occurs 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_akeebabackup` 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_akeebabackup` in your browser's address bar to access Akeeba Backup.

Restoring Joomla's default administrator menus

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 a Save icon which saves settings and returns you to the same editing page.

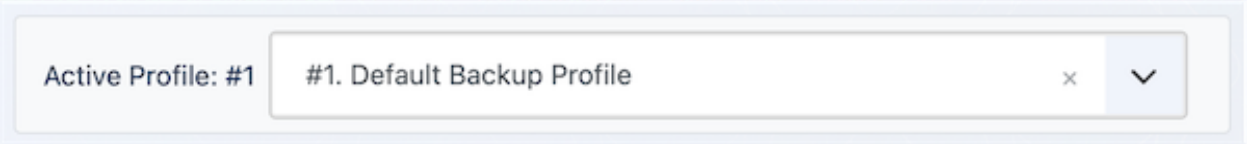
On the bottom of each page, just above the Joomla!™ footer, there is the license information. On the Control Panel page of the Akeeba Backup Core editions there is also a donation link appearing on the right sidebar; if you feel that Akeeba Backup was useful for you do not hesitate to donate any amount 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 any problems or configuration issues are detected, Akeeba Backup will report one or more error or warning messages.

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

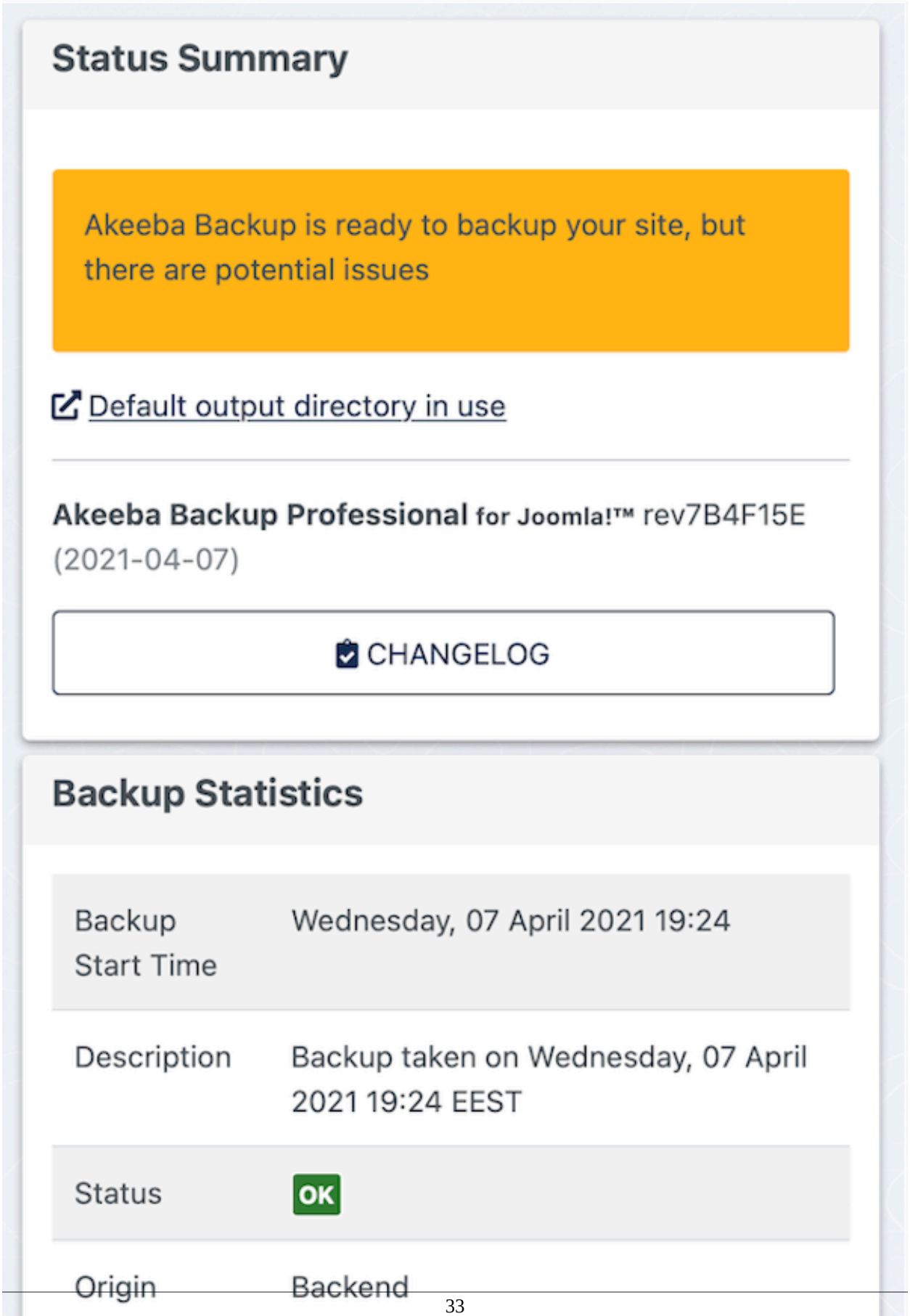


Towards the top of the page, there is the profile selection box. It serves a double purpose, indicating the active profile and letting you switch between available profiles. Clicking on the drop down allows you to select a new profile. Changing the selection (clicking on the drop down list and selecting a new profile) automatically makes this new profile current and Akeeba Backup notifies you about that.

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.

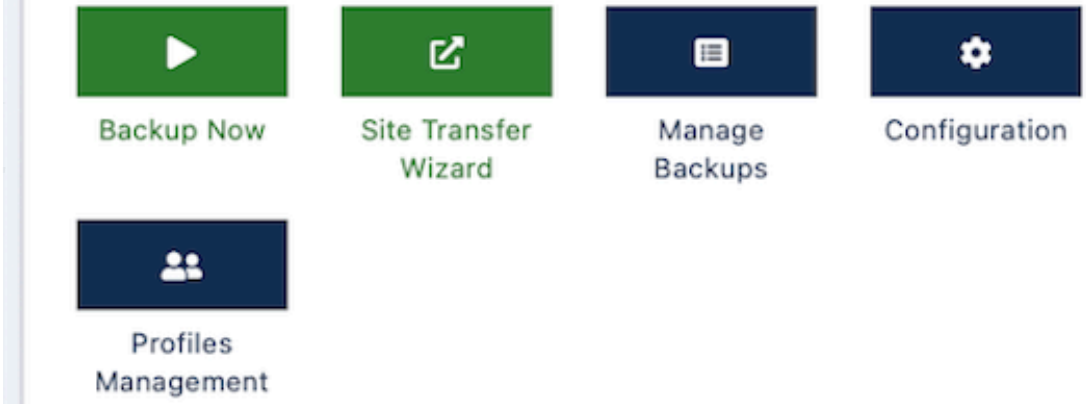

Status Summary
Akeeba Backup is ready to backup your site, but there are potential issues
Default output directory in use
Akeeba Backup Professional for Joomla!™ rev7B4F15E (2021-04-07)
CHANGELOG
Backup Statistics
Backup Start Time
Wednesday, 07 April 2021 19:24
Description
Backup taken on Wednesday, 07 April 2021 19:24 EEST
Status
OK
Origin
Backend
Type
Full site backup

There are two areas:

Status Summary
: Akeeba Backup performs a number of self-checks to make sure that your configuration is consistent and will result in a valid backup. If there are any problems detected they will be communicated to you here, including links to documentation pages on our site explaining how to resolve them.

  If you are a Core (free version) user, you will see a donation link. If you feel that Akeeba Backup has helped you - 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. Thank you!

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.



Troubleshooting

Advanced Operations

Include and Exclude Information

The main navigation panel set allows access to the different functions of Akeeba Backup. You can access them by clicking on the respective each icon. Please note that the screenshot in this documentation displays the Professional version. If you are using the Core version you will have fewer options.

Depending on your backup profile settings, at the top of this area you may find a series of buttons under the header **One-click backup**. Clicking one of these buttons will start a backup with the corresponding backup profile, without asking you for confirmation.

Finally, you can edit the global options of Akeeba Backup by clicking on the Options button towards the top right hand of the page, in the Joomla! toolbar area at the top of the page.

2.2.1. Additional controls, warnings and error messages in the Control Panel

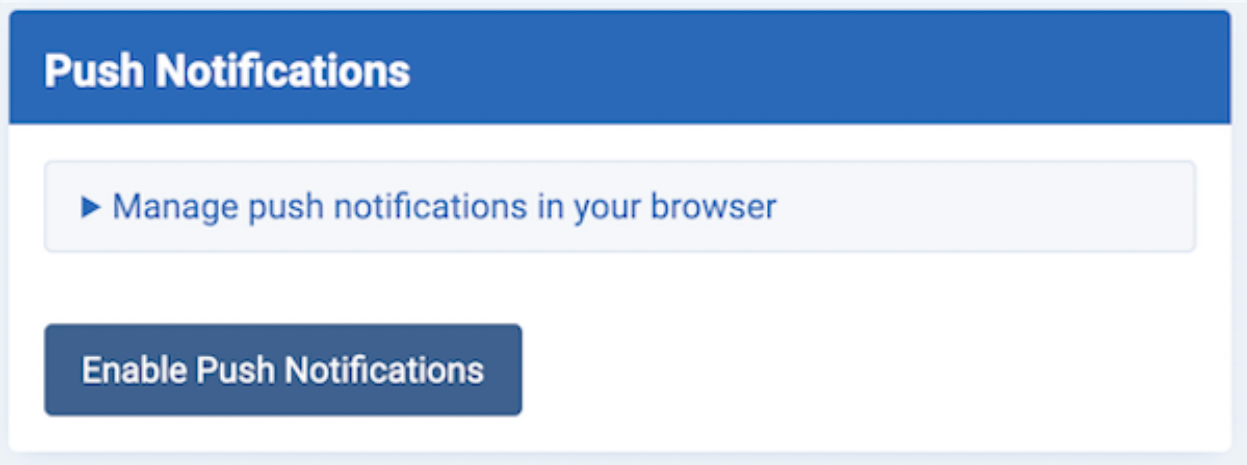
Akeeba Backup performs a number of self-checks every time you visit its Control Panel page. It checks your settings, your profile configuration and your server environment to make sure that you will not be facing any easily preventable issues. When it detects an anomaly it shows you a message to help you identify and resolve a potential issue. This section discusses the various messages you might encounter at the top of the Control Panel page.

2.2.1.1. Web Push controls

Akeeba Backup 9.3.1 and later support using the browser's Push API for sending you push notifications about backup events (backup start, backup end and backup warnings). You can enable that in the component's Options page. If you have done so, you will see the Push Notifications control area right below the backup quick icons.

If your browser reports that notifications are not enabled for this browser and device you will see the Enable Push Notifications button.

Web Push – Notifications disabled



Clicking this button will ask you for permissions to receive push notifications the first time you try using push notifications on this particular site from this particular browser and device. Please note that the browser only understand which site is asking for permission, not which Joomla component. If you had given, or rejected, push notifications from another component running on the same site the browser will remember your preference and will not ask you again.

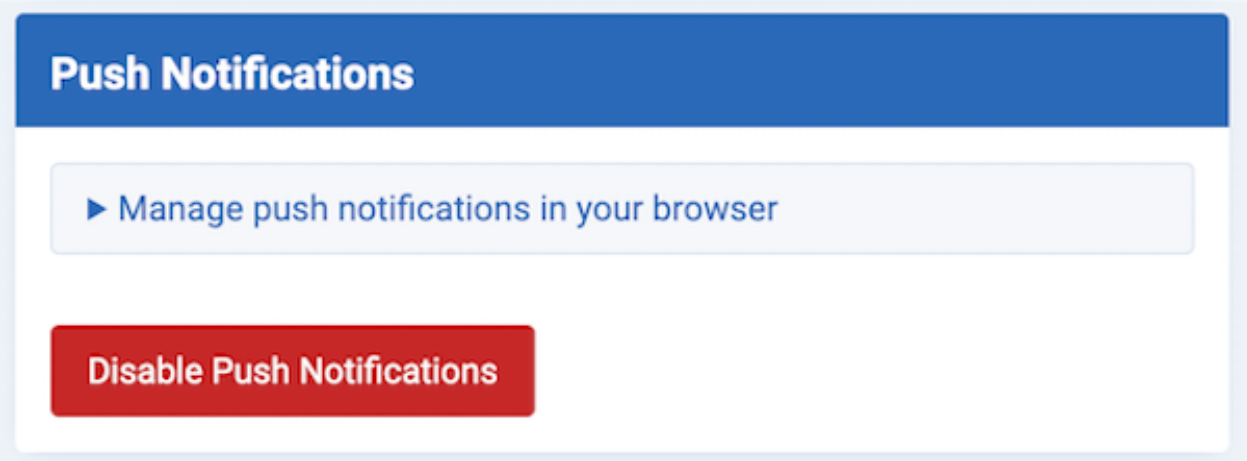
If you reject notifications, or if you had already done so in the past (even for another component on the same site) you will see an error message appear towards the top of the page. If you didn't mean to do that you will need to unblock

notifications for your site on your browser; usually this involves going into the browser's privacy settings for your site. Please refer to your browser's documentation for more information. Also note that browsers do NOT provide an API for overriding your choice or asking you again for consent to push notifications if you have already rejected them. This means that we can NOT implement any kind of feature in our software to help you with that. Sorry! Browsers (correctly) put your privacy ahead of your convenience in this case.

If you accept to receive push notifications you will see a push notification from your site confirming your choice. If this does not happen, or if you receive a server 500 error, please reload the page and click on the Disable Push Notifications button.

When your browser reports that push notifications have been allowed and you are subscribed to them you will instead see the Disable Push Notifications button.

Web Push – Notifications enabled



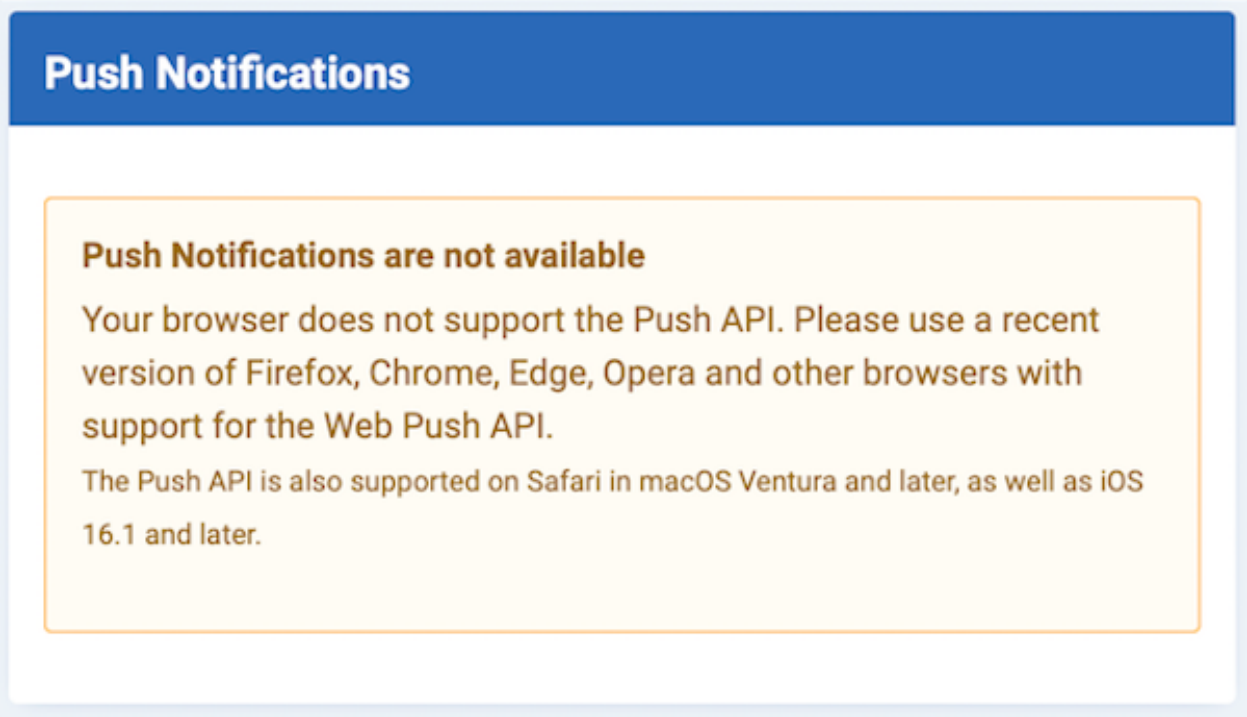
Clicking on that button unsubscribes this specific browser on this specific device from Akeeba Backup's push notifications.

Tip

If you see this button but do not receive push notifications when backups are running it is possible that the browser's push notification state and what is going on in your site are out of sync. This can happen if you have restored an older version of your site, from before you subscribed to Akeeba Backup push notifications on this browser and device. Click the Disable Push Notifications button, reload the page, and click on Enable Push Notifications again.

Finally, if your browser does not support Web Push you will see a message letting you know this is the case.

Web Push – Unsupported



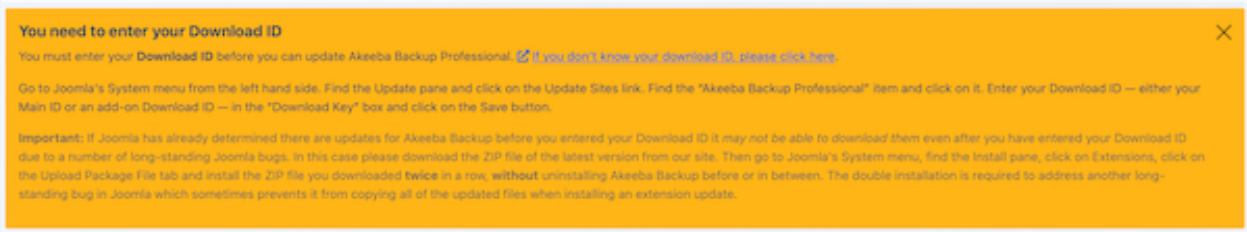
2.2.1.2. Full page errors

In the unlikely event that your server has a major configuration issue, e.g. using an outdated PHP version or PHP module, which prevents you from running our software *at all* instead of the Control Panel page you will see a full screen error message. The page will tell you what the problem is and how to fix it. For example, if you have an outdated PHP version:

Outdated PHP error page

2.2.1.3. Download ID messages

The Download ID message in the Professional

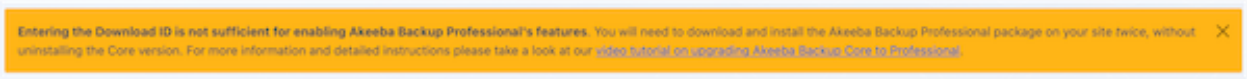


If you are using Akeeba Backup Professional for Joomla! you will be asked to enter your Download ID. This is necessary to receive updates to the software. Without it you will be notified for updates but you will not be able to install them. Click on the link in the message to log in to our site and receive personalized instructions for entering the Download ID.

Tip

You can create Add-On Download IDs free of charge on our site. Just log into our site and click on the "add-on download IDs" link below the header. You can have a different Download ID for each of your clients. If the client stops working with you, you will be able to unpublish (deactivate) their Download ID.

The Download ID message in the Core version



Conversely, if you are using the Akeeba Backup Core for Joomla! but have entered a Download ID you will receive an error message reminding you that this is not the proper way to upgrade to the Professional version. Instead, you should install the Professional version on top of the Core version. If you are not sure how to do this please click the link at the end of the message. It takes you to a video tutorial telling you how to do this.

2.2.1.4. Media files' permissions

Media folder permissions

If Akeeba Backup detects a problem with the permissions of the media folder, where its JavaScript, CSS and image files are stored, it will try to automatically do the necessary changes for you. It requires that you have provided FTP connection information to your site's Global Configuration and enabled the FTP option in that page.

If these changes cannot be done automatically it will display this error message. Please follow the instructions in the message.

If you have already followed the instructions in the message but the interface behaves erratically or appears "broken" one of your system plugins is killing Akeeba Backup's JavaScript. Check your browser's developer tools to see which

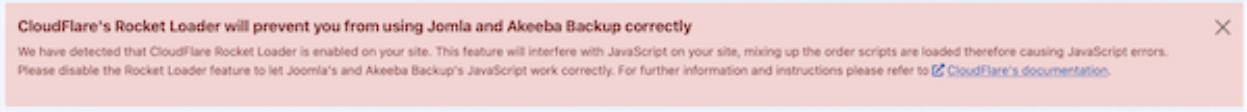
third party JavaScript is causing that. If you can't figure it out yourself please contact us and give us Super User and FTP access to your site so we can help you.

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.

2.2.1.5. CloudFlare RocketLoader

CloudFlare RocketLoader warning



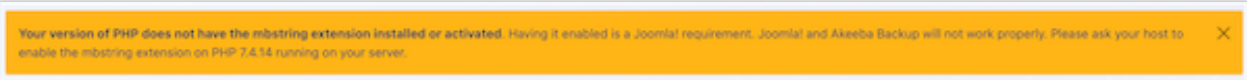
CloudFlare's RocketLoader changes the load order of the JavaScript on every page of your site, deferring the loading of every file at the end of the page load. Unfortunately, this causes applications depending on JavaScript, like Akeeba Backup *or even Joomla! itself*, to fail due to no fault of their developers.

Please do note that Akeeba Backup's JavaScript is always loaded deferred. Likewise, Joomla already defers loading of as much JavaScript as possible. The problem is that there is a minimal amount of core Joomla JavaScript which can not and must not be deferred. CloudFlare's Rocket Loader doesn't know about this, tries to defer this core Joomla JavaScript, causing all JavaScript on the page to break.

Please follow the link at the end of the message for instructions to manually fix this problem.

2.2.1.6. Missing mbstring

Missing mbstring warning



Akeeba Backup, like Joomla! itself, requires the PHP extension called "mbstring" to be loaded and activated. Without it it is impossible to handle extended characters and find the length of binary data. Therefore, if mbstring is missing your backup will fail. Please ask your host to enable mbstring on your site. The PHP version for which mbstring needs to be activated, as reported by PHP running on your server, is printed on the message on your screen. Please copy the message from *your* site to your host's support – do not copy the version displayed in the example screenshot above.

Kindly note that Joomla itself requires the mbstring PHP extension to work properly. Akeeba Backup DOES NOT require something that *Joomla itself* doesn't already require for basic operation. In other words, we are asking you to fix a hosting issue which affects your entire site, not just Akeeba Backup.

2.2.1.7. Obsolete PHP version

Obsolete PHP version

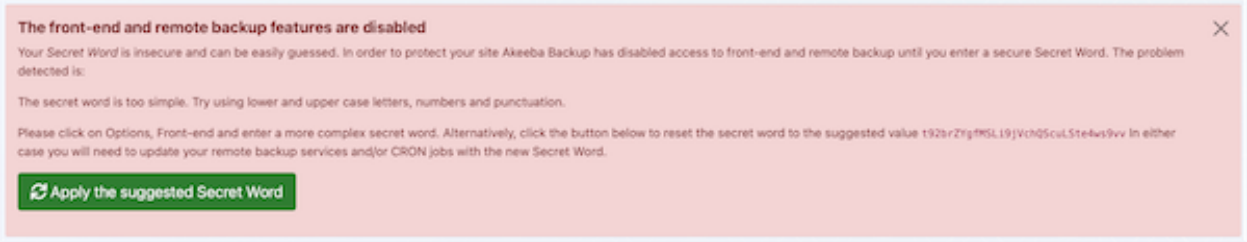
Akeeba Backup always checks the PHP version it runs under. If your PHP version is very old and declared end-of-life (EOL) by the developers of PHP we will warn you. EOL versions of PHP have known bugs which prevent software from running correctly, slow, and insecure. Even if your Linux distribution's vendor claims to still support them, the fact remains that major security and functional flaws are NOT fixed.

Because of these reasons, Akeeba Ltd only officially supports running our software on the versions of PHP which are still under active maintenance per the official PHP site [http://be2.php.net/supported-versions.php]. We only guarantee support End Of Life versions of PHP for 3 to 6 months after their End Of Life date as published in the official PHP site [http://www.php.net/eol.php].

Akeeba Backup will display a slightly different message depending on whether your PHP version has entered Security Maintenance (still minimally maintained by PHP), recently became End of Life or has been End of Life for a very long time.

2.2.1.8. Front-end backup Secret Word

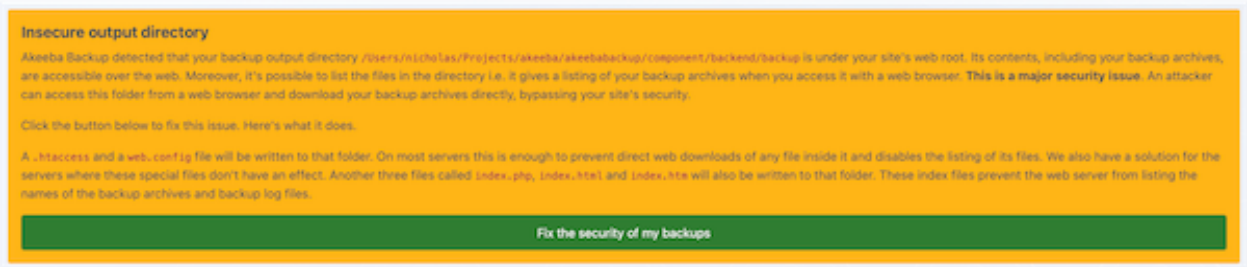
Front-end backup Secret Word



If you have enabled the legacy front-end backup feature or the JSON API feature, Akeeba Backup checks the existence and quality of the Secret Word. If none was found or it's too simple / easily guessable it will decline to run front-end and remote backups until you use a more secure Secret Word. This is a security precaution: an easily guessable Secret Word could be used to launch a Denial of Service attack to your site or steal information from it. If you are not sure how to create a secure secret word click the large button to apply an automatically generated, secure secret word.

2.2.1.9. Insecure output directory

Insecure output directory



Akeeba Backup periodically checks the security of the Backup Output Directory configured in the currently selected backup profile. It does so by trying to write a small file in it and the read it by accessing it through its URL (not by reading the file through the file system). If that works, it means that anyone on the Internet can read any file stored in the backup output directory if they know their name – including log files and backup archives. This can be a major security issue because it may allow an attacker to gain access to privileged information about your site or even a complete copy of your site.

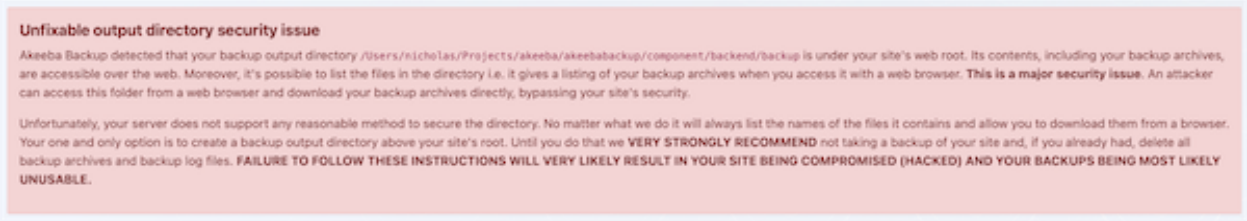
Clicking the Fix the security of my backups button will address the problem by installing a number of files in the backup output directory which block direct web access to the files or, at the very least, make listing the names of the contained files impossible.

It is also recommended that you edit the backup profile's Configuration and add `-[RANDOM]` to the end of the backup archive name. This will append 16 completely random, alphanumeric characters to the backup archive's name. This is a security feature. It makes it extremely hard for an attacker to successfully *guess* the name of your backup archives. Even if your server does not allow Akeeba Backup to prevent direct web access to the files in the output directory, the combination of the files which prevent listing the file names (meaning that an attacker would have to *guess* the name of your backup archive) and the random alphanumeric characters at the end of the backup archive name (meaning that the attacker would need several hundreds of thousands of years to successfully guess the name of your backup archive) protect your backups adequately in the overwhelming majority of use cases.

Important

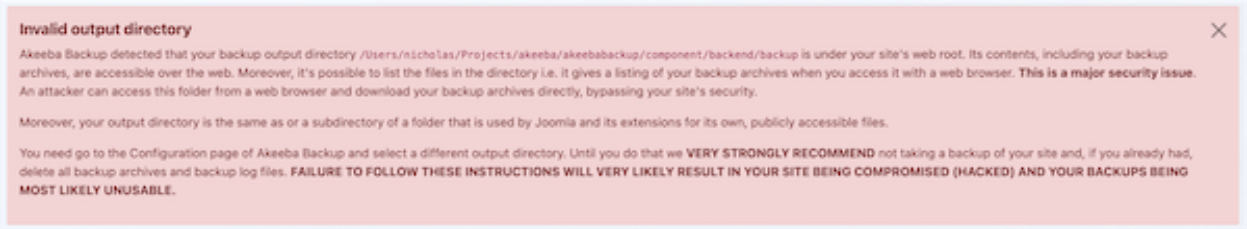
The **BEST** solution to this issue is creating a dedicated folder for your backup placed outside your server's web root. Please consult the Security Information chapter of our documentation to understand how servers work and why this solution is the best one possible.

Insecure output directory, unfixable



If Akeeba Backup cannot find a way to disable access to your backup output directory it will show a sterner error message. It will be forcibly appending `-[RANDOM]` to the end of the backup archive name, even if you haven't done so. However, this is still a not very secure situation. It is possible that an attacker may be able to list the backup archives in the backup output directory which will allow to download them. The best approach here is to create a dedicated folder for your backup placed outside your server's web root.

Invalid output directory



If your backup output directory is your site's root or a folder that's already used by Joomla itself or one of its extensions you will receive an error. This kind of output directory is wrong for two reasons.

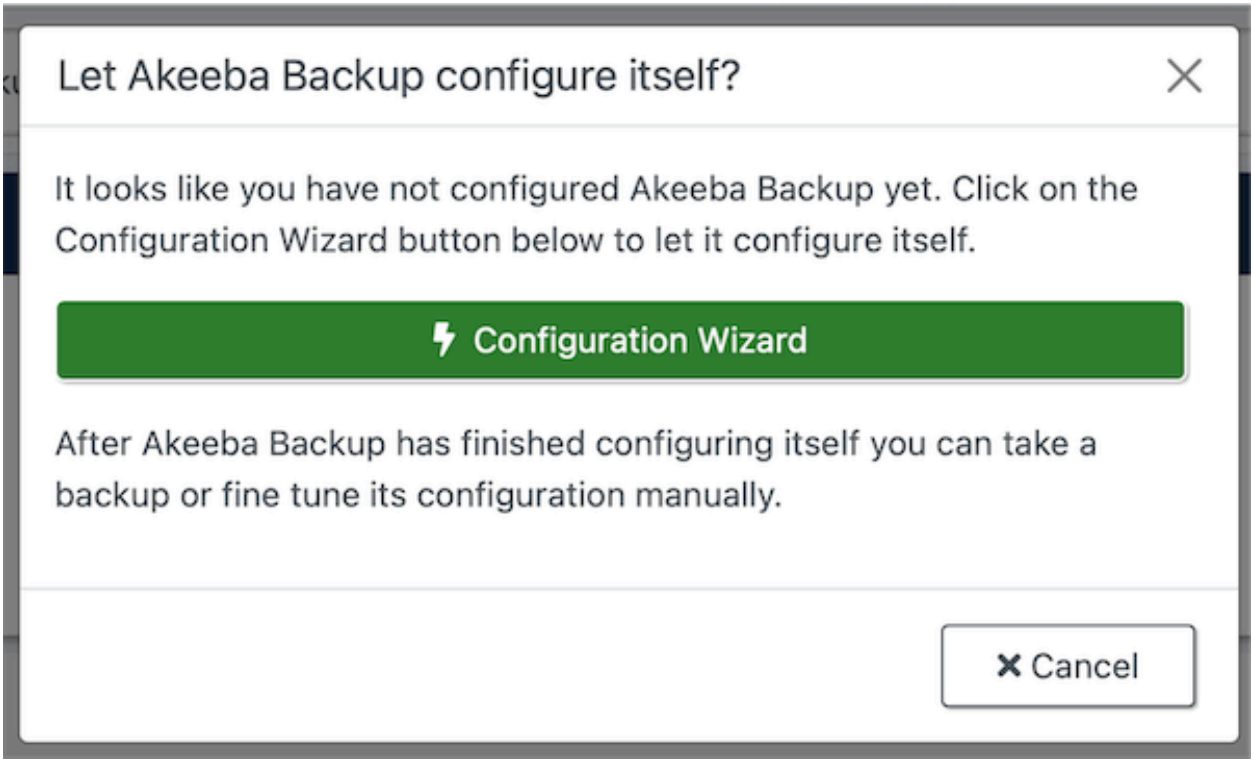
First, the contents of the output directory are automatically excluded from the backup. Using your site's root or a folder used by Joomla or one of its extensions will result in a partial backup which cannot be successfully restored. This is considered a major mistake and Akeeba Backup will not let you take a backup until you fix it.

Furthermore, Akeeba Backup cannot disable web access to your site's root or a folder used by Joomla! itself or one of its extensions. Doing so would break your site. Instead of doing something which would damage your site it instead prefers showing you an error, informing you that you did something wrong.

The best approach here is to create a dedicated folder for your backup placed outside your server's web root.

2.2.1.10. Configuration Wizard

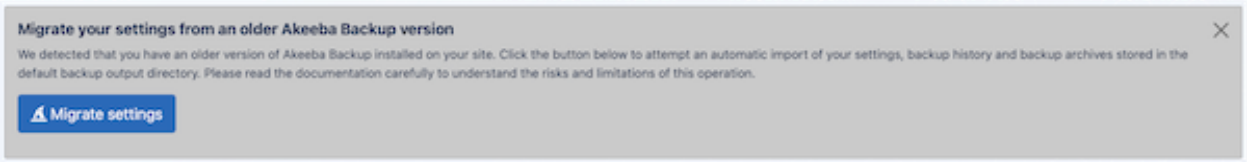
Configuration Wizard notice



If Akeeba Backup detects that you have a brand new backup profile that has not been configured yet it will ask you to run the Configuration Wizard. The Wizard runs without requiring any input from you. Sit back and let Akeeba Backup figure out what are the best backup settings for your site. We recommend all of our users to use the Configuration Wizard as it prevents the most common backup problems you may encounter.

2.2.1.11. Migrate your settings from an older Akeeba Backup version

Migrate your settings from an older Akeeba Backup version



If Akeeba Backup detects that you have installed version 9 or later but you also have version 8 or earlier still installed on your site it will ask you to migrate your settings, backup history and backups from the old version to the new one.

Please refer to Migrating from old versions of Akeeba Backup for more information about why this is necessary, what it does and the possible caveats of the migration process.

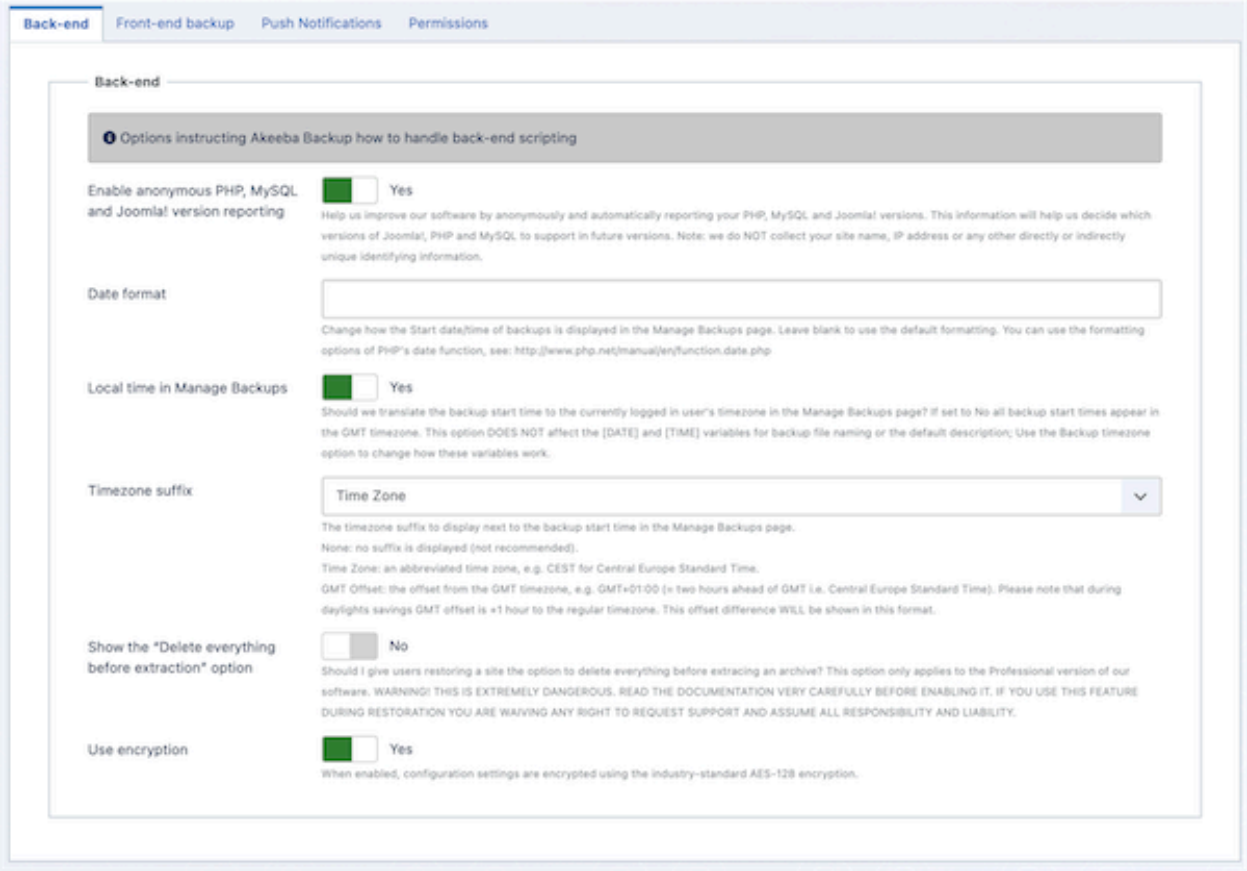
2.2.2. Editing the component's Options

You can edit the global, component-wide options by clicking on the Options button towards the top right hand of the page, in the Joomla! toolbar area at the top of the page. The Options editor opens in a new page. Please note that the

Options page is handled by Joomla itself, namely the built-in com_config component. Further to that, the component Options apply *regardless* of the active profile.

The documentation of the component Options page is organised by each tab displayed by Joomla.

2.2.2.1. Back-end



These options define how Akeeba Backup will display its administration interface

Enable anonymous PHP, MySQL and Joomla! version reporting
: Help us improve our software by anonymously and automatically reporting your PHP, MySQL and Joomla! versions. This information will help us decide which versions of Joomla!, PHP and MySQL to support in future versions.

  Note: we do NOT collect your site name, IP address or any other directly or indirectly personally identifying information. A randomly generated site ID is used to make sure only the latest information submitted by your site is taken into account every month. At the end of each month we generate and store aggregate information, discarding the individual data points collected for that month. Therefore it is impossible for us to link any of the information submitted back to a specific site or individual in full compliance with the European Union's General Data Protection Directive which governs our handling of personally identifiable information.

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].

Local time in Manage Backups
: When this option is set to No the time the backup started is shown in GMT timezone in the Manage Backups page. If you set it to Yes the time will be shown in the logged in user's timezone.

Please note that this feature will not work reliably unless you have set the correct server timezone in Joomla's Global Configuration. Keep in mind that your server's timezone may be different than the timezone you live in or the timezone of the hosting company's offices. For example, it's possible for an Australian to be hosted with a British hosting company whose servers are in Amsterdam. The correct server timezone in this case would be Europe/Amsterdam.

Moreover, you need to have selected your local timezone in your user profile in Joomla!.

If these prerequisites are not met the time displayed will be off. Lack of configuration on your part is not a bug on our part. Please triple check your timezone settings before filing a "bug" with this feature.

Timezone suffix

Choose the suffix to append to the backup time in the Manage Backups page. `None` will result in no suffix. We don't recommend it as it's not immediately obvious which timezone is being used. `Time Zone` is the recommended and default option. It will print the human readable timezone setting, e.g. EEST for Eastern Europe Summer Time, PDT for Pacific Daylight Time and so on. `GMT Offset` will instead display the timezone as an offset from GMT, for example GMT+3 for Eastern Europe Summer Time or GMT-7 for Pacific Daylight Time.

Show the “Delete everything before extraction” option

When this option is enabled, users restoring a backup will see the Delete everything before extraction option. This is a **dangerous** option, meant for advanced users. It will try to delete all files under the backed up site's root before starting the restoration. The obvious danger in this option is that it might delete more than you expected since it cannot and does not know the meaning of each folder under your site's root. It might end up deleting your subdomains, add-one domains, your emails or your cat photos.

Before using this option please make sure that you have kept copies of your backups and any important files outside of your site. If you screw up when restoring your site we take no responsibility. You have been warned.

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.

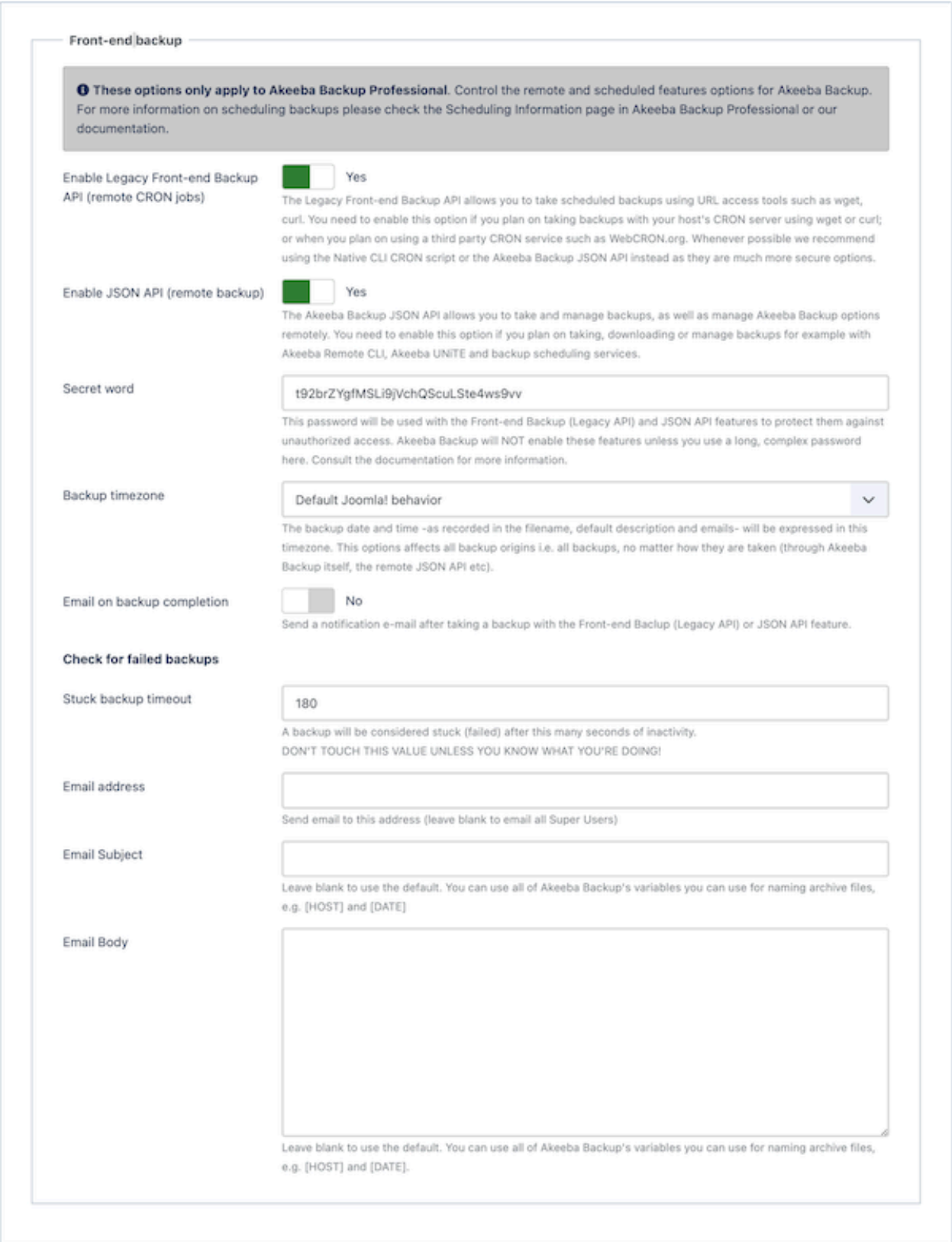
Tip

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.

2.2.2.2. Front-end backup

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



Enable Legacy Front-end Back-

Only applies to Akeeba Backup Professional.

up API (remote CRON jobs)

This option controls whether the legacy front-end backup feature of Akeeba Backup is enabled. This feature allows you to take backups and check for failed backups from the front-end of your site, using standard HTTP redirections. Please remember to enter a Secret Word if you decide to enable this feature.

Enable JSON API (remote backup)

Only applies to Akeeba Backup Professional.

This option controls whether the Akeeba Backup JSON API — used by third party services and the Akeeba Remote CLI command line tool — is enabled. This feature allows you to take backups from the front-end, using standard HTTP redirections. Please remember to enter a Secret Word if you decide to enable this feature.

Secret word

Required to authenticate either of the two previous remote backup methods. Also protects the front-end backup feature from Denial of Service attacks by requiring you to pass this secret word in the front-end backup URL.

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.

For security reasons, you must use a complex enough secret word. Akeeba Backup's JSON API and front-end backup features enforce that by disabling themselves if you are using a Secret Word with a low complexity. We strongly recommend using 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].

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. Always use HTTPS with a commercially signed SSL certificate when configuring or backing up your site.

Backup timezone

The timezone which will be used for all of the backup naming variables processed by Akeeba Backup. These are variables such as [DATE] and [TIME] which you can use in the filename template of backup archives, the backup output directory name, the front-end and remote backup emails and elsewhere.

The default option is called `Default Joomla! behavior` and it will find out the timezone the same way Joomla! does: if there is a logged in user it will use their timezone. If there is no logged in user (e.g. front-end or remote backup) or there is a logged in user but they do not have a timezone set in their user profile the Server Timezone used in the site's Global Configuration will be used instead. If that is not set it will fall back to GMT (Greenwich Mean Time).

We recommend setting this option to the timezone the people responsible for taking and restoring backups are most familiar with. If you are the only Super User use the timezone where you normally live in.

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.

When to send the email
: You can choose when Akeeba Backup will send the email notifying you of the front-end or remote backup completion. If you choose `Always` it will be sent every time a backup runs to completion. If you choose `Upload failed` the email will only be sent for backups which completed but without managing to transfer your backup archive to remote storage. The latter option only has any effect on Akeeba Backup Professional.

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.

  You can enter multiple email addresses separated by commas. For example: `foo@example.com, bar@example.net, baz@example.org`

  Please note that the email addresses are fed directly into Joomla's email library. If Joomla considers your email address to not be well-formed it will not send an email to it at all. We recommend avoiding UTF-8 in email addresses for this reason. Each email message to each email address is sent separately (the are not CC'ed or BCC'ed). The more email addresses you include and the more time it takes for the remote mail server to respond the longer this feature will take to run. We recommend using less than 5 email addresses to avoid PHP timing out. If you need more than 5 email addresses you should consider creating an email forwarder on your server, i.e. an email address which will automatically forward all email messages sent to it to multiple recipients.

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 backup naming variables, 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 backup naming variables, 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

  [FILESIZESLIST]
  : A list of filenames of the archive parts of the backup archive which was just generated and their *approximate* sizes.

    Please note that Akeeba Backup does not store the exact size of each part file. It only stores the total size of all backup parts it has created. Akeeba Backup makes the following assumptions for determining the approximate size of each backup part:

- The size of the only file of a single part backup is equal to the total size of the backup.
- The file size of the whole parts (.j01, .j01, ... for JPA and JPS archives and .z01, .z02, ... for ZIP archives) is equal to the Part Size for Split Archives that you have defined in its configuration. This is true for almost all backup archives. In extremely rare cases parts may be up to 200 bytes short of the part size for split archive. It may also be wrong in case of a filesystem error which caused a backup archive to become truncated. That's why we're calling the size shown by this feature "approximate".
- The file size of the last part (.jpa, .jps, .zip) of a multi-part archive is equal to the total backup size minus the Part Size for Split Archives that you have defined in its configuration times one less than the total number of parts.

Not recording the exact part size is deliberate. Trying to do that on backup archives with hundreds of parts would risk exhausting the PHP memory and cause the backup to fail. We consider being able to take a backup to be far more important than getting down-to-the-byte accurate file sizes reported in an email. Furthermore, byte accuracy is irrelevant since the file sizes are reported in Kb, Mb or Gb (automatically scaled).

[TOTALSIZE] The total size of the backup. If it's a single part backup it reports the size of the only backup archive part. If it's a mulitpart backup archive it reports the sum of the sizes of all backup archive parts. The size is printed in Kb, Mb or Gb (automatically scaled).

[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 or you have the Akeeba Backup Core edition, 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.

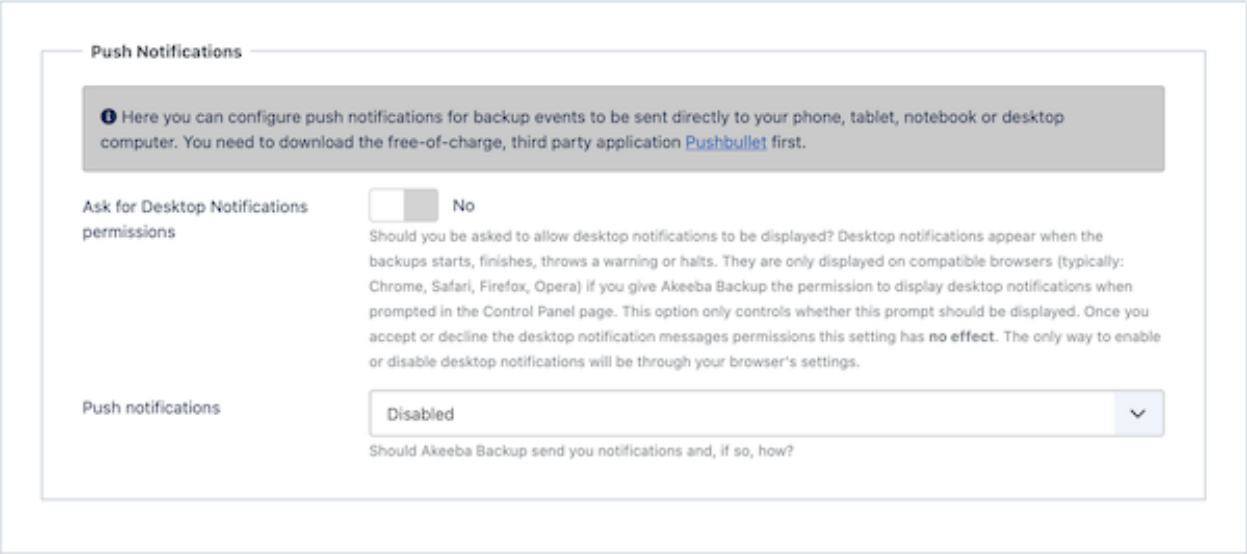
Stuck backup timeout — A backup will be considered stuck (failed) after this many seconds of inactivity. Please note that uploading backup archives to remote storage, such as Amazon S3, using the native CRON mode might take substantially longer than that. We advise you to leave this value as is and schedule 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 the backup naming variables, e.g. [HOST] and [DATE]

Email body — Leave blank to use the default. You can use all of the backup naming variables, e.g. [HOST] and [DATE].

2.2.2.3. Push notifications



Akeeba Backup 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 can only be 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.

Ask for Desktop Notifications permissions
: Enable this option to allow Akeeba Backup to display desktop notifications. Unlike push notifications, these are only shown when you are taking a backup from the backend of your site, though your browser. They are displayed by your browser, not PushBullet. This feature is only compatible with browsers implementing the desktop notifications API for JavaScript such as Firefox, Safari or Google Chrome.

Push notifications
: Select the push notifications type. You can select between PushBullet, Web Push and None. If you choose None the push notifications are disabled.

  PushBullet [https://www.pushbullet.com] is a third party, free of charge, service which sends notifications to your browser or its mobile application. As of mid-2021 they no longer have an iOS / iPadOS application.

  Web Push, or Push API [https://developer.mozilla.org/en-US/docs/Web/API/Push_API], is a web standard supported by most modern browsers. It is supported by Firefox, Chrome, Edge, Opera, Brave and many more browsers. Safari 16 supports Web Push starting with macOS Ventura and iOS 16.1. After setting this option to Web Push please save the settings and go to Components, Akeeba Backup for Joomla. You will now see a small area for managing push notifications, right below the backup quick icons.

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.

Things you need to know about Web Push (Push API)

Using Web Push notifications has the following server requirements:

- The PHP `curl` extension must be installed and enabled.
- The PHP `mbstring` extension must be installed and enabled. This is also a hard requirement for Joomla itself to work correctly.
- The PHP `openssl` extension must be installed and enabled. This is also a Joomla requirement for using WebAuthn and most Multi-factor Authentication methods among other things.
- If you are using PHP 7.2 the PHP `gmp` extension must be installed and enabled. If you are using PHP 7.3 or later you do NOT need this extension but having it won't hurt; in fact, it will make push notifications (and Joomla's WebAuthn and Multi-factor Authentication) faster.

If the requirements are not met you will most likely not be able to receive push notifications from Akeeba Backup.

Web Push is an API implemented by your browser. Push notification endpoints are managed by the company which makes your browser e.g. Mozilla for Firefox, Google for Chrome, Microsoft for Edge, Apple for Safari and so on. We (Akeeba Ltd) have no control over the API or the endpoints being used. If your host needs to whitelist specific domains to access them over HTTPS please do not ask us which are the Web Push endpoints for your browser; we don't know as we are not your browser's maker. If your host cannot figure it out either you will not be able to use Web Push notifications on your site.

When you subscribe a specific browser on a specific device to receive push notifications we use that freshly created push notification subscription to send a notification to your browser. If you do not receive that notification it means that something's wrong between your server and your browser maker's servers. We cannot help you; we are responsible neither for your server's configuration nor the network infrastructure and endpoints maintained by the browser's makers. All we can tell you is that in this case you should click the button to disable push notifications to prevent Akeeba Backup wasting time trying to send you a push notification you cannot receive.

To better protect your privacy, the Push API allows the applications using it — like Akeeba Backup — to encrypt the push notifications using asymmetric cryptography, the same kind of cryptography used by HTTPS to protect your purchases online. We do make use of this encryption, generating a key pair the first time you visit Akeeba Backup's Control Panel after setting the Push Notifications option to Web Push. These keys are stored in a hidden component key in the Options page called `vapidKeys` (VAPID is the official name of the cryptography scheme used by the Push API; yes, we understand it's a negative word, do tell that to the browser makers and the W3C who are responsible for choosing that silly name).

Note

Since the push notification endpoint URL is under the control of the browser's maker it would be possible for them to read your push notifications. Using encryption means that only your server has they keys to create encrypted notifications and only your browser (but NOT the browser's maker!) has the keys to decrypt them. Therefore all the browser's maker sees is “garbage” — encrypted text they can neither read nor modify. This ensures privacy of your push notifications.

The cryptographic keys cannot and must not be changed. If you change them, existing push notification subscriptions will stop working as they cannot be decoded. If you start receiving notifications with garbage data, console errors about being unable to decode the push data or stop receiving notifications altogether what has happened is that the VAPID keys got changed, e.g. because you restored an old backup of your site or some third party software messed with the settings of our component. In this case delete the records from the #__user_profiles table which have a `profile_key` equal to `com_akeebabackup.webPushSubscription`. Do remember that #__ needs to be replaced by the table name prefix for your site.

When you enable push notifications you subscribe *a specific browser on a specific device* to receive push notifications. The information to do that is recorded in the #__user_profiles table (where #__ is the table name prefix for your site) in the records which have a `profile_key` of `com_akeebabackup.webPushSubscription`. There is one record for each user subscribing to push notifications. When Akeeba Backup sends notifications it goes through all records. If you demote a user (move them to a user group which does not have access to Akeeba Backup) their push notification subscriptions are NOT removed. As a result, they will continue receiving notifications. The same applies if the user account is disabled and all of its other information is replaced with anonymous / fake information, like most GDPR tools do. They only way to get rid of notifications in this case is to delete the user account completely.

When Akeeba Backup needs to send push notifications it has to perform one HTTPS request for every push notification subscription of every user with push notifications enabled. These requests typically take a few milliseconds. Given enough users and devices and/or a slow server or a server blocking outbound connections this may cause a long enough delay which will result in the backup failing with a timeout. If this is the case you have no option other than NOT using push notifications with Web Push.

The notification endpoints provided by browsers typically have rate limits i.e. there's an upper limit to how many notifications can be sent and how fast they can be sent. These limits are NOT published anywhere, nor can we wait and retry sending notifications later (the backup would time out and fail). It is possible that if you have a backup that raises multiple warnings shortly before it finishes that you will not receive some notifications such as the backup start, the warnings or the backup end notification. Do NOT rely solely on push notifications to monitor whether your backup executes and/or whether it executes without warnings.

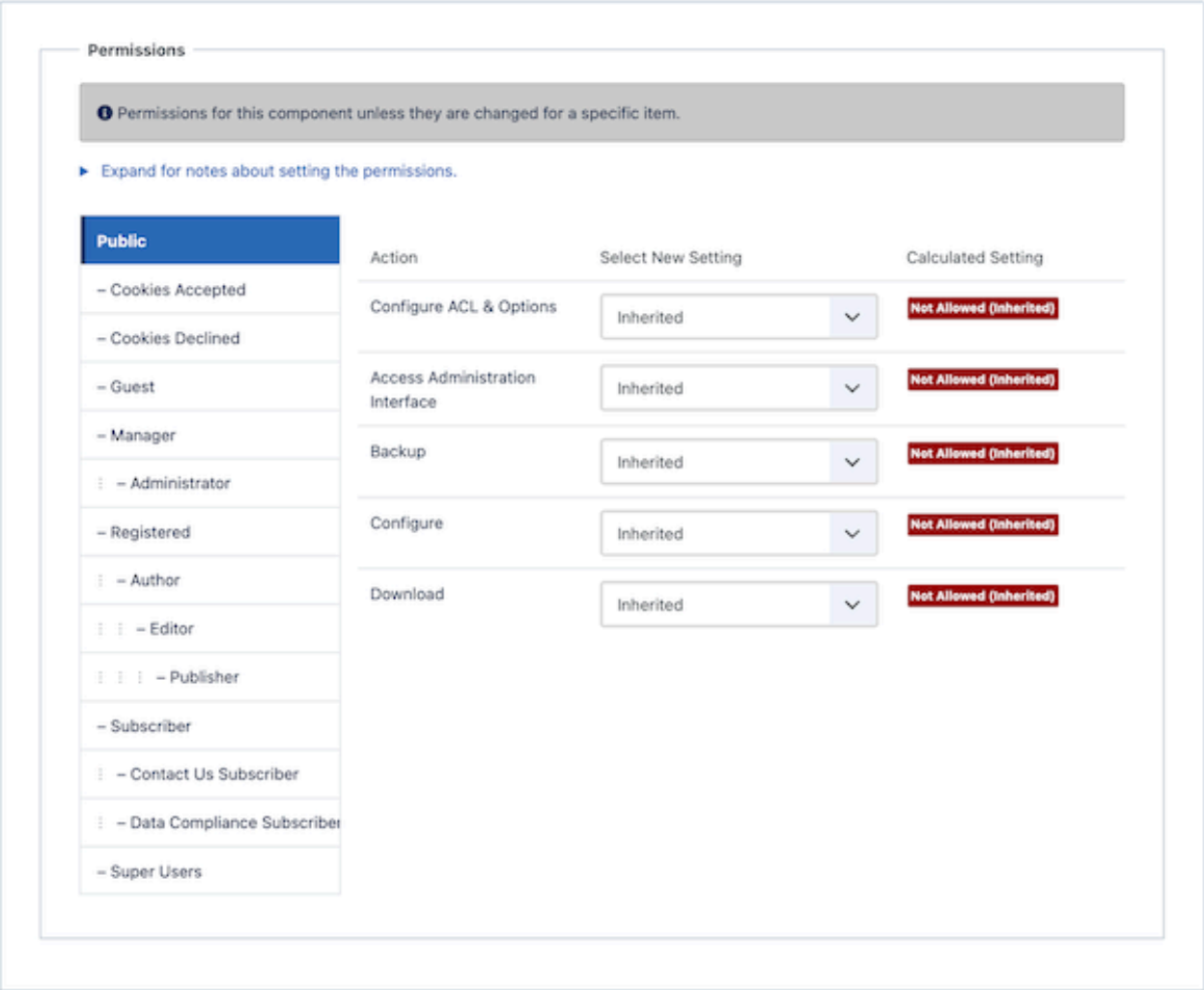
Notifications are also grouped. All notifications sent by Akeeba Backup are tagged as `com_akeebabackup` to help the browser understand they come from the same source. Your browser may elect to group notifications or even only show you only the first or the last notification received with this tag; we have no control over this behaviour. As a result it is possible that you will not receive some notifications such as the backup start, the warnings or the backup end notification. As we said above, do NOT rely solely on push notifications to monitor whether your backup executes and/or whether it executes without warnings.

Depending on your browser and Operating System it is possible that you will receive the push notifications even your browser is not running. For example, this happens with Google Chrome on Android and with most browsers on Windows (they run a background task to receive push notifications, perform background data updates and check for new versions). If this is not the case for your browser and Operating System (e.g. most browsers on macOS and iOS) you will most likely receive the push notifications next time you open your browser. However, your browser may decide that the notifications are “too old” to bother you with them in which case they might not result in a visible / audible notification OR they might not be delivered at all. We are repeating ourselves, but, please, do NOT rely solely on push notifications to monitor whether your backup executes and/or whether it executes without warnings.

Push notifications are handled by a small piece of JavaScript installed in your browser called a Service Worker. The path to Akeeba Backup's Service Worker is `media/com_akeebabackup/js/WebPushWorker.min.js` or `media/com_akeebabackup/js/WebPushWorker.js` under your site's root. Browsers allow you to inspect which Service Workers are installed and remove them. If you remove Akeeba Backup's Service Worker you will no longer receive push notifications. It is possible that your browser removes the Akeeba Backup Service Worker as part of a different operation, e.g. when clearing all caches, switching to a different user profile and so on. If you stop receiving push notifications log into your site and go to Components, Akeeba Backup to automatically reinstall the Service Worker to your browser.

Web Push notifications, just like PushBullet notifications, do NOT rely on your browser running at the time the notification is sent. Therefore you will be sent a notification even from unattended / non-interactive backup operations such as backups running over the legacy front-end backup URL, backups running over the JSON API, backups running through the Joomla CLI integration, and backups running through Scheduled Tasks. If you receive a notification when you are not using Akeeba Backup you now know where it came from and why.

2.2.2.4. 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 — Allows the users of the group to take backups.

Configure — 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.

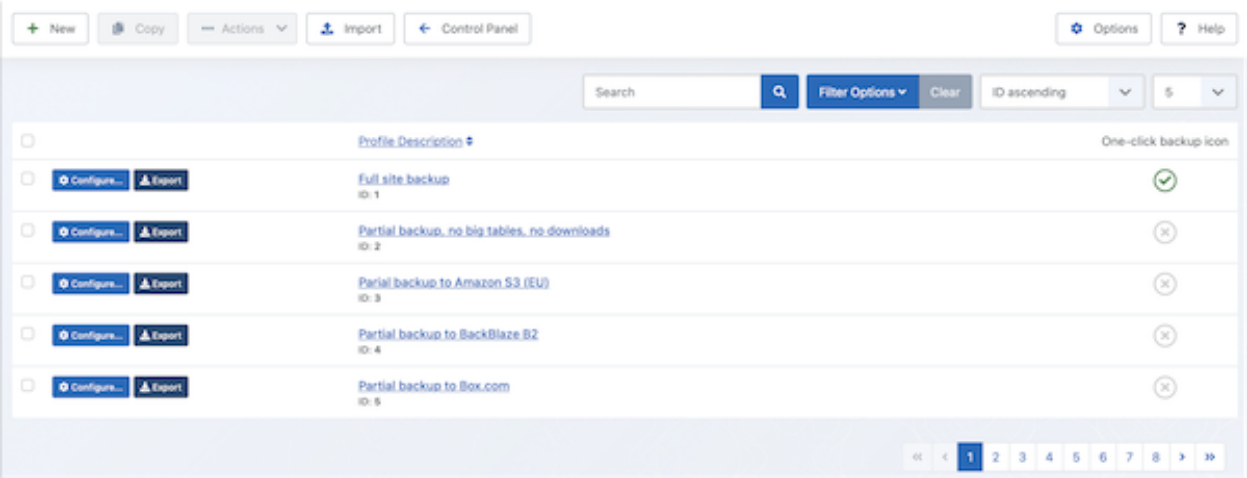
Download — Allows the users of the group to download backup archives from the Manage Backups page. It also grants them access to the Site Transfer Wizard feature.

3. Basic Operations

The Basic Operations group contains the most common functions you will need on your day to day Akeeba Backup use. In fact, you will only use the other features occasionally, 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*. Think of each backup profile as a named group of Akeeba Backup configuration settings and filters. 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.
Enable one click-backup icon	Available under the Actions menu button. Enables the one click backup feature for this backup profile. An icon will be shown in the Control Panel page of the component for this profile. Clicking on it will immediately start a new backup using this profile, without further user interaction.
Disable one click-backup icon	Available under the Actions menu button. Disables the one click backup feature for this backup profile.
Reset	Available under the Actions menu button. Resets the configuration and filter settings of the selected backup profiles to their default values. Use this if you feel like you've messed up your backup profile so bad you'd rather start over.
Delete	Available under the Actions menu button. Permanently removes the 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.

Please note that you cannot delete the default backup profile (profile ID 1). Instead, use the Reset action on it to restore it to the factory default settings and start over.

Import — Opens a modal dialog with the profile import feature. Use it to import a backup profile JSON file you had exported using the Export button described below.

We strongly advise you to review your settings after importing a profile. If the profile comes from another site you may have used an absolute path or overridden the database connection information. In this case you will have to change those settings to reflect the current site's configuration.

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:

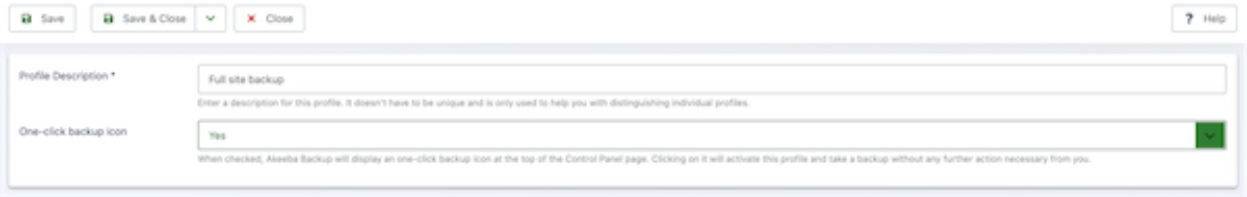
Configure... — Clicking this button makes that profile current and opens the Configuration page. This is equivalent to going back to the Control Panel, selecting that profile in the list, waiting for the page to reload and clicking on Configuration. We figured that having to click to just one button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

Export — 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.

Please note that the file you are downloading contains all of the configuration information **UNENCRYPTED**. We strongly advise you to only use this feature when connected to your site over HTTPS. We also strongly advise against storing exported profile files in media which could reasonably be lost (e.g. USB keys) or cloud services without file encryption that you manage yourself. Whenever possible, encrypt the exported backup profiles e.g. with GPG or even in a password-protected ZIP file before storing them.

Each row displays the profile description, the profile ID and the status of the one click backup icon feature. You can click on the one click backup status icon to toggle that feature on the profile.

The Edit Profile page



When you click on a backup profile you can change its very basic information: the description and the one-click backup icon preference.

The description is something you see throughout the Akeeba Backup interface. Keep it short, unique and descriptive. It will help you understand which backup profile is currently active.

The one-click backup icon refers to the feature explained when describing the toolbar buttons.

3.2. Configuration Wizard

The Configuration Wizard is an automated process which benchmarks your server's performance and tries to fine tune common configuration variables for optimal backup performance on your server. The Configuration Wizard settings

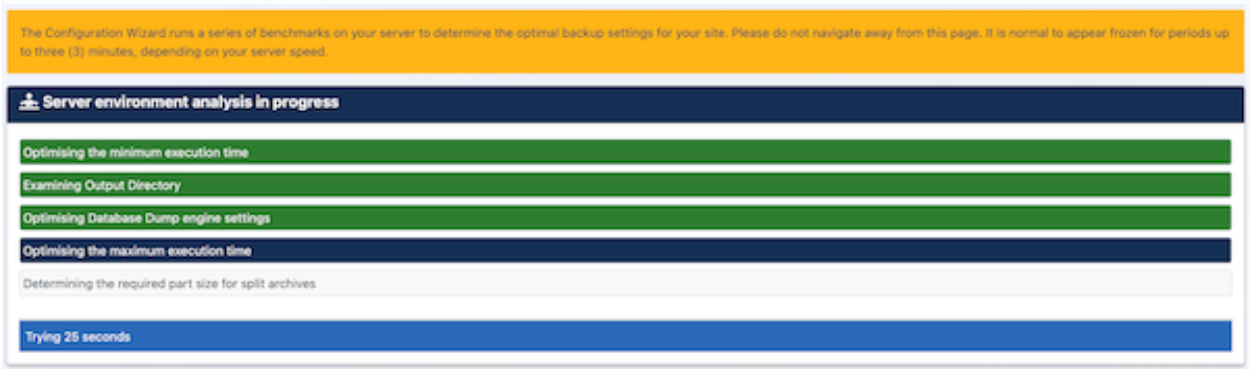
are applied to the current profile only. If you want to automatically configure a different profile, you have to select it from the drop-down list in the Control Panel page before clicking on the Configuration Wizard button.

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)".
- Post processing options are reset to "None" i.e. your backup will not be uploaded to a remote storage location.

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:

- Optimise the minimum execution time so as to make the backup as fast as possible without your server returning an error response.
- Adjust the location and/or permissions of the output directory. Useful if you just transferred your site to a new server or location.
- Optimise the database dump engine settings to make database dump as fast as possible, while avoiding memory outage errors
- Optimise 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 archive splitting to smaller parts which may be required in some cases when you are using a post-processing engine (such as FTP, Amazon S3, Dropbox, etc). If you will be 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

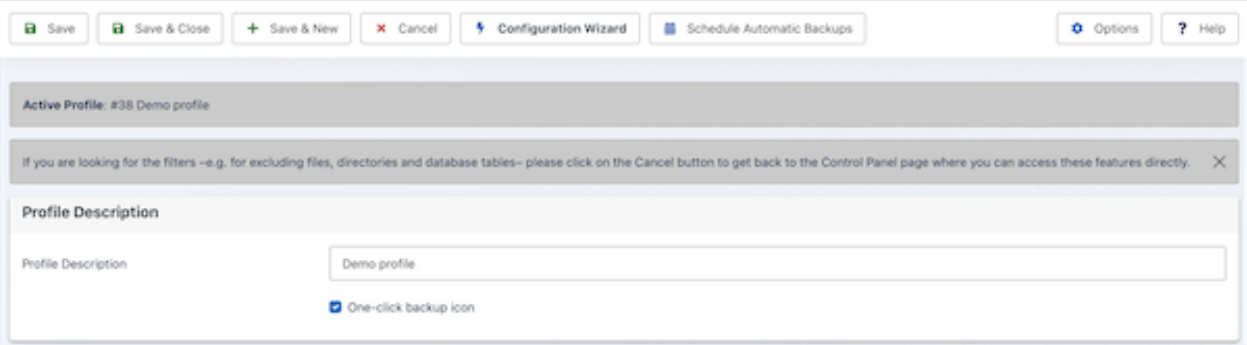
Some of the options discussed below may be only available in the Professional edition which is only available to paying subscribers.

The Configuration page controls how Akeeba Backup works at a very fine detail level. Due to the substantial number of options they are organised in different groups. Each option has a label on the left hand side. You can hover your mouse over the label to see a quick reference (tooltip) for that option. Click on the label to make the quick reference sticky. Click again to let it disappear after you move your mouse away from it.

Some of the settings feature a button at the right had side. These buttons either do some action on the setting's field, like browsing for a folder and testing connection parameters.

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 of that range, or an invalid number, they will automatically revert to the closest value which is within the preset range.

The top of the Configuration page



The top of the page is Joomla's toolbar area. The Save button will save the backup profile settings and get you back to the Configuration page. The Save & Close button will save the backup profile settings and take you to the Control Panel page. The Save & New button will save the backup profile settings, create a new profile which is an exact copy of the saved settings and return to the Configuration page of the *new* backup profile. The Cancel button aborts all unsaved changes and takes you back to the Control Panel page.

The Configuration Wizard button will launch the Configuration Wizard. This is useful if you want to initialize your backup profile or reset any settings you are not sure about.

The Schedule Automatic Backups button will take you to the page where you can find out how to automate your backups.

On the top of the main page you can see a reminder of whether you're using encryption for your backup profile settings (something that you can change in the component's Options, accessible from the Options button in the toolbar). We recommend only saving passwords in the configuration when encryption is enabled.

Note

Encryption is not a panacea. The configuration is stored in the database encrypted and the decryption key is stored in a file. This is meant to protect you from a vulnerability which allows the attacker to only access

the database. If the attacker can read or, worse, write to your site's files your settings can be reasonably considered compromised: the attacker has all the information they need to retrieve both the encrypted data and the decryption key.

Furthermore, whenever you export a backup profile the resulting file is unencrypted. This is on purpose. The decryption key is site-specific, generated whenever you do a clean installation of Akeeba Backup on your site. If the settings were exported encrypted you'd be unable to import them on a different site.

Below you can find 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.

Further down 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.

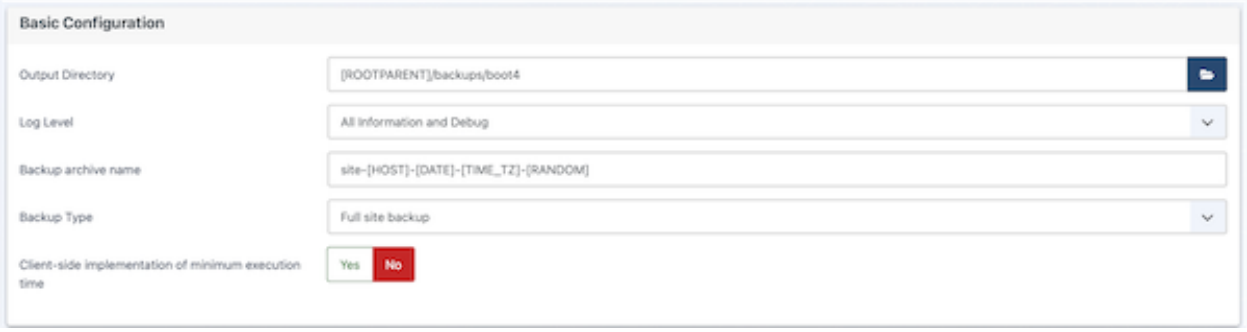
The One-click backup icon box, if checked, will result in a quick icon for this backup profile being displayed in the Control Panel page. Clicking on that icon will start a backup using that profile, without waiting for your confirmation.

Below you'll find a reference for all the options, grouped by the section they belong in.

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 one or more archive or SQL files. This is also where your *backup log file* will be stored. The output directory must be accessible and directly writable by PHP.

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.

The output directory, all of its subdirectories and all files contained therein are *automatically excluded from the backup*. Do not use a folder that contains files you want to back up as your backup output directory. Most importantly, do not use your site's root as your output directory! This will lead to a backup that does not have any of your site's files, making it useless! Akeeba Backup will attempt to warn you in this case.

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_akeebabackup/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)

You can always click on the Browse... button to open a directory picker interface. Inside that interface and 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.

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. We do not recommend using this setting.

- **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. Please take a backup using this log level before requesting support from us.

- **None**. This log level *is not recommended*. It disables logging altogether.

We recommend using Errors and Warnings after you have confirmed that your backup is running properly and All Information And Debug when you need to request support.

Backup archive name

Here you can define the name of your backup files. You must not enter an extension, it's added automatically.

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

Whenever you visit Akeeba Backup's Control Panel we store the host name in the database and try to use it when you take a backup from the command line. If this value cannot be stored or if your site's host name changed since the last time you may get an incorrect host name when taking a backup from the command line, i.e. when you are using the akeeba-backup.php script, typically from a CRON job.

[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 16-character random string.

Using this variable makes it outright implausible that an attacker can successfully guess the filename of your backup archive and access it over the web if you are using a backup output directory that's under your site's root and which is unprotected from direct web access, either because you have not put a .htacces or web.config filename or because your web server does not understand (e.g. NginX) or is not configured to honor such files.

Please note that -[RANDOM] will be appended automatically to the Backup archive name if you are using the default backup output directory and you are not already using the [RANDOM] variable in your Backup archive name. This will happen automatically at backup time. You cannot override this behavior because it is a security feature.

[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.

[TIME_TZ] The current server time, in the format HHMMSSGMTOOOO (hour as two digits, minutes as two digits and seconds as two digits followed by GMT and the the offset to the GMT timezone as four digits), for example 221520GMT +0300 for 10:15:20 pm in Nicosia, Cyprus (which is 3 hours ahead of GMT).

We strongly advise using this instead of [TIME] to remove any ambiguity on which timezone is being used. This is especially important if you rely on the filenames to understand which is the backup you are looking for or when you have multiple people taking and restoring backups in different timezones.

[TZ]	The timezone all dates and times are expressed in. This variable gives you the timezone in a manner that is safe for use in filenames, even on Windows. For example, asia_nicosia for Nicosia, Cyprus.
[GMT_OFFSET]	The timezone all dates and times are expressed in. This variable gives you the timezone as an offset to the GMT timezone. For example +0300 for Cyprus (3 hours ahead of GMT), +0530 for India (5 hours 30 minutes ahead of GMT) or -0600 for Chicago (6 hours behind GMT).
[TZ_RAW]	The timezone all dates and times are expressed in. This variable gives you the raw timezone, e.g. Asia/Nicosia for Nicosia, Cyprus. Kindly note that this results in invalid filenames on Windows.
[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. 1.2.3.
[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.

The date and time options are expressed in the timezone selected in the component's Options page under Backup Timezone. By default this is GMT. You are advised to change this to the timezone your site administrators are most familiar with.

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 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 stores the database dump in a more detailed format and 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 the database dumps of your site's database and an installer script to restore them. It's like a full backup without the files.

- **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.

- **Full site, incremental files**. This backup is a combination of Full site backup and Incremental. It works like a full site backups except for the site files. The site files are treated the same as an Incremental backup, i.e. only modified files are included. This backup type is intended for sites with frequently changing database contents and infrequently changing files. The same warnings about restoration as an Incremental backup apply.

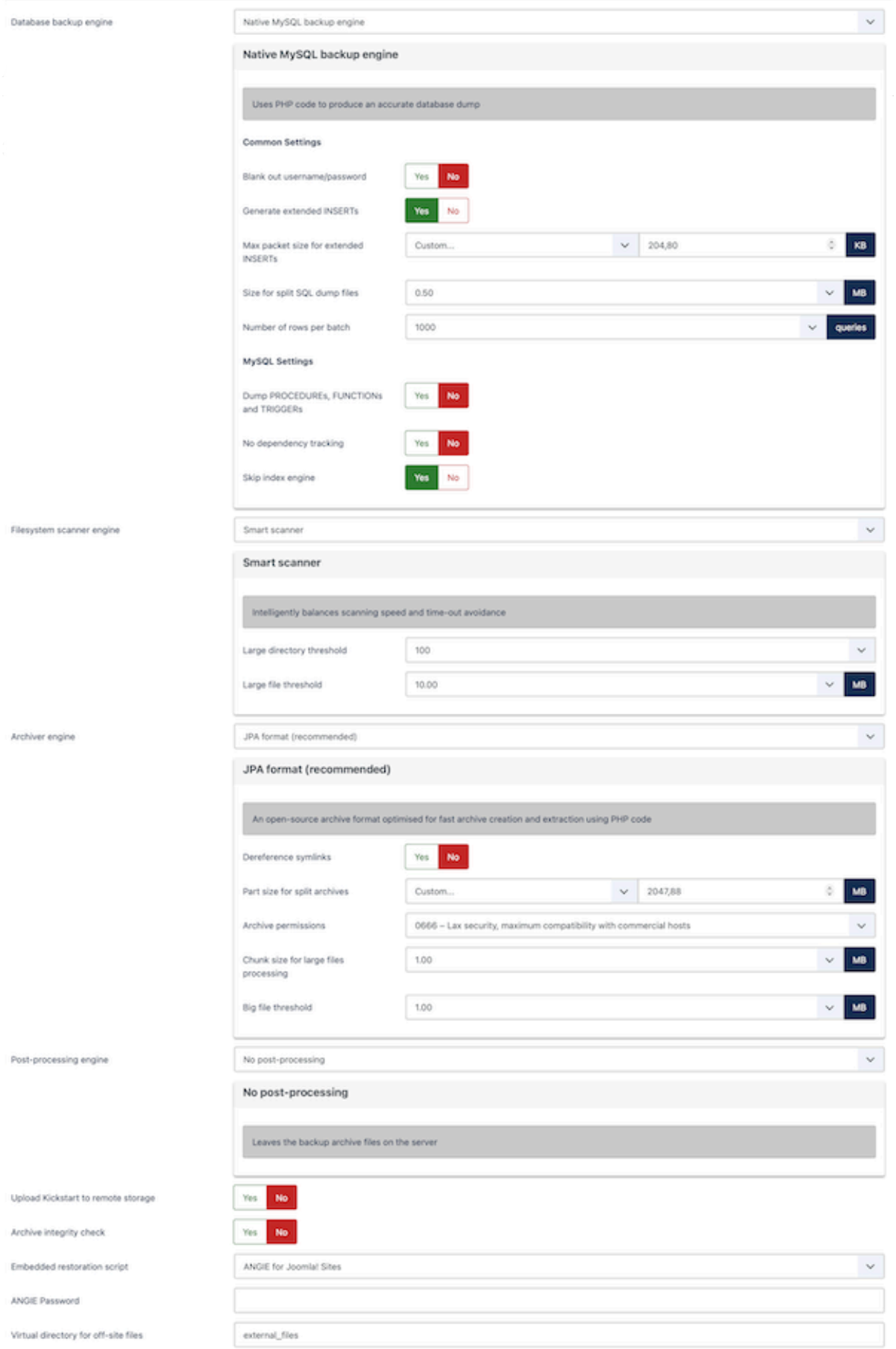
Client-side implementation of minimum execution time

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.


Database backup engine
Native MySQL backup engine
Native MySQL backup engine
Uses PHP code to produce an accurate database dump
Common Settings
Blank out username/password
Yes
No
Generate extended INSERTs
Yes
No
Max packet size for extended INSERTs
Custom...
204,80
KB
Size for split SQL dump files
0.50
MB
Number of rows per batch
1000
queries
MySQL Settings
Dump PROCEDUREs, FUNCTIONs and TRIGGERs
Yes
No
No dependency tracking
Yes
No
Skip index engine
Yes
No
Filesystem scanner engine
Smart scanner
Smart scanner
Intelligently balances scanning speed and time-out avoidance
Large directory threshold
100
Large file threshold
10.00
MB
Archiver engine
JPA format (recommended)
JPA format (recommended)
An open-source archive format optimised for fast archive creation and extraction using PHP code
Dereference symlinks
Yes
No
Part size for split archives
Custom...
2047,88
MB
Archive permissions
0666 – Lax security, maximum compatibility with commercial hosts
Chunk size for large files processing
1.00
MB
Big file threshold
1.00
MB
Post-processing engine
No post-processing
No post-processing
Leaves the backup archive files on the server
Upload Kickstart to remote storage
Yes
No
Archive integrity check
Yes
No
Embedded restoration script
ANGIE for Joomla! Sites
ANGIE Password
Virtual directory for off-site files
external_files

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.
Post-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 your browser. 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, allowing you to transfer sites without using a laptop or desktop computer and without using up a lot of bandwidth on your device.
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.
ANGIE Password	If you are using the ANGIE embedded installer script you can optionally password-protect it, preventing unauthorised access to the installer. When you run the installer you will be asked to enter this password. Please note that the password is case sensitive, i.e. ABC, abc and Abc are three different passwords.
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.

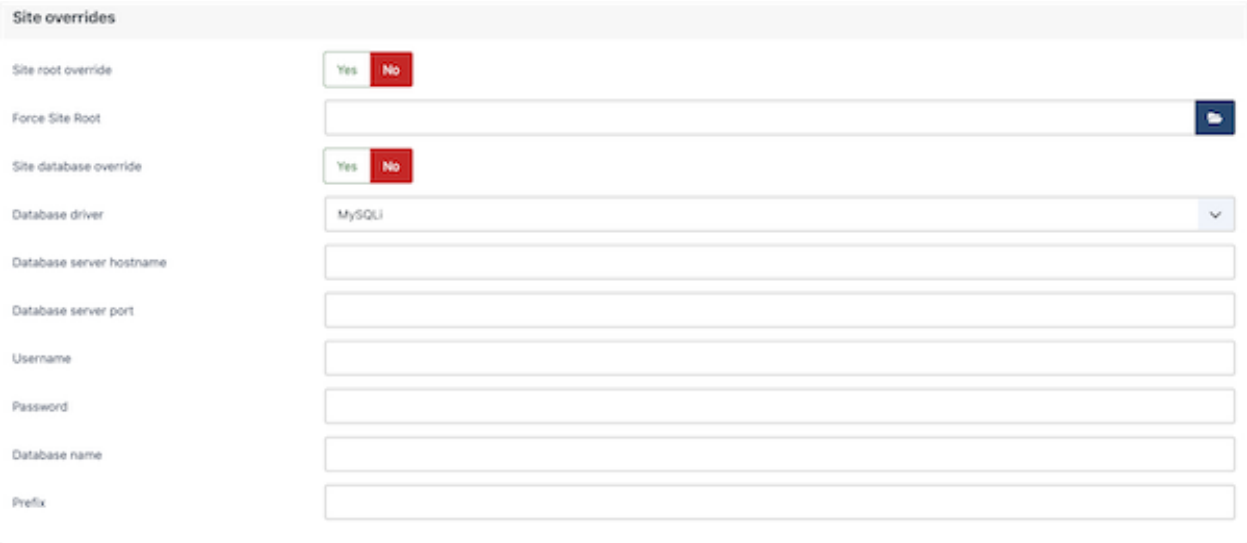
3.3.1.3. Site overrides

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.

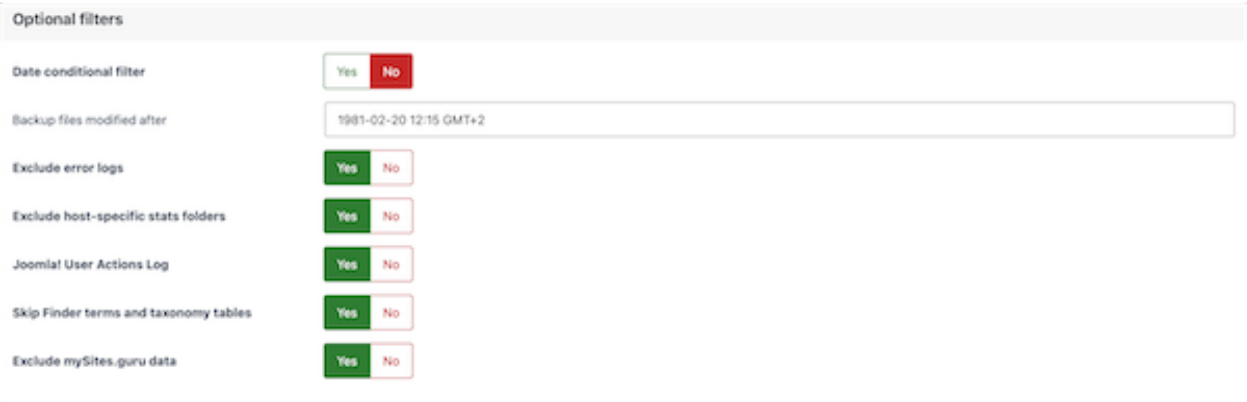
Site overrides



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 host-name	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



These optional filters allow you to exclude files or database tables without having to manually select them in the respective exclusion filter pages.

Date conditional filter

Date conditional filter

Tick the checkbox to activate this 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.

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.

Exclude error logs

Automatically exclude error log files, e.g. `error_log`, no matter where they are on the site being backed up. These files change their size while the backup is in progress which may lead to corrupt backups.

Please note that your host may be using a different naming convention for error logs, e.g. `fatal.log` instead of `error_log`. You will need to exclude these files yourself. This filter won't work on them.

Exclude host-specific stats folders	When enabled, Akeeba Backup will automatically exclude the most common host-specific folders for storing access statistics for your site. These folders are read-only by your web site user, causing restoration issues if they are backed up
Joomla! User Actions Log	Skips over the contents of the Joomla! User Actions Log. The database table holding these records can get quite big on a busy site, slowing your backup down and bloating its size. Skipping over this data does not have an adverse impact to the functionality of the site.
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.
Exclude mySites.guru data	mySites.guru (formerly MyJoomla.com) creates a number of very big database tables on your site as part of their auditing process. This information is really something they should be storing on their own servers. More importantly, these big tables — which would slow down and bloat your backup — do not need to be backed up. They will be regenerated on the restored site next time you run an audit. Even if you have stopped using this service the tables might be left behind as they do not seem to be removed when you uninstall the connector. This means that if you've used this service even once, as a free trial, you are stuck with a number of really big tables. As a result we strongly recommend enabling this filter at all times. It makes sure unnecessary, big tables are not included in the backup, therefore allows your backup archive to be smaller and get created faster.

3.3.1.5. Quota management

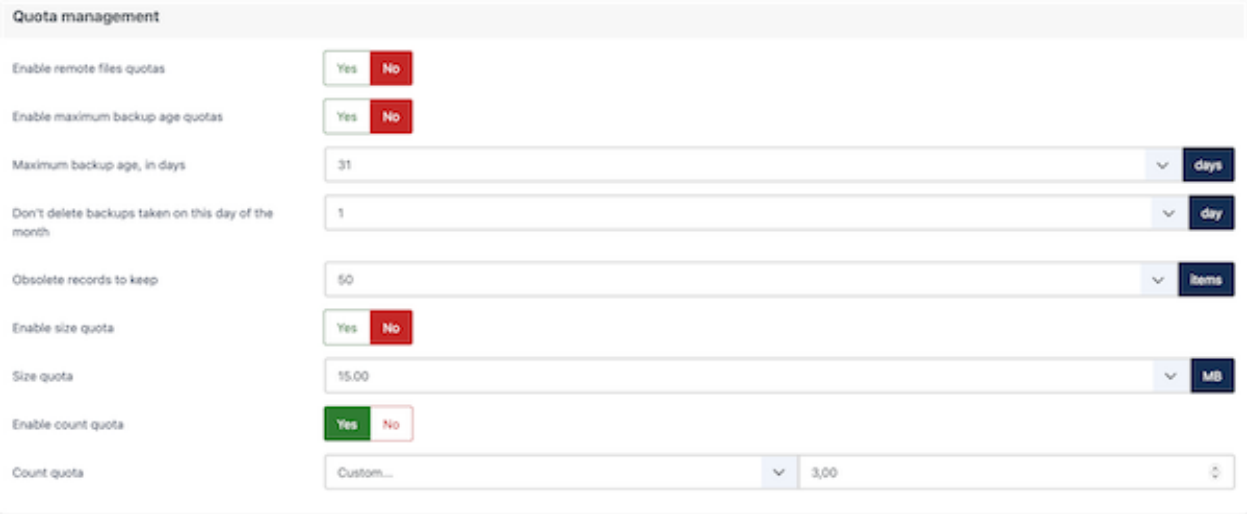
Quotas let you automatically remove backup archives and / or backup records based on specific criteria. Quotas are always calculated against the **backup records**, not the backup archives on disk on or on remote storage. In other words, if you do not see a backup record in the Manage Backups page it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

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

Finally note that the quotas are only being applied at the end of a successful backup, even if post-processing (transferring it to remote storage) failed. It is therefore recommended that you keep an eye out for failed transfers – appearing as warnings in the backup logs and the CLI backup script's output – to avoid an over-zealous quota setting from removing your last full, good backup.

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.

Please keep in mind that remote file quotas, just like local file quotas, *only apply to backup records in the database*. Akeeba Backup cannot and will not consider files present in the remote storage which do not have a corresponding backup record in the same backup profile. Furthermore, if you manually delete the files from the remote storage but leave behind the backup record in Akeeba Backup, these backups will still participate in quotas, even though their files no longer exist.

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 makes Akeeba Backup **completely ignore** 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.

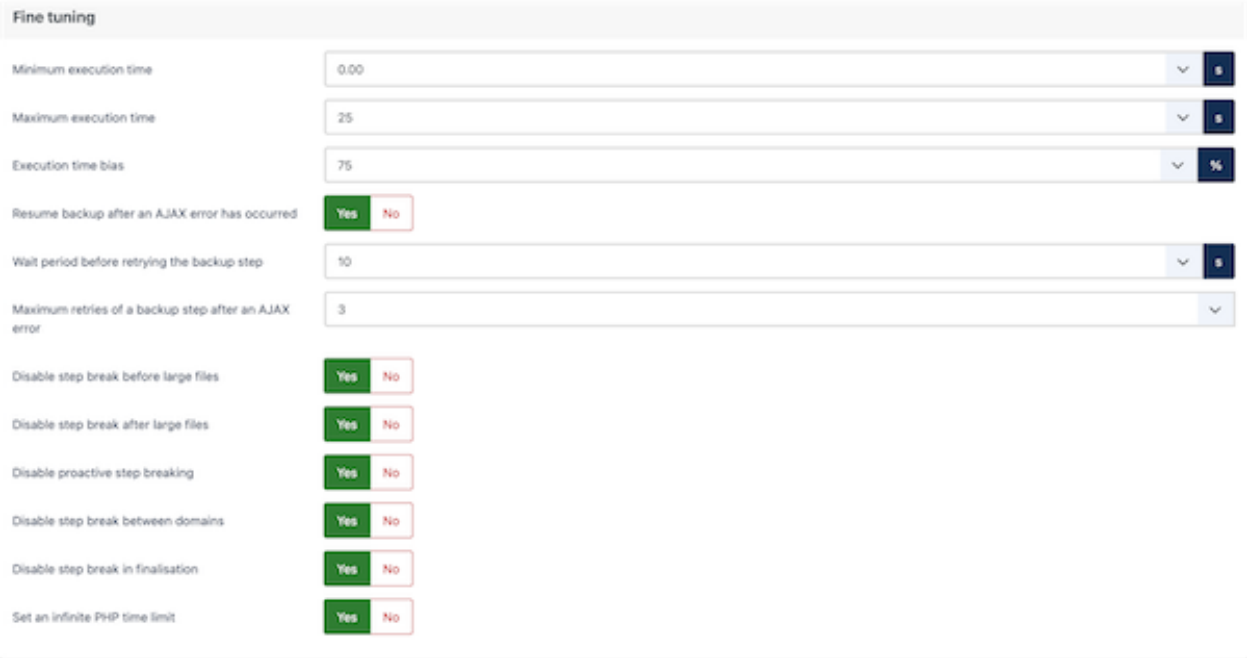
Important

A quota value of zero means "Keep all obsolete records" rather than "delete all obsolete records". To make it abundantly clear: if you set the "Obsolete records to keep" setting to 0 then Akeeba Backup will NOT remove ANY obsolete records EVER.

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

Normally, Akeeba Backup forces the current backup step to finish when it's about to move to a different backup domain, e.g. after finishing backing up the database and getting ready to backup the files of your site. This gives the backup engine the chance to do garbage collection and free up resources.

You can enable that option to make the backup 1-2 seconds faster, risking a backup failure in resource-restricted servers. We consider it a generally unsafe option and we advise against using it.

Disable step break in finalization

Normally, Akeeba Backup forces the current backup step to finish when it's about to move to a different finalization operation e.g. after finishing considering quotas and it's about to start post-processing a backup archive. This gives the backup engine the chance to do garbage collection and free up resources.

You can enable that option to make the backup 1-2 seconds faster, risking a backup failure in resource-restricted servers. We consider it a generally unsafe option and we advise against using it.

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

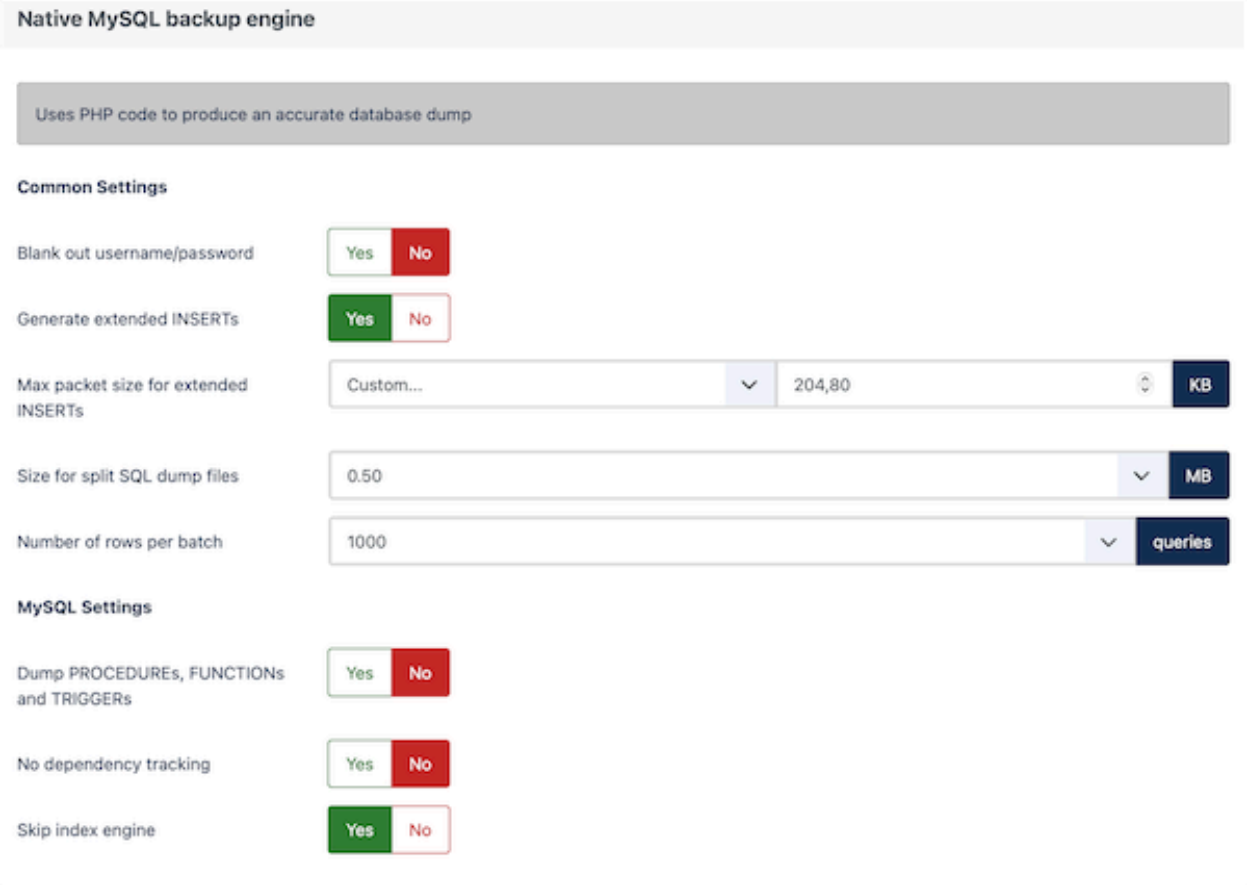
3.3.2.1. Native MySQL Backup Engine

This engine will take a backup of your MySQL database using its native features for reporting the structure of tables, views, triggers etc.

Important

Restoring views, triggers, stored procedures and functions may require elevated 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



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.json (or databases.ini, depending on your Akeeba Backup version) 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 importance on many restrictive shared hosting environments. It is suggested to turn this setting on.

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.

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, 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.

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 possibility that your backup cannot be restored on an empty database due to unsatisfied references to tables not yet created. Always test your backups if you enable this setting.

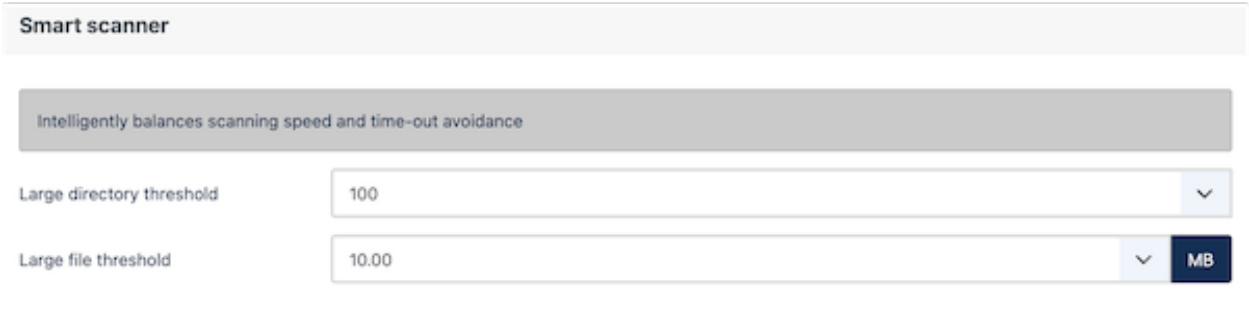
Skip index engine
: Removes USING BTREE and USING HASH from table index definitions in dump files. This is required for restoring to servers which have both indexing engines turned off (e.g. on newest XAMPP versions).

3.3.3. File and directories scanner engines

3.3.3.1. Smart scanner

The Smart Scanner will browse your file system tree for directories and files to include in the backup set, automatically creating a backup step break upon detecting a very large directory which could lead to timeout errors.

Smart Scanner



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.

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 very 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 bloggers, companies with a large downloadable reference library or very active business sites storing for example 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

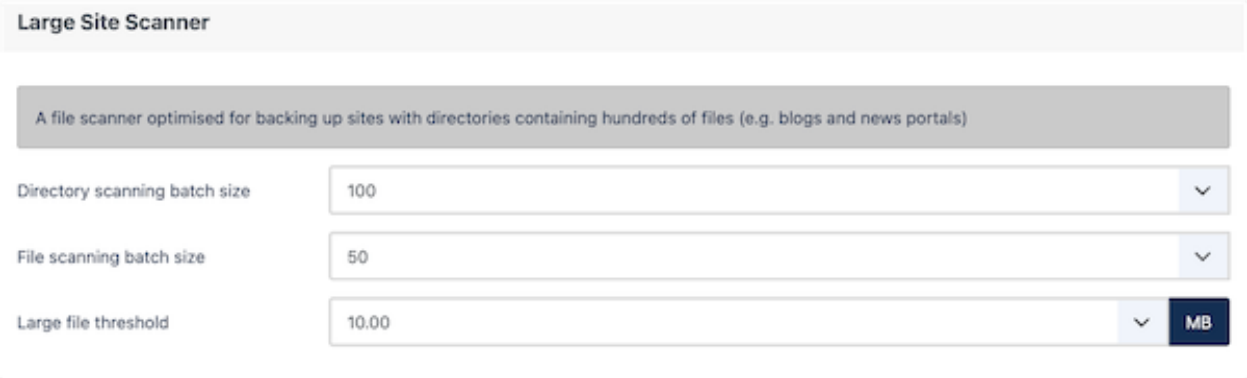
If your backup fails while trying to backup a directory with over 100 files you MUST use the "Large site scanner". It's very likely that this will solve your backup issues.

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.

In most practical situations, servers become noticeably slow in the frontend and very prone to backup errors due to server timeout or resource usage limits exceeded at about 3000 items (files or folders) inside a directory. We do not recommend storing more than 1000 items inside any directory if you value your sanity.

Large Site scanner



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).

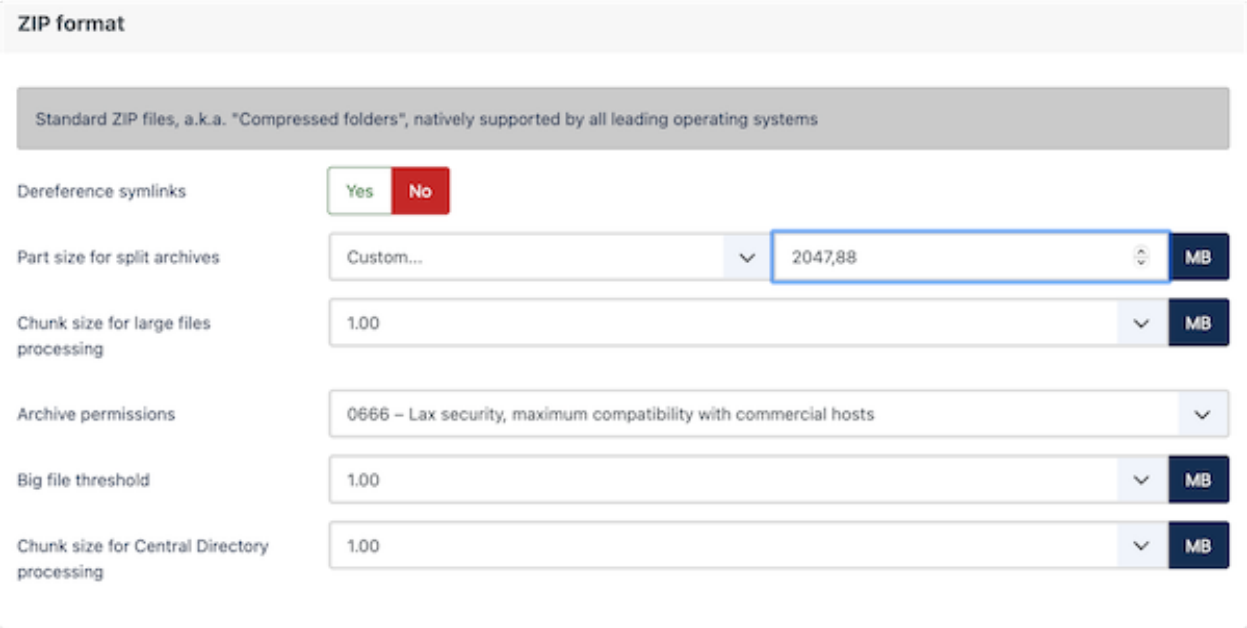
3.3.4. Archiver engines

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™, macOS™, KDE and GNOME.

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

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, macOS, FreeBSD and other compatible UNIX-family 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).

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 or Akeeba Kickstart. 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.

Archive permissions

This options lets you choose the permissions the generated backup archive files will have.

Note

This option only applies on LInux, macOS, FreeBSD, Solaris and other UNIX-based server environments. It does not apply to Windows servers.

The default option is 0666 which allows any user on the server to read and modify / delete the archive. This is *intentional*. Many servers, especially the ones set up by using prepackaged LAMP server virtual machine images on public cloud providers, are set up to run Apache and PHP as an unprivileged user which is *different* to the user the site is hosted under / you log in with. Backup archives are created by the user PHP runs under (the former) whereas FTP / SFTP runs under the user you are logging into as (the latter). This disparity means that any other permissions would make it impossible for you to download and/or delete backup archives via FTP or SFTP. Hence the default value being 0666 which allows *both* users (the one Apache/PHP runs under and the one FTP/SFTP runs under) to read and write to the backup archive files.

Another option is 0600 which only allows the user PHP runs under to have read and write access to the backup archives. This is **very strongly** recommended on properly set up servers where the effective PHP user is the same as the user of your hosting account, therefore the same as the FTP and SFTP user. Commercial hosting environments using PHP-FPM, PHP under FastCGI, chroot jails or one virtual machine per site fall under this category. If unsure try using 0600 and check whether you can download and delete the backup archive via FTP or SFTP. If you can't you may want to use one of the other two permissions options.

The 0644 option sits somewhere in between the two previous options and is meant for the misconfigured servers with different effective users. Unlike 0666, the 0644 permissions will allow you to download the backup archives via FTP / SFTP but **not** delete them. You can still delete the leftover files from your server using Akeeba Backup or your hosting control panel's file manager (if one is provided).

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.

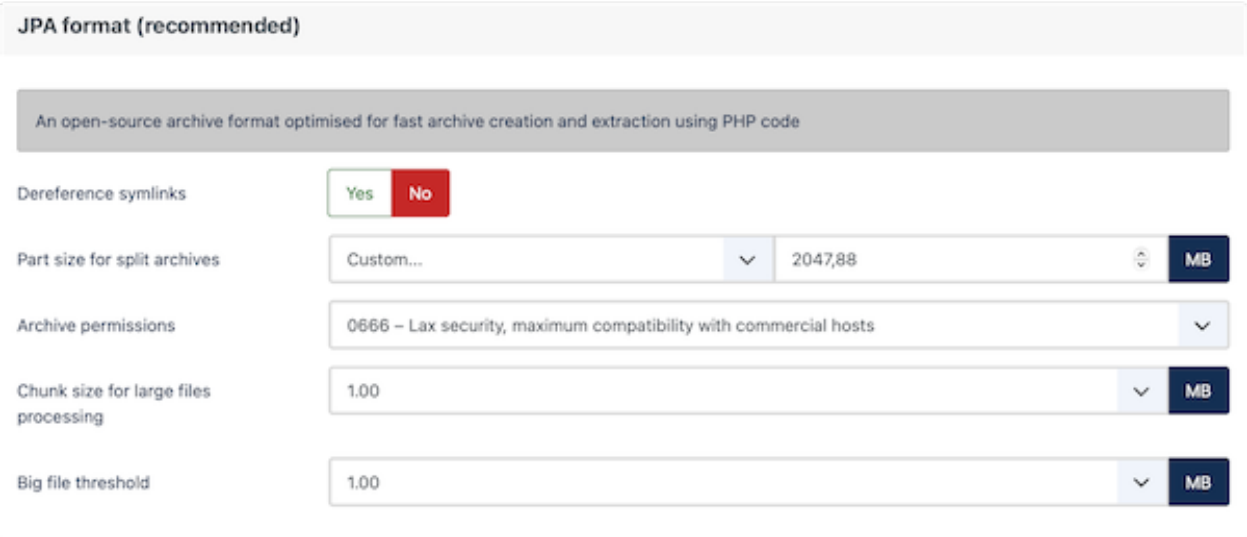
3.3.4.2. JPA format

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 using Kickstart.

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.

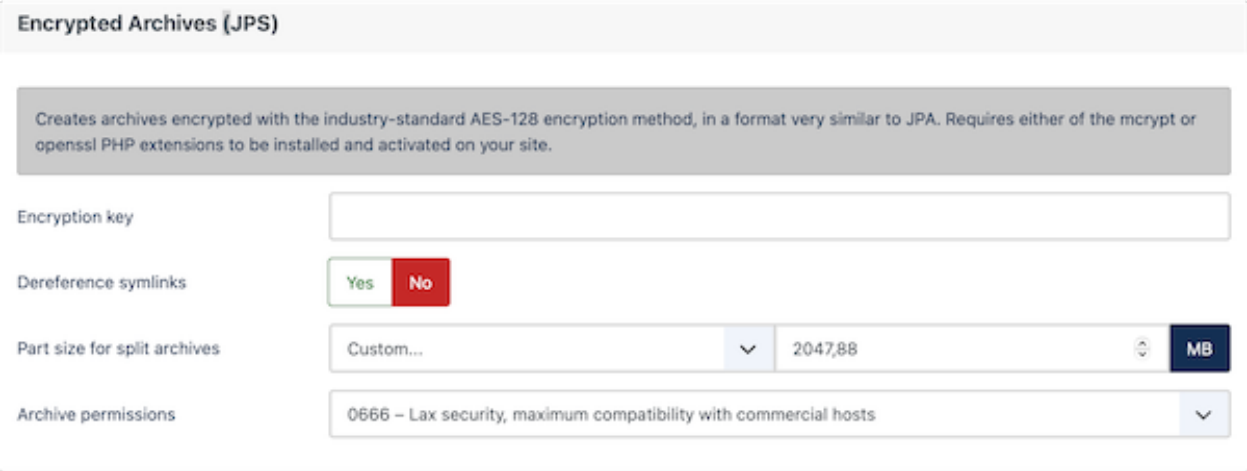
In order for JPS to work it requires that both the zlib and mcrypt or OpenSSL PHP extensions are installed and activated on your server. Please keep in mind that even if your site is using HTTPS this doesn't mean that you have the OpenSSL

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

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!

  You can omit this configuration option but you will have to enter an encryption key when taking a backup. Please note that this is only possible for backups taken with the backend backup and the CLI backup features. You need to enter the JPS password in the Configuration page for your backup to work when using any other backup method such as legacy front-end backup or Akeeba Backup JSON API.

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, macOS, FreeBSD and other compatible UNIX-family 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).

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.

Archive permissions

This options lets you choose the permissions the generated backup archive files will have.

Note

This option only applies on LInux, macOS, FreeBSD, Solaris and other UNIX-based server environments. It does not apply to Windows servers.

The default option is 0666 which allows any user on the server to read and modify / delete the archive. This is *intentional*. Many servers, especially the ones set up by using prepackaged LAMP server virtual machine images on public cloud providers, are set up to run Apache and PHP as an unprivileged user which is *different* to the user the site is hosted under / you log in with. Backup archives are created by the user PHP runs under (the former) whereas FTP / SFTP runs under the user you are logging into as (the latter). This disparity means that any other permissions would make it impossible for you to download and/or delete backup archives via FTP or SFTP. Hence the default value being 0666 which allows *both* users (the one Apache/PHP runs under and the one FTP/SFTP runs under) to read and write to the backup archive files.

Another option is 0600 which only allows the user PHP runs under to have read and write access to the backup archives. This is **very strongly** recommended on properly set up servers where the effective PHP user is the same as the user of your hosting account, therefore the same as the FTP and SFTP user. Commercial hosting environments using PHP-FPM, PHP under FastCGI, chroot jails or one virtual machine per site fall under this category. If unsure try using 0600 and check whether you can download and delete the backup archive via FTP or SFTP. If you can't you may want to use one of the other two permissions options.

The 0644 option sits somewhere in between the two previous options and is meant for the misconfigured servers with different effective users. Unlike 0666, the 0644 permissions will allow you to download the backup archives via FTP / SFTP but **not** delete them. You can still delete the leftover files from your server using Akeeba Backup or your hosting control panel's file manager (if one is provided).

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.

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.

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.

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.

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.

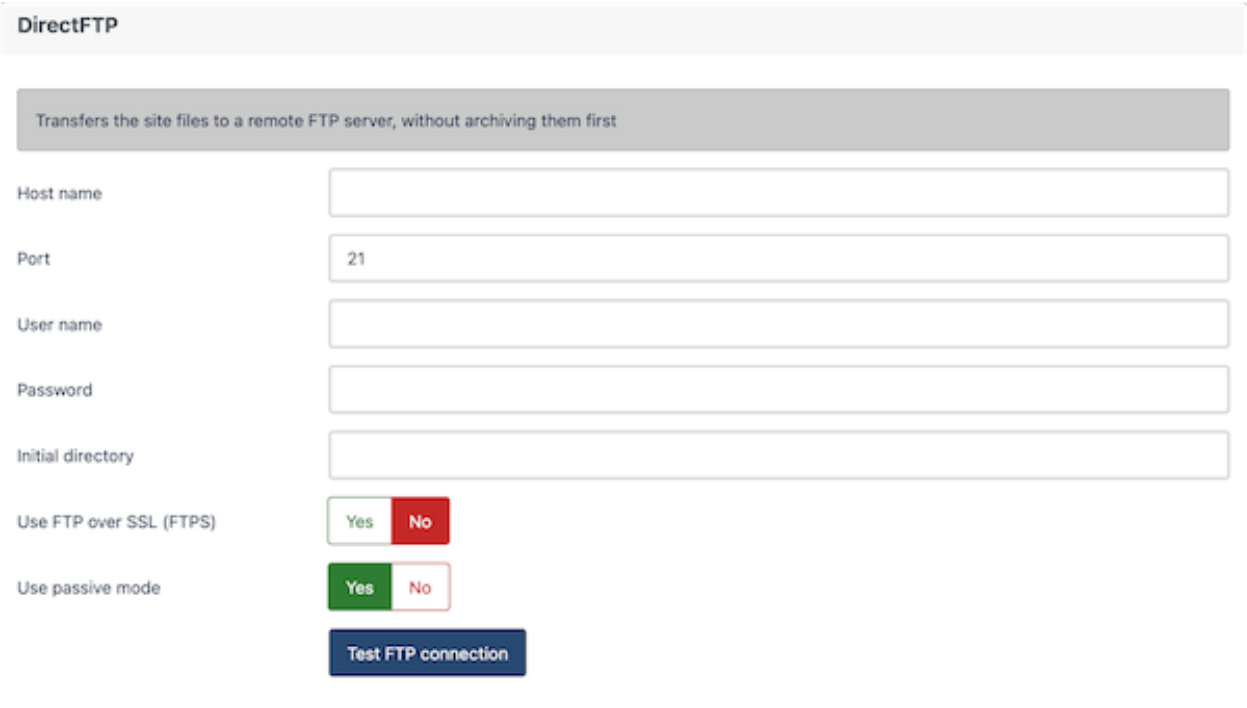
Moreover, bear in mind that the target server *must not* contain any files! If it does, it may not be possible to overwrite them, leading to an incomplete site transfer.

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.

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.

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.

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.

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.

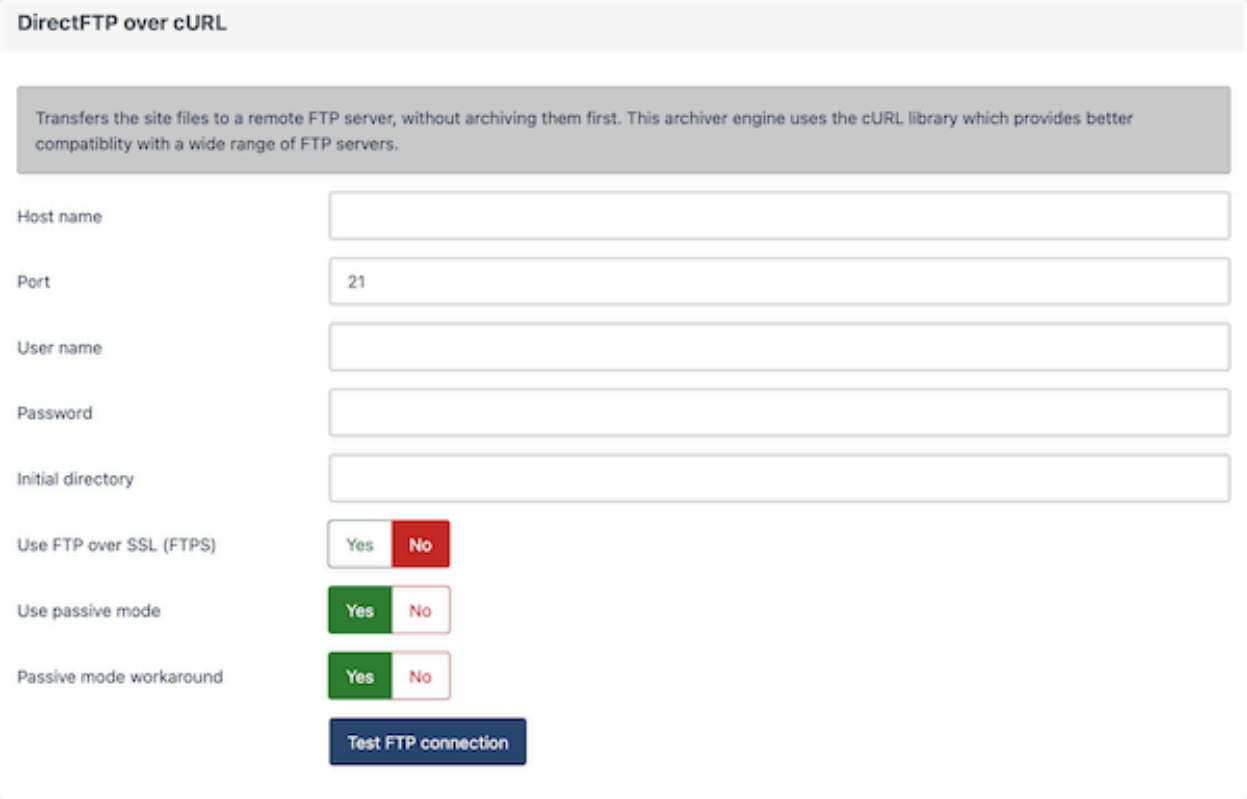
Moreover, bear in mind that the target server *must not* contain any files! If it does, it may not be possible to overwrite them, leading to an incomplete site transfer.

Finally note that due to the nature of the cURL library over 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.

DirectFTP over cURL



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.

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.

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.

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.

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.

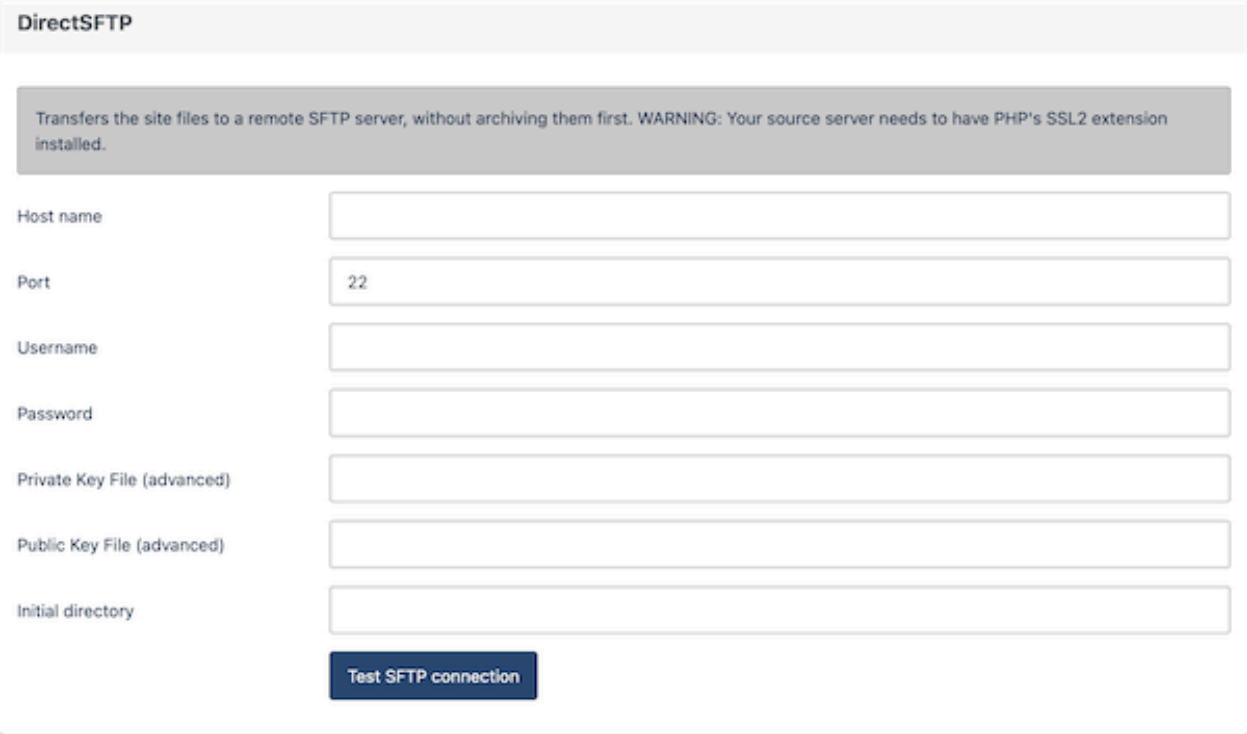
Moreover, bear in mind that the target server *must not* contain any files! If it does, it may not be possible to overwrite them, leading to an incomplete site transfer.

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.

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.

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.

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.

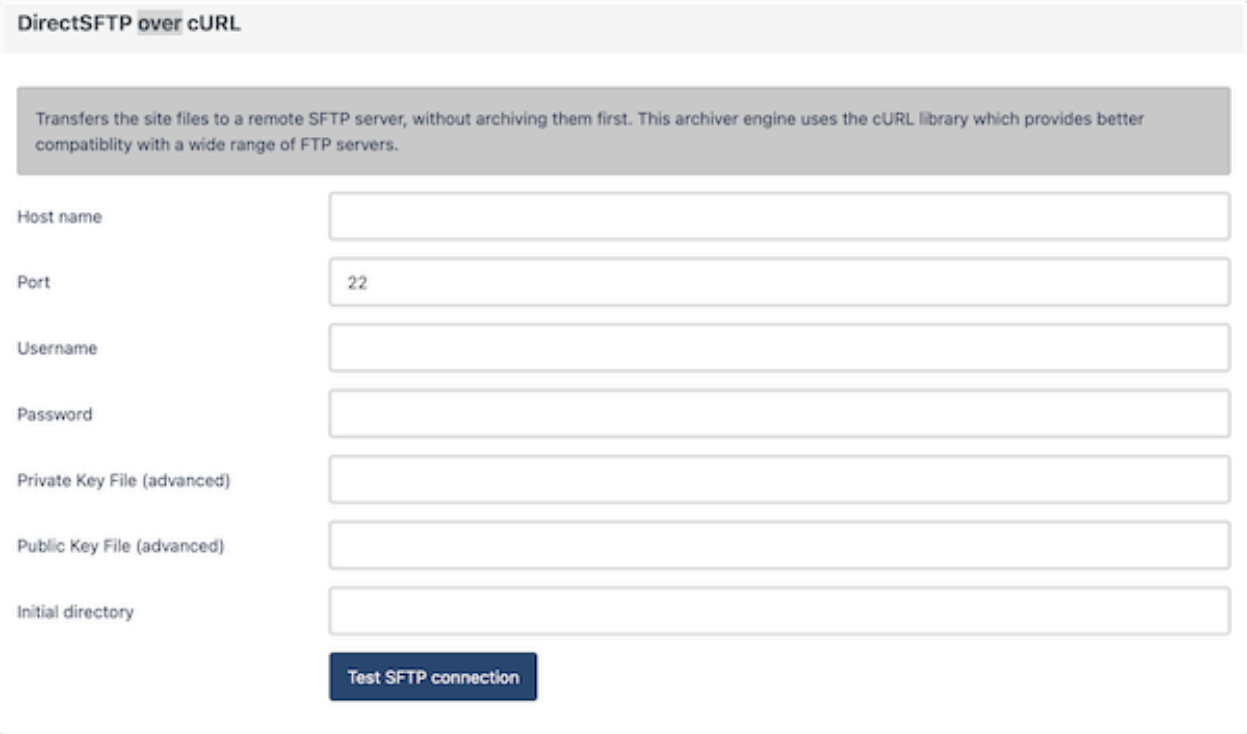
This is considered an advanced feature. Since there are many things which might go wrong in the process and due to the fact that the success of the operation depends on the server configuration of both the originating and target servers, you are advised not to use it unless you know what you're doing.

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.

DirectSFTP over cURL



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

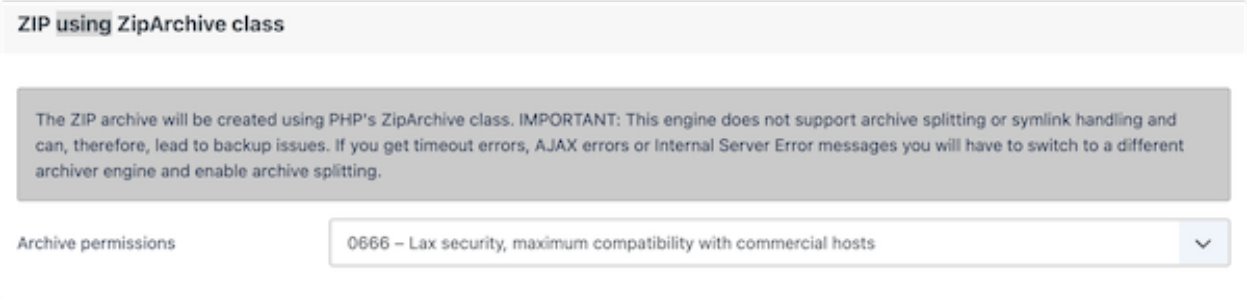
provide a public key file, using the information of the private key file to derive this information. If in doubt, always provide both private and public key files to perform certificate authentication.

- **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. We have not found any practical use case where this is preferable to Akeeba Backup's own ZIP archiver. This is why we wrote the ZIP archiver in the first place.

ZIPArchiver



PHP's ZipArchiver has several limitations which may make it impractical for most sites:

1. It does not support archive splitting. This may make it impossible to use on servers which impose a maximum file size limit or when you want to transfer your backup archives to remote / cloud storage.

2. Files are added to the archive asynchronously. The actual file compression into the archive takes place when we try to close the archive at the end of each backup step. This means that we do not know in advance how much time we will need to back up each file added to the archive. As a result you will end up with PHP timeout error unless you use a very low maximum execution time (between 1 and 3 seconds).

3. The entire list of content of the backup archive is loaded in memory. When backing up large sites this may cause memory outage issues on servers which are memory constrained.

4. It only supports whichever compression method is the default in the ZIP library used by the PHP ZipArchiver extension. This is determined when your host or your host's Linux distribution compiled and packaged the PHP ZipArchiver extension. In most cases it's Deflate which is compatible with Kickstart and third party archive utilities. In some rare cases it may be no compression (leading to massive backup archives) or a compression method which is not supported by Kickstart and / or third party archive utilities.

We only wrote this feature to prove that it's not a very good fit for taking site backups, despite what a stark minority of extremely vocal users claimed. Having both this method and our own ZIP archiver it's trivial to demonstrate the practical limitations of ZipArchiver for managing very large archive files and prove that there is indeed a reason of existence for our own ZIP archiver implementation.

In short, its only reason of existence is to prove it shouldn't exist in the first place. So, please, do not use it. If you do, your backups will fail at worst, you will be wasting server resources (especially lots of memory) at best. You have been warned.

3.3.5. Data processing engines

3.3.5.1. No post-processing

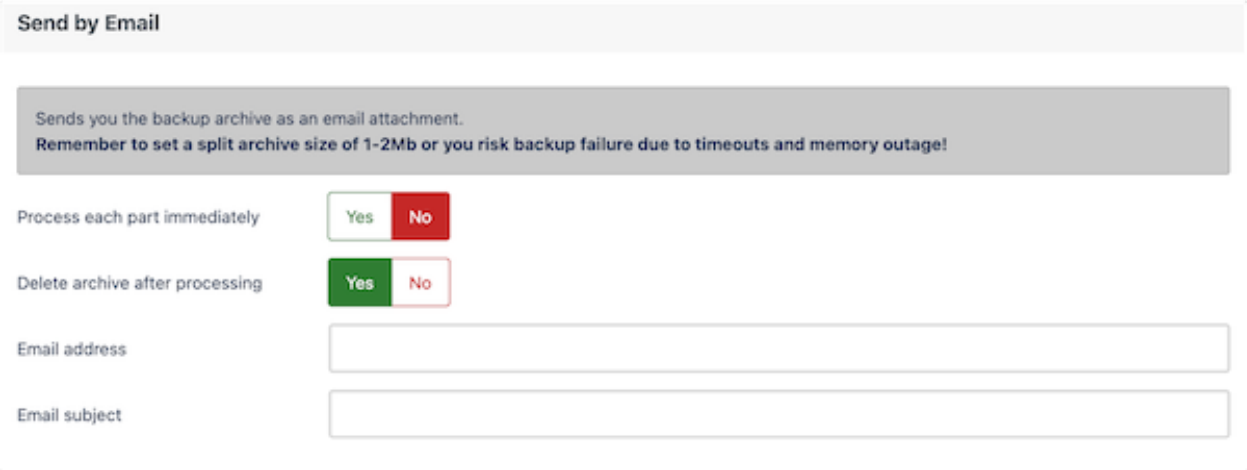
This is the default setting and the only one one available to Akeeba Backup Core. It does no post-processing. It simply leaves the backup archives on your server.

Send by email

3.3.5.2. Send by email

> **Note**
>
> This feature is available only to Akeeba Backup Professional.

Send by email



Akeeba Backup will send you the backup archive parts as file attachments to your email address. You need to be aware of the restrictions:

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.

As a result, this is only suitable for really small sites.

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.

3.3.5.3. 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.

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 limitation on older versions of PHP which can cause the backup or extraction to fail if the backup size is equal to or greater than 2GB.

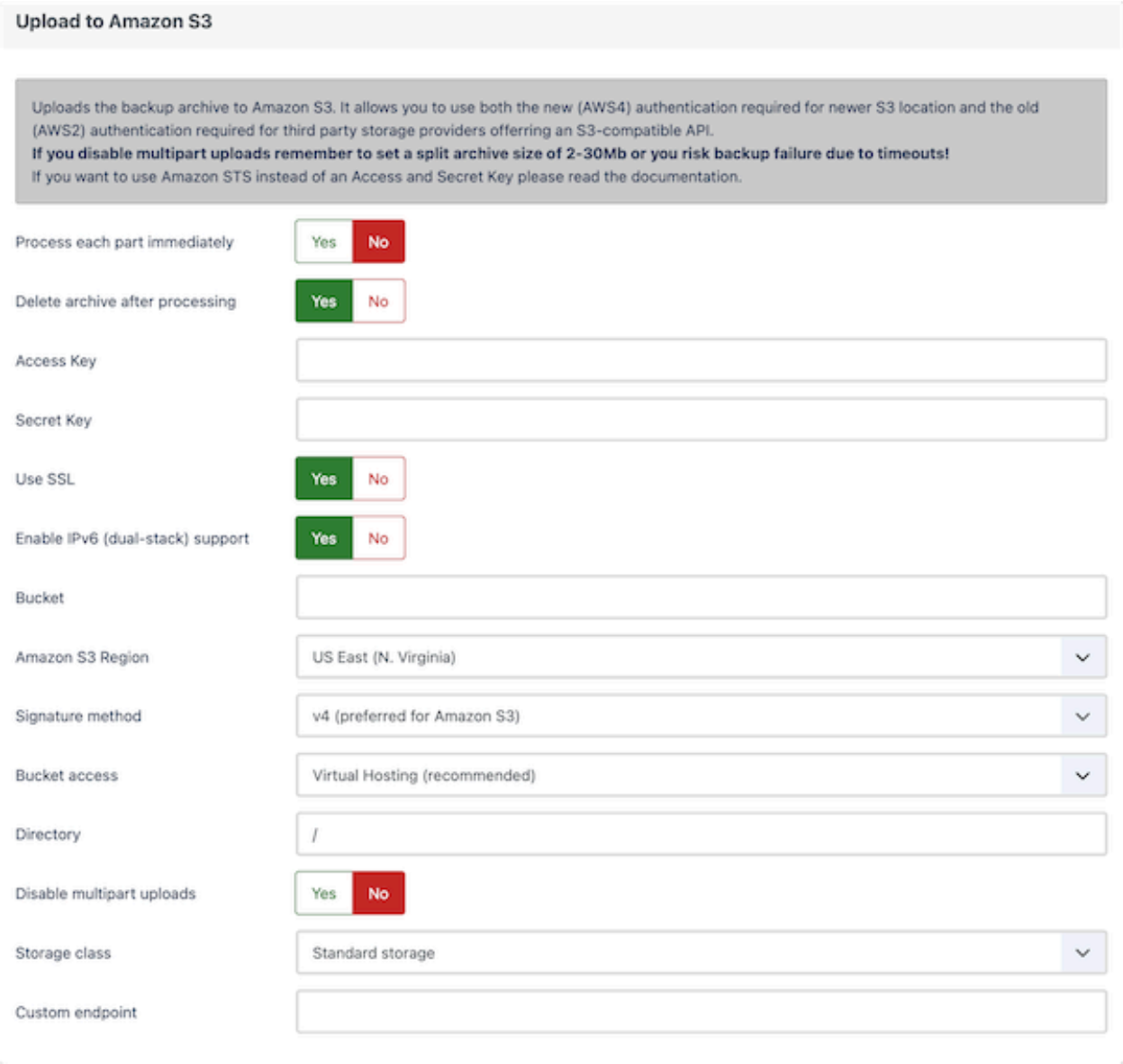
You can also specify a custom endpoint URL. This allows you to use this feature with third party cloud storage services offering an API compatible with Amazon S3 such as Cloudian, Riak CS, Ceph, Connectria, HostEurope, Dunkel, S3For.me, Nimbus, Walrus, GreenQloud, Scality Ring, CloudStack and so on. 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 also supports the Amazon S3 regions in China, e.g. the Beijing Amazon S3 region. Buckets in these regions are only accessible from inside China and have a few caveats:

- You can only access buckets in the Chinese regions 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. Required unless you run Akeeba Backup inside an EC2 instance with an attached IAM Role. Please read about this below.

Secret Key
: Your Amazon S3 Secret Key. Required unless you run Akeeba Backup inside an EC2 instance with an attached IAM Role. Please read about this below.

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.

Enable IPv6 (dual-stack) support
: When this option is disabled Akeeba Backup will use the old Amazon S3 endpoint domain names which only support IPv4. When this option is enabled Akeeba Backup will use the new Amazon S3 endpoint domain names which support both IPv4 and IPv6.

It is strongly recommended to leave this option enabled if you are using Amazon S3 proper (instead of a third party service which is S3 compatible). If your server supports IPv6 it will result in slightly faster uploads to Amazon S3. Even if your server doesn't support IPv6 yet this is a good forwards-compatible option, i.e. when your server is upgraded to support IPv6 your connection to Amazon S3 will be upgraded as well.

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.

  ## Important

  v4 signatures are only compatible with Amazon S3 proper. If you are using a custom Endpoint this option will NOT work.

- **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.

Bucket Access

This option determines how the API will access the Bucket. If unsure, use the `Virtual Hosting` setting.

The two available settings are:

- **Virtual Hosting (recommended)**. This is the recommended and supported method for Amazon S3. Buckets created after May 2019 will only support this method. Amazon has communicated that this method is the only available in Amazon S3's API starting September 2020.

- **Path Access (legacy)**. This is the older, no longer supported method. You should only need to use it with a custom endpoint and ONLY if your storage provider has told you that you need to enable it.

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 backup naming 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.

Storage class

Select the storage class for your data. Standard is the regular storage for business critical data. Please consult the Amazon S3 documentation for the description of each storage class.

Note

Glacier and Deep Archive storage classes are much cheaper but have long delays (several seconds to several hours) in retrieving or deleting your files. Using these storage classes is **not compatible** with the Enable Remote Quotas configuration option and the Manage Remotely Stored Files feature in the Manage Backups page. This is a limitation of Amazon S3, not Akeeba Backup.

We strongly recommend not using these storage classes directly in Akeeba Backup. Instead, use one or more Lifecycle Policies in your Amazon S3 bucket. These can be configured in your Amazon S3 control panel and tell Amazon when to migrate your files between different storage classes. For example, you could use Intelligent Tiering in Akeeba Backup together with the Maximum Backup Age quotas and Remote Quotas to only keep the last 45 days of backup archives and the backups taken on the 1st of each month. You could then also add two lifecycle policies to migrate backup archives older than 60 days to Glacier and archives older than 180 days to Deep Archive. This way you would have enough backups to roll back your site in case of an emergency but also historical backups for safekeeping or legal / regulatory reasons. Feel free to adjust the time limits to best suit your business use case!

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:

- 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.

Automatic provisioning of Access and Secret Key on EC2 instances with an attached IAM Role

Starting with version 6.2.0, Akeeba Backup can automatically provision temporary credentials (Access and Secret Key) if you leave these fields blank. This feature is meant for advanced users who automatically deploy multiple sites to Amazon EC2. This feature has four requirements:

- Using Amazon S3, not a custom endpoint. Only Amazon S3 proper works with the temporary credentials issued by the EC2 instance.
- Using the v4 signature method. The old signature method (v2) does not work with temporary credentials issued by the EC2 instance. This is because Amazon requires that the requests authenticated with these credentials to also include the Security Token returned by the EC2 instance, something which is only possible with the v4 signature method.
- Running Akeeba Backup on a site which is hosted on an Amazon EC2 instance. It should be self understood that you can't use temporary credentials issued by the EC2 instance unless you use one. Therefore, don't expect this feature to work with regular hosting; it requires that your site runs on an Amazon EC2 server.
- Attaching an IAM Role to the Amazon EC2 instance. The IAM Role must allow access to the S3 bucket you have specified in Akeeba Backup's configuration.

When Akeeba Backup detects that both the Access and Secret Key fields are left blank (empty) it will try to query the EC2 instance's metadata server for an attached IAM Role. If a Role is attached it will make a second query to the EC2 instance's metadata server to retrieve its temporary credentials. It will then proceed to use them for accessing S3.

The temporary credentials are cached by Akeeba Backup for the duration of the backup process. If they are about to expire or expire during the backup process new credentials will be fetched from the EC2 instance's metadata server using the same process.

Creating and attaching IAM Roles to EC2 instances is beyond the scope of our documentation and our support services. Please refer to Amazon's documentation.

3.3.5.4. Upload to BackBlaze B2

This provides integration with the low cost, high resiliency BackBlaze B2 storage service.

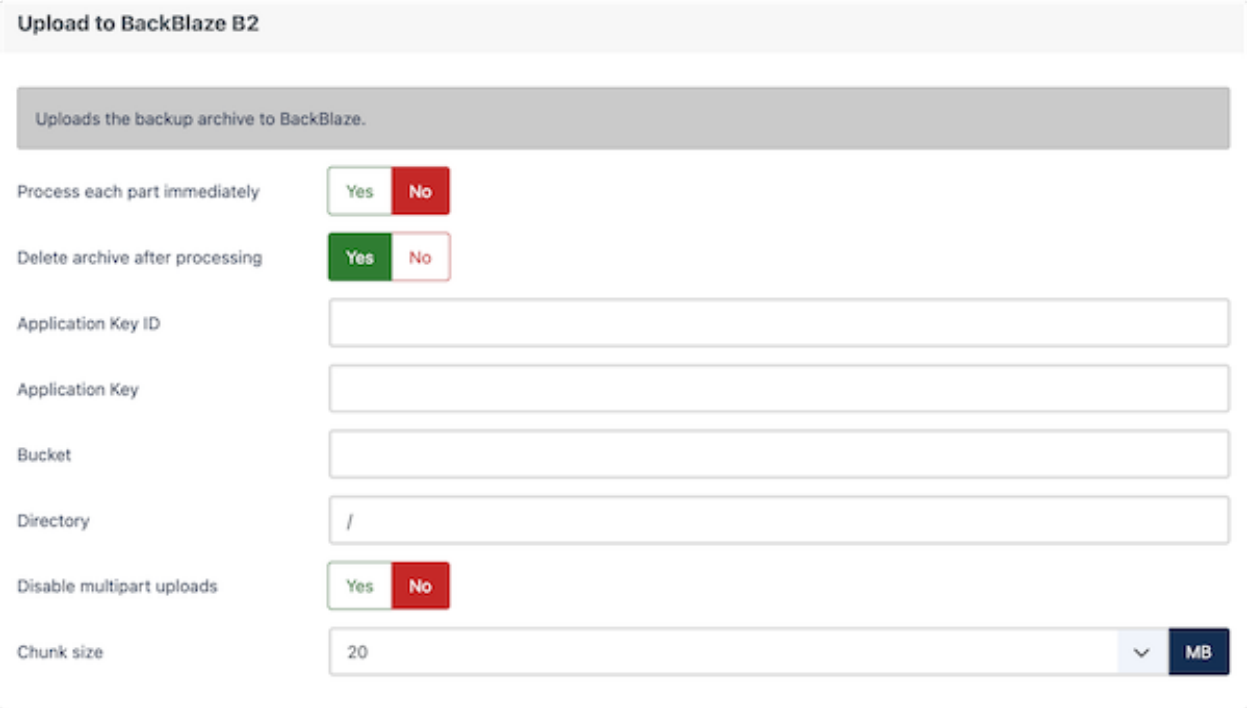
Before you configure Akeeba Backup you need to obtain an application key from the BackBlaze B2 service. Start by logging into your account [https://secure.backblaze.com/b2_buckets.htm]. From the side bar select App Keys. Click the Add a New Application Key button. Remember to select the bucket where your backups will be saved to in Allow access to Bucket(s) and set the Type of Access to `Read and Write`. Click on Create New Key.

You will be presented with a message showing your *`keyID`*, *`keyName`*, *`S3 Endpoint`* and *`applicationKey`*. Write this information down. You will not see it again!

Note

The BackBlaze B2 feature does NOT use the Amazon S3 endpoint. It uses the official BackBlaze B2 API. If you find that the API is slow or unreliable on your host you can use your B2 storage bucket with Akeeba Backup's Upload to Amazon S3 feature. The custom endpoint you need to use is the S3 Endpoint given to you by BackBlaze. Use the keyID as your Access Key and the applicationKey as your Secret Key. The Bucket Name you need is the same one you are using on BackBlaze. Do remember to set the signature method to v2 instead of v4.

Upload to BackBlaze B2



The parameters 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.
Application Key ID	The `keyID` you got when creating the application key per the instructions above
Application Key	The `applicationKey` you got when creating the application key per the instructions above
Bucket	The name of your B2 bucket. Case matters! The bucket names foo, FOO and Foo refer to three *different* buckets.
Directory	A directory (technically: name prefix) of the backup archives in your bucket,
Disable multipart uploads	By default, Akeeba Backup uploads each backup archive to B2 in small "chunks". B2 will then stitch together these chunks to create one, big backup archive. Each chunk transfer takes a small amount of time, preventing timeouts. The downside is that some hosts have weird outbound proxy setups in front of their servers, getting in the way of multipart uploads. In these cases you will see near instant chunk transfers but nothing being uploaded to BackBlaze B2. If this happens you will need to disable multipart uploads by selecting this box and use a fairly small part size for split archives to prevent timeouts while the upload is in progress.

Chunk size The size of each chunk during a multipart upload as explained above.

3.3.5.5. Upload to Box.com

> **Note**
>
> This feature is available only to Akeeba Backup Professional. Using it requires entering your Download ID in the software and having an active subscription on our site which gives you access to one of our backup software.

This uses the official Box.com API to upload archives to this storage service.

Due to the absence of a multipart upload feature in Box' API we strongly recommend using a small Part Size for Archive Splitting, typically in the 10-50MB range, to prevent a timeout of your backups while they are uploading to Box.

Important security and privacy information

The Box.com 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 Box'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 Box. When you are linking Akeeba Backup to Box you are going through the script on our site. Moreover, whenever the request token (a time-limited key given by Box 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 Box account.

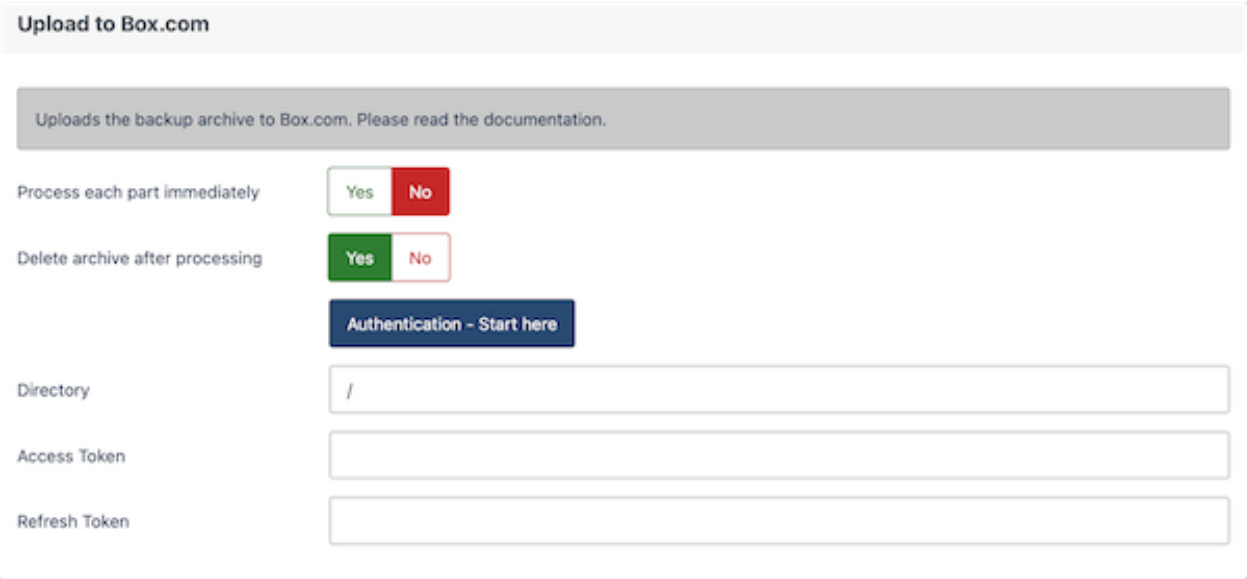
WE DO NOT STORE THE ACCESS CREDENTIALS TO YOUR BOX ACCOUNT. WE DO NOT HAVE ACCESS TO YOUR BOX ACCOUNT. SINCE CONNECTIONS TO OUR SITE ARE PROTECTED BY STRONG ENCRYPTION (HTTPS) NOBODY ELSE CAN SEE THE INFORMATION EXCHANGED BETWEEN YOUR SITE AND OUR SITE AND BETWEEN OUR SITE AND BOX. HOWEVER, AT THE FINAL 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 Box. 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 Box integration.

Moreover, the above means that there are additional requirements for using Box 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 and www.akeeba.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 Box' domains over port 443 to allow the integration to work. These domain names are, unfortunately, not predefined.

Upload to Box



It has the following options:

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.
Authentication - Start here	Click this button to login to Box.com and authorize file transfers to and from it. Follow the instructions on screen. At the end of the process the Access and Refresh Token fields will be filled in for you.
Directory	The directory where you want your archive to be stored.
Access Token and Refresh Token	These are populated through the Authentication - Step 1 button above. Please do NOT copy these to other sites or other backup profiles. If the access and refresh tokens get out of sync your backup archive uploads will fail.

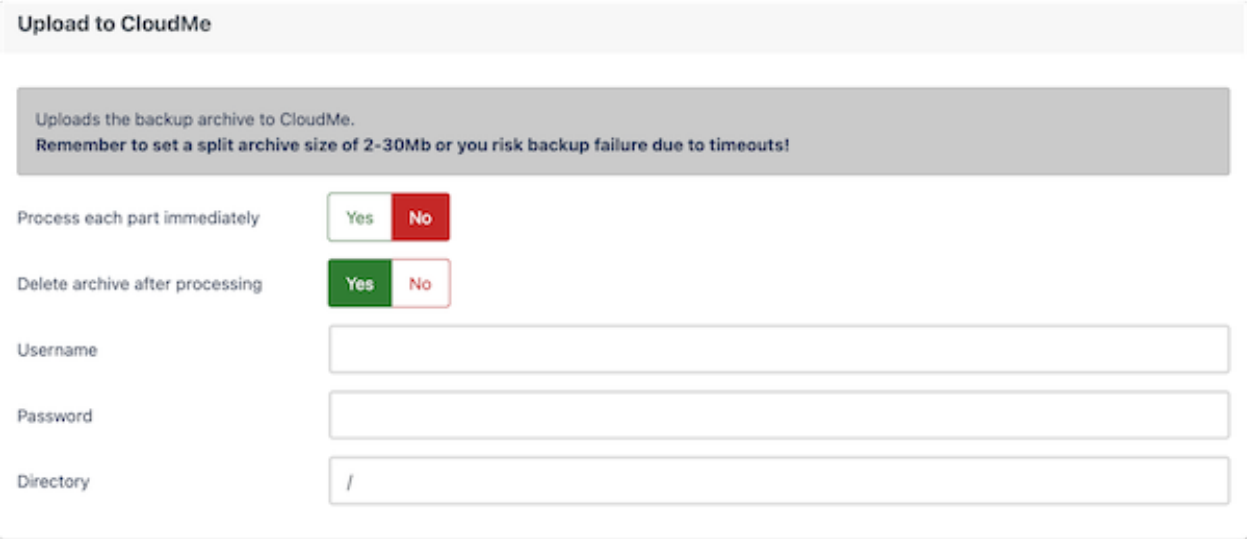
3.3.5.6. Upload to CloudMe

Note

This feature is available only to Akeeba Backup Professional.

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

Upload to 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/sub-subdirectory`. You can use backup naming 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].

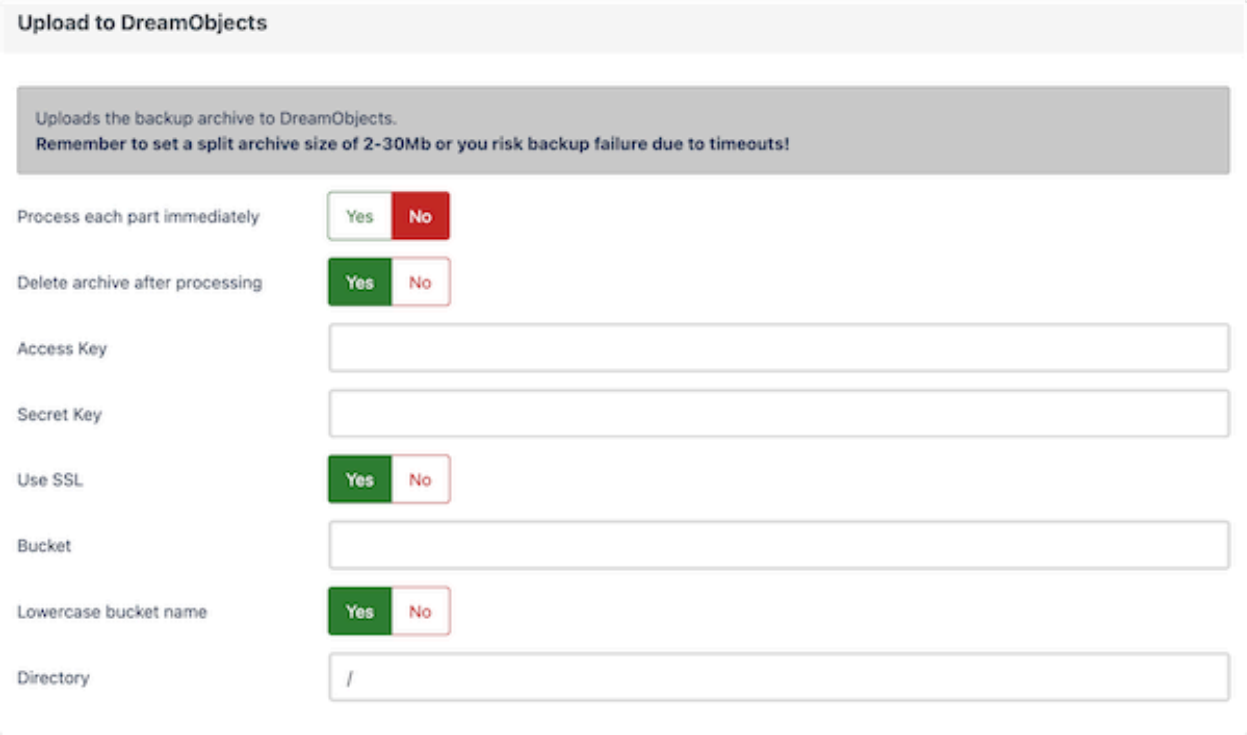
3.3.5.7. Upload to DreamObjects

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.

Upload to DreamObjects



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.

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.

Lowercase bucket name
: When enabled Akeeba Backup will automatically convert the bucket name to all lowercase letters. This addresses a common problem where you've created a bucket named mysite but try to enter it as MySite in Akeeba Backup. This discrepancy causes the backup to fail to upload to DreamObjects. If, however, you have created a bucket with any number of uppercase letters please disable this option and make sure that the bucket name you have entered matches exactly the name of the bucket in DreamObjects, including lowercase and uppercase letters.

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/sub-subdirectory`.

  You can use backup naming 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.

3.3.5.8. Upload to Dropbox (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.

Important security and privacy information

Dropbox 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 Dropbox'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 Dropbox. When you are linking Akeeba Backup to Dropbox you are going through the script on our site. Moreover, whenever the request token (a time-limited key given by Dropbox 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 Dropbox account.

WE DO NOT STORE THE ACCESS CREDENTIALS TO YOUR DROPBOX ACCOUNT. WE DO NOT HAVE ACCESS TO YOUR DROPBOX 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 DROPBOX. 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 Dropbox. 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 Dropbox integration.

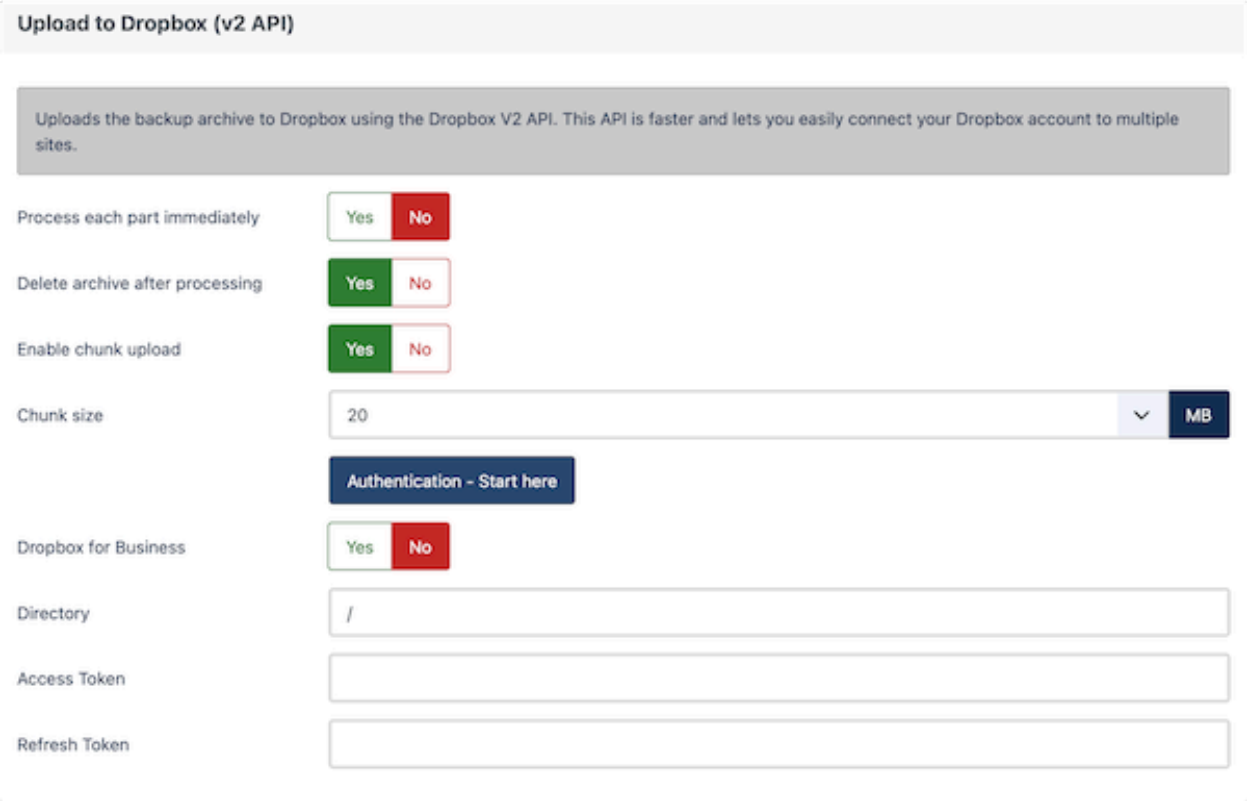
Moreover, the above means that there are additional requirements for using Dropbox integration on your Akeeba Backup installation:

- 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.akeeba.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 Dropbox's domains over port 443 to allow the integration to work. These domain names are api.dropboxapi.com and content.dropboxapi.com per Dropbox' documentation.

Upload to Dropbox (v2 API)



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.

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.

  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.

Authorisation
: Before you can use the application with Dropbox you have to "link" your Dropbox account with your 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 - Start here 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 Backup installation.

  You need to do this on every backup profile you want to link to Dropbox, even on the same site.

Dropbox for Business
: Check if you are a member of a team using Dropbox for Business and want to use your team's folder to store backups. Remember to prefix the Directory below with the name of your team folder. For example, if Dropbox' web interface shows an “Acme Corp Team Folder” in the Files page and you want to put your backups in a folder called “Backups” inside it you should use the Directory name `Acme Corp Team Folder/Backups`, not just `Backups`.

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/sub-subdirectory`.

Access Token
: This is the short-lived access 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.

  It is very important that you DO NOT copy the Access Token to other backup profiles on the same or a different site. It will not work. Instead, please use the Authentication - Start here button on each profile and site you want to connect to Dropbox.

Refresh Token
: This is the long-lived refresh token to Dropbox. It is used to refresh the access token every time it expires. Normally, it is automatically fetched from Dropbox when you click on the Authentication - Start here button above. If for any reason this method does not work for you you can copy the Token from the popup window.

  It is very important that you DO NOT copy the Refresh Token to other backup profiles on the same or a different site. It will not work and will only cause the Dropbox connection to be lost on your site. Instead, please use the Authentication - Start here button on each profile and site you want to connect to Dropbox.

3.3.5.9. Upload to Google Drive

Note

This feature is available only to Akeeba Backup Professional. Using it requires entering your Download ID in the software and having an active subscription on our site which gives you access to one of our backup software.

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:

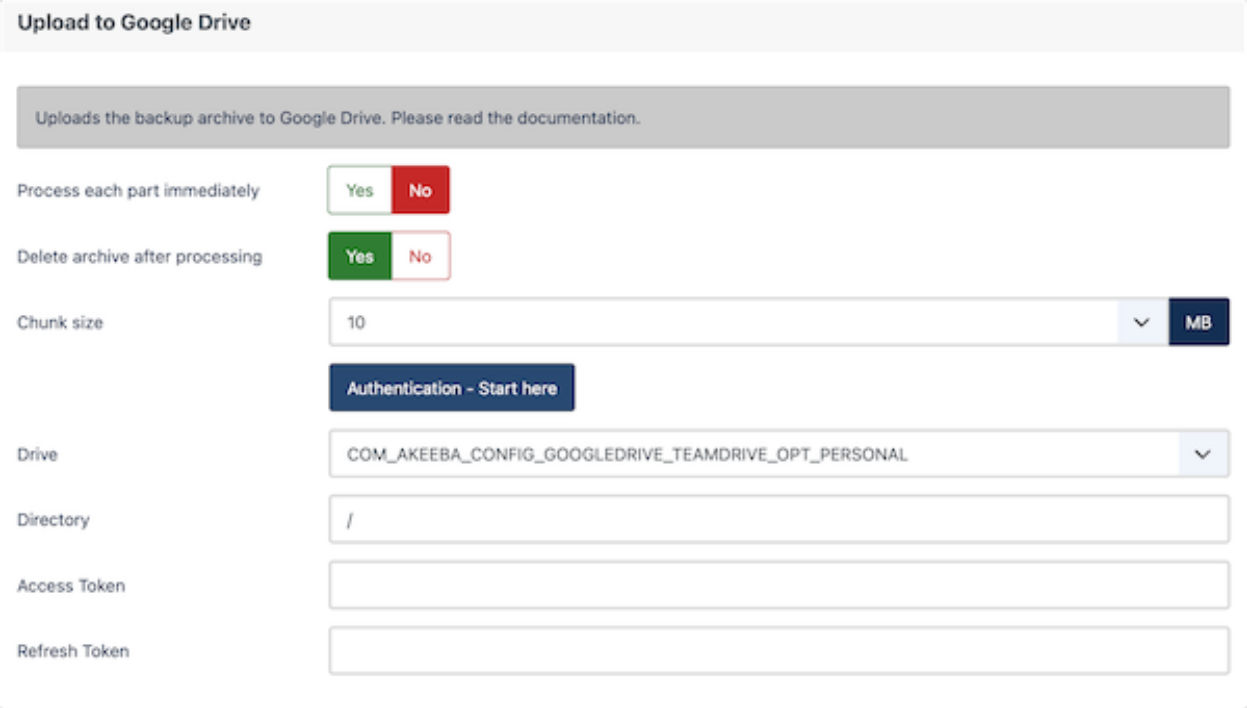
- You need the PHP cURL extension to be loaded and enabled on your server. Most servers do that by default. If your server doesn't have it enabled the upload will fail and warn you that cURL is not enabled.

- Your server's firewall must allow outbound HTTPS connections to www.akeebabackup.com and www.akeeba.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

Upload to Google Drive



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 – Start here

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.

Drive

If your account has access to Google team drives you can select which of these drives you want to connect to. However, this process is a bit more complicated than it sounds.

First, you need to use the Authentication - Step 1 button to connect to Google Drive. Second, you need to click the Save button in the toolbar to apply the Google Drive connection information. Third, you can select your drive using the dropdown. This is the only method for selecting a drive which is guaranteed to work.

Please remember that the list of drives is returned by Google's servers. If you do not see a drive there your problem has to do with permissions and setup at the Google Drive side of things. In this case please be advised that we are not allowed to help you; Google forbids us from providing help about its products through our site. You will have to peruse their community support forums to seek assistance about setting up your team drives to be visible by your Google Drive account.

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 backup naming 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.

3.3.5.10. 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.

Foreword 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. Unfortunately, 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.

Performance and stability

According to our extensive tests in different server environments, the performance and stability of Google Storage is not a given. We've seen upload operations randomly failing with a Google-side server error or timing out when the immediately prior upload of a same sized file chunk worked just fine. We've seen file deletions taking anywhere from 0.5 to 13 seconds per file, for the same file, storage class and bucket with the command issued always from the same server. Please note that you might experience random upload failures. Moreover, you might experience random failures applying remote storage quotas if deleting the obsolete files takes too long to be practical. These issues are on Google Storage's side and cannot be worked around in any way using code in the context of a backup application that's bound by PHP and web server time limits.

We recommend using a remote storage service with good, consistent performance such as Amazon S3 or BackBlaze B2.

Initial Setup

Before you begin you will need to create a JSON authorization file for Akeeba Backup. 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.

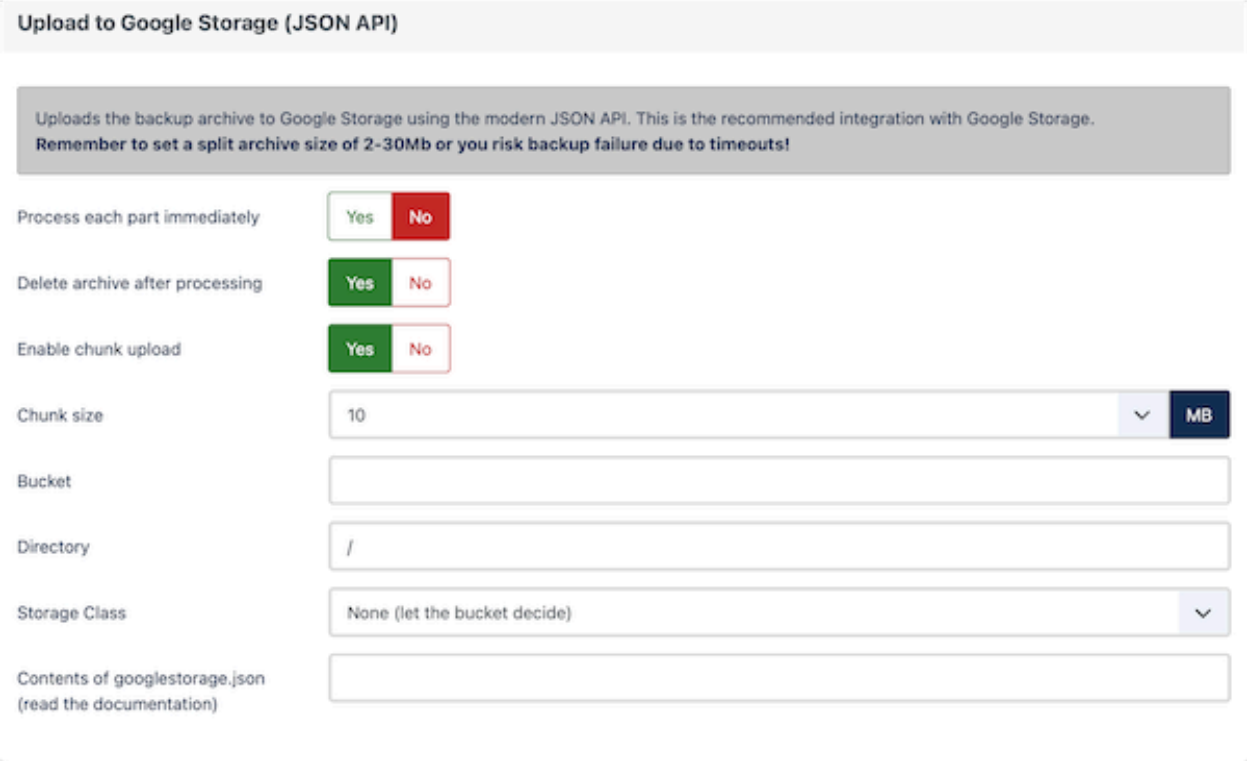
1. Go to https://console.developers.google.com/permissions/serviceaccounts?pli=1
2. Select the API Project where your Google Storage bucket is already located in.
3. Click on Create Service Account
4. Set the Service Account Name to `Akeeba Backup Service Account`
5. Click on Role and select Storage,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

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 will upload your backup archives in smaller chunks, the size of which is specified in the Chunk Size option belo. This is the recommended methods for larger (over 10Mb) archives and/or archive parts.

Chunk size
: Select the maximum size of a file chunk Akeeba Backup will try to upload at once. It's recommended to keep relatively low, e.g. 5 to 20 Mb depending on the transfer speed between your server and Google's servers.

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.

  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/sub-subdirectory`. You can use backup naming 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].
Storage class	Change the storage class of the uploaded backup archives. “None” means that the files will be stored with the default storage class specified in your bucket. Please note that options other than Standard may be cheaper to store but they may incur additional fees if you download them or delete them. Please consult Google for pricing information.
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.

3.3.5.11. Upload to Google Storage (Legacy S3 API)

Note

This feature is available only to Akeeba Backup Professional.

This is an old implementation which might stop working when Google drops support for the S3 API. We recommend using the more modern JSON API integration described later in the documentation.

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)

Please note that 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.

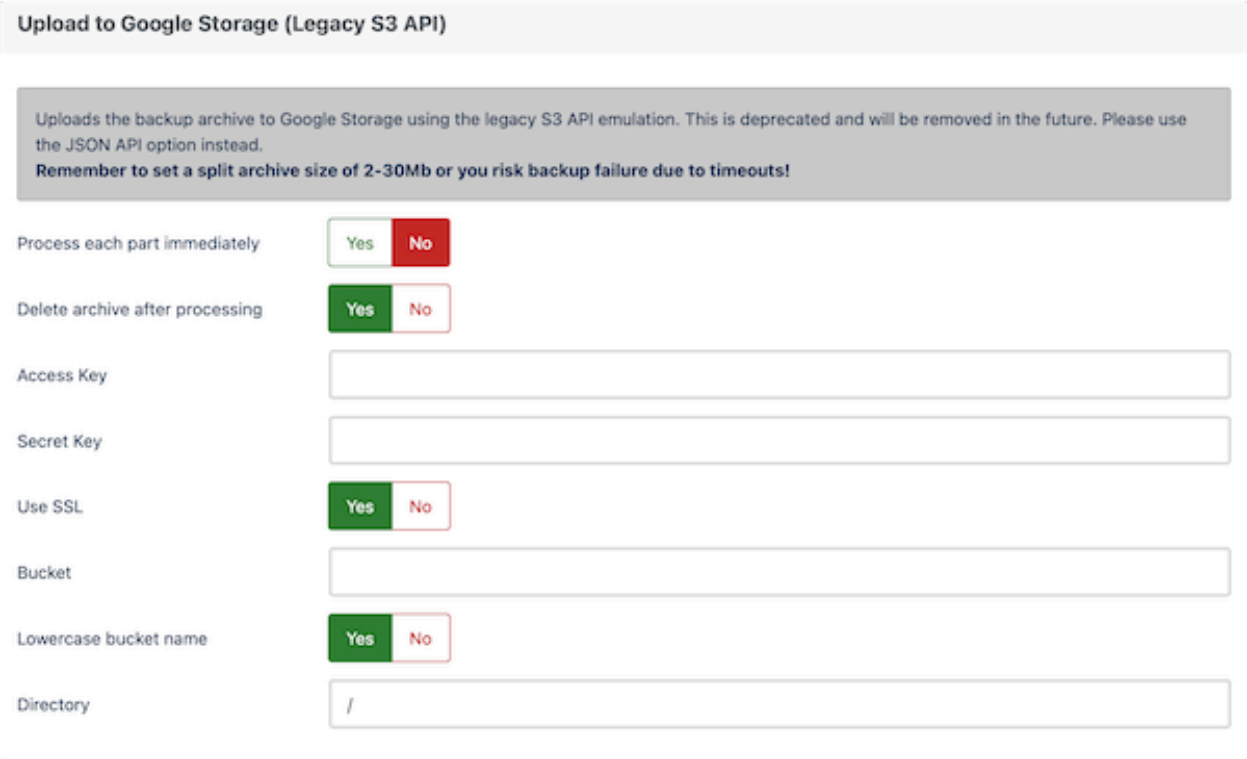
Before you begin you have to go to the Google Developer's Console. After creating a storage bucket, in the left hand menu, go to Storage, Cloud Storage, Settings. Then go to the tab/option Interoperability. There you can go and enable interoperability and create the Access and Secret keys you need for 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 10 and 20 MB.

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.

Upload to Google Storage (Legacy S3 API)



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.

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 API 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.

Lowercase bucket name
: When enabled Akeeba Backup will automatically convert the bucket name to all lowercase letters. This addresses a common problem where you've created a bucket named mysite but try to enter it as MySite in Akeeba Backup. This discrepancy causes the backup to fail to upload to Google Storage. If, however, you have created a bucket with any number of uppercase letters please disable this option and make sure that the bucket name you have entered matches exactly the name of the bucket in Google Storage, including lowercase and uppercase letters.

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 backup naming 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 OneDrive and OneDrive for Business

Note

This feature is available only to Akeeba Backup Professional and **requires an active subscription** to use.

Using this engine, you can upload your backup archives to the low-cost Microsoft OneDrive cloud storage service (https://onedrive.live.com). Moreover, this engine also supports OneDrive for Business which is the kind of free OneDrive storage you get with your Office 365 for Business, work or school account.

OneDrive is an ideal option for small to medium websites with a low budget. It offers a substantial amount of free storage, especially if you are an Office 365 or Office 365 for Business subscriber. 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.

Please note that OneDrive is rather slow. If you have a big site, take frequent backups or otherwise upload performance is of the essence you should use a speedier storage provider such as Amazon S3, BackBlaze B2 or, if you'd rather remain in Microsoft's cloud ecosystem, Microsoft Azure BLOB Storage.

Important security and privacy information

The OneDrive integration 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 this problem by creating a small 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 access token (a time-limited, really long password 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. CONNECTIONS TO OUR SITE ARE PROTECTED BY STRONG ENRYPTION (HTTPS), THEREFORE 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.

Important

Access to the intermediary script on our servers **requires** a. an active subscription to any of our products and b. entering a valid Download ID for your AkeebaBackup.com account in the component's options. If the

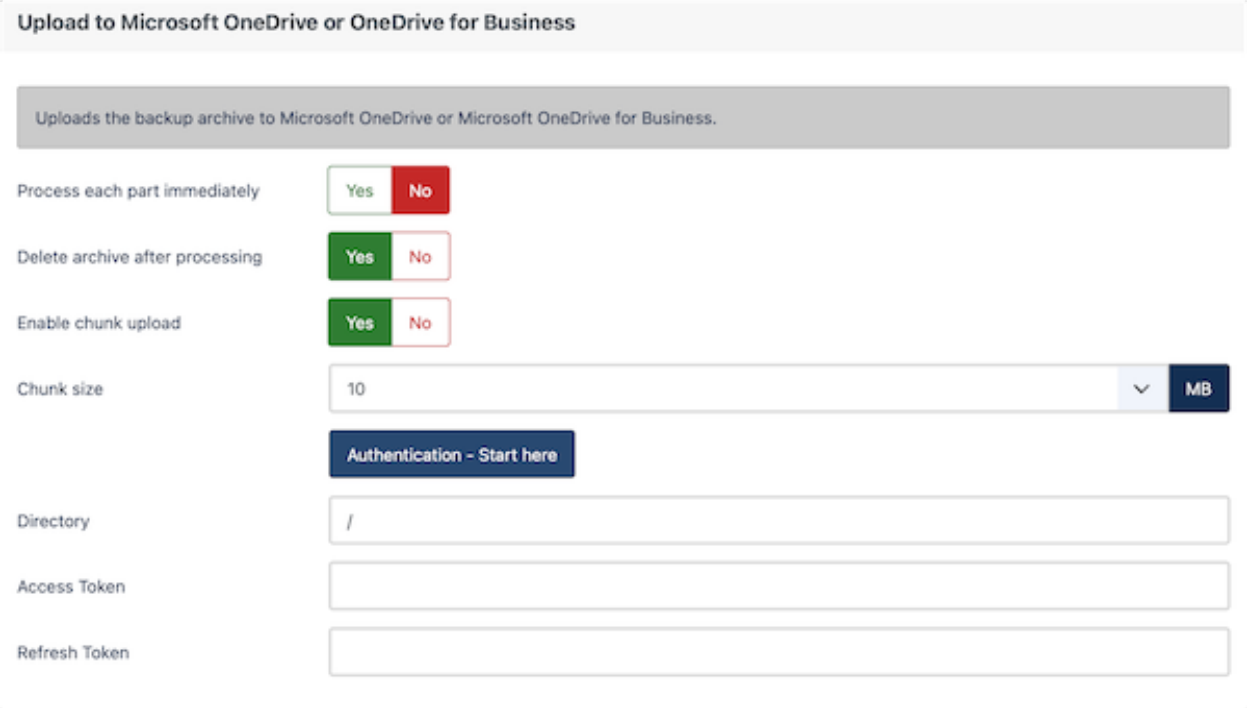
Download ID is invalid or corresponds to an expired subscription you will be unable to use the intermediary script on our servers. As a result you will be unable to upload backup archives to OneDrive.

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 and www.akeeba.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

Upload to OneDrive



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.

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).

Authorisation – Start here
: 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/sub-subdirectory`.

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.

  Unlike other engines, such as Dropbox, you CANNOT share OneDrive tokens between multiple sites or backup profiles. 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.

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

3.3.5.13. 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 cloud storage service from Microsoft is reasonably priced and quite fast, with lots of local endpoints around the globe.

File size limits

Azure BLOB Storage, like most modern file storage services, supports two ways of uploading files: either all at once or in chunks (multipart upload). When uploading files all at once the maximum file size is 5000Mb. When uploading in chunks the maximum file size is 50,000 chunks, therefore the maximum file size is determined by the chunk size. The chunk sizes supported are 1Mb to 4000Mb. This means that the maximum size of a single backup archive file you can upload is between 48.82Gb and 190.7Tb.

Note

Akeeba Backup 9.0.0 to 9.2.1 (inclusive) only supported an old version of the Azure API which had a maximum upload limit of 64MiB. Moreover, they did not support chunked (multipart) uploads of larger files. The information below only applies to Akeeba Backup 9.2.2 and newer versions.

If you are still using an older version of Akeeba Backup you will have to set the Part size for archive splitting to 64MiB or less. Typically, you'd need to use 10Mb to avoid your server timing out during the upload.

Multipart uploads

Do keep in mind that when running a backup over the web (using a browser, the remote JSON API or the legacy frontend backup method) there are several server-related timeouts you cannot ignore. First and foremost there's the PHP timeout itself which, in most cases, we can work around. There are also timeouts in the PHP FastCGI Process Manager (FPM) whenever it's used, your web server (Apache, NginX, IIS, ...) and the operating system itself (maximum CPU usage time per process a.k.a. ulimit -t). These can affect how long an individual upload operation can run. The time required to upload a backup archive file (for all at once uploads) or a chunk of it (for multipart uploads) to Azure equals the size of the file or chunk divided by the available bandwidth.

We want the time to upload a file / chunk to be less than the minimum time limit restriction on your server 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 or chunk size. To this end, there are two thing you can do depending on how you are uploading files:

- For all at one uploads you have to produce split backup archives, by setting the Part size for archive splitting in the archiver engine's configuration pane. The suggested values are between 10 and 20 Mb.

- For multipart uploads you should set the chunk size. Recommended values are 1Mb to 100Mb depending on your server's available bandwidth. We recommend starting with 10Mb. If it times out, decrease it. If it works fine, try increasing it until you hit a timeout; when that happens, walk back to the previous setting.

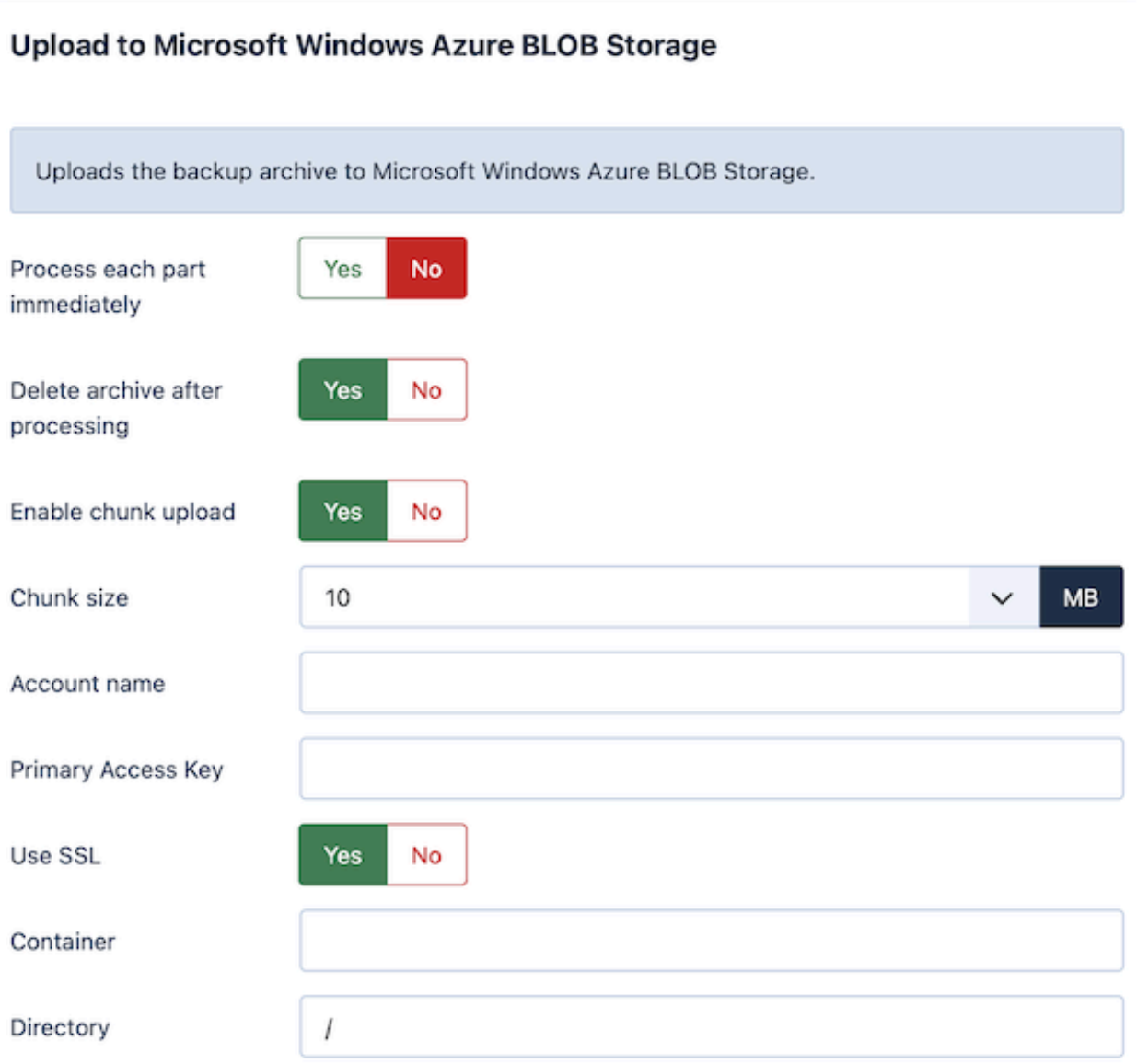
If you use the native CLI backup script there is no applicable time limit except the server's maximum CPU usage time per process which controls the total time the backup and upload process can take. As a result you can select to NOT do chunked uploads if the backup profile is only ever going to be used under the CLI backup script. Also remember that if your CLI backup script fails you should ask your host to “increase the ulimit -t for CLI scripts” — just tell them that and the second level support of your host will know what to do.

Getting the account name and account key

1. Go to your Subscriptions [https://portal.azure.com/#blade/Microsoft_Azure_Billing/SubscriptionsBlade] page on the Azure Portal.

2. Click on your subscription name.
3. In the new blade that opens find the Settings header and click Resources under it.
4. Click on the Storage Account resource your container is located in.
5. Make sure that under the Properties tab, Security header, the “Storage account key access” setting is Enabled. If not, you need to enable it.
6. In the sidebar find the Security + Networking header and click on Access Keys under it.
7. Click at the Show Keys link at the top of the main content area.
8. Copy the “Storage account name” into the Account Name field in Akeeba Backup.
9. Find the key1 heading and copy the contents of the “Key” field into the Primary Access Key field.

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 twice the size of your part size for split archives free in your account. If you don't enable this option, the upload process will take place after the backup is complete and finalized. The drawback to that 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.

Enable chunk upload
: Should we use multipart uploads for large archive files? Archive files over the Chunk Size threshold will be uploaded in multiple chunks, each one at most as big as the Chunks Size.

> **Important**
>
> Microsoft Azure BLOB Storage has a hard limit of up to 50,000 parts per uploaded file. If your backup archive is bigger than 50,000 times the Chunk Size it's not possible to upload it using the multipart. In this case Akeeba Backup will try to upload it all at once which might lead to a timeout. If this happens we strongly recommend setting up the Part Size for Archive Splitting in the Archiver Engine options, setting that to a value under 2000Mb.

Chunk size
: How much of the backup archive to transfer at once. Please refer to the Multipart Uploads information above the image.

Account name
: The storage account name for your Microsoft Azure subscription. See the instructions above the image.

Primary Access Key
: The key1 value for accessing your Microsoft Azure subscription. See the instructions above the image.

Use SSL
: Please leave this enabled. It tells Akeeba Backup to use the secure HTTPS protocol to communicate to Microsoft Azure. Disabling it will use the insecure HTTP protocol, without any encryption of your communication between your site's server and Microsoft Azure. Support for using unencrypted HTTP might be removed by Microsoft without prior warning.

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.

3.3.5.14. Upload to OVH Object Storage

> **Note**
>
> This feature is available only to Akeeba Backup Professional.

Using this engine, you can upload your backup archives to the OVH Object Storage cloud storage service. This allows you to upload files into OVH's public cloud, powered by the OpenStack technology.

Before you begin, you should know the limitations. As most cloud storage providers, OVH 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 OVH 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 10 and 20 MB.

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.

Before you begin

You will need to set up object storage and collect some necessary but not necessarily obvious information from your OVH account. You can do so through OVH's Cloud Manager [https://www.ovh.com/manager/cloud/index.html#/] portal.

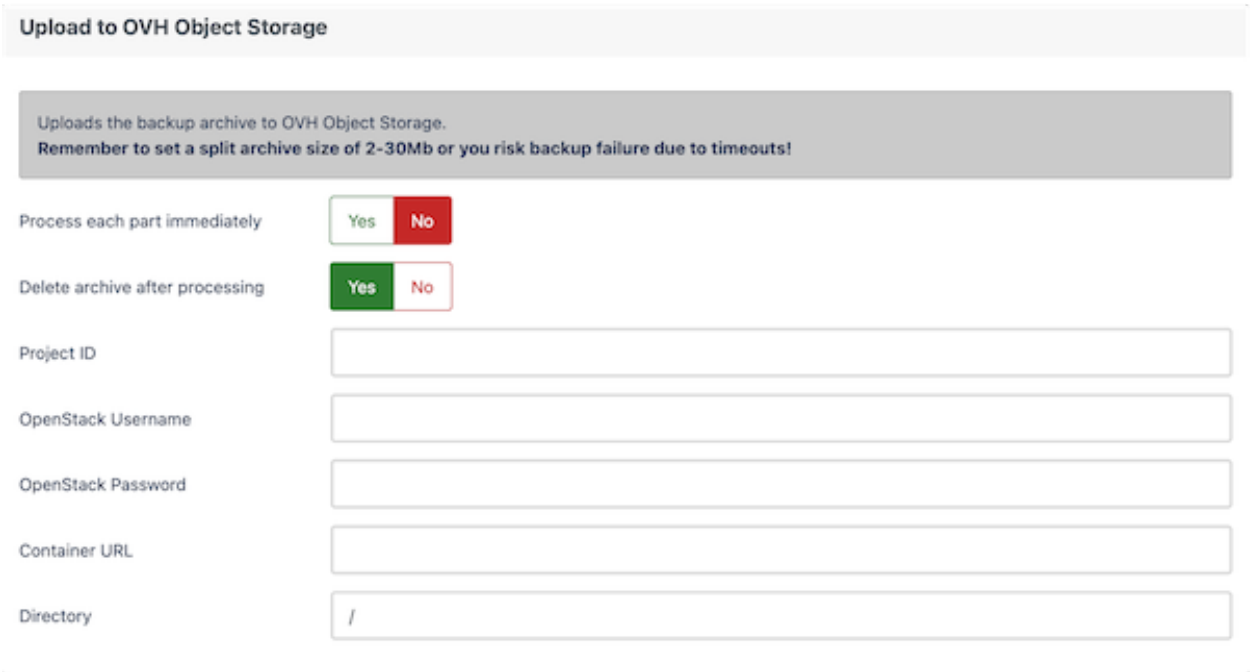
From the left side menu click on Servers and expand your cloud server. If you do not have a server yet you will need to use the Order button to purchase credits. Please note that credits activation can take several days if this is your first order.

Click on the Infrastructure link under your server. In the main area of the manager page you will see the name of your server. Below it, in hard to see grey letters, you will see a 32-digit alphanumeric code such as abcdef0123456789abcdef0123456789. Note it down. This is your *Project ID*.

Click on the Storage link under your server. You will see a list of your containers. If you do not have any containers yet, create a new one. Make sure to select the Private type; you don't want your backups to be publicly accessible! Click on your server's name. The main area changes. You will see a box with information such as objects, container size and Container URL. Note down the *Container URL*.

Click on the OpenStack link under your server. If you have not created an OpenStack user yet, create one now. Copy the values under the ID and Password columns. These are, respectively, your *OpenStack Username* and *OpenStack Password*.

Upload to OVH Object Storage

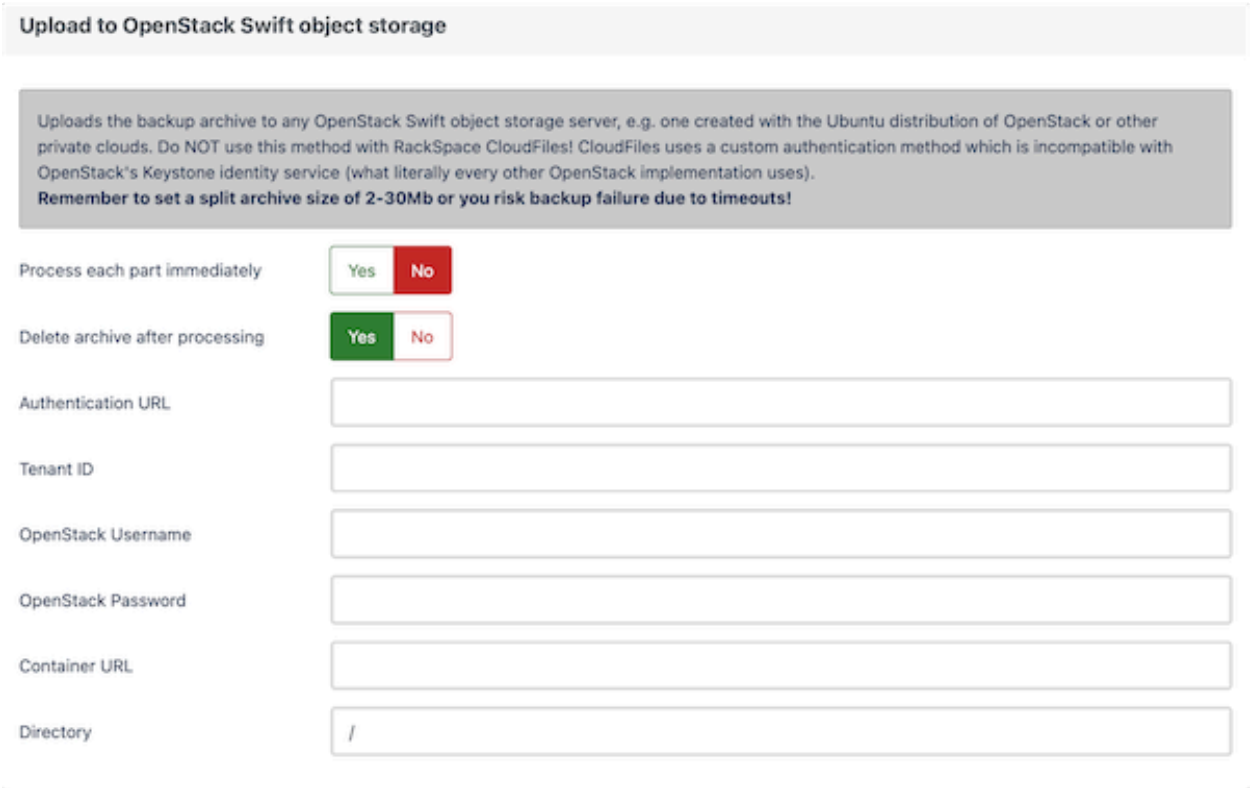


The required settings for this engine are:

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 OVH.
Project ID	See above.
OpenStack Username	See above.
OpenStack Password	See above.
Container URL	See above.
Directory	The directory inside your OVH 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.

3.3.5.15. Upload to OpenStack Swift object storage

Upload to OpenStack Swift



This allows you to upload backup archive to any OpenStack Swift implementation. This is the same thing as the OVH Object Storage (see above) with two additional options:

Authentication URL	The endpoint for the Keystone service of your OpenStack installation. DO include the version. DO NOT include the /token suffix. Example: https://authentication.example.com/v2.0
Tenant ID	Your OpenStack tenant ID, e.g. a0b1c2d3e4f56789abcdef0123456789

3.3.5.16. Upload to RackSpace CloudFiles

Note

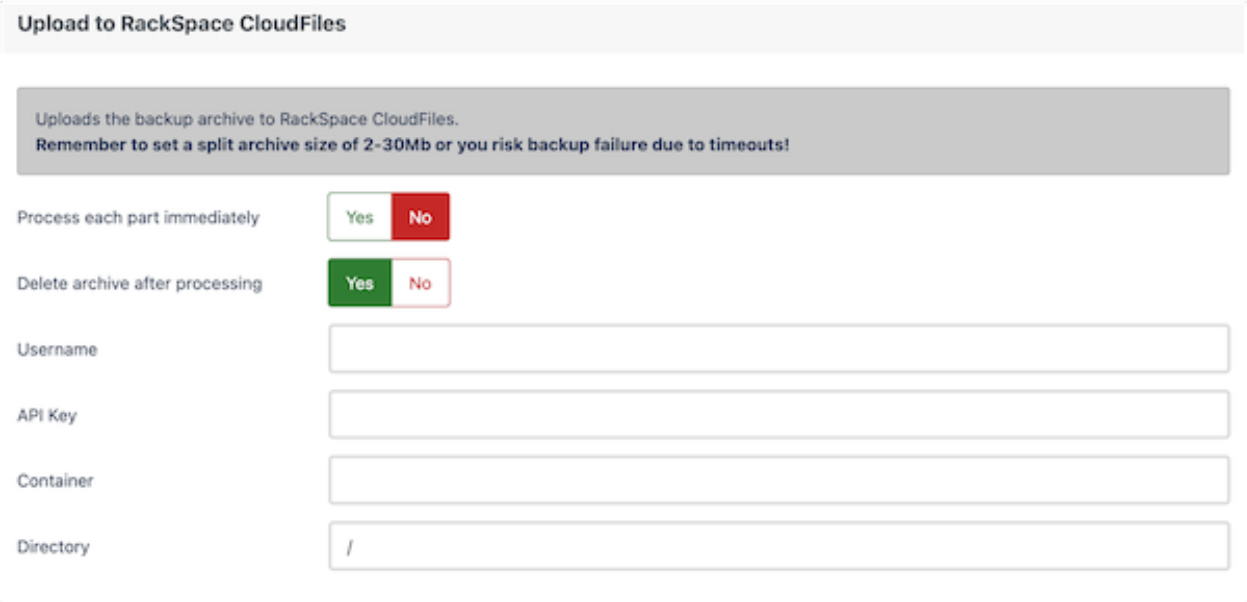
This feature is available only to Akeeba Backup Professional.

Using this engine, you can upload your backup archives to the RackSpace CloudFiles [www.rackspace-cloud.com/cloud_hosting_products/files] cloud storage service. This service had previously been called Mosso.

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 10 and 20 MB.

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.

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/sub-subdirectory`. Leave blank to store the files on the container's root.

3.3.5.17. Upload to Remote FTP server

Note

This feature is available only to Akeeba Backup Professional.

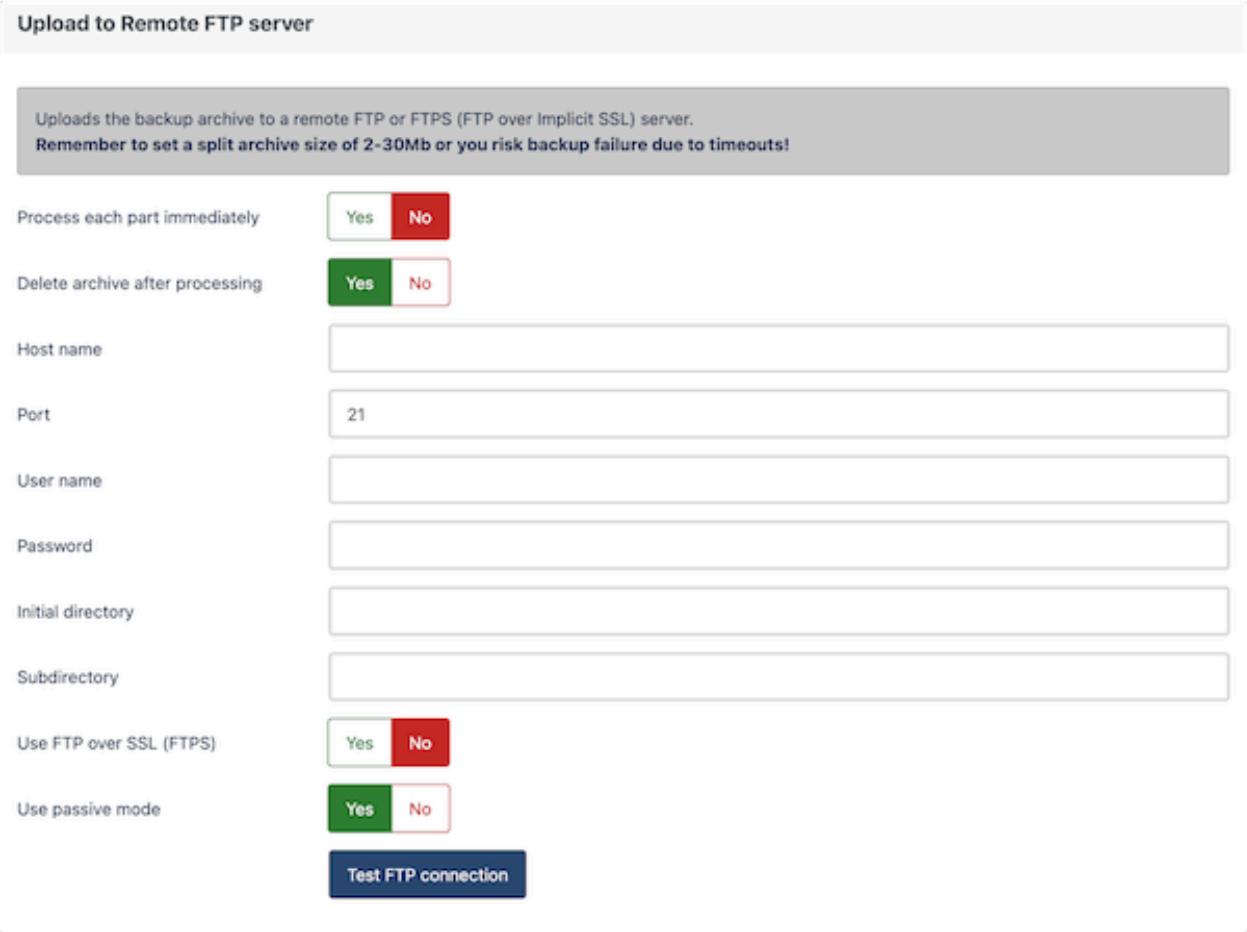
Using this engine, you can upload your backup archives to any FTP or FTPS (FTP over Explicit SSL) server. There are some "FTP" protocols and other file storage protocols which are not supported, such as SFTP, SCP, Secure FTP, FTP over Implicit SSL and SSH variants. The difference of this engine to the 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.

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.

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 10 and 20 MB.

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.
Subdirectory	The relative path to the initial directory, it will be created if it doesn't exists. Leave it empty to upload the archives directly inside the initial directory. You can use all of the backup naming variables, e.g. [HOST] for the hostname of the site being backed up.
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!

3.3.5.18. Upload to Remote FTP server over cURL

Note

This feature is available only to Akeeba Backup Professional.

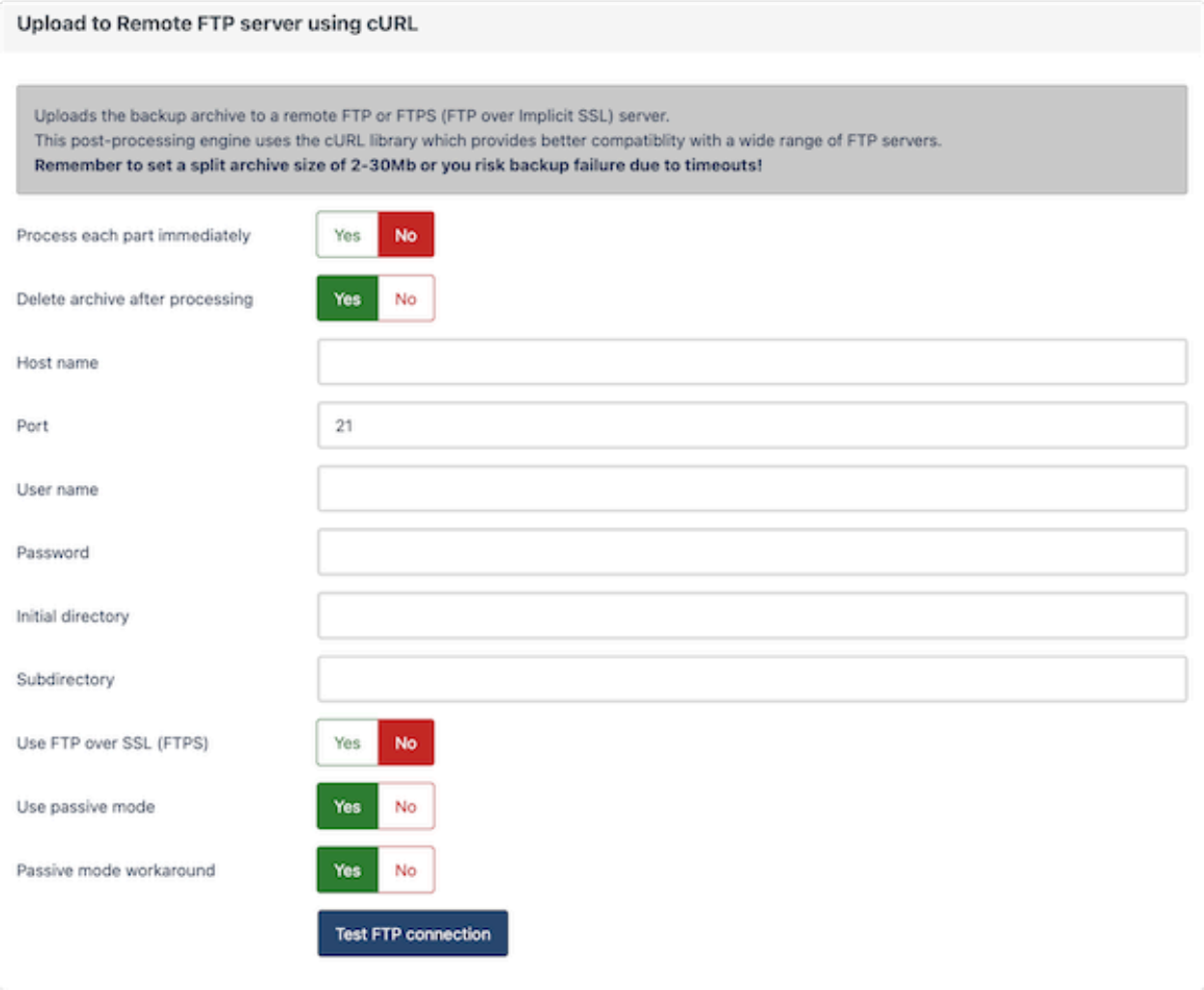
Using this engine, you can upload your backup archives to any FTP or FTPS (FTP over Exclicit SSL) server. There are some "FTP" protocols and other file storage protocols which are not supported, such as SFTP, SCP, Secure FTP, FTP over Implicit SSL and SSH variants. The difference of this engine to the 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.

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.

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 10 and 20 MB.

Upload to Remote FTP Server over cURL



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.
Subdirectory	The relative path to the initial directory, it will be created if it doesn't exists. Leave it empty to upload the archives directly inside the initial directory. You can use all of the backup naming variables, e.g. [HOST] for the hostname of the site being backed up.
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 FTP server's public IP address, as seen by your web 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.

3.3.5.19. Upload to Remote SFTP server

Note

This feature is available only to Akeeba Backup Professional.

Note

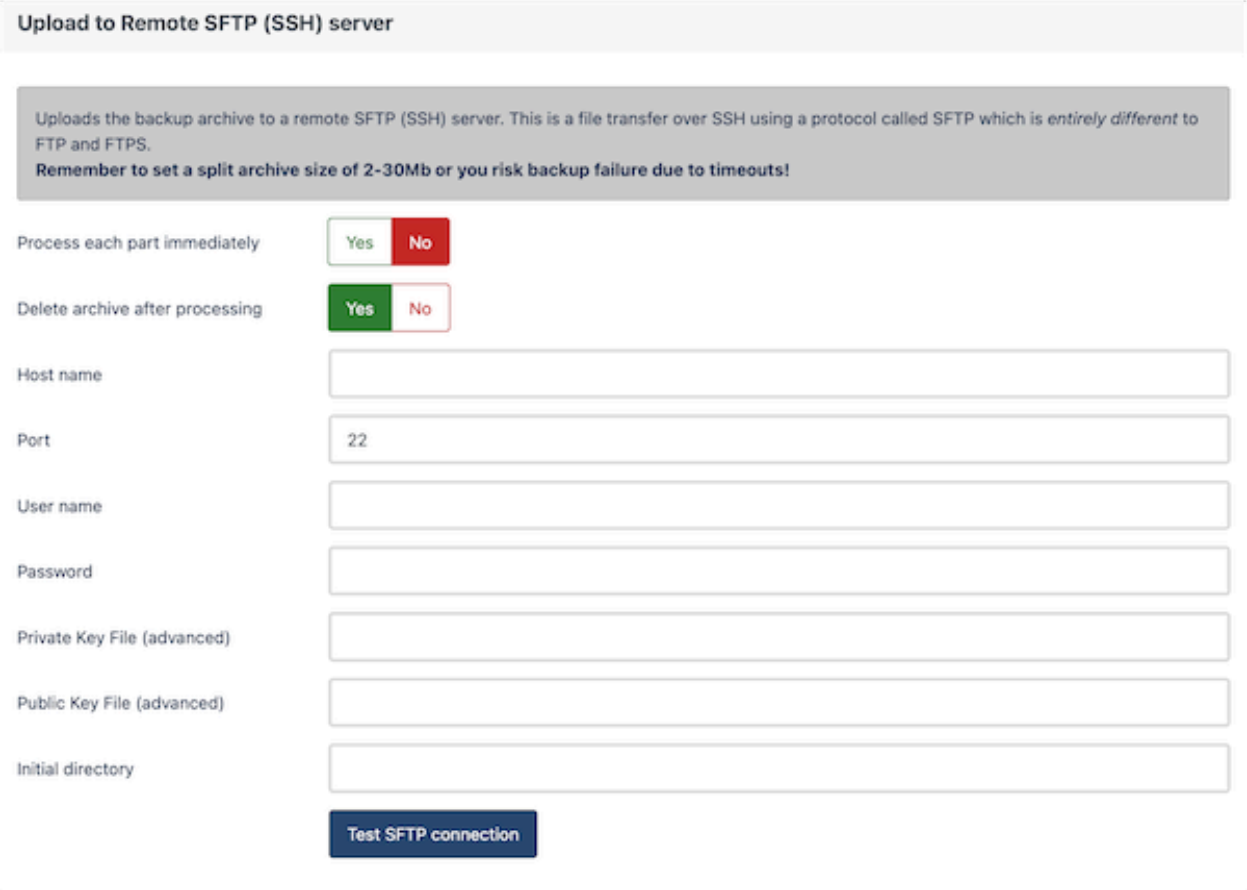
This engine uses the PHP extension called SSH2. The SSH2 extension is still marked as an alpha and is not enabled by default or even provided by many commercial hosts. In this case you may want to use the Upload to Remote SFTP server over cURL engine instead which uses PHP's cURL extension, available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

Upload to Remote SFTP Server



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.

3.3.5.20. 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.

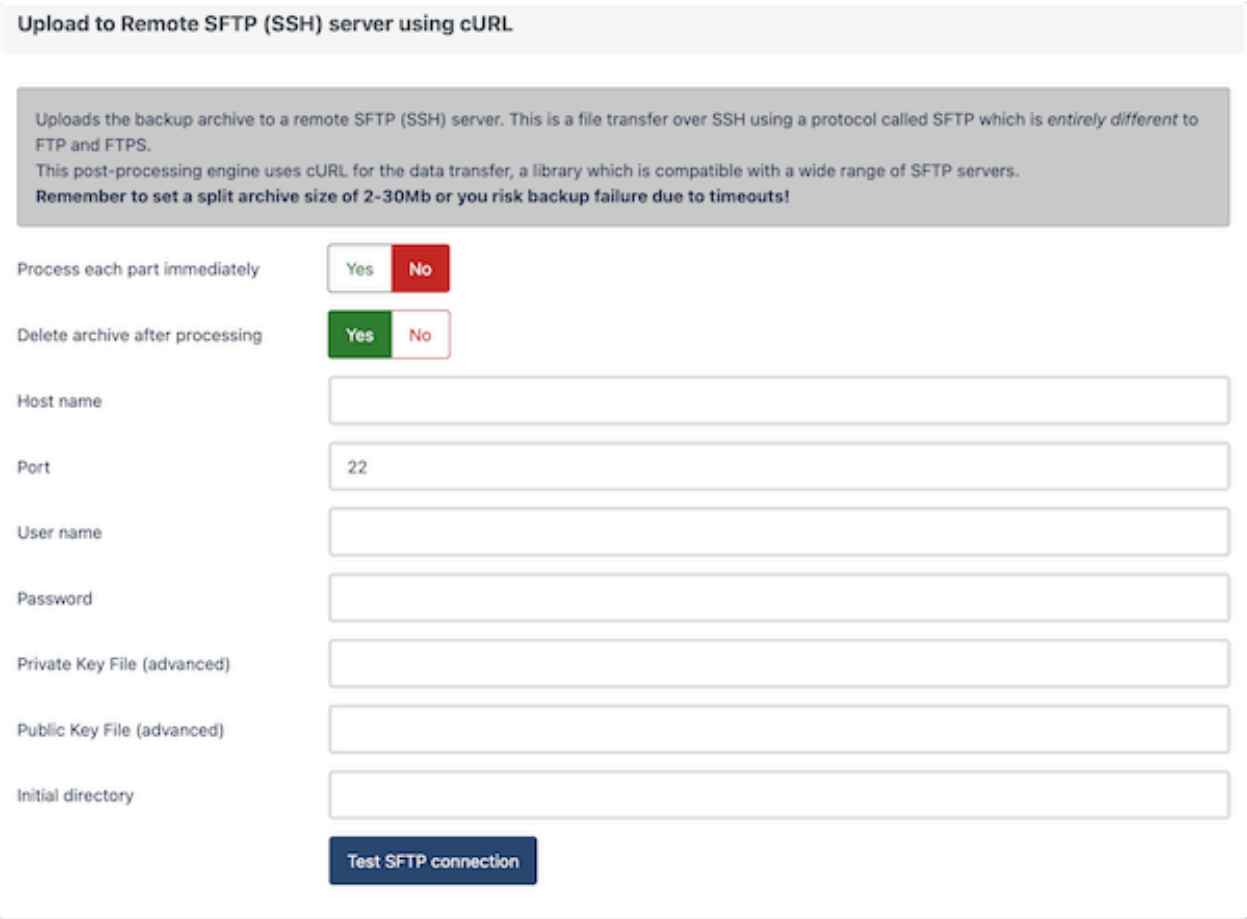
Using this engine, you can upload your backup archives to any SFTP (Secure File Transfer Protocol) server. Please note that SFTP is the *encrypted* file transfer protocol provided by SSH servers. Even though the name is close, it has nothing to do with plain old FTP or FTP over SSL. Not all servers support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

Upload to Remote SFTP Server over cURL



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.

3.3.5.21. 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.

First-time setup

Since Akeeba Backup 7.0 you need to perform an additional step the very first time you set up SugarSync to obtain a set of Access Key ID and Secret Access Key which will be used together with your email and password to access SugarSync. SugarSync's API needs all four pieces of information (Access Key ID, Secret Access Key, Email and Password) to grant access to your files.

First go to SygarSync's site and select the Developer Portal [https://www.sugarsync.com/developer] option at the footer of the site. If this is your first time there select the Join our Program option. It is free of charge.

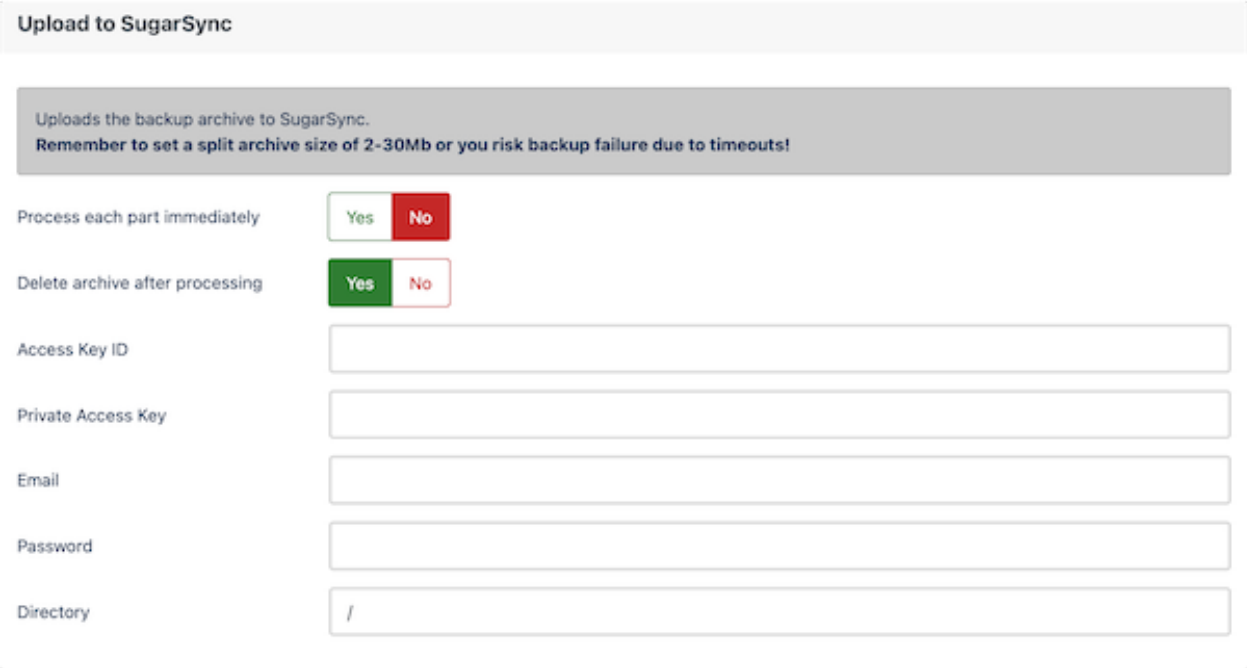
Then go to the Developer Console [https://www.sugarsync.com/developer/account] (it requires you to log into SugarSync). At the top of the page there is the Your Access Keys area. If you already have entries there skip this paragraph. If you do not have any entries there click on Add Keys. This creates an entry.

Note

You can ignore the Your Apps section. In fact, creating an app is optional, makes authentication more complicated and does not offer any security or workflow advantage. Therefore, Akeeba Ltd chose not to implement support for SugarSync's Apps.

You need to copy the Access Key ID and its corresponding Private Access Key in your Akeeba Backup configuration, as explained below.

Upload to SugarSync



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.

Access Key ID
: The Access Key ID you have created in SugarSync's Developer Console [https://www.sugarsync.com/developer/account] page.

Private Access Key
: The Private Access Key that corresponds to the Access Key ID you have created in SugarSync's Developer Console [https://www.sugarsync.com/developer/account] page.

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 backup naming variables, 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.

3.3.5.22. Upload to iDriveSync

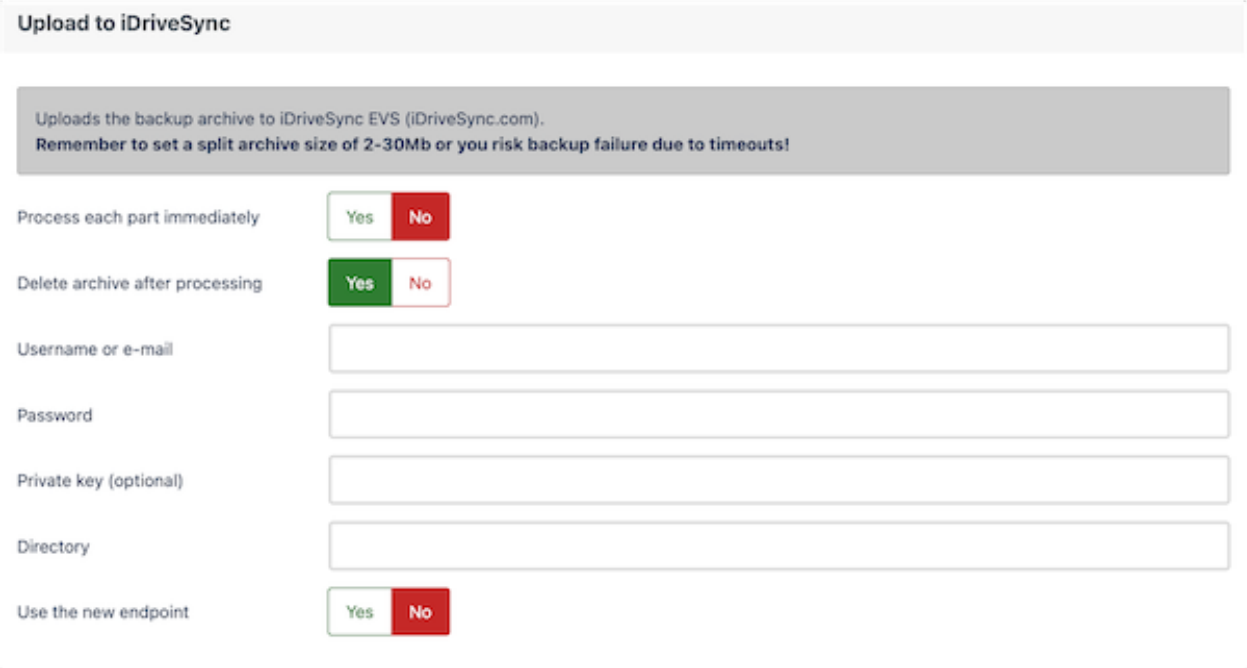
Using this engine, you can upload your backup archives to the iDriveSync low-cost, encrypted, cloud storage service.

Warning

Per the email sent by iDrive Inc., **iDriveSync has reached End of Life in January 2021 and will be turned off December 2021**. Akeeba Backup's iDriveSync integration will be removed between December 2021 and February 2022.

You will need to migrate your data to their iDrive product. iDrive can be used with Akeeba Backup using the Upload to Amazon S3 feature. The information iDrive publishes for Duplicati [https://www.idrive.com/cloud/guides/duplicati] are also relevant for Akeeba Backup. You need to set the Custom Endpoint option to `s3.us-west-1.idrivecloud.io` and change the signature method to v2. You need to create S3 Access Keys in the iDrive Cloud Console and copy them in the Access Key and Secret Key fields in your Akeeba Backup configuration.

Upload to iDriveSync



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 backup naming 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.

3.3.5.23. Upload to WebDAV

> **Note**
>
> This feature is available only to Akeeba Backup Professional.

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

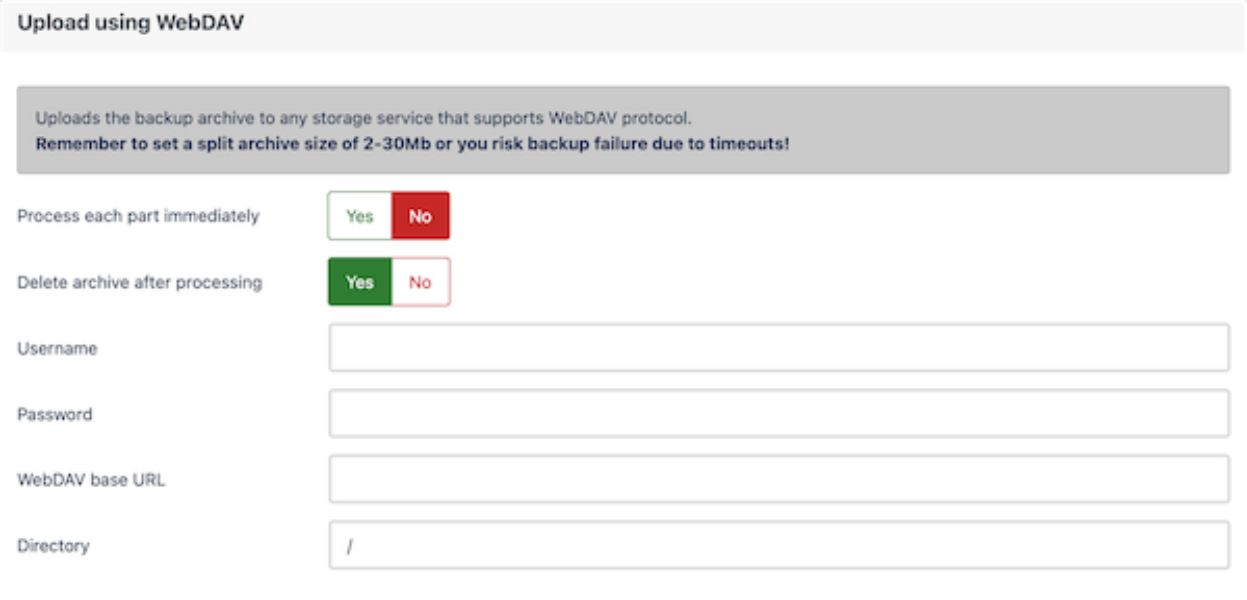
We have not thoroughly tested and do not guarantee that any of the above providers will work smoothly with Akeeba Backup unless you see the notive [TESTED] next to it.

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.

Upload to WebDAV



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 sub-directories, you have to use a forward slash, e.g. `/directory/subdirectory/subsub-directory`. You may use the backup naming variables, 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.

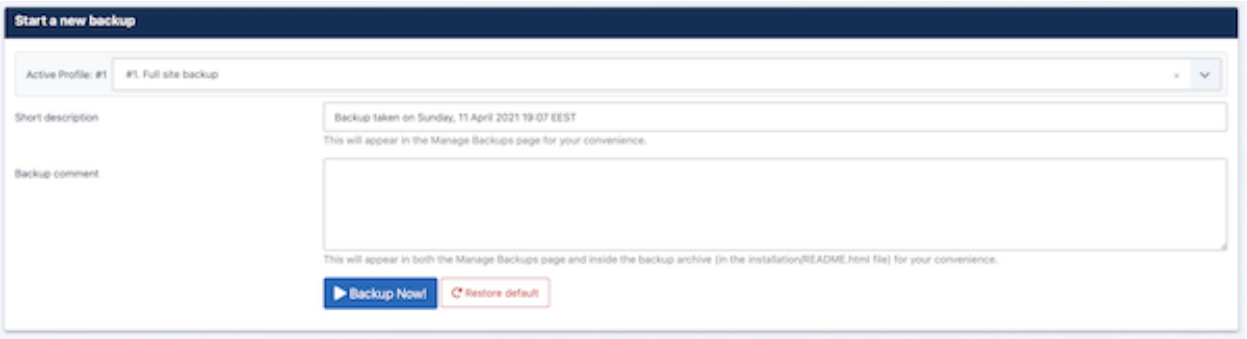
3.4. Backup now

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' some very few webmasters use, 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!

We'd like to note that moving the configuration.php file outside the site's root makes no sense with regards to security anyway. It is trivial for an attacker who can exploit an arbitrary file read or arbitrary code execution vulnerability to find where the real configuration.php file is located in and / or create a listing of all configuration options of your site. Even if an attacker only had access to a file write vulnerability they can break your site regardless of whether you are using the default configuration.php location or not: they can overwrite the defines.php files Joomla uses to find out where the configuration file is stored in, reverting it to the default location. In short, moving the configuration.php file DOES NOT offer any degree of security at all. It's just security theatre which makes management and restoration of your site harder with zero benefit whatsoever.

Moreover we would like to remind you that restoring to a temporary URL (something like http://www.yourhost.com/~youruser) will NOT work. This has to do with the way Joomla!, Apache and PHP session management works. It has nothing to do with Akeeba Backup. Even if you install straight up Joomla! on a temporary URL you'll have problems logging in or, at the very least, with SEF URLs. This is a limitation of your server software, not something which can be addressed by Akeeba Backup and / or its restoration script.

Backup start



The initial backup page lets you define a short description (required) and an optional, lengthier comment for this backup attempt. This information will be presented to you in the Manage Backups 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 Site Backup.

Both the description and the comment support backup naming variables, e.g. `[SITE]`, `[DATE]` and `[TIME]`. 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:

- Encryption key. 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. The installation script which is included in your backup archive, called ANGIE, 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's restoration interface while it's a restoration is in progress and copy your database passwords or obtain other information about your site.

  We STRONGLY advise that you always use an ANGIE Password if you intend to restore your site on a live server. This is the only way to prevent accidental information leak.

  ## 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. It will only prevent people attempting to browse your site while you're restoring a backup from seeing potentially confidential information in the installer.

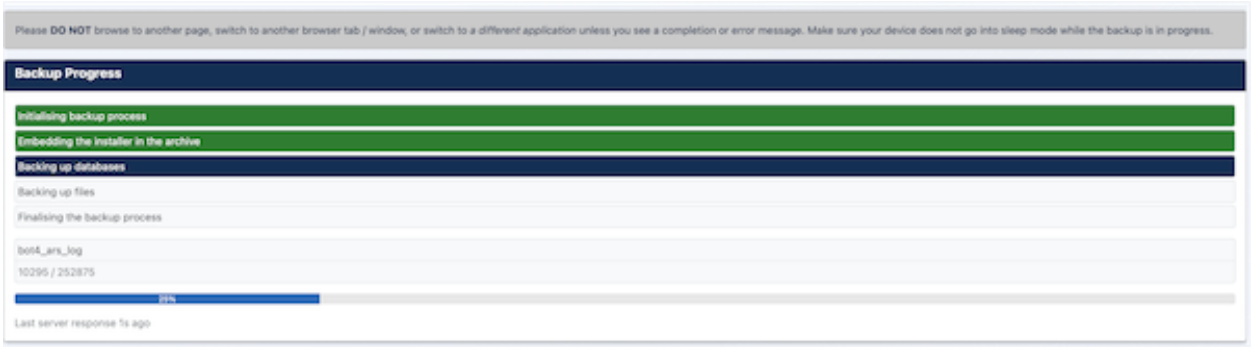
Whenever you are ready to start the backup, just click the Backup Now button. Do note that above the description field, there might be one or more warnings. These are the same warnings appearing in the Control Panel's right-hand pane and act as a reminder.

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, please, we are not your host.

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 best 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.

Please note that browser versions released on or after 2021 also have a feature typically called Sleeping Tabs or something similar. In short, if you have a browser tab in the background or the browser window is not focused then the browser will deactivate some JavaScript features, including timers. This would cause the backup to fail since Akeeba Backup depends on JavaScript timers to step through the backup. For this reason we ask you to not put the backup tab to the background and keep your browser focused. If you want to avoid having your computer essentially unusable during backup please consult your browser's documentation on disabling Sleeping Tabs for your site (example: Microsoft Edge [https://www.thewindowsclub.com/sleeping-tabs-in-edge-browser-in-windows-10]). See the troubleshooting section below for further information.

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 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 **already been backed up in the previous step**. This is very important. When the backup crashes, it hasn't necessarily crashed backing up the table or directory you see on the screen. Most likely that the table/directory which has been *successfully* backed up. The real problem appears in the log file and this is why we ask you for the backup log to be uploaded with your support request. The 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.

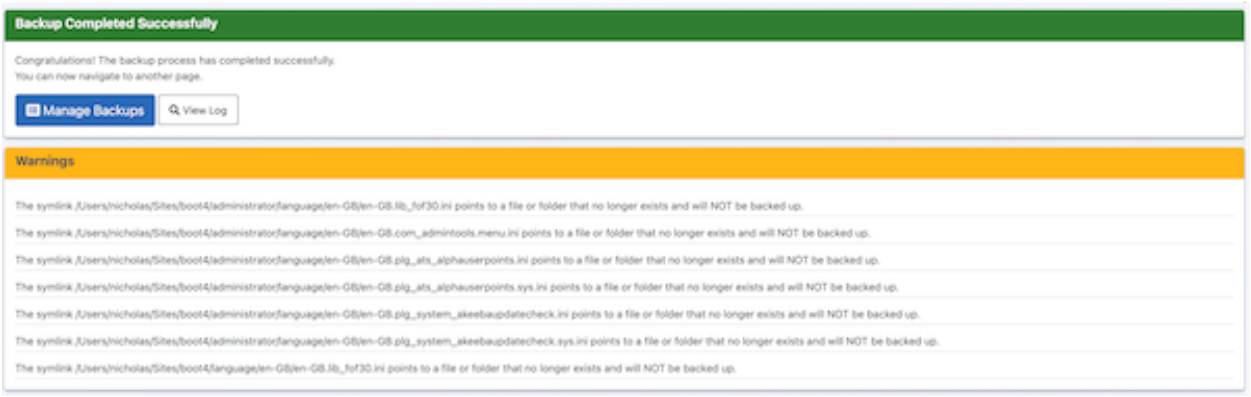
The next thing you see is time elapsed since the last server response. This resets to 0 when a new backup step is started. If you see a last server response over 300 seconds –except when the application is uploading your backup archives– you can assume that your backup has crashed. Only in this case 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.

3.4.1. Troubleshooting backup issues

We have compiled a list of frequently asked questions and their answers, as well as the basic troubleshooting instructions for a backup which appears to not be working. Please do read and follow these instructions. If you need to ask for support because these instructions didn't help you please let us know what you have already tried so we don't waste your time by asking you to do what we've written here and you've already tried.

3.4.1.1. Backup fails after switching to another browser tab, browser window or application

If a backup you start in your browser fails after you have switched to another browser tab, browser window or a different application the problem is caused by **background tab suspension ("Sleeping tabs") feature** in your browser or a third party browser extension with the same effect.

Starting with Microsoft Edge (released in late January 2021) the browser will suspend the JavaScript timers for tabs which have been in the background for a certain period of time. This is a feature meant to save battery life on mobile devices but it will also interfere with your backups if they need more than the configured period of time to complete and you switch to a different browser tab, browser window or application.

The same applies for third party extensions which promise to suspend, put to sleep or otherwise reduce memory and CPU consumption of inactive tabs.

There are two ways to work around this issue.

The simplest and *least practical* way to address this issue is to start a backup and not switch to a different browser tab, browser window or another application until the backup is complete. Make sure that your computer does not go to sleep or gets disconnected from network while the backup is running. This may be impractical for long running backups.

Alternatively, you can tell your browser **not to suspend the background tabs of your own sites**.

For **Microsoft Edge** you need to visit Options, System and click the Add button next to the Never put these sites to sleep label. Enter the domain name of your site like so: **`[*.]example.com`** where `example.com` is the domain name of your site.

For **third party extensions** you need to consult their documentation or disable them in your browser.

Please note that the nature of these features does not allow web developers, like Akeeba Ltd, to communicate to the browser or the browser extension that the tab should never be put to sleep without involving the user. Doing so would allow every site to do that, even when their use of JavaScript timers is not useful to the user (web trackers, advertisements, ...) or outright malicious (e.g. cryptominers). Please do not report this behavior as a bug. It's a deliberate choice of *your browser or the browser extension(s) you are using*. We are not in control of any of it; *you* are.

3.4.1.2. Where are my backup files?

By default they are stored in `administrator/components/com_akeebabackup/backup` under your site's root.

However, you can always set a different Output Directory in the Configuration page of Akeeba Backup. If unsure, click the info icon next to the backup entry. Akeeba Backup will tell you where your backup archive is stored in, relative to your site's root.

Also note that if you are using Akeeba Backup Professional it is possible that your backup archive is stored remotely (e.g. on Amazon S3). In this case the backup record appears with a cloud icon in the status column which is described as "Remote". In this case you can use the Manage Remotely Stored Files button in the rightmost column to manage the remotely stored files.

3.4.1.3. How can I download my backup files?

If you are using Akeeba Backup Professional and your backup is stored remotely as explained in the previous question: you need to use the Manage Remotely Stored Files button. It will allow you to fetch the backup archive back to your web server or download it directly (if supported by the remote storage engine).

If your backup is stored on the same server as your site we *very strongly* recommend using SFTP, FTPS or FTP to download it from your server. You can use a third party tool such as CyberDuck (macOS, Windows), Dolphin (Linux/KDE) or Nautilus (Linux/GNOME). If you are using FTP or FTPS please remember to set your tool to download files **in Binary transfer mode**. If you use ASCII or Auto you might end up with corrupt backup archives. If you are using SFTP you don't need to do that; SFTP always transfers files in binary mode.

If you have no access to your site's files via SFTP, FTPS or FTP you can use the Download button in Akeeba Backup's interface. However, please keep in mind that files over 10MB might end up being corrupt or truncated for reasons that have to do with third party plugins running on your site and / or your server setup. We generally do not recommend this method. If you choose to use this method ALWAYS do a test extraction of your backup archives to make sure they work. You can use Akeeba Kickstart on a local server to do that.

We also do not recommend using your hosting control panel's file manager to transfer files over 10MB for similar reasons. If you find that your backup archives are corrupt please either use SFTP, FTPS or FTP; or upload them to remote storage (e.g. Amazon S3) using Akeeba Backup Professional and download them from the remote storage using any appropriate tool, including your browser. The difference between using your browser to download files

from your site's server and downloading from the site of a remote storage provider is that the former is designed for transferring relatively small files while the latter is specifically designed and optimized for large file transfers without random, silent failures.

In either case please note that your backup archive **may consist of multiple files** with the same name and different, sequentially numbered extensions. For example, a JPA archive may consist of a file with a .jpa extension and zero or more files with the extensions .j01, .j02 and so on. A JPS archive may consist of a file with a .jps extension and zero or more files with the extensions .j01, .j02 and so on. A ZIP archive may consist of a file with a .zip extension and zero or more files with the extensions .z01, .z02 and so on. Whether this happens depends on the size of your site's backup and the Part Size for Archive Splitting in the Configuration page of the backup profile you are using. If you see multiple files you need to download all of them. You cannot restore a site if one or more of these files is missing. These are NOT separate archives; these are archive parts. Think of it as taking a printed page and cutting it in for vertical strips. If any of the strips is missing you cannot read the content of the page. You need all strips to figure out the content. An archive is like a page, a backup archive part file is like a strip of the page.

3.4.1.4. Why do I get warnings about unreadable files or folders?

You get a list of warnings during backup stating "Unreadable file /some/file; check permissions" or "Unreadable directory /some/directory; check permissions". This means that when Akeeba Backup asked PHP to list the contents of the directory or read the contents of the file, PHP replied that this was impossible. Here we will attempt to find out why and fix it.

DOUBLE CHECK A BACKUP IF YOU GET THIS KIND OF WARNINGS! These warnings mean that files or whole directories have not been backed up. It is very likely that some overly important files of your site (or even all of the files of your site!) did not make it into the archive. This will cause the restored site to be broken. That's why we call them warnings: they warn you that something may be broken so that you can investigate and make sure you fix it - or be 100% that you can safely ignore them.

Unreadable directories

A directory becomes unreadable because of its permissions. Please note that some directories are supposed to be unreadable. For example, on a live host, it is common to have directories such as `cgi-bin`, `awstats`, etc which are not part of your Joomla installation and are there to provide some service offered by your host. If you are not sure, ask your host about that. These directories MUST NOT be backed up, or restoration will be impossible. If one of those directories appears as an unwritable directory, please go to Akeeba Backup, click on Files and Directories Exclusion and exclude those directories from the backup (the leftmost icon next to the folder's name must be yellow).

If you get that kind of warning for `<root>`, it means that your site's root is not readable. This will cause Akeeba Backup to be unable to backup your site. In this case, using your favourite FTP application, connect to your site and look for your site's web root. It is usually a directory named public_html, httpdocs, htdocs, www or something like that. If unsure, ask your host. Change its permissions to 0755. If you can not do that yourself, ask your host to do that for you. If they refuse on the grounds of security, explain to them that a. it is necessary to backup your site and b. they are probably doing something wrong if other users are allowed to browse your site's files (and swiftly change to a more secure host!).

If you are on a Windows server, there is no notion of permissions. Instead, you have to edit the Windows ACLs to allow PHP to be able to list the contents of your site's directories. Since this procedure is highly dependent on the server setup, please ask your host to do it for you. If you are on a local server, please read on.

If you are on a **Linux machine**, change the permissions of the root of your website to 0755. For example, if you're trying to restore to `/var/www/mysite` you have to issue a command like **chmod -Rf 0755 /var/www/mysite**

If you are on a **macOS machine**, use the Finder to find your site's root. Right-click or Control-click on it and select Get Info. Scroll down to the Sharing & Permissions slider and expand it if it's not already expanded. Find the row where the Name column reads `everyone` and set the Privilege to `Read & Write`.

If you are on a **Windows** use Windows explorer to find your site's root. Right click on it, select Properties and click on the Security tab. Click on the Edit... button. If you can't see a user named Everyone, you will have to click on the Add... button, type Everyone in the big text box and click OK. Now click on the Everyone user and then take a look at the list of checkboxes below. Click the Accept checkbox on the Full Control row (the topmost one). Click on OK, then again on OK.

The same instructions apply for any other directory which may appear to be unwritable. Instead of your site's root, do this procedure to the directory which appears unwritable.

Unreadable files

Unreadable files are not accessible by PHP for reading. The fact that you can access them with FTP or that you can view them in your browser DOES NOT mean that they are readable by PHP. This is a common misconception. If you feel that this can't be true, please read the Security Information chapter of this documentation. We are not responsible about how your web server and UNIX-based Operating Systems work, but we can help work around the problem!

All you have to do is to find the file referenced by the warning message and give it 0644 permissions (read & write for the user, read only for the group and everyone). That's all there is to it.

Please note that some files (e.g. .ftpquota on your site's root) are not supposed to be readable. These are special files that do not belong to your Joomla installation and are put on your site by your host. These files must not be backed up, otherwise restoration will fail. Go to Akeeba Backup and click on Files and Directories Exclusion to find those files and remove them from the backup. If you are unsure whether a file is put in there by your host, Joomla or one of its extensions please ask your host. If it is a file put there by the host, exclude it. If it is not, change its permissions to 0644.

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

Before you do anything else, make sure that your server fulfils the minimum requirements of the version of Akeeba Backup you are trying to use. The PHP and Joomla! version compatibility matrix is published on our site [https://www.akeeba.com/compatibility.html].

> **Tip**
>
> If you have tried all of the above and your site is hosted on **one.com**, check out the special instructions at the bottom of this answer.

The most usual backup issues manifest themselves by means of an "AJAX Loading Error" message. This error message by itself means pretty much nothing. All it tells us is that the backup failed. Normally, you should post your log file to our ticket system so that we can take a look at it to figure out what went wrong. However, there are some common issues you can work around yourself, without looking at the log file. You should follow the following troubleshooting steps one by one until your backup works.

1. The progress bar you see always tells you *the last successful step Akeeba Backup finished executing*. That's a very common source of misunderstanding. If the last item you see on a stuck progress bar is some of the last database tables on your site it doesn't necessarily mean that the backup got stuck while backing up the database. It is usually the case that the next step, backing up the files of your site, got stuck. If unsure check the log file.

2. Are you on Windows, backing up a local site? Some antivirus and backup software may end up locking the backup archive while it's still being created, leading to an error message about the backup archive not being able to be opened for writing. One very notorious case of this kind is WD SmartWare backup software (kudos to our user, Markus, for letting us know). We strongly advise you to turn off or at least temporarily suspend any backup and antivirus software while the backup is in progress.

   Another issue on Windows is resource usage, especially on old (Windows XP) and 32-bit versions of Windows. These versions of Windows have a limited capacity of system resources, meaning that they can only keep a very

finite number of files open at any time. Combined with a bug in older versions of PHP for Windows this can lead to resource depletion and backup issues, appearing as unreadable files. Please try quitting as many applications as possible, including those running in the background (e.g. those running as system tray icons). It's also a good idea to turn off or temporarily suspend resource-intensive software such as backup and antivirus applications while the backup is running.

Also note that Windows 10 comes with Windows Defender, a combined firewall and antivirus. By default, it will be scanning each and every file that is being read or written by Akeeba Backup. If one of these files is flagged as potentially malicious (for example you have a hacked site that you may or may not know it has been hacked) it might kill the backup. If unsure, please exclude your site's root from Windows Defender's antivirus scanning. Use your favorite search engine to find instructions on how to do that.

3. Sometimes the backup seems to complete, but it hangs when uploading the backup archive to a remote location (e.g. Dropbox, Amazon S3, an FTP server, ...). If you are not sure, the hang we're describing happens *after* the progress bar hits 75%. In this case you will need to lower the Part Size for Split Archives setting. Find the Archiver Engine option of the Akeeba Backup's Configuration page and click on the Configure... button next to it. The Part Size for Split Archives setting is in the pane which opens below. Try smaller settings until the backup completes. Please note that your backup will now be split in multiple files and you need all of them to be present to restore your backup.

4. The first thing you must do is to use the Configuration Wizard button in the Control Panel page to automatically adjust a series of configuration parameters to safer settings. The wizard performs benchmarking of your server to determine those values. It is not always 100% accurate, but the settings it creates are at the very least a good starting point for tweaking them manually. If it seems to get stuck for more than three minutes (180 seconds) the first time you run it, reload the page. If it's still stuck try backing up anyway; some server setups don't like the heuristics used by the configuration wizard. However, as soon as you run the wizard it will revert your backup profile settings to safe defaults which means that you should at least be able to use it as a starting point to run a backup.

5. Make sure that your hosts PHP memory_limit is at least 32MB. Anything lower than that will most likely result to a backup failure. If unsure, ask your host; we can't know this value for sure. If this is an option, ask your host to increase the PHP memory limit. We recommend 128MB or more.

6. Try visiting the Configuration page and clicking on Save. This may be necessary if you just upgraded. This simple move will refresh your configuration and pick up the default values for any new parameters which might have been introduced in the new release.

7. Check your free space. Akeeba Backup is trying to create an archive with your entire database and all of your site's files; it needs adequate free space to do that. If you don't have enough free space, your host will kill the script in mid-process, making Akeeba Backup's interface throw this error. As a rule of thumb, we propose having about 40-50% of your account's allocated quota free.

   Some hosts claim to give you "unlimited" space, or an absurd amount like 100GB. According to our experience, they mostly lie. On many occasions the host only gave 100MB or 1GB of space. If unsure, please ask your host about the real free space you have in your account.

8. If you are using the ZIP archive format it is possible that you run into timeouts. The problem with the ZIP format is that we have to read each file twice. We read it once in order to calculate a "file signature" (properly called a "CRC32 checksum"), then we read it again in order to add it inside the archive. Unfortunately these steps can't be combined and, on top of that, the very slow signature calculation step must be able to run in one go. With larger files and slower hosts this will consistently lead to timeouts. If you suspect this is the case, please use the JPA format setting in the Archiver Engine option of the Akeeba Backup's Configuration page.

9. Some servers have a very strict limit on the maximum execution time of PHP scripts. By default, Akeeba Backup is configured with a maximum execution time allowance of 14 seconds. In order to work around such hosts, please go to your Akeeba Backup Configuration page and scroll all the way down to the Fine Tuning pane. You will find an option labeled Maximum Execution Time. Select the "Custom..." option and type in 5 in the text box that appears

to the right of the drop-down. Click on the Save button and retry backing up your site. WARNING! NEVER SET THE Minimum execution time TO A VALUE HIGHER THAN 2 SECONDS UNLESS EXPLICITLY ASKED TO DO SO BY THE SUPPORT STAFF, OR YOU WILL GET A BACKUP FAILURE.

We have heard of hosts which require settings even lower than that. If in doubt, ask your host what their PHP maximum_exec_time setting is, then subtract one second and use this value in Akeeba Backup's Maximum Execution Time setting.

If you still have issues and you are a paying subscriber with an active subscription to Akeeba Backup Professional please file a support ticket. For your convenience, please make sure you indicate that you have gone through the steps on this page when posting your support request to avoid a canned reply that you should check this page first. Thank you.

Special notes for one.com customers

If your site is hosted on one.com please follow these instructions if the above doesn't work:

1. Run the Configuration Wizard but do not take a backup yet
2. Go to Akeeba Backup's Configuration page and set the following parameters:
   - Logging Level: Only errors (however, if you want to ask for support in our ticket system you must switch it back to All Information and Debug and try taking a new backup before posting us your log file)
   - Click the Configure button in the Archiver Engine and set the "Part size for split archives" to 127Mb or less.
   - Minimum execution time: 10 seconds
   - Maximum execution time: 7 seconds
   - Execution time bias: 50%

Please note that the above settings do cause the backup to run at half speed, but this is required for your backup to run under smoothly on one.com. We'd like to thank one.com's tech support for testing and providing these safe settings.

3.4.1.6. My backup files are not being uploaded to Amazon S3

Before you even begin thinking why the upload failed, please make sure that you have the PHP cURL extension installed and activated in your server's php.ini. If you are unsure, please check the System, System Information page of your site. If you see a cURL section in the PHP Information tab then cURL is installed and all is fine. If you don't see anything about cURL being mentioned in there, well, you have to install and enable it on your server. Ask your host or server administrator about it.

The next thing you should do is a sanity check. Make sure that you have copied your Access Key and Secret Key correctly. Especially the latter, may have up to two equal signs at the end. These are required. Then, check your bucket name. The bucket must already exist; Akeeba Backup can not create it for you. The bucket must also be writable by the user whose Access and Secret Keys you're using; you can check that in the Amazon S3 Console. Moreover, do note that the bucket name is case sensitive. This means that Abc, ABC, abc and AbC are four *different* bucket names, as far as S3 is concerned.

Warning

Amazon warns AGAINST using uppercase letters in your bucket names. If you use an uppercase or mixed case bucket name it is possible that you will get an error message stating that the signatures don't match, or that the bucket does not exist. If this is what you get, please try creating a new bucket with lowercase-only letters **BEFORE** trying any of the following instructions.

If you are using the V2 signature **and only then**, let's try making sure that the SSL setting doesn't cause the problem. First go to Akeeba Backup's Configuration page and find the Data Processing Engine drop down. Click the Configure button next to it. A new pane opens below. Find the Use SSL checkbox and make sure it is not checked. Try a new backup. If your bucket name contains dots you MUST disable the Use SSL checkbox. This is a limitation of the Amazon S3 service due to the way their SSL certificate is set up. It is best to have a bucket name which consists of only lowercase letters (a-z), numbers (0-9) and dashes and which does not begin with a dash.

The other thing you have to check is that your host's firewall allows access to Amazon S3. Ask them if they have a firewall which blocks outgoing connections. In this case, please tell them to allow TCP/IP connections to ports 80 and 443 of `s3.amazonaws.com`. If they request an IP, please tell them that this domain name is a multicast one and they have to run `host s3.amazonaws.com` from their server to obtain the IP. It doesn't matter if this sounds like Chinese to you, your host's support technicians will understand (or should, at the very least).

Finally, some hosts do not play very well with Amazon S3's multi-part upload feature which allows us to upload a very big archive file in 5MB chunks. In this case you will have to follow Plan B which is to have Akeeba Backup split the archive file in small chunks, one file per chunk, and then upload each of those chunks in one go. This is a two-legged solution.

For the first leg of the solution, please go to Akeeba Backup's Configuration page and find the Data Processing Engine drop down. Click the Configure button next to it. A new pane opens below. Check the Disable multipart uploads option and make sure that the Process each part immediately option is not checked.

Now, for the second leg, we have to do some trial and error. Still in Akeeba Backup's Configuration page, find the Archiver Engine drop-down and click on the Configure button next to it. A new pane opens below. Find the Part size for split archives option and select the `49.99` option. Try a new backup. If it crashes while uploading files to Amazon S3, go back to this option and try smaller values, i.e. `20`, `10`, `5`, `2` or even `1`, trying a new backup after setting each one of those values.

In very rare cases (less than 3 tickets a year) we have seen hosts which have a transparent proxy in front of the web server. This proxy gets in the way of Akeeba Backup connecting to Amazon S3. There are two distinct problems with such a setup. If the proxy caches the responses from Amazon S3 – it should NEVER cache them due to their headers, i.e. that's a proxy configuration problem – the proxy responds instead of Amazon S3. From Akeeba Backup's perspective the file transfer went through but in reality nothing was uploaded. There is objectively no way for Akeeba Backup to detect the proxy and the proxy should never be caching those requests. Contact your host and ask them to fix their proxy. The other possible problem is that the proxy is stripping some of the request headers such as the Content-Length which is mandatory for uploading files to Amazon S3. You will get an error from S3 claiming the header is missing. No, it's not a bug in Akeeba Backup. We do send the header but your host's proxy is stripping it. It's a bug in your hosting setup. Again, please do contact your host to fix it.

Do remember that the overwhelming majority of the problems you will ever experience with any kind of upload to remote storage with Akeeba Backup are not issues in Akeeba Backup itself but something in your hosting or server setup. Thank you in advance for understanding that we objectively have no control over it and working with us so we can get your host to fix their infrastructure.

3.4.1.7. How do I know that my backup archive works?

As the saying goes, the proof in the pudding is in eating it.

In fact, we very strongly recommend testing your backup files, at least the first time you produce a backup and whenever you make major changes/upgrades to your site. Testing the backup archives is relatively easy. Download and install a local web server package on your computer to create a local testing server environment. Then try restoring your backup file on the local testing server environment.

You can consult our video tutorials and our documentation for information on restoring your backups.

3.4.1.8. What happens if I have a backup or restoration problem?

If you run into problems please consult our documentation first. We distill our experience in providing support in our documentation to make it easier for you to find solutions and for us to point you out to tested procedures to identify and address problems without having to type everything from scratch every time someone has a problem. If you are unable to find a solution to your problem you may need to file a support ticket. Filing a support ticket is only possible if you are a paying subscriber with an active subscription to Akeeba Backup.

If you file a support ticket we need the following information from you to provide fast and accurate support. Please remember that we're not sitting behind your shoulder nor can we read your mind. Our replies are based on the information you provide.

- A clear description of your problem. Tell us exactly what you were trying to do, at what exact operation it failed and any error messages which were output on the page. If you can get a screenshot of the error message that's even better. If you are not sure if you're describing the issue adequately, read what you wrote to someone who is not familiar with your issue. If they didn't understand you chances are we won't either. If you do not feel comfortable with your level of English try to provide screenshots or a short (less than 1 minute) video.

- Exact version of Akeeba Backup. It's visible on the right-hand pane in Akeeba Backup's Control Panel page.

- Exact Joomla, MySQL and PHP version. You can find all of these version numbers in Joomla site's System, System Information page.

- A copy of the debug log taken with log level set to `All Information and Debug` when the problem happened. You can download a copy of the log from within the View Log page. Please do not copy and paste directly from the log viewer, this will not help us. Please remember to put the log file inside a ZIP archive and attach the ZIP archive to your support ticket. If the ZIP file is over 2MB please upload it to Dropbox, Google Drive or OneDrive and paste a share URL to it in your ticket. We kindly request that you do NOT try to attach the raw log file with a .php extension or a ZIP file containing it; these files will be automatically rejected by our site for security reasons.

- If you are writing about an error related to the restoration process do not forget to tell us how you extracted the archive file, i.e. using Kickstart or using Akeeba UNiTE. Also do tell us about the method you used to download the backup file to your PC - if applicable - and the method you used to upload the backup file to your host, for example "I used FTP in Binary mode", "I just clicked on the Download button in Akeeba Backup". Remember to mention the exact version of Kickstart you are using; it's printed with very big letters on the top of Kickstart's page.

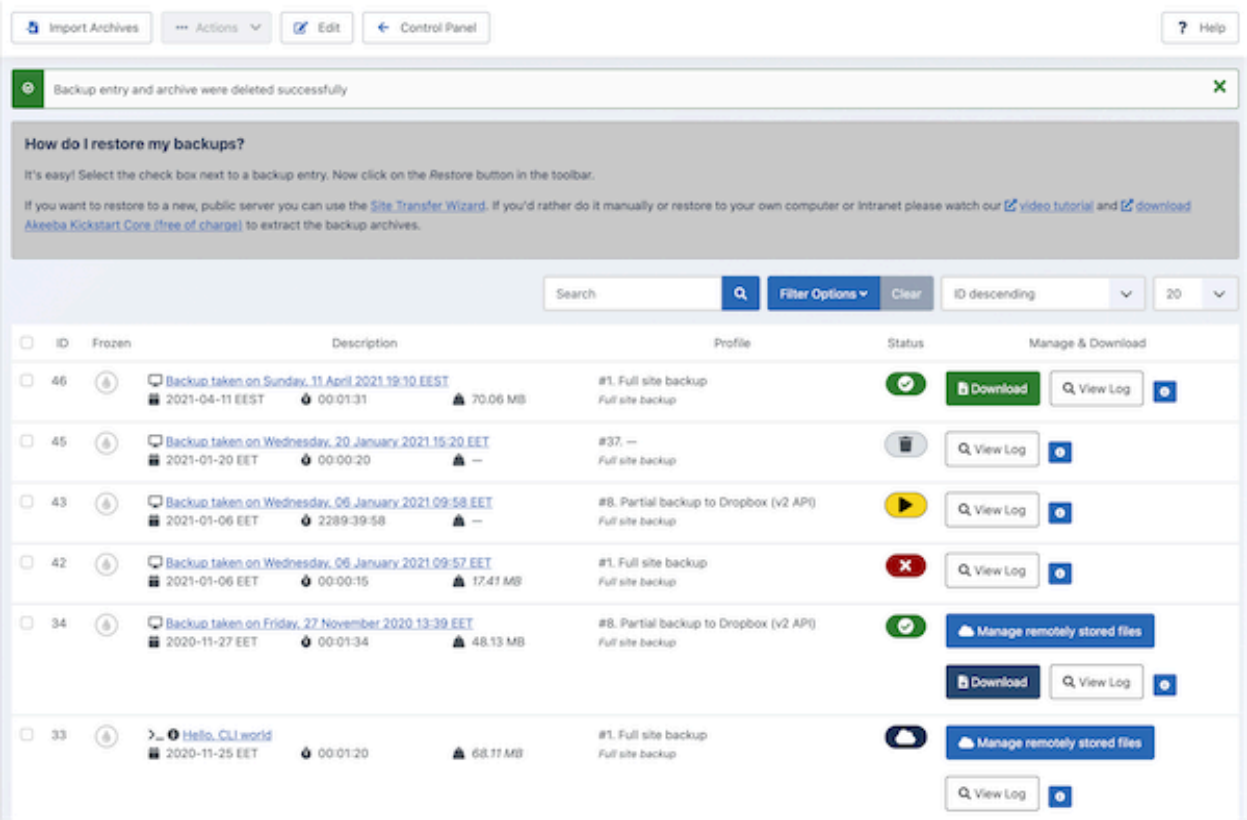
  Finally, do not tell us that "Kickstart x.y.z stuck in the database restoration page" because such a thing is impossible ;) Kickstart extracts the archive. Then it launches ANGIE, the restoration script included in your archive. It's ANGIE restoring the database. Since ANGIE is in the archive, we need to know which version of Akeeba Backup produced the archive. The Kickstart version is completely irrelevant in this case. We need the ANGIE version which is printed in large type at the top of the restoration script's page.

Please try to give us raw information, what you did, what happened, what you saw; not your interpretation of what happened. In most cases it follows that had your interpretation of what happened been correct you would already know how to solve the issue and you wouldn't be filing a ticket. If we have to reverse engineer the facts from your interpretation of what happened it's more than likely that we'll make arbitrary assumptions which are wrong, leading to a mutually frustrating support experience.

Finally please bear in mind that support is provided by the same developers who have written and maintain the software. On the upside this means that you get to talk with people who really know what they are doing. On the downside this means that replies come slower since we manage both the code and the support. We kindly ask for your patience and understanding while we work on your and other's issues and our software. Also bear in mind that we are humans and do need to eat, sleep, bathe and take time off to be with our families. We post our working hours on our site. If you file a ticket outside these hours it will take a bit longer to get a reply. Thank you for your understanding!

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.

ID
: This is a unique numeric identified of the backup attempt. You will only ever need to use it if you are using the Akeeba Backup JSON API or the Joomla! command line client integration to manage your site.

Frozen indicator
: You can mark important backups as "frozen", creating a protected record. Once a record has been marked as frozen, the following will happen:

  - Its files cannot be deleted with the Delete Files button
  - It cannot be deleted with the Delete button
  - For remote backups, you cannot delete the remotely stored archives
  - It doesn't participate in any quotas, local or remote including the obsolete records quotas

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 / time format, timezone and timezone suffix are configured in the component's Options page.

  In the same place below the description you will also see the duration of the backup in Hours:Minutes:Seconds format, as well as the size of the backup.

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.

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

  OK (Checkmark on green background)
  : A complete backup whose backup archive is available for download.

  Obsolete (Trash can on gray background)
  : 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.

  Remote (Clould on blue background)
  : 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.

  Pending (Triangle on yellow background)
  : 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 (Big X on red background) — A backup attempt which failed to complete.

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!

  ## Important

  Some backup archives may consist of more than one files. These are called multipart backup archives.

  You MUST download all part files to have a backup archive set which can be used to extract all files and restore a site.

  All files have the same base name, for example site-www.example.com-20220925-105300-abcdef012345, but different extensions. JPA archives have the extensions .jpa, .j01, .j02, …; JPS archives have the extensions .jps, .j01, .j02, …; ZIP archives have the extensions .zip, .z01, .z02, …

  The part files of a multipart backup archive are NOT separate archives; you cannot extract its one of them individually. The are all parts of the same archive and they all need to be present for the archive to be able to be extracted. The order of the parts is .j01, .j02, …, .jpa for JPA archives; .j01, .j02, …, .jps for JPS archives; and .z01, .z02, …, .zip for ZIP archives. That is to say, the numbered part files come first in the numeric order defined by their extension, the file with the .jpa, .jps or .zip extension comes last.

  You can extract multipart archives using the integrated restoration but also Akeeba Kickstart Core, our free of charge archive extraction tool. It can run on a web server — even a local web server created with MAMP, WAMPserver, XAMPP etc — or, for expert users, on the command line.

- 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 Import Archives button (Akeeba Backup Professional only) takes you to the Import Archives page which allows you to import any ZIP, JPA or JPS file, located anywhere in your server or Amazon S3, in the Manage Backups page in order to restore it on this or any other site.

The Actions drop-down button contains the actions which apply only on the selected backup records:

- Freeze record. Marks the selected backup records as frozen.
- Unfreeze record. Marks the selected backup records as no longer frozen.
- Restore (Akeeba Backup Professional only). Runs the integrated restoration on the first selected backup record. This feature can be used to restore your backup archive on the same server you backed up from or even a different server.

  If you are using Akeeba Backup Core you can always restore your backup archives using Akeeba Kickstart. Please consult the video tutorials on our site.
- Delete files. Removes any backup archives and log files from your server. If you are using Akeeba Backup Professional, it does not delete any archive files stored remotely; you need to use the Manage remotely stored files button instead.
- Delete. Same as Delete files but also removes the backup record itself from the database.

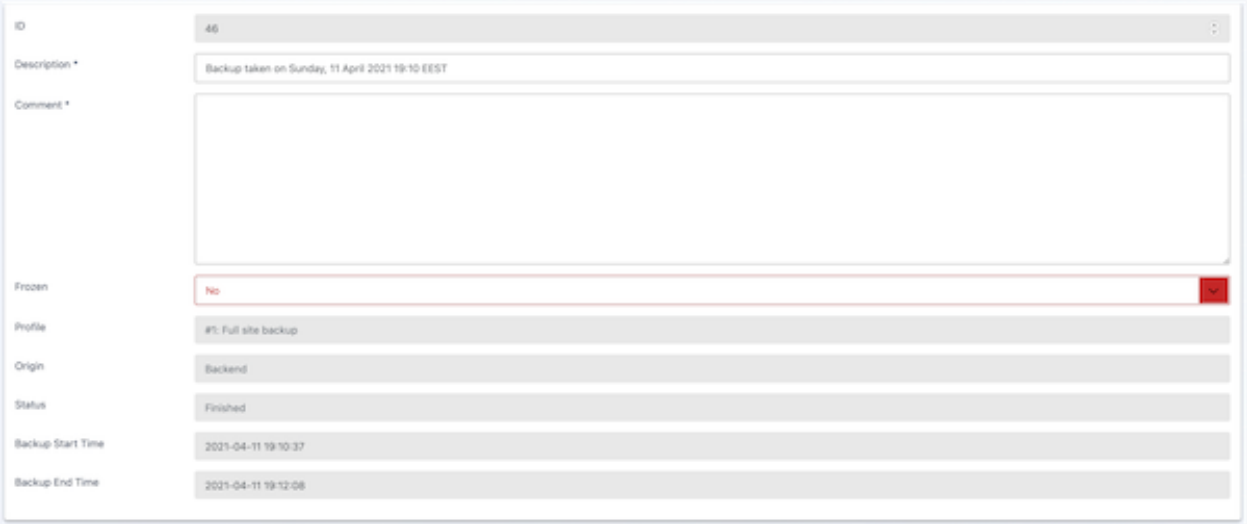
Note

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 to extract the archive and restore it on their server. The procedure is detailed in 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 to easily restore their backups using Kickstart.

Backup description / comment editor



Clicking on the description of a backup record or selecting it and clicking on the Edit button in the toolbar will open the View / Edit backup comment page.

You can see some basic information about the backup record you are editing. You can change its description, comment and its Frozen status.

3.5.1. Integrated restoration

Warning

Using the integrated restoration you are OVERWRITING your site with the one contained in the backup archive. The same precautions apply as with any backup restoration.

Note

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

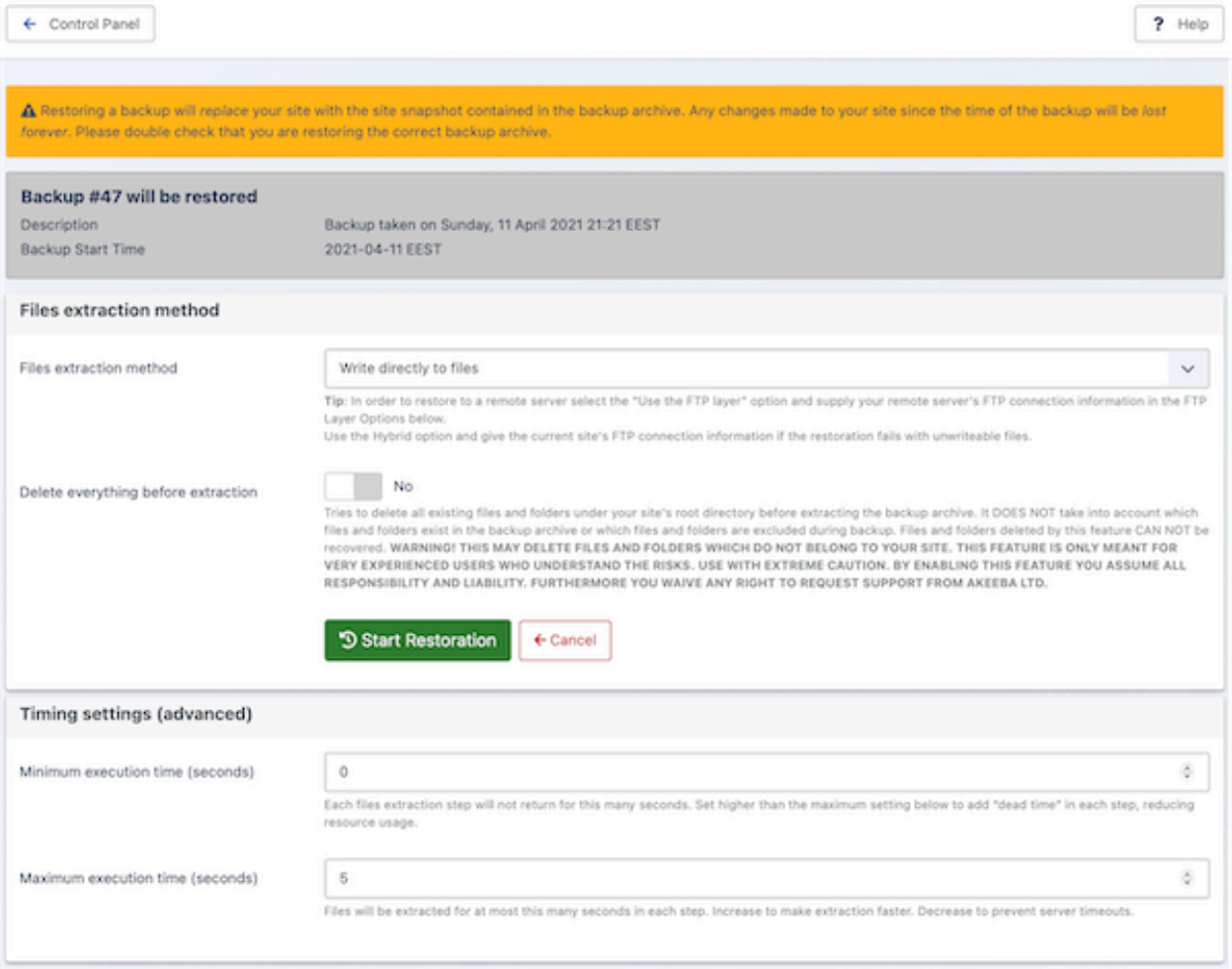
Restoring a backup from inside Akeeba Backup itself requires that the backup is of the type “Full site” or “All configured databases (archive file)” and that the backup archive is stored on your server (not in remote storage). If your backup archive is stored remotely you will need to fetch it back to your server using the Manage remotely stored files feature documented further below.

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 (ANGIE) to complete the restoration procedure.

The communication between your browser and the archive extraction script uses a randomly generated secret key which prevents random people from abusing this feature to upload arbitrary code to your site and/or interfere in the backup restoration process. They secret key is only active during the restoration. When the file with the secret key is not present the extraction script is completely inert i.e. it refuses to accept any commands. The key is transmitted verbatim. As a result you need to use HTTPS on your site and a trusted connection (your home / office Internet connection or a VPN as opposed to a public, shared Internet connection) for security resons.

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 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 extracted, 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). 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.

The Delete everything before extraction option only appears if you have enabled the respective option in the component's Options page. This is a VERY dangerous options which tells Akeeba Backup to try and delete all existing files and folders under your site's root directory before extracting the backup archive. It DOES NOT take into account which files and folders exist in the backup archive or which files and folders are excluded during backup. Files and folders deleted by this feature CAN NOT be recovered.

Warning

THIS MAY DELETE FILES AND FOLDERS WHICH DO NOT BELONG TO YOUR SITE. THIS FEATURE IS ONLY MEANT FOR VERY EXPERIENCED USERS WHO UNDERSTAND THE RISKS. USE WITH EXTREME CAUTION. BY ENABLING THIS FEATURE YOU ASSUME ALL RESPONSIBILITY AND LIABILITY. FURTHERMORE YOU WAIVE ANY RIGHT TO REQUEST SUPPORT FROM AKEEBA LTD.

If you chose to use the FTP or Hybrid 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`, `components`, `language` 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.

At the bottom of the page you will find the Timing settings (advanced) settings. These settings control the extraction process and work the same way as the respective settings for taking a backup. If you find that the extraction fails you may want to use a minimum execution time of 7 seconds and a maximum execution time of 2 seconds (yes, max is less than min) to create a roughly 30% duty cycle.

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_akeebabackup/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 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_akeebabackup/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. You can change these permissions after the restoration is over, of course.

   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 0755 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 Restoration Script page on a new tab or window. DO NOT CLOSE THE INTEGRATED RESTORATION PAGE'S TAB/WINDOW! After you have competed the Akeeba Backup Restoration Script 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_akeebabackup/restoration.php` setup file to nullify the, already non-existent, potential risk of a malicious user abusing this script.

3.5.2. Manage remotely stored files

Note

This feature is only available in the Akeeba Backup Professional edition

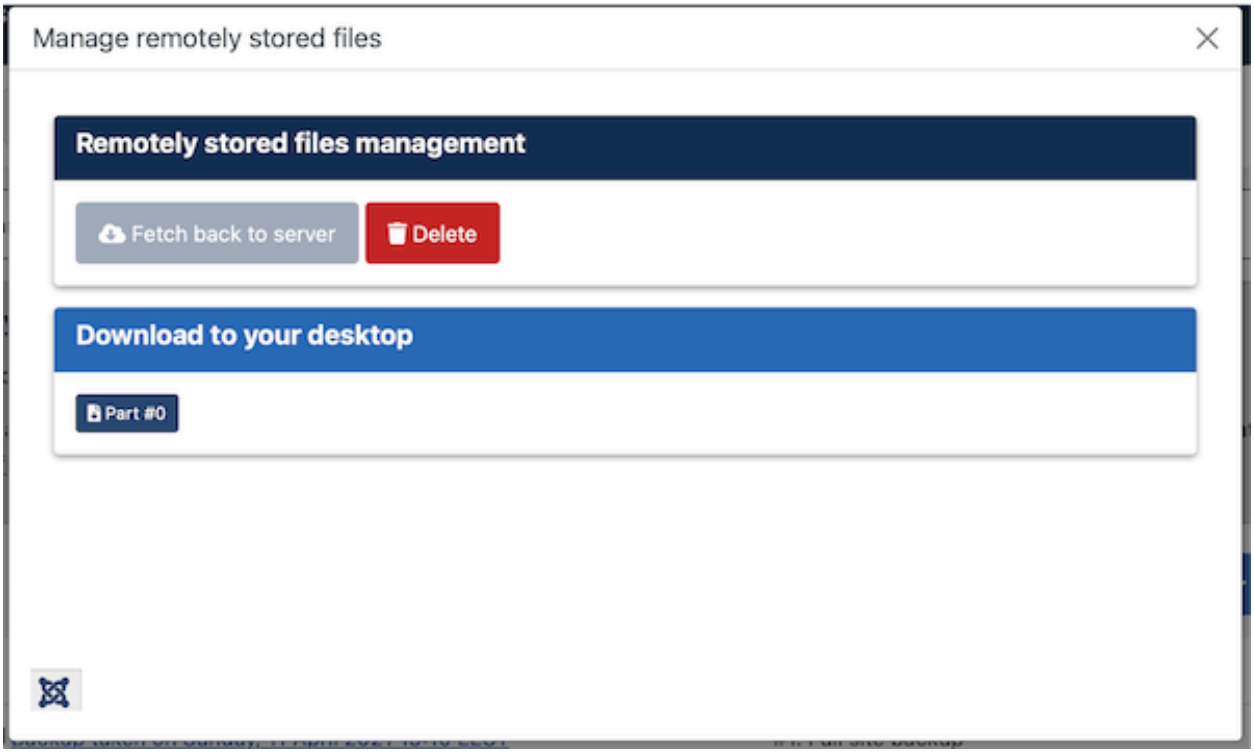
Akeeba Backup Professional allows you to set up so-called post-processing engines. They tell Akeeba Backup to upload the backup archives to remote storage such as cloud storage providers, FTP/SFTP servers and so on. Depending on the post-processing engine you have you can manage the files stored on the remote location. 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 page. Clicking on that link opens a 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.

Important

If you change the remote storage settings in your backup profile — e.g. a different remote storage provider or storage account (which doesn't have access to the same files), a different bucket / container / drive, a different directory etc — the Manage Remote Stored Files actions may fail. Akeeba Backup only stores the type of remote storage engine used and the path to the file in the remote storage in the database, not the settings to connect to the remote storage provider.

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 is typically necessary when you want to restore a backup archive: you can only restore backup archives stored locally on your server. Please note that this requires download privileges for the third party storage service user you have set up in the backup profile. If you are using a service with fine-grained access control such as Amazon S3 it is possible that this will fail. 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. This is not available for all storage engines. For the FTP and SFTP engines this may fail, depending on the capabilities of your browser (for example: as of 2020 Google Chrome has removed support for FTP and will fail to download files over FTP). 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 telling you that no actions are avaialble. 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. When you to that your browser will *reload the Manage Backups page*. While the page is reloading the changes you made WILL NOT be visible. This is not a bug, it is the way it is meant to be and the reason why the page is automatically being reloaded.

3.5.3. How the Manage Backups page works with local and remote backup archive files and where quota management fits in

This is a longer page explaining the non–obvious connection between what you see in the Manage Backups page, locally and remotely stored files and the Quota settings in the Configuration page of each backup profile. This is based on numerous conversations we've had with clients, taking into account their challenges, questions and considerations. We hope it will shed some light on why certain things are implemented they way they are and exactly how they work.

The Manage Backups page lists backup attempts, not files

Akeeba Backup keeps a history of all of your *backup attempts* in your site's database. This is what is displayed to you in the Manage Backups page, not the backup archives on your server or remote server.

There are several reasons for this. Having the information of all in–progress, successful and failed backups allows to implement features beyond taking a backup from the backend of the site where we are guaranteed to have a working user session. For example, this allowed us to create the front–end backup URL and the Akeeba Backup JSON API which can be used to schedule backups using services external to your site, running without the need for a logged in Joomla user.

Furthermore, this implementation is much better for performance. Querying the database records for a thousand backups takes less than 0.2 seconds. Querying the local filesystem for this many backup records takes about 1 second. Querying a remote storage service takes several minutes. The only way to have a workable Manage Backups page is to query database data; anything else would either time out (making the page completely unusable) or would be so slow that it would be practically unusable.

Local vs Remote files

Each backup record saves, among other things, the directory on your server where the backup archive was created in, the base name of the backup archive and the remote path (remote storage service type, path and filename); the latter is stored only if you have set up a Post-processing Engine other than None in the Configuration page and only applies to Akeeba Backup Professional.

By default, Akeeba Backup will create your backup archive on the same server the backup is running. This is intentional. Creating a backup archive requires being able to not only append to the archive file but also go back and change things towards the start of the backup archive (the very beginning of the first part) where the total number of files, the total compressed and uncompressed size of these files and whether this is a multi–part backup archive information is stored. Both of these features are only supported when dealing with files on your server, i.e. **local files**.

The backup archive file or, in the case of a multi–part backup archive, the part files of the backup archive file can be optionally transferred to remote storage based on your configured Post-processing Engine options. The files stored outside of your server are called **remote files**.

It is possible for a backup record to have only local files (its status is shown as OK), only remote files (its status is shown as Remote), both (its status is also shown as OK) or neither (its status is shown as Obsolete).

Whether local files exist is something which is determined every time you access the Manage Backups page. As noted above, this is a very cheap operation which only takes fractions of a second.

Whether remote files exist is something which is ONLY updated when you manage those files through Akeeba Backup. That is to say, if you delete your remotely stored backup archive files outside of Akeeba Backup your backup record in Akeeba Backup's Manage Backups page will STILL appear as Remote, indicating there are remote files. This is not a bug. As explained, trying to do otherwise would take a very long time and make the Manage Backups page unusable.

Deleting files: local vs remote

The Delete Files button in the Manage Backups page only deletes the locally stored files. Likewise, the Delete button in the Manage Backups page will only delete locally stored files and the backup record itself; it won't touch the remotely stored files. This happens for several reasons.

The least important of all reasons is that deleting a remotely stored backup takes a lot of time. If you have a remotely stored backup with dozen of parts it would time out. That's why the Manage Remotely Stored Files page has this large spinner and reloads several times when you have a remotely stored backup archive consisting of many parts (therefore many files which need to be deleted one at a time).

However, this is not the most important reason. That could be worked around. What cannot be worked around is the need to manage locally and remotely stored files separately from each other and remotely stored files separately from backup records. It's easier to give you two common use cases.

You start taking a backup. The backup does complete successfully but the files are not transferred to remote storage because of a temporary network issue, the remote storage service is down or you had used the wrong credentials. At this point the backup archive is stored locally. You can rectify the error and use the Manage Remote Files to upload the backup archives to remote storage. Now you have both local files and remote files. One of the points of storing backups remotely is that you don't want them taking up space on your web server. If you cannot delete the locally stored files without deleting the remotely stored files you'd have to use FTP/SFTP to do that, therefore Akeeba Backup *would not* be the one and only administration interface you need for your backups which beats the reason for having it in the first place. Therefore, having local and remote files managed separately solves this problem.

It's also possible that you want to delete the backup record from your database but keep the backup archive stored remotely. Here's an example from our own use case. Before July 2019 we were issuing invoices ourselves instead of using a third party service. We are legally required to keep copies of these invoices for at least seven years, preferably ten (7 is the minimum set by law, 10 is the recommendation of the tax office in case they have a case that dragged on due to bureaucracy). We need to keep that July 2019 backup until at least 2029. However, we don't want it to be listed in our backend to prevent accidental restoration to a version of the site that won't even run correctly on our current server.

There are many use cases like that where people want to keep an old version of the site in remote storage but not let another less experienced administrator try to restore it, breaking everything. The way to handle it is deleting the backup record *without* deleting the remotely stored files. If we deleted the remotely stored files when deleting the backup record you'd have to use a database client such as phpMyAdmin to achieve that which would again be a problem as you cannot manage your backup records with just Akeeba Backup.

Having backup records and local files managed independently from remotely stored files solves these problems.

As to why locally stored backups are deleted together with the backup record, it once again comes down to the same requirement to have Akeeba Backup as the sole tool necessary to manage backup records and backup files. If you delete a backup record without deleting the locally stored file you won't know that there's something taking up space or how to delete it.

You might wonder, why not do that for remote stored files too? Apart from the use cases we gave you above, there's a big difference to the context of locally and remotely stored files. Locally stored files take up space on your web server where space comes at a premium. Remotely stored files are stored on far cheaper external storage. Locally stored files are only required if you want to restore your site (i.e. they are “hot” files). Remotely stored files are used for archiving backups in case you need them (they are “cold” files). It makes sense to treat hot and cold file storage pools differently since the context of storing a file in each one of them is different.

You can of course delete the remotely stored archive files or move files between local and remote storage using the Manage Remotely Stored Files in the Manage Backups page. That's why this feature exists. You can delete remote files, download remote files back to your server (remote files to local files) and uploading archive files stored on your server to the remote storage (local files to remote files).

Quota management

Akeeba Backup offers a feature called Quota Management. It can automatically delete old local and remote files you don't need at the end of the backup process.

Quota management works on the *backup records* in the database, not the backup archive files. In other words, Akeeba Backup will NOT check the contents of your backup output directory for local files or the folder of the configured remote storage for remote files when processing quotas. It will trust what is stored in its database. There are, again, many reasons for that.

Beyond the performance issues we already mentioned, it is possible that different backup profiles or even *different sites* share the same backup output directory or remote storage directory. If quotas were applied on files stored on disk / remote storage you would never be able to have different quota settings per backup profile or even different quota settings across files. According to our experience this would lead to backup archives you need to end up getting automatically deleted. A backup gone when you do not expect it is a disaster waiting to happen. That's another reason we erred on the side of caution.

Quotas are not a substitute for manual management. They are an optional feature on top of manual backup archive management. They allow you to implement a custom retention policy for backup archives and backup records.

In fact, backup files quota management and backup record quota management are separate and work independently. Yes, this means that you can shoot your feet since quota management requires the backup records of said backups to exist in the database to figure out what to remove and you can accidentally configure Akeeba Backup to remove these records, essentially making the quotas never fire. There is method to what on first glance appears to be madness. We implemented it this way because it also gives you far greater flexibility. In the end of the day we follow the time–honored UNIX dogma that the user is a responsible adult in ultimate control. Our software won't question your intentions, it will execute your instructions with precision and efficiency.

For example, you may choose to retain remotely stored backup archives for the last 30 days and the 1st of each month but also remove backup records older than the last 45. Assuming daily backups, this lets you see when backups were taken in the last month and a half, making sure backups were not skipped due to a scheduling problem. It also lets you keep the first of the month backup archives (typically, for regulatory reasons) without seeing them in the Manage Backups page (so you don't accidentally mess up and restore a backup that shouldn't be restored).

3.6. 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, 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, there are use cases where this may not be convenient, e.g. when you want to make a backup available for restoration by a non-technical client who can't reasonably use FTP. This is where Import Archives comes in.

This feature allows you to find archives stored anywhere on your hosting account and import them as backup records in Akeeba Backup, visible in its Manage Backups page. This means that you can upload backup archives anywhere in your site's folder structure, or even in a directory outside your site's root and Akeeba Backup will be able to import

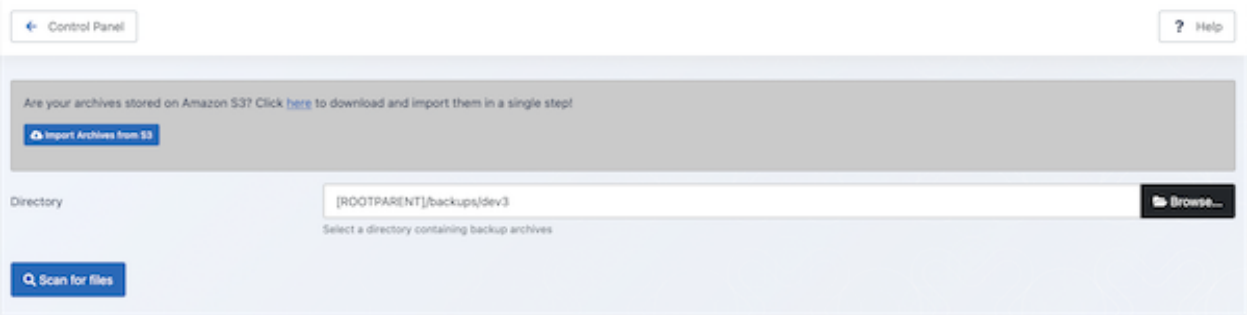
them. All backup archives are imported as backup records of the default backup profile (profile with ID #1) and can be restored just like any other backup archive.

In order to launch this feature, go to the Manage Backups page and click on the Import archives button in the toolbar. A new page appears which lets you select a directory. Alternatively, go to the Control Panel page and click on the Import archives button.

Tip

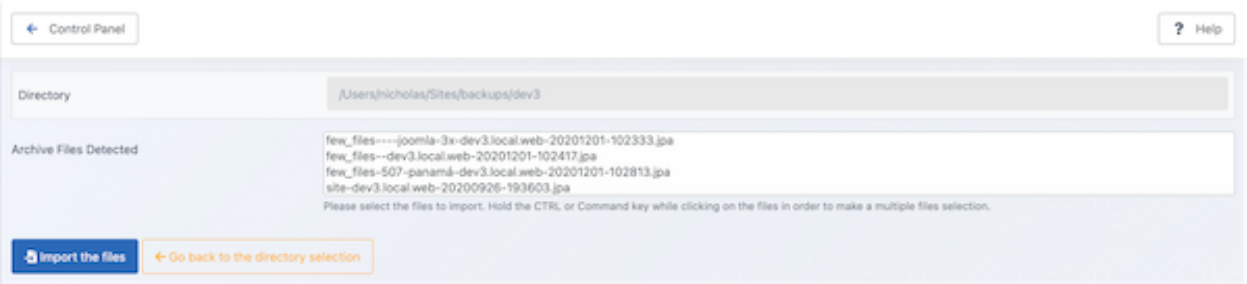
You also have the option to import archives from Amazon S3. Click the Import Archives from S3 button directly above the directory selection box. It will take you to a slightly different page where you can enter the connection credentials to your S3 account and allow you to browse for ZIP, JPA or JPS files to import.

The "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.

3.7. Import archives from S3

This feature works very similarly to the discover and import archive feature. The main difference is that whereas the import archive feature is used to import archives stored on your server, the Import archives from S3 feature is used to import archives stored in the Amazon S3 storage service, even if you haven't used the application to store them there.

Start by entering your Access Key and Secret Key in the fields and click on the Connect to S3 button. The page reloads and you see a new drop-down listing all your S3 buckets. Select a bucket and click on the Change bucket button. You now see a list of subdirectories and files in your bucket. Just navigate inside the directory containing your backup archives and select the backup archive you want to import. That's it!

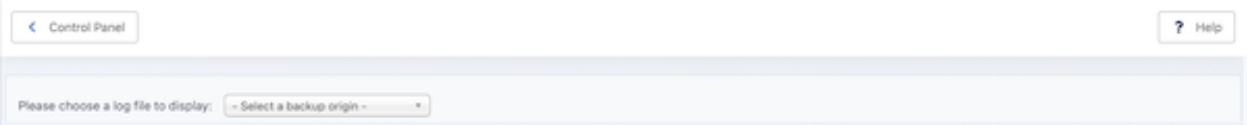
3.8. View Log

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

Note

The verbosity of the log file is controlled by the Log Level option in the Configuration page. If you had set this to a lower level (None, Errors only, Errors and Warnings) it may not be very useful. In this case you need to set the log level to All Information and Debug and retry your backup to collect more information.

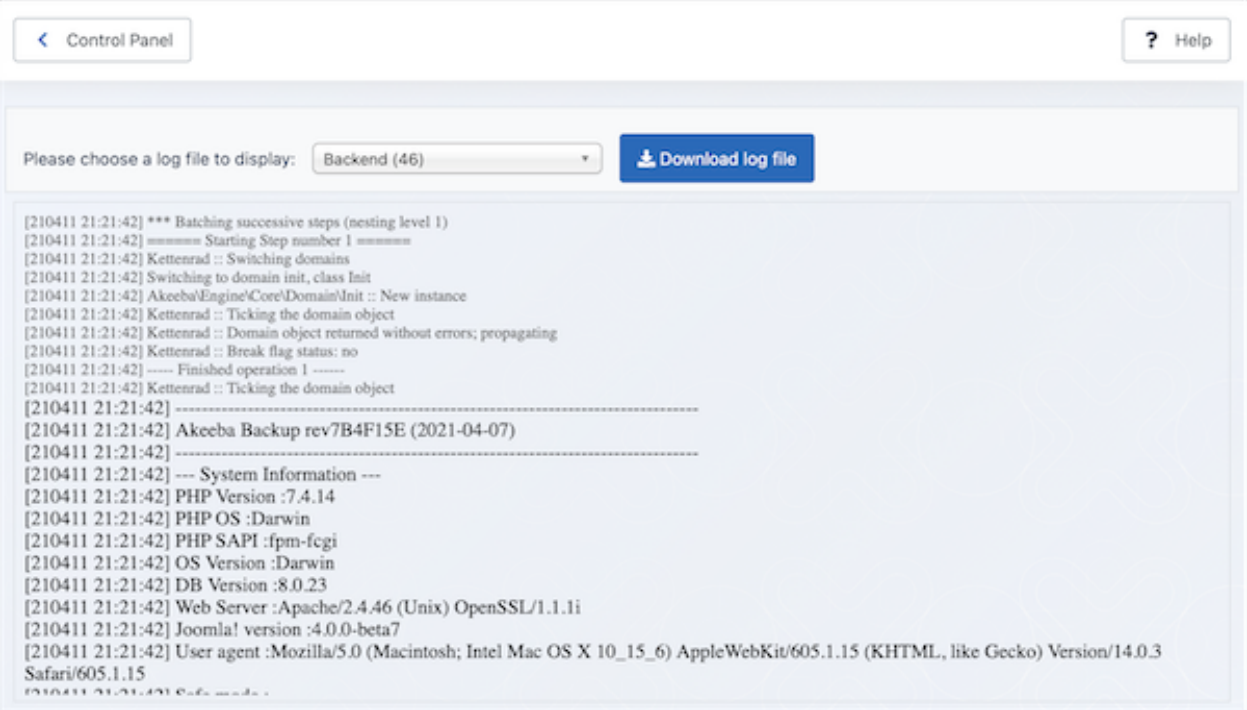
Selecting an origin



The first page allows you to select a log file to display. Each option in the drop-down is a backup origin followed by the backup ID in parentheses. If you are not sure which backup is the one you're looking for you can always go to the Manage Backups page, find the backup there and click the View Log button next to it. This way you will bypass this selection 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 Log File button. Please do not copy and paste the text appearing in the log viewer, do not take a screenshot, do not save the page source as a PDF. We need the text log file to help you.

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 not help us help you.

If the backup log file is too big you will see warning about it and you will need to click on the Show log button to display the log file. Once you do that the message will be replaced by the log viewer areas. 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). The time stamp is in the GMT timezone. 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 the information in the log 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.

Tip

If you have a failed backup but do not understand what the log file tells you, Akeeba Backup can try reading it and tell you what it thinks is going on. Just go to the Control Panel page, click on Troubleshooter - ALICE and select the log file with the same ID as the one you were viewing in the View Log page. Akeeba Backup will figure out what the log means and give you a better idea of what is going on. Please note that you should only do that for failed backups and only if the Log Level in the Configuration setting is set to “All Information and Debug”. Trying to process successful backups or backups with a lower log level will confuse the troubleshooter and the response you get will be useless and invalid.

4. Include data to the backup

Note

This feature is available only in Akeeba Backup Professional.

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 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.

Important

You DO NOT need to add your site's root or your site's database to the backup. These are always added automatically and they are not listed in the Multiple Databases Definitions or Off-site Directories Inclusion pages.

4.1. Multiple Databases Definitions

Note

This feature is available only in Akeeba Backup Professional

Sometimes your site may use more than one databases. Taking a full site backup requires backing up those external databases too. Normally, Akeeba Backup only backs up the tables from your Joomla! database. The solution to this problem is the Multiple databases definitions option of Akeeba Backup. You can define an unlimited number of additional MySQL databases which will get to be backed up (and restored) along with your regular Joomla! database.

Important

Please note that you do not need to and, in fact, must not use this feature to add the main database of your site (the one used by Joomla! itself). The main database of your site is backed up automatically. If you ignore this warning and add your main database as an additional database in this page you **will cause errors during the restoration of your site**.

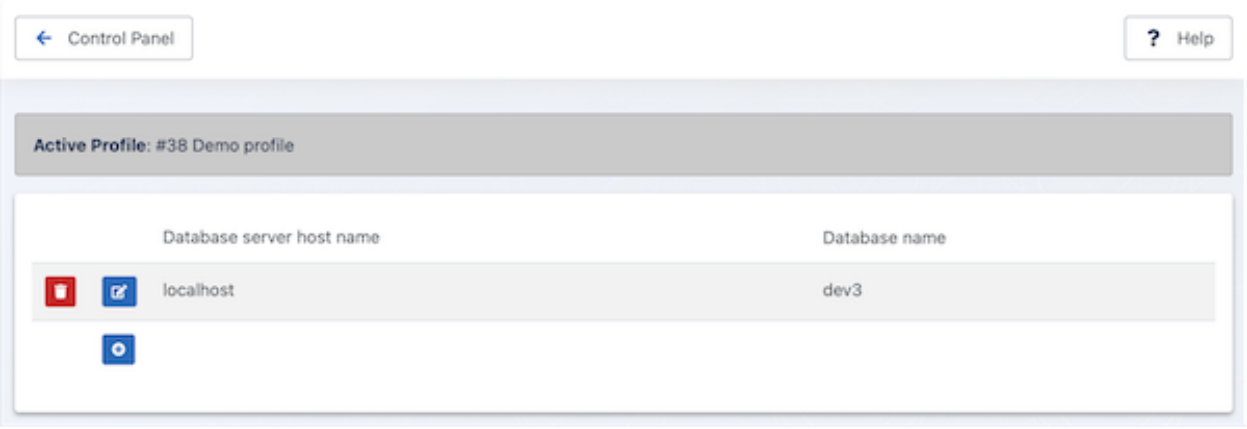
Moreover, you should not confuse the term "database" with your Joomla!™ tables. It is possible that a single *database* contains tables for the current Joomla!™ site, tables from a third party script, tables from another Joomla!™ site on the same server (e.g. a subdomain) and so on and so forth. As far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

Finally note that adding an empty database (one which has no tables) will result in backup errors.

Note

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

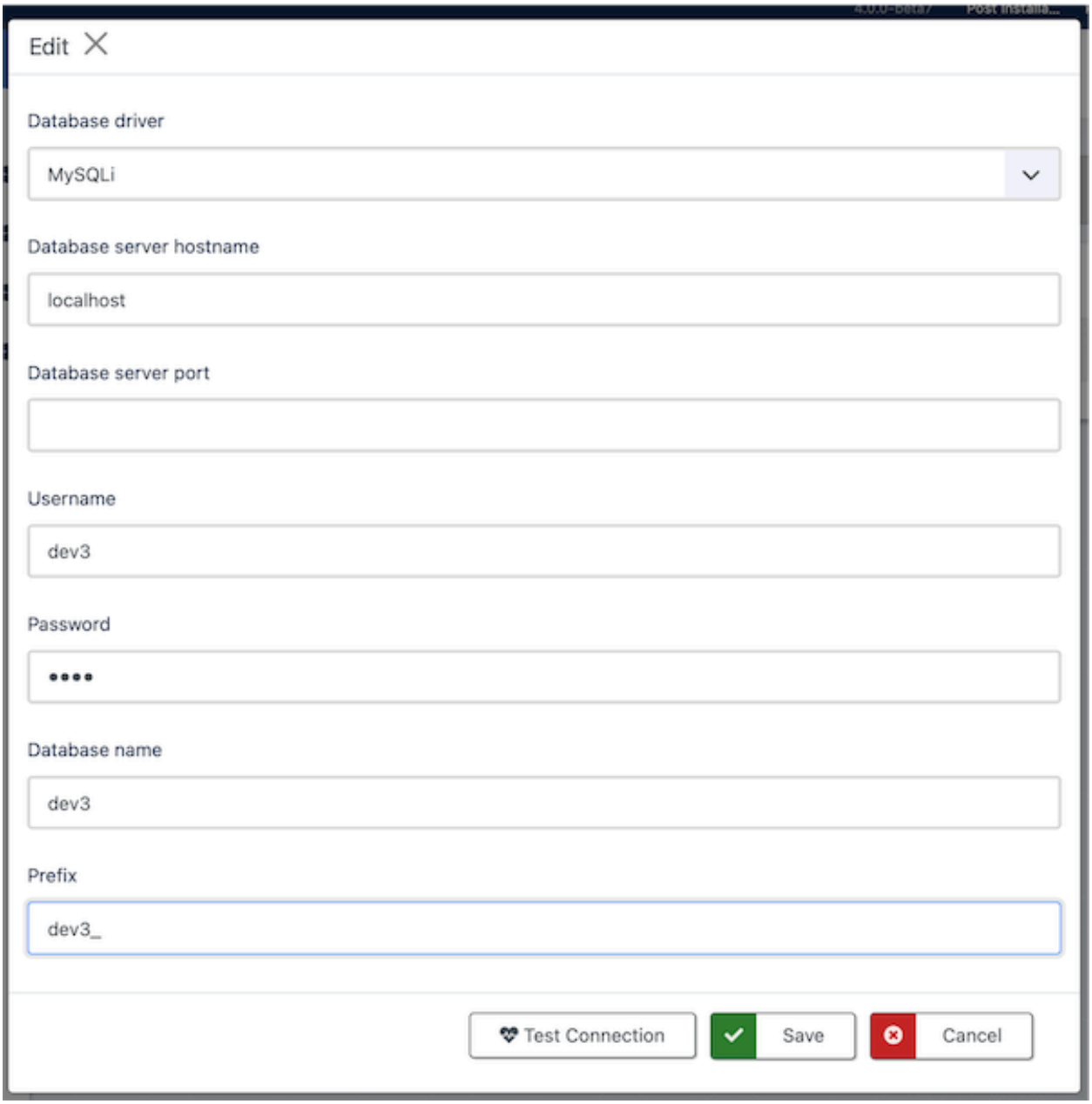
Multiple Databases Definitions



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

- **Trashcan** (red background). Clicking on this icon will remove the current database definition.
- **Pencil** or **Add** (blue background). 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 PHP's standard MySQL connection driver (the name stands for MySQL Improved). We recommend using it for MySQL, Percona and MariaDB databases on most server.
  - **MySQL (PDO)**. This is the alternative modern MySQL connection driver (the name stands for MySQL – PHP Data Object). This is for some servers which do not include MySQLi support but do include PDO support.

- **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. Please remember that for MySQL servers the settings `localhost` and `127.0.0.1` are NOT the same. The first means "connect using a socket or named pipe" the second means "connect using TCP/IP networking". If you are on Windows they have a massive performance difference as well.
- **Database server port**. Leave it blank, unless your host has told you to use a non standard port for connecting to their database server. The default TCP/IP port for MySQL is 3306. If none is specified this is what Akeeba Backup will try to use.
- **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. **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.

Some hosts use your account name as a prefix for the database and username. **This is not the same as the Prefix setting above.** That database and username prefix is actually part of the actual database and username that you need to fill into this page. 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 does NOT take a prefix. If in doubt, please contact your host. We can't guess the right values for you because we are not your host. If you ask your host to give you the connection information to your database, they must be able to do so - except for the password which they obviously cannot see for security reasons.

When you think you have all the connection information ready, click on Test Connection. This will check all settings except the Prefix. The connection test will tell you if it succeeded or failed.

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

It's very likely that advanced site owners will place files outside the site's root to prevent web visitors from having direct access to those files. These directories typically contain files that need complex access control and are, therefore, only made available for download through PHP code, e.g. a download manager extension for Joomla!. Akeeba Backup Core will only backup files under the site's root, which would make these files impossible to backup.

The solution to that problem is the Off-site Directories Inclusion feature of Akeeba Backup Professional. Using this feature you can tell Akeeba Backup to look for files in arbitrary locations outside the site's root and include them in the backup archive. 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 parent folder the "virtual folder".

For example, let's say you want to backup an off-site directory named `images`. If we weren't using the virtual folder its contents would end up being backed up inside the Joomla! `images` directory. This is not desirable. If your

virtual folder is called `my_offsite_includes` , this directory would end up being backed up as something like `my_offsite_includes/123ABC-images` . Notice the stuff and the dash before the actual directory name? This is a smart feature which allows you to backup many directories that have 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.

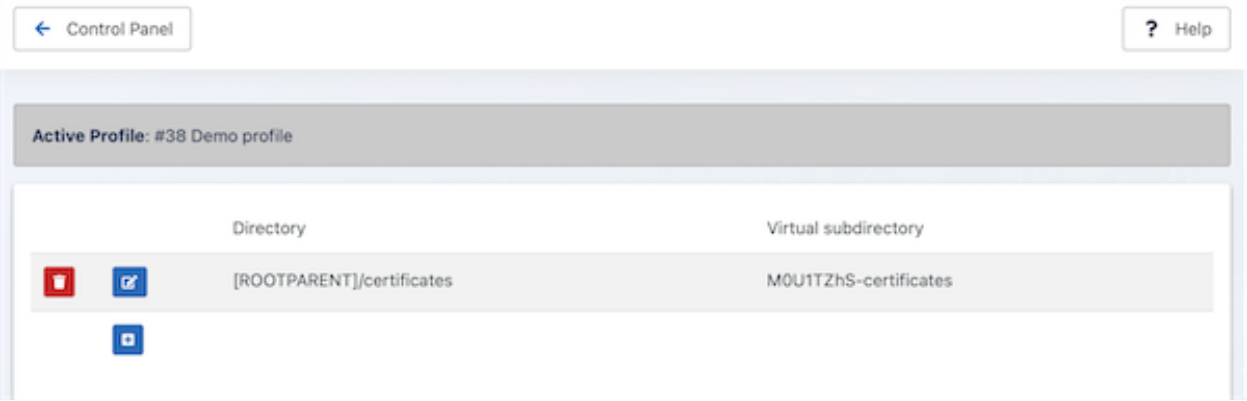
Moreover, ANGIE -the restoration script included in the Akeeba Backup archives- can semi-automatically restore the off-site directories to their original location. You will need to confirm the destination directory or, if you don't want to do this, just tell it to skip over that directory.

Important

You MUST NOT 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. If you manually add your site's main directory as an off-site directory inclusion you will be backing up the same files twice, doubling your backup size. For the same reason you must not add an folder already under the site's root as an off-site directory inclusion: you'd be backing up files already backed up, bloating the backup size.

Finally note that if your backup output directory is somewhere under your site's root (this is the case with the default backup output directory) and you add your site's root as an off-site directory using this feature your backup will fail. That's because Akeeba Backup won't know that your backup output directory is under the off-side directory you included, therefore it will try to back up the output folder. This will try adding the backup archive it's writing to into itself which will cause all sorts of problems and ultimately a backup failure.

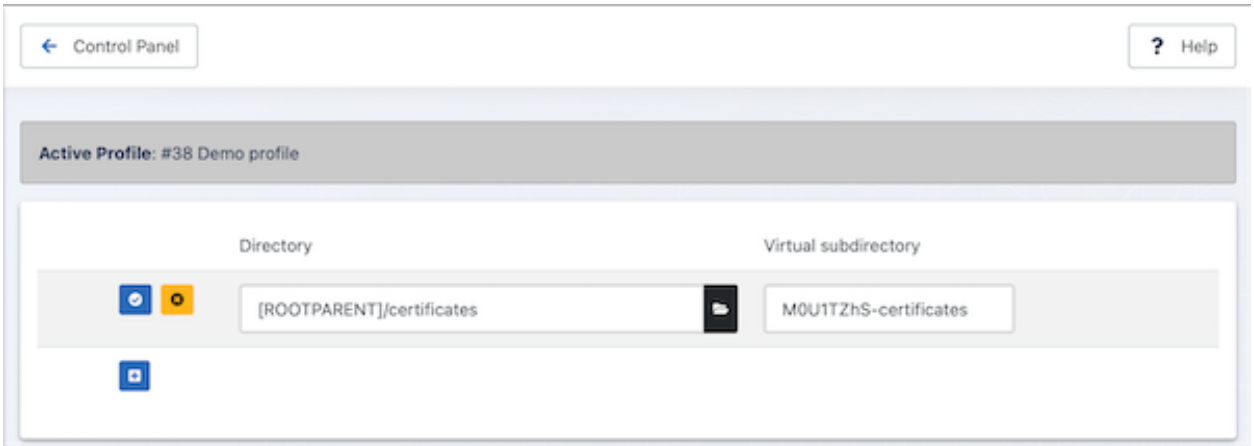
Off-site Directories Inclusion



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** (red background). Clicking on this icon will remove the current directory definition from the backup set.
- **Pencil** or **Add** (blue background). 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 checkmark** (blue background). Clicking on this icon will save any changes you have made.
- **Cancel** (yellow background). 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. You should never use this for the reasons explained earlier in this section.
- **[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. 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

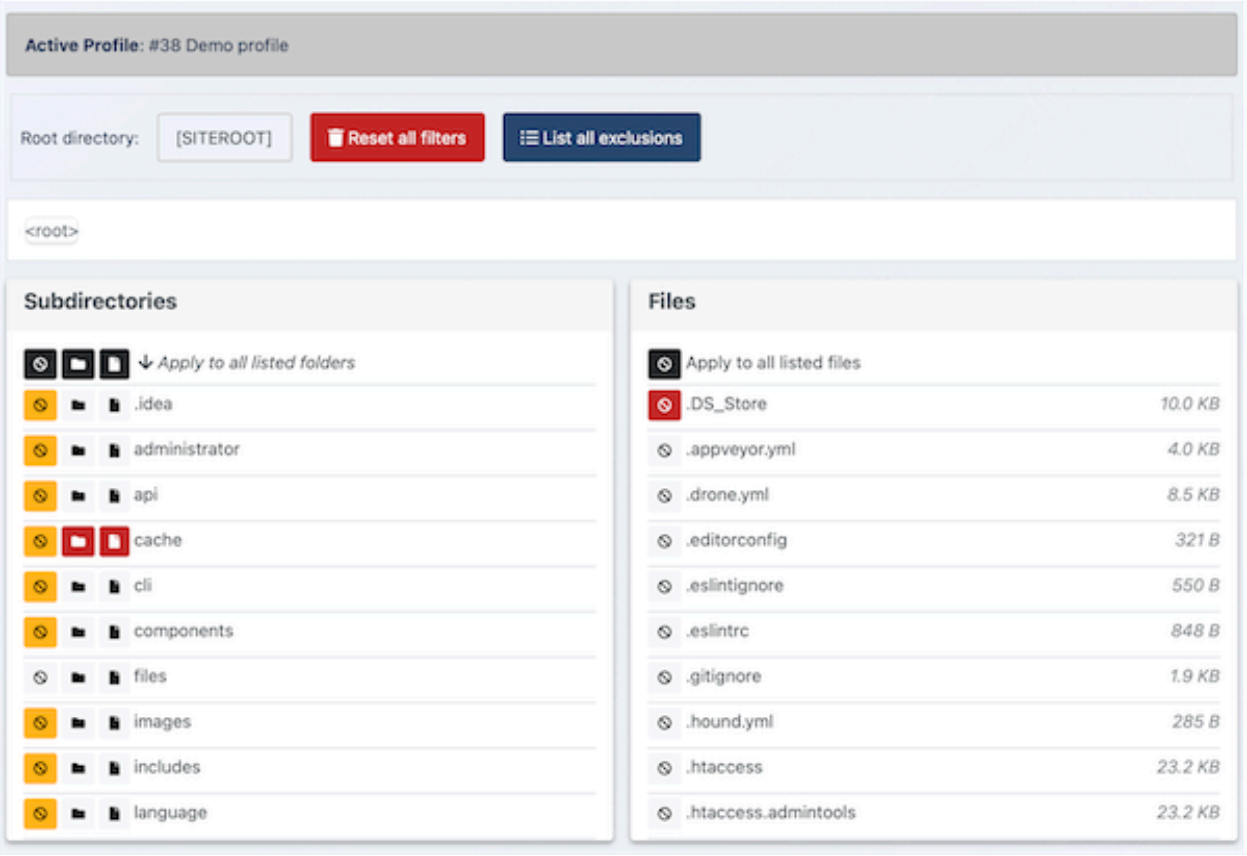
Very often our sites have files or folders which don't really belong to the backup. A few examples are:

- Additional sites whose root folders are subdirectories of your site's root. As explained elsewhere in this documentation, backing them up and restoring them would end up overwriting more sites than you bargained for.
- Directories with large amounts of videos, images, download repositories or other infrequently changing files. In most cases it makes sense to exclude them from your daily backups and only include them in a separate weekly or monthly backup profile.
- Leftover files you had forgotten about until the time came to back up your site. For example, that really big ZIP file with the previous version of your site you meant to delete two years ago.

Akeeba Backup lets you exclude files and folders to solve these problems.

Before discussing this feature, you should be aware of some automatic file and folder exclusions applied by Akeeba Backup. Akeeba Backup will automatically exclude your site's temp-folder and logs folder as configured in your site's Global Configuration; the "cache" directories under your site's root and administrator directory; and all files and directories inside the Akeeba Backup's output directory. This means that you should **never use a folder whose contents you intend to back up as your backup output directory, your site's temp-folder or your site's logs folder**. Moreover, do not leave the temp-folder and / or log folder blank or set them to your site's root in your site's Global Configuration. Doing so will result in your backup archive NOT having any of your site's files since the site's root will be automatically excluded by the automatic filters as explained above.

Files and Directories Exclusion - Browser View



At the top of the page there are two tabs, allowing you to switch between the Browser and Summary views.

The middle area contains a few controls you need to know about:

- 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 Reset all filters button will, as the name implies, remove all of the file and directory exclusion filters *for the selected root directory*.

- The List all exclusions button takes you to the Summary View page.

- 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.

The lower area consists of two panes, showing the folders and files in the current directory. The icons next to each item are an exclusion type each. You can use them to enable / disable filters on each folder or file. The top row of each panes has controls (icons) which apply the filters to all of the listed folders or files below it.

Each icon can have three states: on (yellow background), off (gray 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. Force enabled filters cannot be changed through this page.

The available filters for directories are:

- **Exclude Directory (circle with diagonal line running through it)**. When enabled, the folder and all of its contents (subdirectories and files) will not be included in the backup. This filter overrides the Skip Subdirectories and Skip Files filters.

- **Skip subdirectories (folder icon)**. When enabled, the subdirectories of this directory will not be included in the backup. However, the directory itself and its files will be included in the backup.

- **Skip files (file icon)**. When enabled, the files inside this directory will not be included in the backup. However, the directory itself and its folders (and the files inside these folders) will be included in the backup.

If both Skip Subdirectories and Skip Files filters are enabled on a folder then an empty folder will be included in the backup. If you do not want the folder to be included *at all* use the Exclude Directory filter.

Clicking on a folder name in the Folders pane will navigate inside it.

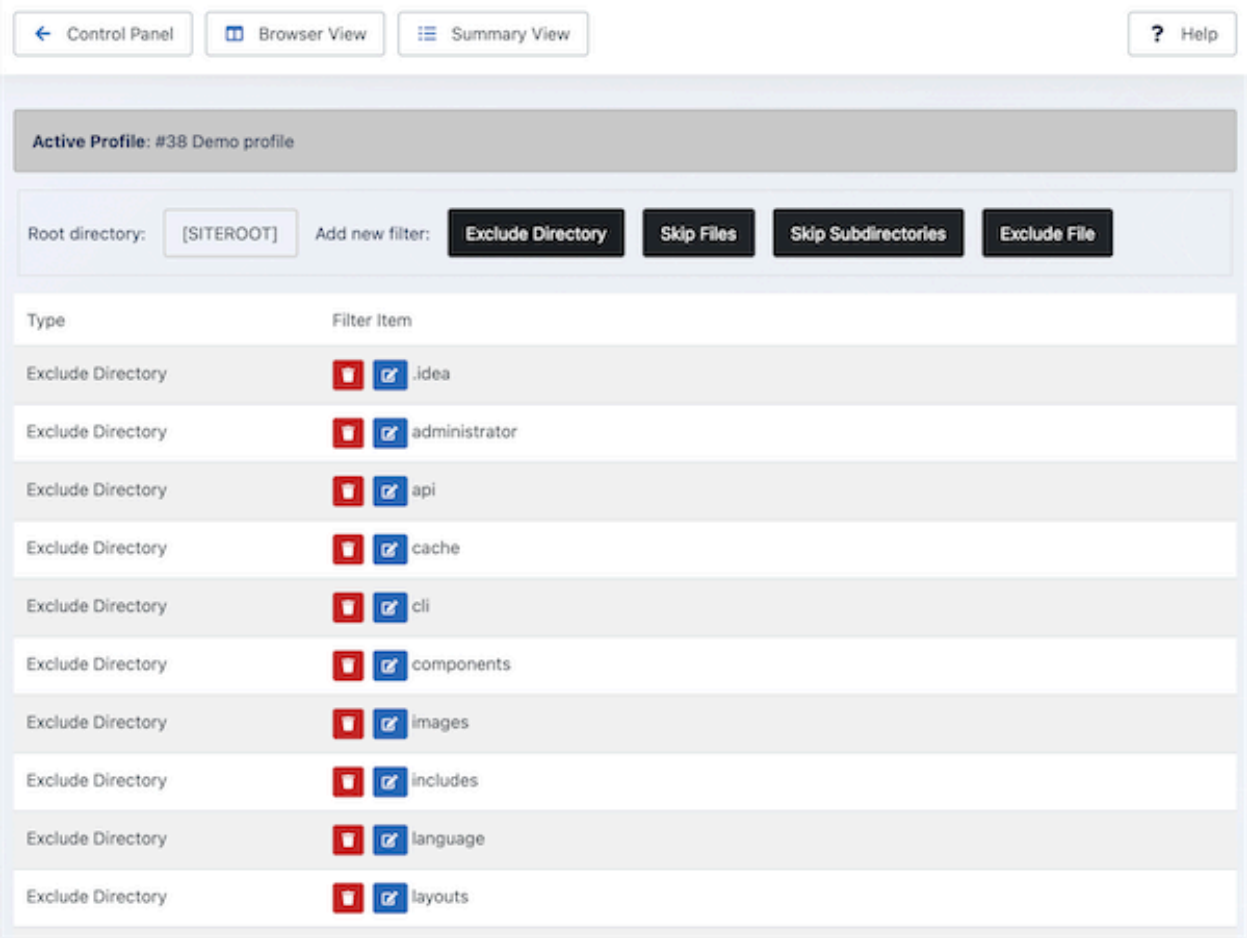
The available filters for files are:

- **Exclude File (circle with diagonal line running through it)**. When enabled, the file will not be included in the backup.

- The file name.

Each file name displays its size to the right. The file size will be displayed in the unit which is more convenient, i.e. bytes, KB, MB or GB. If you see no unit of measurement, the size is displayed in bytes.

Files and Directories Exclusion - Summary View



The Summary View displays a list of filters instead of a directory browser.

At the top you have 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.

On the top side of the grid you have the Add new filter buttons:

- **Exclude Directory**. The named folder and all of its contents (subdirectories and files) will not be included in the backup. This filter overrides the Skip Subdirectories and Skip Files filters.
- **Skip Subdirectories**. The subdirectories of the named folder will not be included in the backup. However, the folder itself and its files will be included in the backup.
- **Skip Files**. The files inside the named folder will not be included in the backup. However, the folder itself and its subfolders (and the files inside these folders) will be included in the backup
- **Exclude File**. The file will not be included in the backup.

The same notes regarding use of folder filters described in the Browser View apply.

Each line of the grid displays the following information:

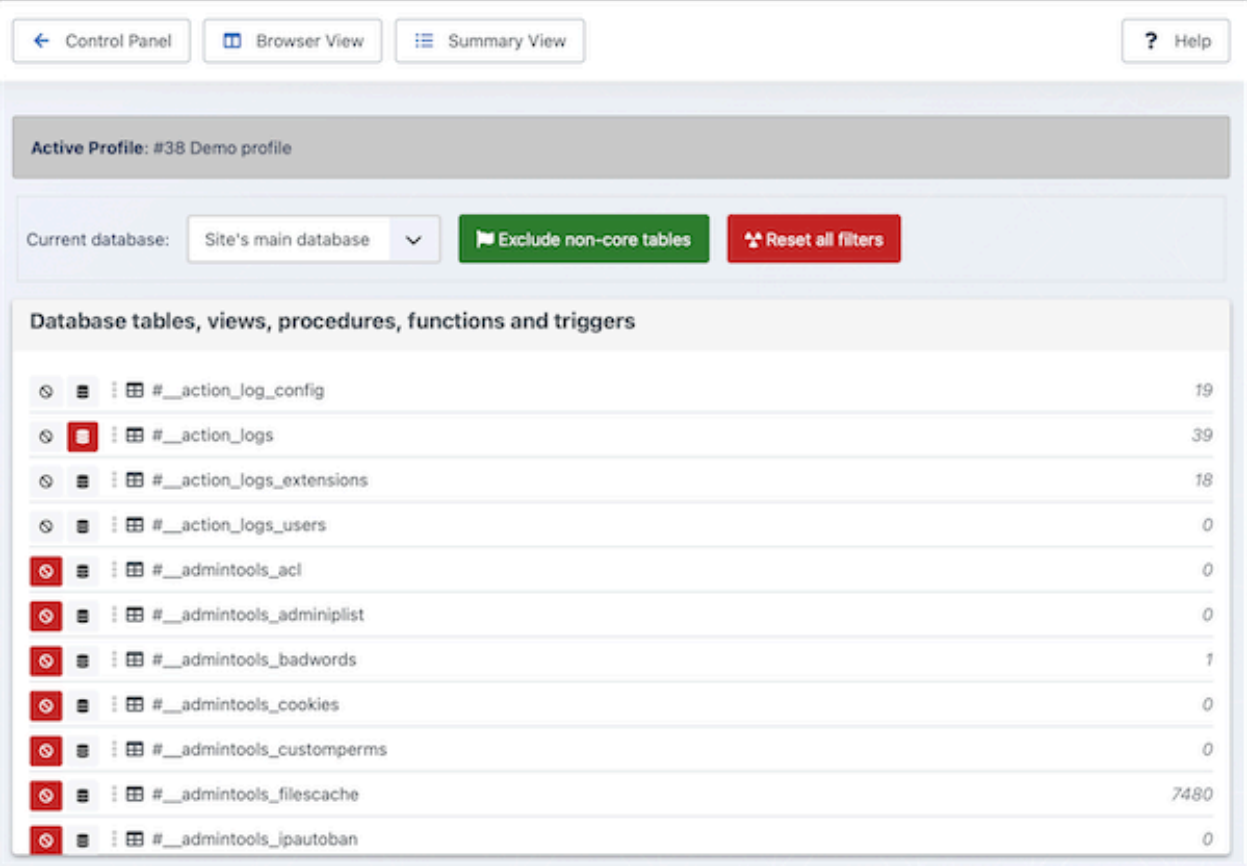
- **The filter type**. As described above in the add new filter buttons.
- **Trashcan**. When you click it, the filter row will be removed.
- **Pencil**. When you click it, the row switches to edit mode
- The **filter item** itself. It is the relative path to the directory or file which the filter row 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.

5.2. Database Tables Exclusion

There are cases where you need to exclude either entire database tables or their contents from the backup. For example, if you are using a single database for the tables of two or more sites you will want to exclude all tables not belonging to the site you're backing up to prevent accidental overwriting of the wrong site when restoring the backup. Moreover, if you have large tables with not very important data, such as log entries, you may want to exclude their contents - but not the entire table- from the backup for performance reasons. This is what the Database tables exclusion feature lets you do.

Database Tables Exclusion - Browser View



At the very top of the page you can see two tabs which let you switch between the Browser View and the Summary View.

Below the tabs you will see the backup profile number and title as a reminder. Remember that database table exclusion filters, just like all Akeeba Backup filters, are set up per backup profile.

Further down there is 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 additional databases you can select the appropriate database from the drop-down list to define filters for that database.

There are another two buttons here. The **Exclude non-core tables** button. Clicking it will automatically apply the Exclude This filter on all tables whose name doesn't begin with your site's prefix. These are usually tables which do not belong to the current Joomla! installation. Be warned of a major pitfall: the effects of this button are static. That is to say, if new tables with a different prefix are added in the future (e.g. tables are added in the other sites using the same database) you will have to come back here and click on this button again. Instead of that and if you have the Akeeba Backup Professional version you can use the Regular Expressions Database Tables feature to automatically deal with such configurations, without having to click this button.

The **Reset all filters** button will remove all database table filters for the currently selected database.

The main area of the page displays the contents of the database: tables, views, triggers, stored procedures and functions. Each row represents one database entity. The two leftmost icons represent an exclusion type, explained below. The third icon tells you what kind of database entity (table, view, trigger, ...) it is; hover over it to find out.

Each of the exclusion type icons may have one of three states: on (yellow background), off (gray background), or force enabled (red background). You can toggle between the on and off states by clicking on the icon. The force enabled state cannot be changed. It means that it is active (on) because another feature of Akeeba Backup, such as regular expressions, have it turned on. In case of non-table database entities the red filter type means that this operation is not applicable to this entity. For example, there is no point excluding the contents of a view since only its structure is being backed up anyway.

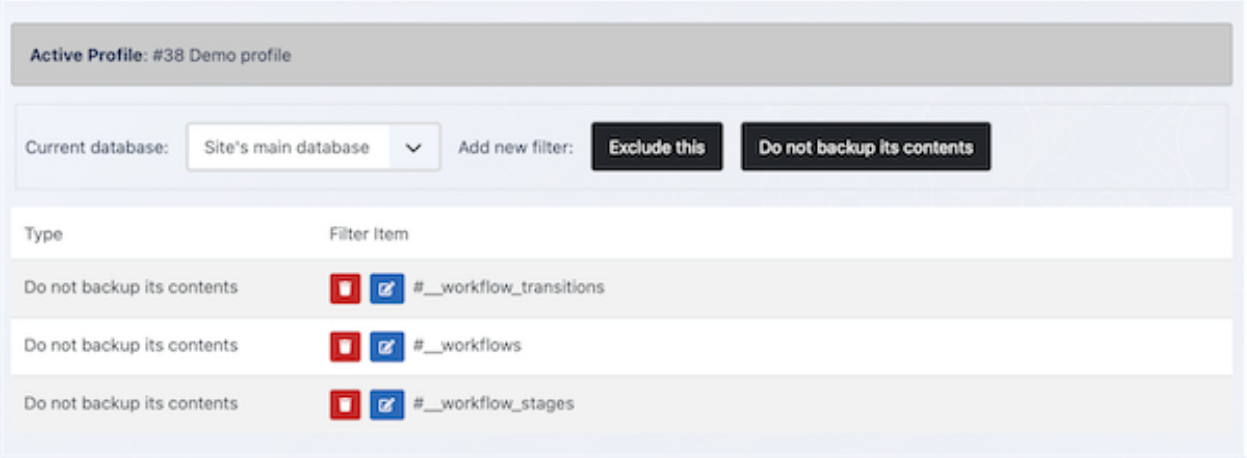
Important

The prefixes of the entities' names appear "abstracted". If your site's prefix is `abc1_` , the table `abc1_users` will appear as `#__users`. This helps you quickly identify the tables your site runs on.

The available filters are:

- **Exclude This (circle with diagonal line running through it)**. This database entity will not be backed up at all.
- **Do not backup its contents (three stacked disks / database icon)**. Only the structure of the database entity will be backed up, but not its contents. When restoring, this table will be created empty.

Database Tables Exclusion - Summary View



The Summary View displays a list of all active filters, allowing to quickly modify them.

At the very top of the page you can see two tabs which let you switch between the Browser View and the Summary View.

Below the tabs you will see the backup profile number and title as a reminder. Remember that database table exclusion filters, just like all Akeeba Backup filters, are set up per backup profile.

Further down there is 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 additional databases you can select the appropriate database from the drop-down list to define filters for that database.

Above the grid you have the Add new filter buttons. The filter types correspond to the icons in the Browser View, as discussed further above.

Each line of the grid displays the following information:

- **The filter type**. As discussed above.
- **Trashcan**. When you click it, the filter 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 #__ as discussed above.

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

Sometimes you know that you have to exclude files or directories following a specific naming pattern, but they are so many that it's 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 too complicated tracking them down and excluding them one by one. Regular expression filters let you create pattern-based filters to deal with that. What are regular expressions? Let's consult Wikipedia:

> 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, we recommend taking a look at the Regular Expressions Cheat Sheet (V2) [https://www.cheatography.com/davechild/cheat-sheets/regular-expressions/] from Cheatography. Some practical examples will be presented at the end of this section.

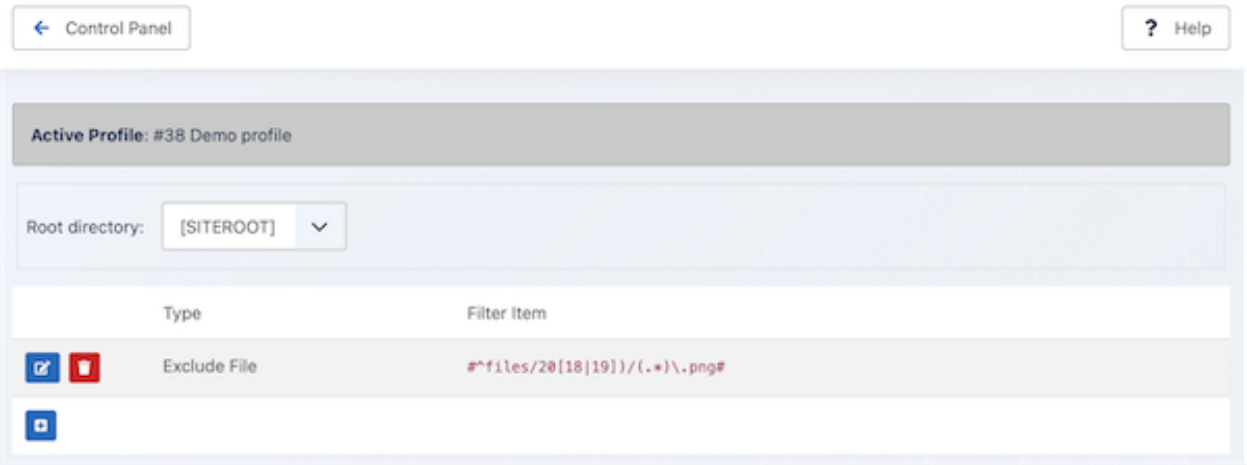
There are some special considerations experienced regular expressions users must have in mind:

- 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



At the very top of the page you will see the backup profile number and title as a reminder. Remember that database table exclusion filters, just like all Akeeba Backup filters, are set up per backup profile.

Right below it is 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.

Each row in the grid below represents a filter. The three columns on each row are:

Icons column
: You can perform the basic operation by clicking on this column's icons:
  - **Trashcan** (red background). When you click it, the filter row will be removed.
  - **Pencil** (blue background). When you click it, the row switches to edit mode

- **Add** (blue background; 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:

- **Disk** (dark background). When you click it, the changes will be saved.
- **Cancel** (yellow background). When you click it, any changes will be cancelled and the row will resume its previous state.

You can easily make sure that your filters match the directories and/or files you meant to. 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

Sometimes you know that you have to exclude database tables following a specific naming pattern, but they are so many that it's impractical going to the normal exclusion filters page and click them one by one. Or, more frequently, you want to exclude database tables not following a specific pattern, e.g. tables whose name doesn't begin with your site's table naming prefix. Regular expression filters let you create pattern-based filters to deal with that. What are regular expressions? Let's consult Wikipedia:

> 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, we recommend taking a look at the Regular Expressions Cheat Sheet (V2) [https://www.cheatography.com/davechild/cheat-sheets/regular-expressions/] from Cheatography. Some practical examples will be presented at the end of this section.

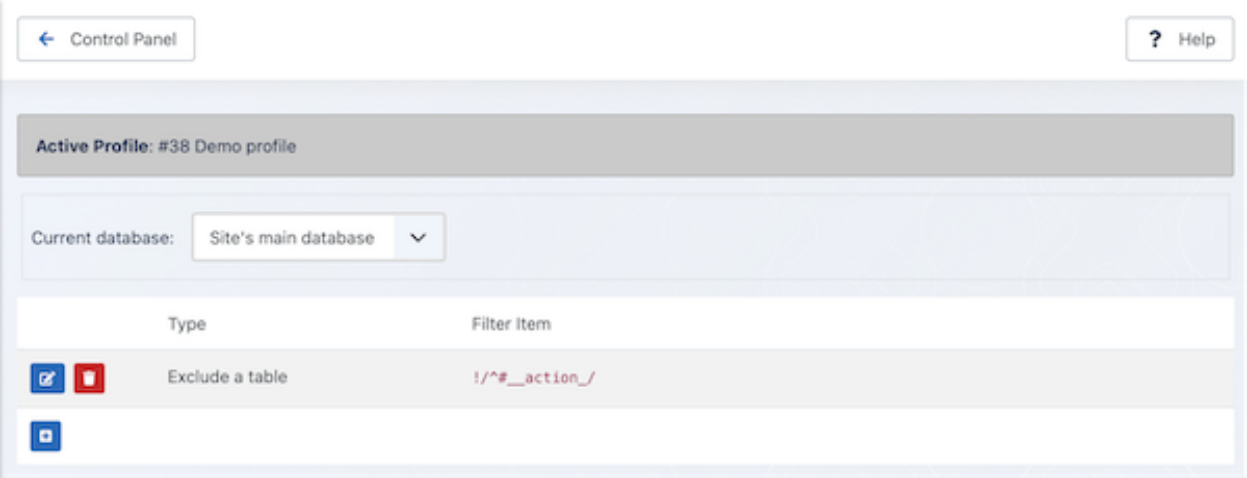
There are some special considerations experienced regular expressions users must have in mind:

- 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 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 `abc1_`, the table `abc1_users` is internally referenced as `#__users`. This is called the "abstracted" name in Akeeba Backup's documentation. You must take this into account when writing regex filters. The abstracted name of the table is the name you will have to match with your regular expressions!
- The prefix replacement described above takes place in Full Site and All Configured Databases backup modes. However, it *does not* take place in the Database Only backup mode. As a result, you have to reference the tables by their full, normal name, e.g. `abc1_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



At the very top of the page you will see the backup profile number and title as a reminder. Remember that database table exclusion filters, just like all Akeeba Backup filters, are set up per backup profile.

Below that you can find the Current Database 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.

Each row represents a filter. It has three columns:

Icons column
: You can perform the basic operations by clicking on this column's icons:
  - **Trashcan** (red background). When you click it, the filter row will be removed.
  - **Pencil** (blue background). When you click it, the row switches to edit mode
  - **Add** (blue background; 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:

- **Disk** (blue background). When you click it, the changes will be saved.
- **Cancel** (yellow background). When you click it, any changes will be cancelled and the row will resume its previous state.

You can make sure that your filters match the tables you meant to. 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

   ```
   !/^#__/
   ```

2. Exclude Akeeba Backup tables. We know that these tables have `ak_` in their name after the table prefix, e.g. `abc1_ak_foobar` becomes `#__ak_foobar`, so you only need to filter `#__ak`.

   ```
   /^#__ak_/
   ```

6. Automating your backup

6.1. Taking backups automatically

Why you need scheduled backups

Akeeba Backup is an excellent tool for taking backups of your site every time you want to transfer it to a different server, are ready to make substantial changes to it (like upgrading Joomla) or are about to or have just made changes to it. This way is something goes wrong you can easily roll back to the last known state of your site. The corollary to that is that if your backups are too far in between they might not encompass all useful changes to your site, leading to lost work and frustration.

This poses two concerns, depending on the type of site you have. If you have an infrequently changing site (low traffic blog, company presentation, organization bulletin board, ...) it takes a lot of self-discipline to take backups every time something changes, something that's not a given when you have multiple people managing a site. If you have a fast changing site (such as an e-commerce site, a busy blog with loads of comments, a community / forum site etc)

you'd have to frequently log into your site and take backups, probably multiple times a day. This is admittedly time-consuming and boring. In the end of the day you might skip a few or a lot of backups and that could prove detrimental to your ability to rescue your site should things go awry.

The solution to that is **scheduling** your backups, i.e. having backups being taken automatically, at regular intervals. This is not something that will happen without any intervention by you, the site owner. Backups take too long to run in a single page load and need your site to maintain a fairly consistent state throughout the backup. Therefore having traffic to the site trigger the backups is a bad idea as it's neither guaranteed to come at the right frequency and quantity to guarantee a backup when you'd like it to happen, nor does it guarantee a consistent state, by definition (people accessing your site may indeed cause things to change). Ideally, your backups should run when your site experiences the least amount of traffic. If you're not sure when: check your hosting control panel's traffic logs and look for the time of the day you receive the lowest number of requests. From that, it should be fairly obvious that you need *something else* to trigger the backup.

An overview of scheduling methods

Akeeba Backup offers several options which are explained further below in this section of the documentation. This is a quick overview which will let you pick a method suitable for your site.

The first group of methods are designed to run on the same server as your site.

Depending on your hosting company you might be able to use CRON jobs. This is the preferred option. True CRON jobs allow you to run custom scripts or commands accessible from the command line (CLI) at regular intervals. However, some hosts may limit the amount of time CRON jobs are allowed to run. If this time is less than the time it takes to run a backup this method is unsuitable for you. If you're not sure how long the backup takes: take a backup from the back-end of your site, go to the Manage Backups page and look at how long it took to run it. You can use the Native CRON Script and the Alternative CRON Script with this method.

On other servers you might be able to use pseudo-CRON jobs. In this case you can tell your host to access a specific URL at regular intervals. Please ask your host whether their pseudo-CRON follows HTTP redirections and if it does, whether it has a limit of redirections followed. If they answer that they do not follow HTTP redirections or that they impose a limit of redirections followed you cannot use this method. Furthermore, the time limits explained in the previous paragraph apply.

If the previous methods which rely on your own server are not suitable for your site, Akeeba Backup does offer two alternatives.

The first alternative is the Front-end Legacy Backup API. You can use a special URL with third party CRON services which follow HTTP redirections and do not impose a limit of redirections followed such as WebCRON or even a shell script using wget, curl or something similar on any server or Internet connected computer under your control. For example, you could set up a CRON job on your Linux machine, a Scheduled Task on your Windows computer or even have a Raspberry Pi or your NAS access that URL. If you use a third party service to trigger the backups be advised that they charge fees which are usually pretty low.

The second alternative is the Akeeba Backup Remote JSON API. This is a more advanced way of taking backups remotely. It is used in two distinct cases. First, in conjunction with our remote backup software, namely Akeeba Remote CLI and Akeeba UNiTE. These can be used to schedule taking backups remotely and, in the case of UNiTE, restoring them on a computer under your control as well. Second and most common, with a third party service that can schedule and take remote backups using Akeeba Backup's Remote JSON API such as BackupMonkey or Watchful. Third party services charge fees according to their commercial policy.

Availability of scheduling methods

Backup scheduling features are only available in Akeeba Backup Professional, our for-a-fee edition. They are not included in Akeeba Backup Core, our free of charge edition.

The target audience of Akeeba Backup Core has always been the small, hobbyist site owner and those who need a relatively easy way to transfer their sites between servers. This target audience does not need backup scheduling, either because their sites change too infrequently or because they perform one-off backups for site transfers.

If you find yourself with a site that changes often enough to warrant automatic backups we kindly ask you to consider an Akeeba Backup Professional subscription. There are several good reasons to do so. For starters, a site valuable enough to warrant automatic backups should *also* have these backups stored off-site, a feature exclusive to the Professional edition. Moreover, your small investment in Akeeba Backup Professional will save you copious amounts of time and money should something go awry. Furthermore you will have access to our acclaimed support, provided by the same developers writing the software, should you have any issue using it. It's worth noting that if you are a web professional you only need one subscription for unlimited sites, including those of your clients. Finally, by purchasing an Akeeba Backup Professional subscription you are financing the painstaking and expensive research, development and maintenance that goes into Akeeba Backup – thank you for making this software possible!

Scheduling backups

Each available backup method has a different way of scheduling. This documentation section explains in detail how each method works and how you can schedule it.

You should also go to Akeeba Backup's Schedule Automatic Backups page. You will get a condensed version of these instructions, specialized -to the degree possible- for your own site and *the currently active backup profile*. Trust us, this page will save you a lot of headache. In fact, it's what *we* reference on *our own* sites.

6.1.1. Front-end backup, for use with CRON

Tip

This feature is only available in Akeeba Backup Professional.

Requires the Enable Legacy Front-end Backup API (remote CRON jobs) option to be enabled in the component's Options, Frontend tab.

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 report a "bug" stating 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 and you've been sufficiently warned.
- 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

If you want to automate your backups despite your host not supporting proper CRON jobs you can use a third party service, such as Webcron.org [http://webcron.org/]. Just make sure you set up the 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? Just take a regular backup from your site's back-end, then go to the Manage Backups page and take a look at the Duration column.

We **VERY STRONGLY** recommend using the Front-End Backup feature only with sites configured to use HTTPS with a properly signed SSL certificate *for security reasons*: plain HTTP sites and self-signed HTTPS certificates can, under certain circumstances, lead to your Secret Word leaking. If a malicious user obtains the Secret Word they can launch a Denial of Service attack on your site and / or abuse Akeeba Backup's feature to obtain a copy of your site, including all privileged information. Getting a properly signed SSL certificate no longer costs any money. The Let's Encrypt certificate authority [https://letsencrypt.org/] offers free of charge SSL certificates. Most likely your hosting control panel already supports automatically acquiring and installing SSL certificates from Let's Encrypt. For example two of our favorite hosts, SiteGround and Rochen, have supported this since late 2015. If you are not sure, ask your host. Using HTTPS not only makes your site safer, it will even make it more popular with search engines. It's a win-win proposition!

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. Do not use symbols, accented characters, non-Latin character sets (like Green or Cyrillic letters) etc. Such 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: they do NOT follow redirections. In these cases you can schedule a CRON job on your own computer. The downside is that your computer will need to be powered on (not turned off or even in sleep / hibernate) at the time you've set up the backup to run and for the entire length of time it take to run the backup.

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_akeebabackup&

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!

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.

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.

Do not forget to surround the URL in double quotes. If you don't the backup will fail. The reason is the way operating systems parse command lines. Special characters such as question marks and ampersands have special meanings.

If you're unsure whether your command line makes sense please check with your host. Sometimes you have to get from them the full path to wget in order for CRON to work. For example:

/usr/bin/wget --max-redirect=10000 "http://*www.yoursite.com*/index.php?option=com_akeebabackup&

view=backup&key=*YourSecretKey*"

Again, please do contact your host; they usually have a help page for all this stuff. Read also the section on CRON jobs below.

Optionally, 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_akeebabackup&

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.addictive-code.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. If you are on an incredibly outdated server with `wget` version 1.10 and earlier the backup will most probably result into an error message concerning the maximum redirections limit being exceeded. This is *not* a bug in our software, it's a limitation of an ancient version of the third party wget software. Kindly note that version 1.11 which lifts that limitation was released ages ago, *in 2008 (two thousand eight!)* to be more precise.

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 in our software, it's how the Internet works.

We would like to note that some SEO or SEF URL extensions for Joomla! may get in the way of front-end backups, *especially on multi-language sites*. If you get inexplicable 403 or 404 errors towards the beginning of a front-end backup right after a redirection (HTTP code 301, 302 or 307) please consult with the developers of the SEO / SEF URL extensions you are using. Usually you can add an exception for Akeeba Backup's front-end backup URLs.

Furthermore, some hosts with very finicky web server firewalls may automatically block the front-end backup URL. Typically you get a 403 error at the very beginning of the backup process or after 2-3 redirections (at which point the front-end backup will no longer work). This typically happens when they misunderstand the front-end backup Secret Word as a security threat. Try changing your Secret Word to something else. If the problem persists please contact your host and ask them to take a look and add an exception for the front-end backup Secret Word you are using.

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_akeebaback-up&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_akeebabackup&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.

The ampersands above should be written as a single ampersand, not as an HTML entity (&).

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. Remote JSON API

Tip

This feature is only available in Akeeba Backup Professional.

Requires the Enable JSON API (remote backup) option to be enabled in the component's Options, Frontend tab.

Important

Akeeba Backup 9.0 and later only includes support for the Akeeba Backup JSON API version 2 (API level 400). If the service or software you are using only supports the version 1 API you will need to ask them to update their service or use a newer version of the software you are using.

This feature is designed to let third party services and software take and optionally download backups of your site remotely (from any computer or a server other than the one your site is hosted on). This API is currently used by popular Joomla management and backup scheduling services such as Backup Monkey and Watchful.

The JSON API uses optional encryption to ensure confidentiality of the API calls even on plain HTTP sites. However, we very strongly recommend that you use the JSON API over HTTPS connections. On most servers you can install an SSL certificate from Let's Encrypt at no additional cost. Please consult your host.

Things you should know:

- **It is not designed to be run from a web browser**, but from an application or service which understands the API.
- The API is available through the Akeeba Backup component which runs inside Joomla. While we take reasonable precautions, third party plugins may interfere with it. These problems are very rare.
- Since it's running over the web it is subject to the PHP and web server timeout limits. Make sure your backup runs to completion through the back-end of your site *before* using the JSON API. If your backup fails through the back-end it will definitely fail through the JSON API as well.
- Since it's running over the web your host may intercept, alter or block the requests to the JSON API and its replies. If that happens taking a backup will be impossible or may terminate before it is complete. In this case please contact your host. They should be able to determine why they interfered with the request and help you work around it.
- Giving access to the Remote JSON API is not to be taken lightly. Software and services given access to your site's Akeeba Backup Remote JSON API have the authority to see information about your past backups, get information about the version and update availability of Akeeba Backup, read and modify your backup settings (including remote storage services' access information), take backups and download backup archives. Only give access to services and software you trust.
- Downloading a backup archive through the JSON API is by no means guaranteed to work. It largely depends on the size of the backup archive and your host. Our advice to third party services and software is to always offer an alternative for downloading the backup archives using SFTP or FTP.
- Access to the Remote JSON API has many intermediate layer which are not under the control of Akeeba Ltd. Indicatively these are: the code which produces the request to the JSON API; the operating system and server / computer configuration where the code producing the request to the JSON API is installed in, including its firewall configuration; network connectivity between that server / computer and your site's server; your site server's firewall and system configuration; your server's web server and PHP configuration; any third party plugins running on your Joomla site. If you can take a backup through your site's back-end but not through the remote JSON API do not assume it's a bug with our software (99.9% of the times it's not). First contact the developer of the third party service or software which implements the remote backup. That's a small community, we know them and they know us. They can identify whether the problem is on their end, on your server or if it's a backup issue. If you do have a backup issue and are already a subscriber with access to Akeeba Backup's support (that is to say, you have access to Akeeba Backup Professional) please file a support ticket on our site's Support section and we will help you.
- If you are trying to run remote backups from your own server or computer we have limited troubleshooting options for clients who qualify for support. Namely, we will try to run the backup through our own computers and servers and try to determine the root cause of the issue. If it's outside our client software or the Akeeba Backup installation on your site we can only tell you that it's not a bug in our software and tell you about what could be interfering (see the above bullet points). In this case the root cause is outside our control and we cannot reasonably be expected to resolve it.
- Your secret key must be non-empty and meet a minimum complexity metric for this feature to be enabled. If you have not set up a secret key or if the secret key is too easy to guess the JSON API will respond with an access denied message and Akeeba Backup will display a prominent warning in its control panel page with instructions to fix it.

Setting up third party services

Each third party service has a slightly different way to set up the connection with Akeeba Backup's Remote JSON API. In most cases you install an extension on your site and the third party service can set up the connection automatically. They do so by reading the Secret Key you have set up in Akeeba Backup's Options page and using the known endpoint URL to Akeeba Backup's JSON API.

In some cases you may need to provide either or both of these pieces of information to the third party service. The set up for third party software below explains how to obtain that information.

Setting up software

Setting up software for remote backups through the JSON API requires providing either or both of the following information:

Endpoint URL: It is `https://www.example.com/index.php?option=com_akeebaback-up&view=Api&format=raw` where `https://www.example.com` is the full URL to your site's front page. Akeeba software which implements the JSON API (Akeeba Remote Control and Akeeba UNiTE) can only be given the full URL to your site; they will figure out the rest of the URL automatically.

Secret Key: This is the secret key you set up in Components, Akeeba Backup, Options, Front-end Backup tab, Secret word option. We recommend using a random, 64-character password generated with a random password generator as explained in the documentation page of the Options page of Akeeba Backup. If the secret key you provide to the third party software or service does not match the one you set up in Akeeba Backup, or if the secret key you set up in Akeeba Backup is too easy to guess (based on automated password complexity metrics) the JSON API will always respond with an access denied message.

Support for backup issues when using the JSON API for remote backups

If you are using Akeeba Remote CLI or Akeeba UNiTE and qualify for support according to our Terms of Service we can provide substantial but not unlimited assistance. We can identify whether your connection information is correct (or let you know how to correct it). Moreover, we can determine whether the problem is with our software or outside or of our control. In the first case we will identify the issue in our software and fix it. In the latter case we will tell you our analysis, where we believe the problem lies to, produce evidence that the backup can be run from our computers or servers and try to point you to the correct direction for resolving the issue that's not coming from our software itself.

When it comes to **third party services** we kindly ask you to always **seek support with the service provider before contacting us**. In many cases the problem lies with the service itself or its connection to your server. We do not have access to these third party services and cannot troubleshoot or resolve such issues. The service providers can. If you ask us to assist with such third party service issues we will have to decline support for objective reasons.

If the service provider tells you that you have a **backup issue** your support options are determined by the Support Policy which is part of our Terms of Service. That is to say, if you qualify for support according to our Terms of Service we can provide assistance for your backup issue with the JSON API in the same way we would provide assistance for any other backup issue.

6.1.3. Native CRON script

Tip

This feature is only available in Akeeba Backup Professional.

Important

Unlike Akeeba Backup 3.x to 8.x inclusive, the CRON script is no longer a standalone PHP application. Instead, it's implemented as a command for Joomla CLI application.

First of all, you need to make sure that the Console – Akeeba Backup plugin is published and its Access is set to Public. If you do not do that, Joomla does not know about Akeeba Backup's CLI commands.

The `joomla.php` script you see below is part of Joomla itself, not something we have written or have any control over. This means that any code that runs before you see any output from Akeeba Backup is handled by Joomla itself, not our code. If you get an error before reaching that point you will need to file a bug report with the Joomla project which controls this code, not us (we can't fix Joomla's code).

Further to that, keep in mind that the Joomla CLI Application DOES NOT run at all under the PHP-CGI binary. It will only run under the PHP-CLI binary. Our custom CLI scripts in versions 3.x to 8.x did run under PHP-CGI, with some caveats which could cause the backups to fail with a timeout error. Therefore the Joomla CLI application may fail to run in some cases our scripts used to work. THIS IS NOT A BUG IN OUR SOFTWARE AND WE HAVE ABSOLUTELY ZERO CONTROL OVER IT. Please file a bug report with the Joomla project so they can fix it.

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. The CLI command can be executed from the command-line PHP interface (PHP CLI) using the Joomla CLI Application. It doesn't require the front-end backup in order to work; it is a native backup solution 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/joomla.php akeeba:backup:take
```

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 command 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 backup naming variables, e.g. `[SITE]`, `[DATE]` and `[TIME]`. 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.

The command 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/joomla.php akeeba:backup:take
   ```

2. Backup with profile number 2 and default description:

```
usr/local/bin/php /home/johndoe/httpdocs/cli/joomla.php akeeba:backup:take --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/joomla.php akeeba:backup:take --description
```

4. Backup with profile number 2 and a description reading "My automated backup":

```
usr/local/bin/php /home/johndoe/httpdocs/cli/joomla.php akeeba:backup:take --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 how Joomla's CLI application works, following 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.
- This script is not meant to run from a web interface. If your host only provides access to the CGI or FastCGI PHP binaries or pseudo-CRON (accessing a URL on a schedule), `joomla.php` will not work with them. The reason is actually the same as the time constraint above.
- 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

Note

This section depends on your host's control panel software. It is included it for informational purposes only and we cannot guarantee its accuracy. If you have questions about this please ask your host.

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/php-cli /home/myusername/public_html/cli/joomla.php akeeba:backup:ta
```

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/php8-cli`. This is an example for the location of the correct executable file for PHP CLI. 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.

CLI backups on restrictive hosts

Some hosts, especially shared hosting environments, have a maximum CPU usage time limit per process. If your backup takes longer to complete — including any uploading of the backup archives to remote storage — than this time limit the backup process will be terminated prematurely by your host. The log file will show no error but it will also not show that the backup is complete.

There are two ways to resolve that.

Option 1: Talk to your host (hardest to do, faster backups).

Ask your host to increase the maximum CPU usage limit to a higher value that's enough for your backup to run. How much is that? Take a backup from the backend of your site. Go to the Manage Backups page. Read the Duration of the backup and add 15% to it. For example, if the backup took 1 hour 13 minutes and 42 seconds (a total of 4422 seconds) you need to tell your host to increase the CPU usage time limit to at least 5085 seconds.

Please note that this may impossible to explain to first level support techs. If they seem to misunderstand you and start talking about the PHP time limit (something completely different) ask for your ticket to be escalated to a second level support tech. You can tell the second level support tech that the issue is with the **ulimit -t** which is applied to your CRON jobs. They should be able to either raise the hard limit or tell you how to modify your CRON job command line to set a higher soft limit. It's okay if you do not understand what that means; the second level tech does and will be able to help you.

Option 2: Use Joomla's Scheduled Tasks (easiest to do, slower backups).

Use Joomla's Scheduled Tasks with the CLI scheduling option. When choosing the task type select the "Akeeba Backup – Take a Backup".

This method is slower than taking a backup with the native CRON script *but* it works around the maximum CPU usage time limit your host has applied by spreading the backup execution across multiple executions of the backup task. In this case do keep in mind that unlike the native CRON script you MUST set up Joomla's Scheduled Task CRON job to run every minute, NOT just at the desired backup time. You can control the backup time and frequency of the backups from Joomla itself (System, Manage, Scheduled Tasks).

Please note that when you go for this option you can no longer override configuration variables.

Overriding configuration variables

You can 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 Super Users 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 JSON files throughout the component's back-end. The first location you should look at is `administrator/components/com_akeebabackup/engine/core`, where you will find four JSON files with general settings. Inside the `administrator/components/com_akeebabackup/engine` subdirectories you will find one JSON file per engine.

If you are using Akeeba Backup Professional you will find more JSON files under the `administrator/components/com_akeebabackup/platform/Joomla/Config` directory. These files include some of the features only available in the Professional version of the software.

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

For your information, the configuration keys for cloud storage services can be found in the .json files under `administrator/components/com_akeebabackup/engine/Postproc`.

Applying them on the command line is easy. Take this command line as an example:

```
usr/local/bin/php /home/johndoe/httpdocs/cli/joomla.php akeeba:backup:take
--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. 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. Do note that your command line 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.

6.1.4. Alternative CRON script

Tip

This feature is only available in Akeeba Backup Professional.

Requires the Enable Legacy Front-end Backup API (remote CRON jobs) option to be enabled in the component's Options, Frontend tab.

Important

Unlike Akeeba Backup 3.x to 8.x inclusive, the CRON script is no longer a standalone PHP application. Instead, it's implemented as a command for Joomla CLI application.

First of all, you need to make sure that the Console – Akeeba Backup plugin is published and its Access is set to Public. If you do not do that, Joomla does not know about Akeeba Backup's CLI commands.

The `joomla.php` script you see below is part of Joomla itself, not something we have written or have any control over. This means that any code that runs before you see any output from Akeeba Backup is handled by Joomla itself, not our code. If you get an error before reaching that point you will need to file a bug report with the Joomla project which controls this code, not us (we can't fix Joomla's code).

Further to that, keep in mind that the Joomla CLI Application DOES NOT run at all under the PHP-CGI binary. It will only run under the PHP-CLI binary. Our custom CLI scripts in versions 3.x to 8.x did run

under PHP-CGI, with some caveats which could cause the backups to fail with a timeout error. Therefore the Joomla CLI application may fail to run in some cases our scripts used to work. THIS IS NOT A BUG IN OUR SOFTWARE AND WE HAVE ABSOLUTELY ZERO CONTROL OVER IT. Please file a bug report with the Joomla project so they can fix it.

This command uses the front-end backup feature of Akeeba Backup to run a backup. This may work on some hosts where the regular CRON script doesn't.

As already stated in the Front-End Backup feature, we **VERY STRONGLY** recommend using the Front-End Backup feature -including the case where it's used by the alternative CRON script- only with sites configured to use HTTPS with a properly signed SSL certificate *for security reasons*: plain HTTP sites and self-signed HTTPS certificates can, under certain circumstances, lead to your Secret Word leaking *even if you are only using it with the alternative CRON script*. If a malicious user obtains the Secret Word they can launch a Denial of Service attack on your site and / or abuse Akeeba Backup's feature to obtain a copy of your site, including all privileged information. Getting a properly signed SSL certificate no longer costs any money. The Let's Encrypt certificate authority [https://letsencrypt.org/] offers free of charge SSL certificates. Most likely your hosting control panel already supports automatically acquiring and installing SSL certificates from Let's Encrypt. For example two of our favorite hosts, SiteGround and Rochen, have supported this since late 2015. If you are not sure, ask your host. Using HTTPS not only makes your site safer, it will even make it more popular with search engines. It's a win-win proposition!

You will have to a command line similar to this with your host's CRON interface:

```
/usr/local/bin/php /home/USER/webroot/cli/joomla.php akeeba:backup:alternate
```

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.

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/joomla.php akeeba:backup:alternate
   ```

2. Backup with profile number 2

   ```
   /usr/local/bin/php /home/johndoe/httpdocs/cli/joomla.php akeeba:backup:alternate --profi
   ```

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 Options button on the top right corner of the toolbar.

- Before using the alternative CRON script for the first time, or after moving your site to a new domain name and / or enabling HTTPS, you must visit the Akeeba Backup's Control Panel page at least once. This will cache the URL to your site for use by the alternative CRON script.
- 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 fopen() URL wrappers

  If none of these methods is available, the backup will fail.
- Taking a backup with the front-end backup feature must be possible on your site. See the Discussion in the chapter about the front-end backup, especially the issues with SEO / SEF URL extensions and host firewalls.
- Your host may have a firewall setup which doesn't allow the CRON script to access the front-end backup URL if it's launched from the same server. In this 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 a few hosts.

If you are seeking assistance regarding a failed CRON job please indicate if and which of these steps you have already tried. We don't want to ask you to do something you've already tried.

6.1.5. Joomla Scheduled Tasks (without CLI)

Note

This feature is only available on Joomla 4.1.0 and later. Make sure the plugin Task - Akeeba Backup is published.

Akeeba Backup is a component which takes backup when someone or some*thing* tells it to do so. Automating backups is all about having *something* tell Akeeba Backup when to take a backup at a predefined time, as well as keep telling it to step through the backup until the backup is done. We can call this the “execution controller”.

In the previous sections we have covered different ways this can be done: using a native CLI CRON job; using your host's or a third party's scheduler which accesses a special URL; or using a CLI CRON job which accesses a special URL. There is also the possibility of using a third party service, Akeeba UNiTE or Akeeba Remote Control CLI using the JSON API.

All of these methods require an execution controller external to your site. While this is the best way to run something complex and long running — like a backup — it is not the *only* way. Joomla 4.1.0 and later have a feature called Scheduled Tasks which allows you to set up stuff to run on a schedule which you define and modify *within Joomla itself*. In its turn, the Scheduled Tasks feature has three execution controllers: using a native CLI CRON job; using a special URL; or using traffic on your site as the access trigger.

Akeeba Backup 9.2.0 and later offer an integration with Joomla Scheduled Tasks with two different task types, one that works with all execution controllers and another one which only works with the CLI execution controller. Please make sure to read the next two sections which explain the differences between the two task types and the caveats from using Joomla Scheduled Tasks before you decide whether to use this feature.

6.1.5.1. The two Akeeba Backup task types

The Task - Akeeba Backup plugin makes two task types available to you:

Akeeba Backup – Take a Backup

This is the regular backup task type which works with all three execution controllers for Joomla Scheduled tasks (lazy, URL and CLI). Technically speaking it is a “resumable” task, meaning that it will do some work and notify Joomla that it's pausing for now. Joomla will note that down along with the fact that it needs to resume this task at the first opportunity. Depending on how many other tasks you have scheduled, their priority and when the execution controller is triggered it may take a while until the task is resumed and the backup continues. Rinse and repeat until the backup is done.

While this can be used to take backups using the lazy scheduling offered by Joomla Scheduled Tasks it MAY cause backup consistency issues, backups may run at odd times or not at all, backups may take an unrealistic amount of time to run and so on and so forth. Please read the DANGER AHEAD section below to understand all the risks you are taking by using this scheduled task.

We consider this backup method to only be suitable for small to medium-sized sites with a fairly steady flow of traffic which does not cause major changes in the database or file content e.g. news sites, blogs and the like. If you have a high value, high traffic, high rate of change site such as an e-commerce site or a community site we VERY STRONGLY RECOMMEND AGAINST USING JOOMLA SCHEDULED TASKS TO TAKE A BACKUP. The only realistic options for you are native CLI CRON jobs, the URL CRON jobs, third party services / remote JSON API backups and *maybe* the CLI-only Scheduled Task type.

Akeeba Backup – CLI-only Backup

Unlike the previous task type, this backup task will run start to finish without pausing. As a result this will only run when using the CLI execution controller for Joomla Scheduled Tasks. This is on purpose! Only under CLI are we confident there are no execution time limits which could be of concern.

You might wonder, why use this task type instead of a native CLI CRON job? Well, there are pros and cons.

On one hand, using this type of Joomla Scheduled Task means you can control its execution schedule and / or disable it from within Joomla itself, without having to mess with your host's CRON jobs after you've set them up.

On the other hand, the only option you can choose is the backup profile to use when taking a backup. You have none of the other options, including configuration overrides and the description and comment customisation you have with the native CLI CRON job.

We consider this task type as a useful tool for an advanced site integrator to allow their clients to exercise a modicum of control over the backup schedule, or as a tool to be used for convenience purposes. If you want the full power of the CLI CRON jobs use that feature instead.

6.1.5.2. DANGER AHEAD: Caveats on using Joomla Scheduled Tasks.

Start of execution is fuzzy and unreliable

First and foremost, the Joomla Scheduled Tasks feature does not have the scheduling resolution you get from real CRON jobs. A real CRON job service — either the one built into your server's operating system or an external, third party one — has a resolution of a few seconds. This means that scheduled tasks will always run on the scheduled time give or a take a few seconds. Joomla's Scheduled Tasks have a resolution which can be several hours depending on which execution controller you are using:

- Lazy Scheduling. This is triggered by traffic to your site and at most once in as many seconds as you set up in System, Scheduled Tasks, Options, Lazy Scheduler, Request Interval. The default is 300 seconds, meaning you only get to execute one task (or one *partial* task) every 5 minutes. If you have a steady flow of traffic your backups may start several minutes after you have scheduled them. If you don't have a steady flow of traffic your backups may start hours later, or not at all.

- URL (“Web Cron”) or CLI: this only triggers the execution whenever you access the special URL or run the CLI CRON job for Joomla's Scheduled Tasks. The maximum resolution you can do by scheduling the execution of these execution controllers is one minute. This means that your backups may start several minutes after you have scheduled them.

Joomla Scheduled tasks are sequential, not parallel

Moreover, unlike real CRON jobs which run in parallel, Joomla Scheduled Tasks run *sequentially*. This means that depending on the number of scheduled tasks you have defined, their schedule and their priority it is possible that the backup task does NOT start / does NOT resume the very first time the Joomla Scheduled Tasks execution controller is triggered. It may take several triggers of the execution controller to circle back to the pending backup scheduled task. In fact, it is possible to come to a scheduling situation where the backup NEVER starts or NEVER finishes. This CAN be worked around when using the URL or CLI CRON job execution controller; it can NOT be worked around when you are using the lazy backup scheduling.

The combination of the previous two issues may lead to backups taking forever or not finishing at all

The next thing to note is that backups do not run start to finish, they run in small chunks of work. This is intentional. A backup can be a very long operation which may last from a dozen seconds on small sites hosted on really fast servers to several hours for big sites and/or sites hosted on slower hosts. When accessing a URL you are executing a PHP script which is subject to limits: the PHP execution time limit, the PHP memory limit, the web server time limit waiting for PHP to send it an indication on whether the script has finished running, the maximum CPU usage time per process controlled by your server's operating system etc. The only way to work around these limits is split the backup process in smaller chunks of work.

Once a small chunk of work is done the Akeeba Backup task will tell Joomla Scheduled Tasks that we are pausing for now and please tell; us to resume at the first available opportunity. This is the same thing we do during a backend backup. There is, however, a big difference!

When taking a backend backup the execution controller is a piece of JavaScript running on your browser. Once it sees that “I'm done but I need to do more work” signal it *immediately* tells the Akeeba Backup component to continue working on the next chunk of backup work.

As we explained above, Joomla Scheduled Tasks does not work like that. Based on the execution controller you are using and the other scheduled tasks you have on your site it may take a minute to several hours for the backup to resume running (i.e. execute the next chunk of backup work). At best, this will make the backups take MUCH, MUCH LONGER than they do when taking a backup from the backend of your site or when using any of the other backup scheduling options. We ran a test on a real world site which normally takes 2 minutes to take a backup from the backend. When using Joomla Scheduled Tasks and its CLI execution controller scheduled to run once a minute (the maximum frequency allowed for a CRON job) it took over 20 minutes for the backup to complete. On another site that normally takes 14 minutes the backup took approximately 2 hours to complete.

Lazy scheduling may lead to inconsistent backups

The best time to run a backup for your site is when there is as little traffic as possible to minimise the chance that there will be database or filesystem changes while the backup is executing. This ensures backup consistency.

By its nature, Joomla Scheduled Task's Lazy Scheduling only runs backups when there is traffic on your site. Even worse, it only really works when there's a lot of traffic on your site. This means that your backups will run only when your site is under load with two major consequences.

First, your site will be slower since you are running a process that's heavy on I/O and memory during the peak usage time of your site. For the same reason it is possible that some requests may fail outright — including the one which runs your backup! Therefore you end up with either a slow site or a failed backup.

The other problem is that you can no longer expect consistency of your backups since you are running them exactly when you expect a lot of things to change on your site as a result of the traffic it is receiving.

Therefore using Lazy Scheduling is A VERY BAD IDEA unless you have a mostly “read-only” site such as a news site or a blog with no comments or using a third party comments service outside of your site. If you try to use that on a site where major changes are expected as a result of your user's traffic — such as a community site, a forum, or an e-commerce site — your backup will most likely be inconsistent and possibly unusable.

These issues are objectively outside our control which is why we offer no support for lazy scheduling and limited support for URL and CLI CRON jobs triggering the Joomla Scheduled Tasks.

6.1.5.3. Setting up your site for lazy backup scheduling

Warning

We very strongly advise against using the Lazy Scheduling method for Joomla Scheduled Tasks to take backups of your site. We will offer no support for taking or restoring backups taken using this method — use at your own risk. There are a lot of issues which can arise from the use of this kind of unreliable backup scheduling method which are objectively outside our control as explained in the “DANGER AHEAD” section above.

If you decide to ignore this very strong warning we advise you to at least test each and every backup taken with this method to catch the inevitable issues before they become insurmountable problems leading to data loss.

Backup profile configuration

Unlike backend backups , scheduled backups don't have each backup step run in quick succession to each other. This means that we don't need to have any wait time between them — the nature of the scheduling method adds enough wait time, ranging from several seconds to several hours. Therefore we are going to change the fine tuning options to cram as much work as we can within a given amount of time.

Go to the Configuration page of your backup profile and use the following settings:

- Minimum execution time: 0 seconds
- Maximum execution time: `120` seconds

  Depending on your server you may want to modify this value. Setting it higher you can do more work per backup step and your backup finishes faster. However, this means that your server is slower during that time and your backup may fail if you hit the PHP or web server time limit, or if you hit your host's maximum CPU usage time. We recommend trying high values first — but not higher than about *half* of the Task Timeout set up in Joomla Scheduled Tasks' Options page which is 300 seconds by default — and lower them if you notice your site being too slow or your backups are failing.
- Execution time bias: 80%
- Disable step break before large files: Yes
- Disable step break after large files: Yes
- Disable proactive step breaking: Yes
- Disable step break between domains: Yes
- Disable step break in finalisation: Yes
- Set an infinite PHP time limit: Yes

Joomla Scheduled Tasks setup for Lazy Scheduling

In your site's administrator backend go to System, Manage, Scheduled Tasks.

Click on the Options button in the toolbar.

Click on the Configure Tasks tab.

Set the Task Timeout (seconds) to 300. As noted above, this is the time limit beyond which Joomla will kill a task already running. This needs to be at least twice as much as the Maximum Execution Time in your backup profile and no smaller than 60 seconds. The default value of 300 seconds should be enough for most practical purposes and servers. On some servers you may have to lower this to 60 seconds. If you need to lower it even more DO NOT use the Lazy Scheduling for Joomla Scheduled Tasks — they won't work right. You will need to use a real CLI CRON job. If your host doesn't offer that option, well, you will have to use a third party service to schedule your backups or consider moving to a decent host (there's only so much anyone can do to work around really restrictive and badly set up hosting environments running atop oversold servers which are permanently on the verge of crashing due to too much load...).

Click on the Lazy Scheduler tab.

Set Lazy Scheduler to `Enabled`.

Set the Request Interval (seconds) to *`30`* (thirty).

Click on Save & Close.

The request interval needs to be relatively low to allow backups to run at a reasonable amount of time *as long as there is enough traffic on your site*. The default value of 300 (5 minutes) is too coarse for the backup to run in a reasonable amount of time, leading to most definitely corrupt or inconsistent backups.

Set up the scheduled task itself

In your site's administrator backend go to System, Manage, Scheduled Tasks.

Click on the New button in the toolbar.

Choose the “Akeeba Backup – Take a Backup” task type.

Give it a title of your liking, e.g. “Lazy Backup”.

Set up the execution rule and it configuration. For example, if you want your backup to run every day at midnight set the Execution Rule to `Interval, Days`, set the Interval in Days to `1` and set the Execution Time (UTC) to `12:00 AM`.

Tip

The backup date and time is always expressed in the UTC time zone in Joomla. If you want to convert your local time zone to UTC you can use a free time zone converter [https://www.timeanddate.com/world-clock/converter.html].

Under Task Parameters you can see a drop-down list with your backup profiles. By default, the default backup profile (#1) is selected. If you would like your backup to run under a different profile select it in the drop-down.

In the Advanced tab we very strongly recommend setting the Priority to `High`, making the backup task take priority over other scheduled tasks. You should leave other tasks in the Normal or Low priority. This ensures that the backup completes in as little time as possible.

Regarding the Notifications section we strongly recommend enabling all of the options in there. This will keep you appraised of the execution of all backup tasks, regardless of their status.

You may change the Permissions if you'd like to limit who can edit the backup schedule. We recommend removing all privileges for non-Super User groups who have some editing permissions by default e.g. Editor, Publisher, Manager and Administrator.

Click on Save & Close.

Further steps and caveats

Tempting as it may be DO NOT try to use the Run Test button next to the backup task. It does nothing for backup tasks; it returns immediately. This is expected and it does NOT mean that your backup task is broken. You will have to wait for it to run for real. Yes, it's a bit confusing but the Run Test button was only really designed to work with very short tasks which complete in a single shot, not resumable tasks which run over a long period of time. Backup tasks are resumable tasks exactly because they need to run over a long period of time!

Another thing to note is that when you first create a daily backup task it's scheduled to run *the upcoming (next) day, at the time you specified*. This is pretty much the case with all execution rules, even if you use an interval e.g. once every 30 minutes. The next task run date and time is calculated with the date and time you click on the Save & Close as the base of calculation. So, a task set to run on an interval of 30 minutes will only execute for the first time 30 minutes after you save it *at the earliest*. Yes, it is confusing but there's a good reason Joomla works that way: it prevents mishaps where long running and potentially dangerous tasks run all at once once you save them. In any case, you can see the next task run date and time by editing the task and clicking on the Execution History tab, then look at Next Execution.

If you edit a backup task that's not yet finished you will see that the Last Exit Code is 123 and the Next Execution is in the past. **THIS IS NORMAL**. The exit code 123 means “I have more work to do”. When Joomla sees that exit code it does not update the Next Execution date and time. This allows Joomla to resume the task as soon as possible.

6.1.5.4. Setting up your site for scheduling using a URL CRON job

Warning

We advise against using the “Web Cron” (URL-based CRON jobs) method for Joomla Scheduled Tasks to take backups of your site. Doing so may result in the backups not starting at the right time or at all, not completing in a reasonable amount of time or at all, or end up with consistency issues.

As a result we can only offer very limited support for taking or restoring backups taken with this method. When it comes to taking backups we can only point you to this documentation. For restoring backups we will evaluate whether there is something we can do to help you restore your site but most likely we will simply point you back to the “DANGER AHEAD” section of this documentation to help you understand why you are experiencing consistency issues.

If you decide to use this method we recommend to test your backups frequently to catch the inevitable issues before they become insurmountable problems leading to data loss.

Backup profile configuration

Unlike backend backups , scheduled backups don't have each backup step run in quick succession to each other. This means that we don't need to have any wait time between them — the nature of the scheduling method adds enough wait time, ranging from several seconds to several hours. Therefore we are going to change the fine tuning options to cram as much work as we can within a given amount of time.

Go to the Configuration page of your backup profile and use the following settings:

- Minimum execution time: 0 seconds
- Maximum execution time: `120` seconds

Depending on your server you may want to modify this value. Setting it higher you can do more work per backup step and your backup finishes faster. However, this means that your server is slower during that time and your backup may fail if you hit the PHP or web server time limit, or if you hit your host's maximum CPU usage time. We recommend trying high values first — but not higher than about *half* of the Task Timeout set up in Joomla Scheduled Tasks' Options page which is 300 seconds by default — and lower them if you notice your site being too slow or your backups are failing.

- Execution time bias: 80%
- Disable step break before large files: Yes
- Disable step break after large files: Yes
- Disable proactive step breaking: Yes
- Disable step break between domains: Yes
- Disable step break in finalisation: Yes
- Set an infinite PHP time limit: Yes

Joomla Scheduled Tasks setup for Web Cron (URL) scheduling

In your site's administrator backend go to System, Manage, Scheduled Tasks.

Click on the Options button in the toolbar.

Click on the Configure Tasks tab.

Set the Task Timeout (seconds) to 300. As noted above, this is the time limit beyond which Joomla will kill a task already running. This needs to be at least twice as much as the Maximum Execution Time in your backup profile and no smaller than 60 seconds. The default value of 300 seconds should be enough for most practical purposes and servers. On some servers you may have to lower this to 60 seconds. If you need to lower it even more DO NOT use the Web Cron (URL) scheduling method for Joomla Scheduled Tasks — they won't work right. You will need to use a real CLI CRON job. If your host doesn't offer that option, well, you will have to use a third party service to schedule your backups or consider moving to a decent host (there's only so much anyone can do to work around really restrictive and badly set up hosting environments running atop oversold servers which are permanently on the verge of crashing due to too much load...).

Click on the Web Cron tab.

Set Web Cron to `Enabled`.

Click on Save & Close.

You are back on the same page. Click on the Web Cron tab again.

Copy the URL you now see in the Webcron Link (Base) field. It's something like:

```
https://www.example.com/component/ajax/?
plugin=RunSchedulerWebcron&group=system&format=json
&hash=wC9qeUErwcW2mZekDxBh
```

(line breaks added for clarity; the URL does NOT contain line breaks). You will need this URL in the next step.

Click on Cancel.

Set up the Web Cron (URL) CRON job to trigger Joomla's Scheduled Tasks

You need to set up a URL CRON job in the same way described in the Frontend Backup documentation section.

There are only two differences: when to run this CRON job and what is the URL to execute.

Use the Webcron Link (Base) URL you copied in the previous step, when setting up Joomla Scheduled Tasks for use with Web Cron.

You need to access this URL *every minute of every hour of every day*. That's because Joomla will run tasks — including starting and stepping through the backup — only when you access that URL. Therefore you need to access that URL very frequently (once a minute).

We only recommend using this method if your host offers real CRON jobs or URL CRON jobs. We DO NOT recommend using this with third party URL CRON services such as WebCron.org. If you need to use a third party service it's much cheaper and much easier using either the front-end backup URL with WebCron.org or a third party service which integrates with Akeeba Backup (such as BackupMonkey.io, mySites.guru or Watchful.net).

Side note: Yes, there is a way to only run the backup task with a Joomla Scheduled Tasks URL. However, it's too convoluted to set up and manage even for us, let alone our clients with much less or outright non-existent experience in systems administration. To make matters worse, it ended up being substantially more expensive than using a third party service which integrates with Akeeba Backup but also offers site monitoring features. This brings us to the old adage of "just because you can doesn't mean you should".

Set up the scheduled task itself

In your site's administrator backend go to System, Manage, Scheduled Tasks.

Click on the New button in the toolbar.

Choose the "Akeeba Backup – Take a Backup" task type.

Give it a title of your liking, e.g. "Daily Backup".

Set up the execution rule and it configuration. For example, if you want your backup to run every day at midnight set the Execution Rule to `Interval, Days`, set the Interval in Days to `1` and set the Execution Time (UTC) to `12:00 AM`.

Tip

The backup date and time is always expressed in the UTC time zone in Joomla. If you want to convert your local time zone to UTC you can use a free time zone converter [https://www.timeanddate.com/world-clock/converter.html].

Under Task Parameters you can see a drop-down list with your backup profiles. By default, the default backup profile (#1) is selected. If you would like your backup to run under a different profile select it in the drop-down.

In the Advanced tab we very strongly recommend setting the Priority to `High`, making the backup task take priority over other scheduled tasks. You should leave other tasks in the Normal or Low priority. This ensures that the backup completes in as little time as possible.

Regarding the Notifications section we strongly recommend enabling all of the options in there. This will keep you appraised of the execution of all backup tasks, regardless of their status.

You may change the Permissions if you'd like to limit who can edit the backup schedule. We recommend removing all privileges for non-Super User groups who have some editing permissions by default e.g. Editor, Publisher, Manager and Administrator.

Click on Save & Close.

Further steps and caveats

Tempting as it may be DO NOT try to use the Run Test button next to the backup task. It does nothing for backup tasks; it returns immediately. This is expected and it does NOT mean that your backup task is broken. You will have to wait for it to run for real. Yes, it's a bit confusing but the Run Test button was only really designed to work with very short tasks which complete in a single shot, not resumable tasks which run over a long period of time. Backup tasks are resumable tasks exactly because they need to run over a long period of time!

Another thing to note is that when you first create a daily backup task it's scheduled to run *the upcoming (next) day, at the time you specified*. This is pretty much the case with all execution rules, even if you use an interval e.g. once every 30 minutes. The next task run date and time is calculated with the date and time you click on the Save & Close as the base of calculation. So, a task set to run on an interval of 30 minutes will only execute for the first time 30 minutes after you save it *at the earliest*. Yes, it is confusing but there's a good reason Joomla works that way: it prevents mishaps where long running and potentially dangerous tasks run all at once once you save them. In any case, you can see the next task run date and time by editing the task and clicking on the Execution History tab, then look at Next Execution.

If you edit a backup task that's not yet finished you will see that the Last Exit Code is 123 and the Next Execution is in the past. **THIS IS NORMAL**. The exit code 123 means “I have more work to do”. When Joomla sees that exit code it does not update the Next Execution date and time. This allows Joomla to resume the task as soon as possible.

6.1.5.5. Setting up your site for scheduling using a CLI CRON job

Please note that you can only use this method if your host offers real CRON jobs. If your host only offers URL CRON jobs or does not offer CRON jobs at all use one of the other Joomla Scheduled Tasks methods or another backup automation method such as using a third party service.

We recommend using this method instead of the Native CRON Script if you are on a host which has a maximum CPU usage time limit per process which is lower than the time it takes to run a full backup of your site. In all other cases this method and the Native CRON Script method are functionally equivalent; you can use whichever one feels easier to you. Both methods are *fully supported and highly recommended over any other backup scheduling method*.

Backup profile configuration

You do not need to make any configuration changes to your backup profiles when using the “Akeeba Backup – CLI-only Backup” scheduling method.

Joomla Scheduled Tasks setup for CLI scheduling

You do not need to make any changes to your Joomla Scheduled Tasks configuration when using the “Akeeba Backup – CLI-only Backup” scheduling method.

Set up the CLI CRON job to trigger Joomla's Scheduled Tasks

You need to set up a URL CRON job in the same way described in the Native CRON Script documentation section.

There are only two differences: when to run this CRON job and what is the URL to execute.

Use the following command:

```
/usr/local/bin/php /home/USER/webroot/cli/joomla.php scheduler:run --all
```

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.

You need to have this CRON job run every minute.

Set up the scheduled task itself

In your site's administrator backend go to System, Manage, Scheduled Tasks.

Click on the New button in the toolbar.

Choose the “Akeeba Backup – CLI-only Backup” or the “Akeeba Backup – Take a Backup” task type. Please read the note below.

Note

We recommend trying to use the ‘Akeeba Backup – CLI-only Backup’ task type first. If you notice that your backup process seems to be terminated prematurely, before it's fully finished, without an error in the log file please use the ‘Akeeba Backup – Take a Backup‘ task type instead.

Here's the reason for this. The ‘Akeeba Backup – CLI-only Backup’ task type tries to run the entire backup process, start to finish, in a single task execution — it works the same as the Native CRON Script method. This is the most efficient way to run a backup as there is no dead time between consecutive backups steps. However, on larger sites this may take several minutes to hours. Many hosts, especially those offering shared hosting, have a maximum CPU time limit per process. If the backup takes longer than that it is terminated prematurely and the backup does not complete.

The ‘Akeeba Backup – Take a Backup‘ task type, however, works differently. It runs a small chunk of the backup (a backup step) and returns, notifying Joomla that it needs to do more work at the first available opportunity. Since each step is contained in its own task execution it is unlikely to run longer than the maximum CPU time limit per process your host has applied. On the other hand, each backup step runs for just a few seconds every minute (the maximum frequency you can call the Joomla Scheduled Tasks CLI script to trigger the execution of tasks). This means that there is a lot of dead time between backup steps which makes the backup take much longer to complete.

Therefore our recommendation is to first try using the faster ‘Akeeba Backup – CLI–only Backup‘ task type. If it fails you can use the ‘Akeeba Backup – Take a Backup‘ task type to work around your server's limitation, something you cannot do with the Native CRON Script!

Give it a title of your liking, e.g. “Daily Backup”.

Set up the execution rule and it configuration. For example, if you want your backup to run every day at midnight set the Execution Rule to `Interval, Days`, set the Interval in Days to `1` and set the Execution Time (UTC) to `12:00 AM`.

Tip

The backup date and time is always expressed in the UTC time zone in Joomla. If you want to convert your local time zone to UTC you can use a free time zone converter [https://www.timeanddate.com/world-clock/converter.html].

Under Task Parameters you can see a drop-down list with your backup profiles. By default, the default backup profile (#1) is selected. If you would like your backup to run under a different profile select it in the drop-down.

In the Advanced tab we very strongly recommend setting the Priority to `High`, making the backup task take priority over other scheduled tasks. You should leave other tasks in the Normal or Low priority. This ensures that the backup completes in as little time as possible.

Regarding the Notifications section we strongly recommend enabling all of the options in there. This will keep you appraised of the execution of all backup tasks, regardless of their status.

You may change the Permissions if you'd like to limit who can edit the backup schedule. We recommend removing all privileges for non-Super User groups who have some editing permissions by default e.g. Editor, Publisher, Manager and Administrator.

Click on Save & Close.

Further steps and caveats

Tempting as it may be DO NOT try to use the Run Test button next to the backup task. If you are using the “Akeeba Backup – CLI-only Backup” task type it will not work; that task is set up to only work when tasks are triggered through the CLI. If you are using the “Akeeba Backup – Take a Backup” task type it will also do nothing as this button does nothing for backup tasks; it returns immediately. This is expected and it does NOT mean that your backup task is broken. You will have to wait for it to run for real. Yes, it's a bit confusing but the Run Test button was only really designed to work with very short tasks which complete in a single shot, not resumable tasks which run over a long period of time. Backup tasks are resumable tasks exactly because they need to run over a long period of time!

Another thing to note is that when you first create a daily backup task it's scheduled to run *the upcoming (next) day, at the time you specified*. This is pretty much the case with all execution rules, even if you use an interval e.g. once every 30 minutes. The next task run date and time is calculated with the date and time you click on the Save & Close as the base of calculation. So, a task set to run on an interval of 30 minutes will only execute for the first time 30 minutes after you save it *at the earliest*. Yes, it is confusing but there's a good reason Joomla works that way: it prevents mishaps where long running and potentially dangerous tasks run all at once once you save them. In any case, you can see the next task run date and time by editing the task and clicking on the Execution History tab, then look at Next Execution.

If you edit a backup task that's not yet finished you will see that the Last Exit Code is 123 and the Next Execution is in the past. **THIS IS NORMAL**. The exit code 123 means “I have more work to do”. When Joomla sees that exit code it does not update the Next Execution date and time. This allows Joomla to resume the task as soon as possible.

6.2. Checking for failed backups automatically

Tip

This feature is only available in Akeeba Backup Professional.

While you can automate backups with any of the methods explained 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).

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 potentially dangerous feature: if you check for failed backups while a backup is still running 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 feature is only available in Akeeba Backup Professional.

Requires the Enable Legacy Front-end Backup API (remote CRON jobs) option to be enabled in the component's Options, Frontend tab.

The front-end backup failure check feature lets you 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 as explained in the relevant section of this documentation.

The URL to use to trigger the front-end backup failure check feature is

http://*www.yoursite.com*/index.php?option=com_akeebabackup&view=check&key=*YourSecretKey*

where *YourSecretKey* is the Secret Word for front-end backups, as configured in the component's Options page.

Please note that *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 want to automate the check you can schedule this URL just like you would a front-end backup URL. Furthermore, the front-end backup failure check URL does NOT use redirections. Therefore it can even be used with hosts and services which do not follow redirections.

6.2.2. CRON script for backup failure check

Tip

This option is only available in Akeeba Backup Professional.

Important

Unlike Akeeba Backup 3.x to 8.x inclusive, the CRON script is no longer a standalone PHP application. Instead, it's implemented as a command for Joomla CLI application.

First of all, you need to make sure that the Console – Akeeba Backup plugin is published and its Access is set to Public. If you do not do that, Joomla does not know about Akeeba Backup's CLI commands.

The `joomla.php` script you see below is part of Joomla itself, not something we have written or have any control over. This means that any code that runs before you see any output from Akeeba Backup is handled by Joomla itself, not our code. If you get an error before reaching that point you will need to file a bug report with the Joomla project which controls this code, not us (we can't fix Joomla's code).

Further to that, keep in mind that the Joomla CLI Application DOES NOT run at all under the PHP-CGI binary. It will only run under the PHP-CLI binary. Our custom CLI scripts in versions 3.x to 8.x did run under PHP-CGI, with some caveats which could cause the backups to fail with a timeout error. Therefore the Joomla CLI application may fail to run in some cases our scripts used to work. THIS IS NOT A BUG IN OUR SOFTWARE AND WE HAVE ABSOLUTELY ZERO CONTROL OVER IT. Please file a bug report with the Joomla project so they can fix it.

In order to schedule a backup failure check, you will have to use the following command line to your host's CRON interface:

/usr/local/bin/php /home/USER/webroot`/cli/joomla.php akeeba:backup:check`

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 same considerations and scheduling information as the backup CRON script apply.

6.2.3. Alternative CRON script for backup failure check

Tip

This feature is only available in Akeeba Backup Professional.

Requires the Enable Legacy Front-end Backup API (remote CRON jobs) option to be enabled in the component's Options, Frontend tab.

Important

Unlike Akeeba Backup 3.x to 8.x inclusive, the CRON script is no longer a standalone PHP application. Instead, it's implemented as a command for Joomla CLI application.

First of all, you need to make sure that the Console – Akeeba Backup plugin is published and its Access is set to Public. If you do not do that, Joomla does not know about Akeeba Backup's CLI commands.

The `joomla.php` script you see below is part of Joomla itself, not something we have written or have any control over. This means that any code that runs before you see any output from Akeeba Backup is handled by Joomla itself, not our code. If you get an error before reaching that point you will need to file a bug report with the Joomla project which controls this code, not us (we can't fix Joomla's code).

Further to that, keep in mind that the Joomla CLI Application DOES NOT run at all under the PHP-CGI binary. It will only run under the PHP-CLI binary. Our custom CLI scripts in versions 3.x to 8.x did run under PHP-CGI, with some caveats which could cause the backups to fail with a timeout error. Therefore the Joomla CLI application may fail to run in some cases our scripts used to work. THIS IS NOT A BUG IN OUR SOFTWARE AND WE HAVE ABSOLUTELY ZERO CONTROL OVER IT. Please file a bug report with the Joomla project so they can fix it.

This script uses the front-end backup failure check feature outlined above. 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 failure check, you will have to use the following command line to your host's CRON interface:

```
/usr/local/bin/php /home/USER/webroot/cli/joomla.php akeeba:backup:alternate_check
```

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 same considerations and scheduling information as the alternative backup CRON script apply.

7. Site Transfer Wizard

Note

This feature is only available in Akeeba Backup Professional.

As discussed below, you DO NOT need the Site Transfer Wizard to transfer your site between folders, subdomains, domains and servers using Akeeba Backup. Just take a backup, upload it together with Kickstart on the new location and run Kickstart. This is the method demonstrated in our video tutorial which is freely

available on our site. Site Transfer Wizard does not make transferring your site possible, it makes it easier as long as both hosts (source and target) support it. The typical problem is that your source server cannot connect to the target server because either or both servers have a firewall. In this case you'll have to transfer the site manually, per the video tutorials.

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.

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.

Important

The Site Transfer Wizard IS NOT the only way to transfer your site with Akeeba Backup and IS NOT guaranteed to work on all servers. If your site is very big, your server too slow or simply doesn't support the requirements of the Site Transfer Wizard then the wizard will fail to transfer the backup archive for you. **We cannot do anything against your host's technical limitations. However, you can still transfer your site with the Manual method available in the Site Transfer Wizard**. In a nutshell: you can take a backup; download the backup archive files to your site; upload the backup archive files and Kickstart where you want to restore the site to; run Kickstart. The Wizard will display a video tutorial about this when you select the Manual method.

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.

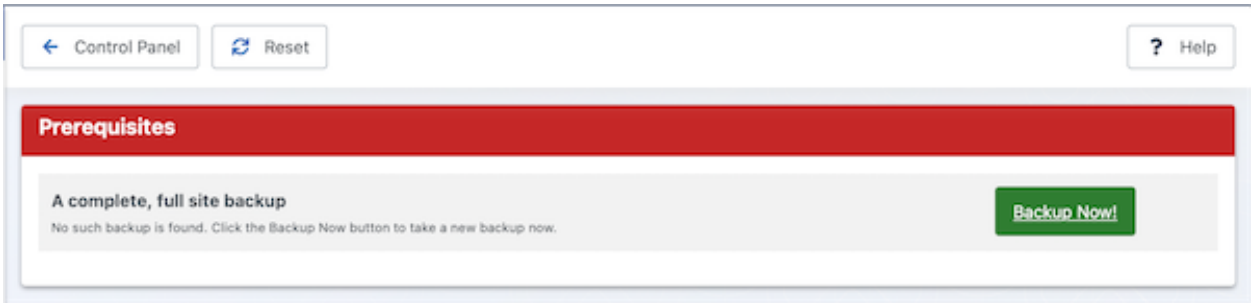
If you will be using SFTP to transfer your site your current server must either have the PHP cURL extension installed with SFTP support of the PHP SSH2 extension installed. It must not block outbound connection to the remote server's FTP port (typically port 22). The remote FTP server must allow connections 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

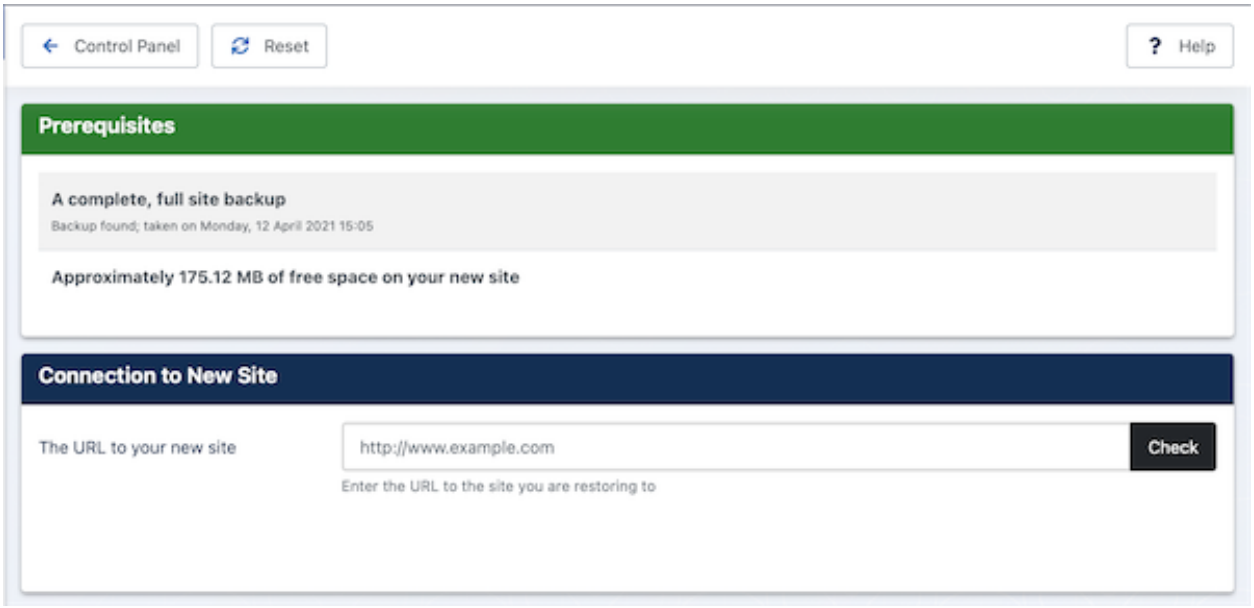


Click the Backup Now! button to take a new backup with the current backup profile. After the backup is complete you will need to go back to the main page and then click on Site Transfer Wizard again.

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!

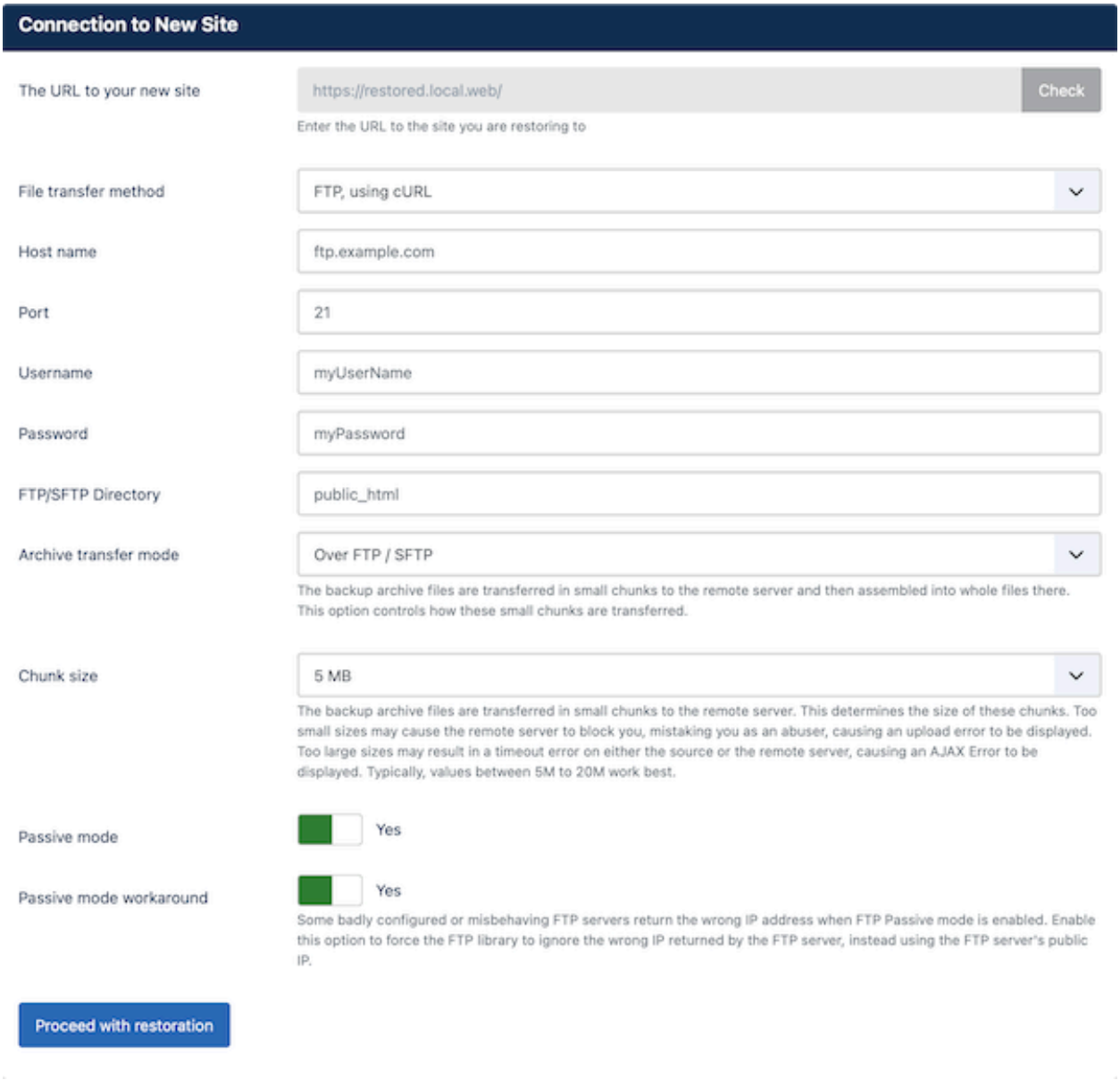
Tip

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, do not file a "bug" report** (as noted above, we can't override your hosts' technical limitations since *we are not your host, therefore cannot reconfigure your servers with more sane limits*). The manual method 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 hundreds of thousands of site integrators and site owners 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 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. Akeeba Backup Command Line Interface (CLI)

Akeeba Backup integrates with Joomla's CLI application (**joomla.php**) which made its first appearance in Joomla 4.0. This is a full-blown command line client for Joomla and its extensions, allowing us to provide a much richer experience than what was possible using the separate command line scripts we have been offering since 2010 and documented under backup automation.

Using the Joomla CLI client you can of course take backups but you can also manage all aspects of Akeeba Backup itself. You can import, export and configure backup profiles. You can get information about backups, access their log files, retry uploading them to remote storage if that failed during the backup itself, delete backup files or backup records and even configure all of the component options. Most commands allow some form of machine-readable content for their output and provide exit status codes, making it ideal for automation e.g. with an Ansible playbook.

Please note that *restoring* backups is not possible with the command line client. You need to use our standalone CLI tool called Akeeba UNiTE. Restoring a backup requires overwriting the current site. Due to Joomla loading some of its PHP files on-demand ("lazy-loading"), even when running under CLI, we can't offer a consistent backup restoration experience without running into the very real possibility that the restoration would fail because of Joomla lazy-loading a PHP file of the wrong version or the wrong path because the site has changed while running the restoration command. Akeeba UNiTE sidesteps this problem by being self-contained and running parallel to but outside of Joomla itself.

Also note that the Akeeba Backup Command Line Interface is only available on Joomla 4.0 or later and requires having the Console - Akeeba Backup plugin published. The requirement for Joomla 4.0 or later is due to the fact that the Joomla CLI application was not available in previous versions of Joomla. The requirement for the plugin has to do with how Joomla itself works under the hood to detect and make available third party commands for the Joomla CLI application.

1. Common conventions

You can access the Akeeba Backup CLI using the Joomla CLI application. This is the file `joomla.php` in your site's `cli` directory.

First of all, you need to know two things: how you can run PHP CLI on your server and the path to your site. If unsure, ask your host or system administrator. Assuming that you can run PHP on your server by typing **/usr/bin/php** and that your site's path is `/home/mysite/public_html` you can run the following command to list all available Akeeba Backup CLI commands:

```
/usr/bin/php /home/mysite/public_html/cli/joomla.php list akeeba
```

If you get an error reading

```
There are no commands defined in the "akeeba" namespace.
```

you will need to publish the Console - Akeeba Backup plugin on your site first. If you get a different error you need to make sure that the PHP command you are using is correct, it is in fact the PHP CLI (not PHP CGI) binary and that the PHP version it runs is the same you are using on your web site.

Please note that most servers have multiple PHP versions installed. It's possible that the PHP version you are accessing with the command you are using is different than the one you are using on your web site. If unsure, please do ask your host. If the first level support of your host is unsure do ask them to escalate to an engineer. Server engineers can give you the correct path for PHP; it's trivial for them but may be indeed beyond the training or information available to a first level support agent.

The commands listed are in the format **akeeba:foo:bar**, i.e. three parts separated by semicolons. The first part is called the namespace of the command and it's always **akeeba** for Akeeba Backup CLI commands. A keen observer may notice that the namespace for various Joomla CLI commands tends to be the component directory without com_ in front. We noticed that too and we're following the same convention. The other two parts further identify the command you need to run. For example, listing the latest backups taken with Akeeba Backup requires running a command like this:

```
/usr/bin/php /home/mysite/public_html/cli/joomla.php akeeba:backup:list
```

If you need a quick refresher on what each command does and/or what arguments and options it takes you can use the built-in **help** command on it:

```
/usr/bin/php /home/mysite/public_html/cli/joomla.php help akeeba:backup:list
```

All commands set the exit status code upon completion. Commands that executed successfully return an exit status of 0 (zero). Non-zero results mean that an error occurred while executing the command. The exit status reference can be found in the following sections of this documentation.

Some Akeeba Backup CLI commands accept a `format` parameter. This tells Akeeba Backup CLI which output format to use for the information it returns. The following formats are supported (not all format are supported by all commands; please consult the command reference further ahead in this documentation):

table, text
: Human readable text. This is meant for humans to read. It's not very useful for automation.

json
: The output is returned as a JSON string. You can feed this through commands such as jq [https://www.baeldung.com/linux/jq-command-json] or consume it in automation tools such as Ansible's json_query [https://docs.ansible.com/ansible/latest/user_guide/playbooks_filters.html#selecting-json-data-json-queries] filter.

csv
: The output is returned as CSV (Comma Separated Values) data. This can be fed to Excel, Apple Numbers, Google Sheets, LibreOffice Calc and other spreadsheets. It can be a very useful format for generating reports.

yaml
: ## Important

  You need to have installed and enabled the optional PHP YAML extension. This is not enabled by default on most servers. If the extension is not installed or installed but not enabled you will get an error.

  The output is returned as YAML. This is a lightweight, structured, machine-readable data format which can be consumed by automation tools.

var_dump
: The output goes through PHP's built-in var_dump() [https://www.php.net/manual/en/function.var-dump.php] function. This is mostly useful for debugging and we may ask you to use it if you request support and we suspect something odd is going on with the output data.

var_export
: The output goes through PHP's built-in var_export() [https://www.php.net/manual/en/function.var-export.php] function. It can be used as-is in a PHP script's variable assignment. This can be useful for home-grown automation scripts written in PHP.

count
: Returns the number of items which would be output. This is useful for automation when used with the various **akeeba:*:list** commands. For example, if you need to know whether there are any backups taken the last week you can use the akeeba:backup:list command using the --from and --to options to limit the search within the last week and --format=count to get the number of backups taken in that period. If it's 0 no backups were taken. If it's non-zero this many backups were taken.

2. Command reference

Akeeba Backup CLI offers a relatively large number of commands which allow you to do almost everything you can do through the graphical user interface. For simplicity's sake we have grouped them under different categories, depending on what is the target of the command.

2.1. Backup record management

These commands allow you to manage backup records. A backup record is any backup attempt be it fully successful, partially successful (the backup ran but it didn't upload to remote storage), failed or pending. They are equivalent to the Backup Now and Manage Backups pages of the component.

2.1.1. akeeba:backup:take

Note

Only works with Akeeba Backup Professional.

What it does: Takes a backup with Akeeba Backup.

Syntax: akeeba:backup:take [--profile=PROFILE_ID] [--description=DESCRIPTION] [--comment=COMMENT] [--overrides=OVERRIDES]

Arguments

This command takes no arguments.

Options

--profile=PROFILE_ID	Optional. Which backup profile to use when taking a backup. Uses the default backup profile (1) when omitted. You need to specify the numeric backup profile. For example, to take a backup with profile #2 you need to specify `--profile=2`.
--description=DESCRIPTION	Optional. Short description for the backup record, accepts backup naming variables. Uses the default description when omitted. The description is displayed in the Manage Backups page and when listing backup records. It's meant to offer you a quick refresher of what the backup is for. For example: `--description="Taking a backup before updating Joomla"`
--comment=COMMENT	Optional. A longer, plain text comment describing the backup. This is displayed as a tooltip in the Manage Backups page and returned when listing backup records. It's meant for more in-depth information. Typically you'd set a shell variable and pass it to this option. For example `--comment="$MY_VARIABLE"` for the Bash (default Linux) shell.
--overrides=OVERRIDES	Optional. Override configuration parameters while taking the backup. Please consult the Native CRON Script documentation for more information on how to use this option.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 All good. The backup is successful and emitted no warnings.

1 The backup failed to complete. An error occurred while taking the backup.

2 The backup is successful but warnings were emitted.

2.1.2. akeeba:backup:list

What it does: Lists backup records known to Akeeba Backup.

This lists all backup attempts regardless of whether they are successful, ended in an error, are locally stored, remotely stored or their files no longer exist. It's the equivalent to the Manage Backups page.

Syntax: akeeba:backup:list [--from=FROM] [--limit=LIMIT] [--format=FORMAT] [--description=DESCRIPTION] [--after=AFTER] [--before=BEFORE] [--origin=ORIGIN] [--profile=PROFILE] [--sort-by=SORT-BY] [--sort-order=SORT-ORDER]

Arguments

This command takes no arguments.

Options

--from=FROM	Optional.How many backup records to skip before starting the output. This is typically used in conjunction with `--limit`. For example `--limit=20 --from=40` to display up to 20 backup records starting with the 40th record.
--limit=LIMIT	Optional. Maximum number of backup records to display. Up to this many backup records will be returned at a time. It is possible that less or no records are returned. However, no more than this many records will be returned at a time.
--format=FORMAT	Optional. The output format, as described above. Possible values: `table`, `json`, `yaml`, `csv`, `count`
--description=DESCRIPTION	Optional. Listed backup records must match this (partial) description. Note that this is a partial, case-insensitive match. --description="day" will match backup records with the descriptions "Before **day**light savings", "**Day**-to-day backup" and "Added **Day**ton to the site"
--after=AFTER	Optional. List backup records taken after this date. The date and time must be specified in MySQL format, e.g. "2020-12-01 13:45:23" for December 1st, 2020 at 23 seconds past 1:43pm. The date and time are **always** expressed in the GMT (UTC) timezone.
--before=BEFORE	Optional. List backup records taken before this date. The date and time must be specified in MySQL format, e.g. "2020-12-01 13:45:23" for December 1st, 2020 at 23 seconds past 1:43pm. The date and time are **always** expressed in the GMT (UTC) timezone.
--origin=ORIGIN	Optional. List backups taken from this origin only. The possible origins are:

- **backend**. Backups taken from the backend of the site.
- **frontend**. Backups taken with the legacy frontend backup or the akeeba-altbackup.php script.
- **json**. Backups taken with the Akeeba Backup JSON API or the Joomla JSON API.
- **cli**. Backups taken with the akeeba-backup.php CLI script or the Akeeba Backup CLI.

--profile=PROFILE
: Optional. List backups taken with this profile.

  Give the numeric profile ID. For example, `--profile=2`

--sort-by=SORT-BY
: Optional. Sort the backups by this column. Possible columns:

  - **id**. The backup identity number. This is a positive integer that increases monotonically, i.e. each new backup attempt gets an ID that is 1 higher than the previous backup attempt.
  - **description**. The short description of the backup.
  - **profile_id**. The profile number.
  - **backupstart**. The date and time (UTC) the backup started on.

--sort-order=SORT-ORDER
: Optional. Sorting order.

  Use `asc` for ascending order, `desc` for descending order.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

Note that a zero exit code is returned even if the command returns no records. If you want to test whether records matching your criteria exist use --format=count and check whether the output of the command is a non-zero integer.

2.1.3. akeeba:backup:info

What it does: Lists a backup record known to Akeeba Backup given its backup identity (ID) number.

Syntax: akeeba:backup:info ID [--format=FORMAT]

Arguments

This command takes one mandatory argument.

ID The numeric ID of the backup.

Options

--format=FORMAT
: Optional. The output format, as described above.

  Possible values: `table`, `json`, `yaml`, `csv`, `count`

Example

```
php joomla.php akeeba:backup:info 123 --format=json
```

Returns the information for backup number 123 encoded as JSON.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

No error is returned if you give it an invalid backup ID.

2.1.4. akeeba:backup:modify

What it does: Modifies a backup record known to Akeeba Backup. This is used to change the description and / or comment of a backup record.

Syntax: akeeba:backup:modify ID [--description=DESCRIPTION] [--comment=COMMENT]

Arguments

This command takes one mandatory argument.

ID The numeric ID of the backup.

Options

--description=DESCRIPTION Optional. Change the short description to this value.

--comment=COMMENT Optional. Change the backup comment to this value.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 The backup ID you provided does not exist.

2 You gave neither --description nor --comment. You need to provide one or both for the command to do anything at all.

3 Cannot modify backup record. Further information is provided in the error message. This usually happens when there's a database issue preventing you from saving the modified backup record back to the database.

2.1.5. akeeba:backup:delete

What it does: Deletes a backup record known to Akeeba Backup, or just its files.

Important

This command only deletes **locally** stored files, i.e. backup archives stored on the same server Akeeba Backup is installed. It will NOT delete any backup archives stored remotely e.g. on Amazon S3, Dropbox and so on.

Syntax: akeeba:backup:delete ID [--only-files]

Arguments

This command one mandatory argument.

ID The numeric ID of the backup.

Options

--only-files Optional. Only delete the backup files stored on the site's server, not the record itself.

If you do not give this option both the locally stored backup archives and the backup record itself are deleted.

When you give this option only the locally stored backup archives will be deleted. The backup record remains in the database and is now marked as Obsolete.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 Cannot delete. Further information is provided in the error message.

2.1.6. akeeba:backup:upload

Note

Only works with Akeeba Backup Professional.

What it does: Retry uploading a backup to the remote storage.

When a backup fails to upload to the remote storage engine you can rectify the problem that caused the upload failure – usually a server configuration issue or just a dropped network connection – and use this command to upload the remaining parts from your server to the remote storage. It can also be used to re-upload a previously remotely stored backup you have retrieved locally using the akeeba:backup:fetch command.

Syntax: akeeba:backup:upload ID

Arguments

This command takes one mandatory argument.

ID The numeric ID of the backup to retry uploading.

Options

This command does not take any options.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 You have supplied an invalid backup record ID.

2 Retrying the upload failed. Further information is provided in the returned error message.

2.1.7. akeeba:backup:fetch

Note

Only works with Akeeba Backup Professional.

What it does: Download a backup from the remote storage back to your server.

Please note that not all remote storage engines may implement this feature. It is also possible that the credentials you use to connect to the remote storage server / service do not give you permissions to download backup archives, for example if you're using Amazon S3 with an IAM user which only allows writing to but not reading from files. Do keep in mind that it's always possible that the remotely stored file has been deleted outside Akeeba Backup, therefore trying to retrieve it will result in a reasonably self-understood error.

Please also keep in mind that this is only meant to work with Post-processing engines, NOT archiver engines. Akeeba Backup Professional includes archiver engines which let you upload the raw site files instead of a backup archive to a remote FTP or SFTP server. These backups CAN NOT be retrieved back to the server. They are immediately considered "Obsolete" since no backup archive was generated during the backup process.

Syntax: akeeba:backup:fetch ID

Arguments

This command takes one mandatory argument.

ID The numeric ID of the backup to retrieve from the remote storage back to the server.

Options

This command does not take any options.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 You have supplied an invalid backup record ID.

2 Trying to retrieve the remotely stored file failed. Further information is provided in the returned error message.

2.1.8. akeeba:backup:download

What it does: Returns a backup archive part for a backup record known to Akeeba Backup.

The command name is a bit of a misnomer. It doesn't really "download" anything rather than read a backup archive file which is already on the server and return its contents or write them to a file.

This can be used for automating restorations together with Akeeba UNiTE. Use akeeba:backup:list to find the ID of the backup archive you want to restore. Use akeeba:backup:download to store it with a predictable name in a directory whose name you already know. Use that information in an Akeeba UNiTE XML file and run Akeeba UNiTE against it to restore the site.

Syntax: akeeba:backup:download ID [PART] [--file=FILE]

Arguments

This command takes one mandatory and one optional argument. Please remember that arguments are positional, i.e. they need to appear in the order described below.

ID The numeric backup record ID you want to retrieve files for. Please remember that the backup archive files MUST be present on your server. If you have a remotely stored backup archive remember to use akeeba:backup:fetch beforehand.

PART The part number to download. Default: 1. This is useful when your backup archive consists of more than one files, called parts. You need all the parts to be present to extract the backup archive and restore it.

Options

--file=FILE Optional. The absolute path to the file where the backup archive's contents will be written to.

If omitted, akeeba:backup:download returns the backup archive's contents as its output (in STDOUT). This behaviour can be useful for using shell pipes to transfer the file between hosts [https://possiblelossofprecision.net/?p=444].

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 You have supplied an invalid backup record ID.

2 There are no locally stored backup archive files for this backup record.

3 The part number you supplied is less than 1 or greater than the total number of parts in the backup archive.

4 The part number you supplied exists but its corresponding file cannot be found on the server. This typically means that you have accidentally deleted part files (.j01, .j02, ... or .z01, .z02, ...). Another possibility is that your backup archive failed to completely upload to remote storage or failed to be completely retrieved from remote storage.

5 The part number you supplied exists, its file is found on the server but Akeeba Backup CLI cannot open it for reading. This typically means that the user CLI is running under does not have read permissions on the backup archive. Most likely you took your backup using the backend, frontend or remote JSON API i.e. through a web server and the web server runs under a *different* user than your command line.

6 The output file you specified with --file=FILE cannot be written to. Please check that the path is correct and that the user you are currently logged in as has write permissions to that folder.

2.2. Log management

These commands allow you to manage the log files which are generated when taking backups.

2.2.1. akeeba:log:list

What it does: Lists log files known to Akeeba Backup.

This command lists all log *files* which are found in the backup output directory of a selected backup profile. In short, it lists files on your server's disk. This has a few corollaries you should be aware of.

You may see some log files which do not directly correspond to a backup record such as the generic log file kept for each backup origin (backend, frontend, CLI and JSON API).

You may also see log files from backup records which have been deleted outside Akeeba Backup, e.g. after restoring an older backup.

If you have changed the output directory of a backup profile the log files in the old output directory will NOT be listed.

If you have deleted a log file outside of Akeeba Backup it will not be listed.

LIkewise, this command may not list all log files for all of your backup profiles if one or more backup profiles use a different output directory.

It is possible that one or more backup profiles share the same backup output directory. It's also possible that more than one Akeeba Backup installations (sites) share the same backup output directory – even though that's not recommended and can lead to difficult to troubleshoot issues. In case of a shared output directory you will see a list of all log files regardless of which Akeeba Backup installation or profile they came from.

Syntax: akeeba:log:list [PROFILE_ID] [--format=FORMAT]

Arguments

This command takes one optional argument.

PROFILE_ID Optional. The numeric profile ID. The profile's output directory will be used to list backup log files. If omitted the default backup profile ID (1) will be used.

Options

--format=FORMAT Optional. Change the output format as explained earlier.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

2.2.2. akeeba:log:get

What it does: Retrieve a log file known to Akeeba Backup.

This is the same as using the Download link in Akeeba Backup's View Log page.

The log file is output in STDOUT.

Syntax: akeeba:log:get PROFILE_ID LOG_TAG

Arguments

This command takes two mandatory arguments. Please remember that arguments are positional, i.e. they need to appear in the order described below.

PROFILE_ID The numeric profile ID. The profile's output directory will be used to locate the backup log file. If unsure, use the default profile ID (1).

LOG_TAG The tag of the log file you want to download. This tag is returned from the akeeba:log:list command.

Options

This command takes no options.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

2.3. Exclude and include filters management

Each Akeeba Backup profile has several filters which tell it which files, directories and database tables to exclude. These are called exclusion filters. This is equivalent to the Files and Directories Exclusion, Database Tables Exclusion, RegEx Files and Directories Exclusion (Pro version) and RegEx Database Tables Exclusion (Pro version) pages of the component.

Akeeba Backup Professional also has another type of filters called inclusion filters. These tell Akeeba Backup to include folders outside of your site's root or databases other than the one used by your site. These are equivalent to the Multiple Databases Definitions and Off-site Directories Inclusion pages of the component.

2.3.1. akeeba:filter:list

What it does: Get the filter values known to Akeeba Backup.

Syntax: akeeba:filter:list [--root=ROOT] [--profile=PROFILE] [--target=TARGET] [--type=TYPE] [--format=FORMAT]

Arguments

This command takes no arguments.

Options

--root=ROOT
: Optional. Which filter root to use.

  Defaults to [SITEROOT] or [SITEDB] depending on the --target option. Ignored for --type=include. The names of the inclusion filters *are* the roots you can use when listing exclusion (--type=exclude or --type=regex) filters.

  Tip: the filesystem and database roots are the "filter" column for --type=include. There are two special roots, [SITEROOT] (the filesystem root of the Joomla site) and [SITEDB] (the main database of the Joomla site).

--profile=PROFILE
: Optional. The backup profile to use. Default: 1.

  Keep in mind that filters are set *per backup profile*. They are not reused across backup profiles.

--target=TARGET
: Optional. The target of filters you want to list: `fs` (files and folders) or `db` (database tables).

--type=TYPE
: Optional. The type of filters you want to list: exclude, include or regex. The different types are:

  - exclude. Exclusion filters. Equivalent to the Files and Directories Exclusion (when --target=fs) or Database Tables Exclusion (when --target=db) pages of the component.
  - regex. Regular expression based exclusion filters. Equivalent to the RegEx Files and Directories Exclusion (when --target=fs) or RegEx Database Tables Exclusion (when --target=db) pages of the component. Only available in Akeeba Backup Professional.
  - include. Inclusion filters. Equivalent to the Multiple Databases Definitions (when --target=db) and Off-site Directories Inclusion (when --target=fs) pages of the component. Only available in Akeeba Backup Professional.

--format=FORMAT Optional. Sets the output format of the command as explained earlier.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 Unknown root specified.

2.3.2. akeeba:filter:delete

What it does: Delete a filter value known to Akeeba Backup.

Syntax: akeeba:filter:delete [-root=ROOT] [--filterType=TYPE] [--profile=PROFILE]

Arguments

This command takes no arguments.

Options

--root=ROOT Optional. Which filter root to use.

Defaults to [SITEROOT] or [SITEDB] depending on the --filterType option. Ignored for include filters. The names of the inclusion filters *are* the roots you can use when managing exclusion filters.

Tip: There are two special roots, [SITEROOT] (the filesystem root of the Joomla site) and [SITEDB] (the main database of the Joomla site).

--filterType=TYPE Optional. The type of filter you want to delete:

- files. Exclude individual files.
- directories. Exclude an entire folder, including its files and subdirectories.
- skipdirs. Exclude the subdirectories of this folder but not its files.
- skipfiles. Exclude the files contained in this folder but not its subdirectories.
- regexfiles. Exclude individual files matching this regular expression. Only available in Akeeba Backup Professional.
- regexdirectories. Exclude the subdirectories of and all files container in all folders matching this regular expression. Only available in Akeeba Backup Professional.
- regexskipdirs. Exclude the subdirectories of all folders matching this regular expression but not their files. Only available in Akeeba Backup Professional.
- regexskipfiles. Exclude the files contained in all folders matching this regular expression but not their subdirectories. Only available in Akeeba Backup Professional.
- tables. Exclude individual database tables and their contents.
- tabledata. Exclude the contents of individual tables but not the table itself (it backs up the table's structure).

- regextables. Exclude all database tables whose name matches this regular expression and their contents. Only available in Akeeba Backup Professional.
- regextabledata. Exclude the contents of all database tables whose name matches this regular expression but not the table itself (it backs up the tables' structure). Only available in Akeeba Backup Professional.
- extradirs. Include folders outside of the site's root. Only available in Akeeba Backup Professional.
- multidb. Include databases other than the one used by the site. Only available in Akeeba Backup Professional.

--profile=PROFILE — Optional. The backup profile to use. Default: 1.

Keep in mind that filters are set *per backup profile*. They are not reused across backup profiles.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 The root you entered cannot be found.

2 You have Akeeba Backup Core but the filter you're trying to manage is only available in Akeeba Backup Professional.

3 Could not delete filter. The returned error message contains further information as to why.

2.3.3. akeeba:filter:exclude

What it does: Set an exclusion filter to Akeeba Backup.

Syntax: akeeba:filter:exclude VALUE [-root=ROOT] [--filterType=TYPE] [--profile=PROFILE]

Arguments

This command takes one mandatory argument.

VALUE — The value of the filter to set. This is the folder or table name to exclude, or the regular to use (for regex* filter types).

Options

--root=ROOT — Optional. Which filter root to use.

Defaults to [SITEROOT] or [SITEDB] depending on the --filterType option. Ignored for include filters. The names of the inclusion filters *are* the roots you can use when managing exclusion filters.

Tip: There are two special roots, [SITEROOT] (the filesystem root of the Joomla site) and [SITEDB] (the main database of the Joomla site).

--filterType=TYPE — Optional. The type of filter you want to set:

- files. Exclude individual files.

- directories. Exclude an entire folder, including its files and subdirectories.
- skipdirs. Exclude the subdirectories of this folder but not its files.
- skipfiles. Exclude the files contained in this folder but not its subdirectories.
- regexfiles. Exclude individual files matching this regular expression. Only available in Akeeba Backup Professional.
- regexdirectories. Exclude the subdirectories of and all files container in all folders matching this regular expression. Only available in Akeeba Backup Professional.
- regexskipdirs. Exclude the subdirectories of all folders matching this regular expression but not their files. Only available in Akeeba Backup Professional.
- regexskipfiles. Exclude the files contained in all folders matching this regular expression but not their subdirectories. Only available in Akeeba Backup Professional.
- tables. Exclude individual database tables and their contents.
- tabledata. Exclude the contents of individual tables but not the table itself (it backs up the table's structure).
- regextables. Exclude all database tables whose name matches this regular expression and their contents. Only available in Akeeba Backup Professional.
- regextabledata. Exclude the contents of all database tables whose name matches this regular expression but not the table itself (it backs up the tables' structure). Only available in Akeeba Backup Professional.

--profile=PROFILE — Optional. The backup profile to use. Default: 1.

Keep in mind that filters are set *per backup profile*. They are not reused across backup profiles.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 The root you entered cannot be found.

2 You have Akeeba Backup Core but the filter you're trying to manage is only available in Akeeba Backup Professional.

2.3.4. akeeba:filter:include-database

Note

Only works with Akeeba Backup Professional.

What it does: Adds an additional database to be backed up by Akeeba Backup.

Syntax: akeeba:filter:include-database [--profile=PROFILE] [--dbdriver=DBDRIVER] [--dbport=DBPORT] [--dbusername=DBUSERNAME] [--dbpassword=DBPASSWORD] [--dbname=DBNAME] [--dbprefix=DBPREFIX] [--check]

Arguments

This command takes no arguments.

Options

--profile=PROFILE	Optional. The backup profile to use. Default: 1. Keep in mind that filters are set *per backup profile*. They are not reused across backup profiles.
--dbdriver=DBDRIVER	Optional. The database driver to use to connect to the database. One of • mysqli. Use PHP's MySQLi database driver. This is the modern, MySQL-specific driver. Recommended. • pdomysql. Use the PHP Data Object (PDO) connector for MySQL. This may be required on servers where the MySQLi driver is not available. • mysql. Effectively, this is a synonym to mysqli. There are two reasons for having that. First, it's all too easy to mistype mysqli as mysql (without the trailing i). Second, sites updated from Joomla 3 running on PHP 5.x may have been using that old driver that was removed in PHP 7.0 and later. We don't want to break existing installations so mysql is now nothing more than an alias to mysqli.
--dbport=DBPORT	Optional. The database server port. It uses the default MySQL port (3306) if not defined.
--dbusername=DBUSERNAME	Optional. The database connection username.
--dbpassword=DBPASSWORD	Optional. The database connection password,
--dbname=DBNAME	Optional. The database name.
--dbprefix=DBPREFIX	Optional. The common prefix of the database table names. Setting this is not required for taking a backup of the database. However, without setting up a prefix you will NOT be able to change the common table name prefix when restoring the backup.
--check	Optional. Check the database connection before adding the filter. We strongly advise you to use this option as it will catch mistyping something in the options. There are, however, legitimate reasons to not use this option e.g. if you are setting up a backup profile before setting up the external database it will be backing up.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 You have Akeeba Backup Core. This feature is only available in Akeeba Backup Professional.

2 A database inclusion filter with the same connection information already exists.

3 You passed the --check option but PHP reports it cannot connect to the database you specified using the connection information provided.

4 Setting the filter failed.

2.3.5. akeeba:filter:include-directory

> **Note**
>
> Only works with Akeeba Backup Professional.

What it does: Add an additional off-site directory to be backed up by Akeeba Backup.

Syntax: akeeba:filter:include-directory DIRECTORY [--profile=PROFILE] [--virtual=VIRTUAL]

Arguments

This command takes one mandatory argument.

DIRECTORY	The absolute path to the off-site directory (folder) you want to include in the backup.

Options

--profile=PROFILE	Optional. The backup profile to use. Default: 1. Keep in mind that filters are set *per backup profile*. They are not reused across backup profiles.
--virtual=VIRTUAL	Optional. The subfolder inside the backup archive where these files will be stored. All off-site folders are backed up inside the backup archive as a subdirectory inside a folder called "external_files" (the name of this folder can be configured in the Configuration page of the backup profile). So, if you were to enter `example` here, the contents of the directory would be backed up inside the backup archive under the path `external_files/example`. If you do not specify this option the name of the virtual folder will be determined automatically.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 You have Akeeba Backup Core. This feature is only available in Akeeba Backup Professional.

2 A folder inclusion filter with the same connection information already exists.

3 Setting the filter failed.

2.4. Profile configuration options management

These commands allow you to manage the configuration settings of your backup profiles, used to take backups.

2.4.1. akeeba:option:list

What it does: Lists the configuration options for an Akeeba Backup profile, including their titles.

Syntax: akeeba:option:list [--profile=1] [--filter=FILTER] [--sort-by=SORT_FIELD] [--sort-order=SORT_ORDER] [--format=FORMAT]

Arguments

This command takes no arguments.

Options

--profile=PROFILE	Optional. The backup profile to list configuration options for. Provide the numeric profile ID. If omitted, the default backup profile ID 1 is used.
--filter=FILTER	Optional. Only return records whose keys begin with the given filter. Use text e.g. `akeeba.core.`
--sort-by=SORT_FIELD	Optional. Sort the output by the given column: none, key, value, type, default, title, description, section Default: none
--sort-order=SORT_ORDER	Optional. Sorting order. `asc` for ascending, `desc` for descending.
--format=FORMAT	Optional. Output format: table, json, yaml, csv, count.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 Invalid profile ID specified.

2 Your filter resulted in no options being listed.

2.4.2. akeeba:option:get

What it does: Gets the value of a configuration option for an Akeeba Backup profile

Syntax: akeeba:option:get [--key=KEY] [--profile=PROFILE] [--format=FORMAT]

Arguments

This command takes no arguments.

Options

--key=KEY	Optional. The option key to retrieve.
--profile=PROFILE	Optional. The backup profile to use. Default: 1.
--format=FORMAT	Optional. Output format: text, json, print_r, var_dump, var_export.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 Invalid profile ID.

2 You have entered a partial key prefix. You need to provide a specific key which returns exactly one value.

2.4.3. akeeba:option:set

What it does: Sets the value of a configuration option for an Akeeba Backup profile

Syntax: akeeba:option:set --key=KEY --value=VALUE [--profile=PROFILE] [--force]

Arguments

This command takes no arguments.

Options

--key=WHATEVER	Mandatory. The option key to set.
--value=WHATEVER	Mandatory. The value to set the option to. Integer options must be within the acceptable range returned by akeeba:option:list. Boolean options accept the values 0, false, no or off for FALSE values and 1, true, yes or on for TRUE values. Enumerated (enum) options only accept the enumerated keys returned by akeeba:option:list. Any other option type cannot be set to any value. These are typically hidden values used internally, action buttons or other non-value interface elements listed in the backup engine's options definition (JSON) files so that they are rendered in the Configuration user interface.
--profile=SOMETHING	Optional. The backup profile to use. Default: 1.
--force	Optional. Some options are "protected" and cannot be normally changed. Specify --force to allow their values to be set anyway.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 Invalid profile ID.

2 The specified key does not exist.

3 The value you have entered is outside the acceptable range.

4 This is a protected key. You need to specify --force to set it.

5 Internal error. Cannot set the value.

2.5. Backup profile management

These commands let you manage the backup profiles themselves.

2.5.1. akeeba:profile:list

What it does: Lists the Akeeba Backup backup profiles.

Syntax: akeeba:profile:list [--format=FORMAT]

Arguments

This command takes no arguments.

Options

--format=FORMAT	Optional. Output format: table, json, yaml, csv, count.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

2.5.2. akeeba:profile:modify

What it does: Modifies an Akeeba Backup profile

Syntax: akeeba:profile:modify --id=ID [--description=DESCRIPTION] [--quickicon=QUICKICON]

Arguments

This command takes no arguments.

Options

--id=ID	Mandatory. The numeric profile ID to modify.
--description=DESCRIPTION	Optional. Change the description of the backup profile to the given string. Either --description or --quickicon must be specified for the command to be valid.
--quickicon=QUICKICON	Optional. Should there be a one click backup icon for this backup profile? Set to 0 to disable it, set to 1 to enable it. Either --description or --quickicon must be specified for the command to be valid.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 Invalid profile ID.

2 Internal error. Could not modify the backup profile.

2.5.3. akeeba:profile:reset

What it does: Resets an Akeeba Backup profile. This resets the Configuration options to the factory default values and removes all filters.

Syntax: akeeba:profile:reset --id=ID [--filters]

Arguments

This command takes no arguments.

Options

--filters	Optional. Should I reset the filters as well? When present the filters will be reset. When not present only the Configuration options will be reset.
--id=ID	Mandatory. The numeric backup profile ID to reset. More information.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 Invalid profile ID.

2 Internal error.

2.5.4. akeeba:profile:create

What it does: Create a new, not yet configured, backup profile

Syntax: akeeba:profile:create [--description=DESCRIPTION] [--quickicon=QUICKICON] [--format=FORMAT]

Arguments

This command takes no arguments.

Options

--description=DESCRIPTION	Optional. Description for the new backup profile. Default: "New backup profile".
--quickicon=QUICKICON	Optional. Should the new backup profile have a one-click backup icon? Default: 1
--format=FORMAT	Optional. The format for the response. Use JSON to get a JSON-parseable numeric ID of the new backup profile. Values: text, json

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

2 Internal error

2.5.5. akeeba:profile:copy

What it does: Creates a copy of an Akeeba Backup profile

Syntax: akeeba:profile:copy --id=ID [--filters] [--description=DESCRIPTION] [--quickicon=QUICKICON] [--format=FORMAT]

Arguments

This command takes no arguments.

Options

--id=ID	Mandatory. The numeric ID of the profile to copy
--filters	Optional. Include filters in the copy? When this option is not present the filters will NOT be copied.
--description=DESCRIPTION	Optional. Description for the new backup profile. Uses the old profile's description if not specified.
--quickicon=QUICKICON	Optional. Should the new backup profile have a one-click backup icon? Copies the old profile\'s setting if not specified.
--format=FORMAT	Optional. The format for the response. Use JSON to get a JSON-parseable numeric ID of the new backup profile. Values: text, json

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 Invalid profile ID.

2 Internal error.

2.5.6. akeeba:profile:delete

What it does: Deletes a backup profile.

Syntax: akeeba:profile:delete --id=ID

Arguments

This command takes no arguments.

Options

--id=ID Mandatory. The numeric ID of the profile to delete.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 Invalid profile ID.

2 Internal error

2.5.7. akeeba:profile:export

What it does: Exports the configuration information and filters of a backup profile as a JSON file. You can import it from the Akeeba Backup user interface or using the akeeba:profile:import command on this or any other site.

This command echoes a JSON document.

Syntax: akeeba:profile:export --id=ID [--filters]

Arguments

This command takes no arguments.

Options

--id=ID Mandatory. The numeric ID of the profile to modify.

--filters Optional. Include the filter settings?

If this option is not present only the configuration settings are exported, not the filters.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 An error occurred.

2.5.8. akeeba:profile:import

What it does: Imports a previously exported backup profile.

You can either specify a JSON file using the --fileOrJSON option OR you can pipe JSON data to the standard input.

Syntax: akeeba:profile:import [--fileOrJSON=FILEPATH] [--format=FORMAT]

Arguments

This command takes no arguments.

Options

--fileOrJSON=FILEPATH Optional. A path to an Akeeba Backup profile export JSON file or a literal JSON string. Uses STDIN if omitted which allows you to pipe raw JSON into the command's standard input.

Pipe example: **cat profile.json | php /path/to/joomla.php akeeba:profile:import**

--format=FORMAT	Optional. The format for the response. Use json to get a JSON-parseable numeric ID of the new backup profile. Values: json, text

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

2.6. Component options management

These commands let you manage the component Options.

2.6.1. akeeba:sysconfig:list

What it does: Lists the Akeeba Backup component-wide options

Syntax: akeeba:sysconfig:list [--format=FORMAT]

Arguments

This command takes no arguments.

Options

--format=FORMAT	Optional. Output format: table, json, yaml, csv, count.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

2.6.2. akeeba:sysconfig:get

What it does: Gets the value of an Akeeba Backup component-wide option

Syntax: akeeba:sysconfig:get --key=KEY [--format=FORMAT]

Arguments

This command takes no arguments.

Options

--key=KEY	Mandatory. The option key to retrieve
--option=SOMETHING	Optional. Output format: text, json, print_r, var_dunp, var_export.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 Invalid option key.

2.6.3. akeeba:sysconfig:set

What it does: Sets the value of an Akeeba Backup component-wide option

Syntax: akeeba:sysconfig:set --key=KEY --value=VALUE [--format=FORMAT]

Arguments

This command takes no arguments.

Options

--key=KEY	Mandatory. The option key to set.
--value=VALUE	Mandatory. The option value to set.
--format=FORMAT	Optional. Output format: text, json, print_r, var_dunp, var_export.

Exit codes

One of the following exit codes will be set when the command finishes running:

0 The command completed without an error.

1 Invalid key.

Chapter 5. Miscellaneous Extensions (Modules, Plugins)

1. Action Log

Note

This feature is only available in Akeeba Backup Professional.

Displayed on the Plugin Manager as Action Log - Akeeba Backup.

Joomla includes the User Actions Log feature which allows you to keep an audit log of important actions taken by (typically administrative) users of your site. This plugin lets you record and display any action taken by the user in Akeeba Backup in your site's administrator area.

Please note that due to the way Joomla works you must have this plugin published not just to record user actions but also *to display them*. If you unpublish this plugin you may see untranslated strings in the actions log. This is how Joomla works, not a bug.

2. Quick Icon

Note

Displayed on the Plugin Manager as Quick Icon - Akeeba Backup Notification.

This plugin displays a notification icon in Joomla's administrator notification module, displaying the backup status. If your latest backup, taken with any profile, was taken longer than the configured amount of time in this plugin the icon will change color and display a warning message that your backup is out of date. Clicking on the icon will start a backup with the backup profile you have configured in this plugin.

3. Backup on Update

Note

Displayed on the Plugin Manager as System - Backup on update

All current Joomla! 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 major and minor versions of Joomla! some extensions on your site might experience problems or make your site completely inaccessible. It's always a good idea to take a backup of your site *before* upgrading Joomla!. Yet, how many times did you forget to do it only to end up with an inaccessible site and a furious client? 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.

When the plugin is enabled you will see a message reminding you of the status of Backup on Update when you visit the Joomla! Update component. You can temporarily disable (and re-enable) the backup on update feature by clicking the button in that message.

Chapter 6. Restoring backups and general guidelines

We kindly request our users to read the general guidelines before filing a support ticket, a bug report or giving up. Most frustrating and "inexplicable" problems end up being a simple case of misunderstanding how things work, making the wrong configuration choice or a server issue which can typically be resolved by asking your host nicely. The guidelines below aim to prevent most of the common issues or, if prevention is unlikely, at least help you identify the root cause and tell you how to fix it.

1. General guidelines for backing up and restoring your site

Restoring sites with Akeeba Backup is easy. Sometimes it may even be *too easy* which makes you prone to making obvious mistakes due to the implied complacency of using third party software to restore your site. In the following paragraphs we explain how to avoid the most common mistakes. This is a long read but we recommend that you read and understand it. We will not accept any "bug" reports about these issues which are, ultimately, user errors. If you need clarifications about any of these issues, however, do feel free to ask us (and be specific about what you didn't understand). We will be happy to help you better understand the issue.

Make sure you have backups before you need to restore your site. Over the years we've come across many people who were furious that having installed, but not having used, our backup software didn't help them when their site got destroyed. Don't be that person. Take frequent backups of your site, at the very least before updating Joomla! and its extensions and after making any substantial change to your site. It's dull, it's boring, it's a chore, *it will save your life one day*. Always keep at least one copy (ideally: three copies) of your backups *outside of your site* and ideally outside of your computer too. What is the point of having your backups stored on your site's server when the server's disk crashes or the hosting company goes bankrupt?

Test your backups periodically. Maybe you excluded a database table you didn't mean to. Maybe a folder was unreadable but you ignored the warning. Maybe the FTP program you are using to download your backups is broken. Maybe you accidentally set the temp-directory or logs file directory of your site to your site's root or another important system folder, ending up excluding it automatically in the process. Maybe there was a bug in the version of Akeeba Backup you were using, it created a corrupt archive and you didn't update to the next version that fixed it (it happens once every 3-4 years, we usually fix the issue within 24 hours). No matter what happened, a corrupt backup can ruin your day. Test your backups periodically to make sure that they actually work.

Multipart backup archives. Some backup archives may consist of more than one files. These are called multipart backup archives. You MUST download all part files to have a backup archive set which can be used to extract all files and restore a site. All files have the same base name, for example site-www.example.com-20220925-105300-abcdef012345, but different extensions. JPA archives have the extensions .jpa, .j01, .j02, …; JPS archives have the extensions .jps, .j01, .j02, …; ZIP archives have the extensions .zip, .z01, .z02, … The part files of a multipart backup archive are NOT separate archives; you cannot extract its one of them individually. The are all parts of the same archive and they all need to be present for the archive to be able to be extracted. The order of the parts is .j01, .j02, …, .jpa for JPA archives; .j01, .j02, …, .jps for JPS archives; and .z01, .z02, …, .zip for ZIP archives. That is to say, the numbered part files come first in the numeric order defined by their extension, the file with the .jpa, .jps or .zip extension comes last. You can extract multipart archives using the integrated restoration but also Akeeba Kickstart Core, our free of charge archive extraction tool. It can run on a web server — even a local web server created with MAMP, WAMPserver, XAMPP etc — or, for expert users, on the command line.

Akeeba Backup archives are self-contained. The backup archive contains a copy of your site's files, an export of its database data *and the restoration script to put everything together*. This is very different to most other backup software

in that the restoration script is inside the backup itself, not a separate thing you need to install. The only thing you need to do before restoring a backup is extract the backup archive. This can be done with Akeeba Kickstart (single file web application), Akeeba UNiTE (automatic site restoration running under the CLI) etc.

Restoring a backup overwrites the entire site (Joomla! installation). It cannot be used to transfer content between different Joomla! installations. An Akeeba Backup archive contains your entire site, files and database contents. Restoring a backup will overwrite the files and database tables that have the same name as those included in the backup. As a result it cannot be used to transfer content (e.g. articles) between two different Joomla! installations. It will transfer the entire Joomla! installation instead.

You DO NOT need to install Joomla! before restoring a backup. As explained above, an Akeeba Backup archive contains your entire site and the restoration script required to restore it. You do not need to have a functional site to restore a backup. The whole point of Akeeba Backup is restoring a backup when the site is no longer functional. Moreover, we recommend NOT installing Joomla! before restoring a backup because of the point described below about mixing different versions of Joomla!.

Restoring a backup does NOT delete files on your site which don't exist in your backup. If a file exists on the site you are restoring to but not inside the backup archive itself it will not be overwritten. This is important in two cases. First, when you are restoring a backup after your site is hacked (*unhacking a site*). The hacker may have left behind files which can be used to re-hack your site. These files will not be deleted automatically. Use a component, such as Admin Tools Professionals with its PHP File Change Scanner, or a third party service such as Sucuri to detect and remove such files. Furthermore, if you are trying to *replace a site* with a new one. If the old site is based on an older version of Joomla!, a different CMS (e.g. WordPress), is a static site or had different extensions / templates installed these files will be left behind. In both of these uses cases you should take a copy of your site's files, then delete all files and folders, create a new database and *finally* restore the backup.

Make sure you are backing up only the files that belong to the site you want to back up. This is very important when you are backing up a site in the root of your domain but you have additional sites in subdirectories (or subdomains whose root directory is a subdirectory of your main site). Akeeba Backup does NOT assign meaning to your directory names! It does not know that the directory `old_site` has a copy of your site from two years ago or that the folder `gju4rl` has the files for a subdomain you use to share files with your cousin who lives in another country. Akeeba Backup will backup all files and folders under the root of your site unless you tell it otherwise with the Files and Directories Exclusion feature. In any other case restoring your main site would overwrite the files of all the sites in subdirectories under your main site's root!

Make sure you are backing up only the database tables that belong to the site you want to back up. This is very important when you are backing up a site that shares a database with another site. Akeeba Backup does NOT assign meaning to the prefix used by your database tables. All tables in the database used by your site will be backed up *regardless* of their database prefix. If you use the same database for the tables of other sites, e.g. sites installed in subdorectories or subdomains on the same hosting account, you MUST exclude them manually with the Database Tables Exclusion feature. Otherwise restoring your site would overwrite the database of all of the other sites sharing the same database.

Keep copies of your backup archives outside of your site. The reason is simple: human error. If you mess up the restoration and, at the same time, somehow delete your only backup archive you are in deep trouble. Always keep several copies of your backup archives before doing a restoration. We recommend having at least THREE copies: on your computer, on a cloud storage provider (e.g. Dropbox, OneDrive, Google Drive, ...) and on a USB stick you keep in a sealed envelope in your desk's drawer. You can never have too many copies of your backups - but you *can* have too few!

Always practice the restoration on a test server / subdomain before doing it on your live site. Do you know why the military has so many drills? By repeating the same task over and over they get to perform it near perfectly, every single time, without thinking - even when bullets are flying around them. You don't want to start learning how to restore backups when your site is down. At that point you want your site restored, pronto. That's why you should practice restoring backups before you *need* to restore them. Practice on a test server, e.g. a MAMP, WAMPServer or

XAMPP installation on your computer, or a subdomain on a server under your control. Once you have done this a few times you can restore any site, anywhere, without much thinking and without mistakes.

Make sure you have a database. Even though Akeeba Backup's restoration script can try to create a database for you this WILL NOT work on most database servers for security reasons. Creating a database requires database server administrator (root) privileges which you typically do not have and, even if you do, *should not* use when restoring a site to a live server. Instead, create a database for your site in advance. You will also need to create a database user and give it the correct privileges as described in our troubleshooter [https://www.akeeba.com/documentation/troubleshooter/abidatabase.html].

Do not restore in a subdirectory of your main site. For example, if your site's root is in `public_html` do not restore to `public_html/dev`. The reason is that the `.htaccess` files, which tell Apache (your web server) how to server your site, *cascade*. That is, Apache will read all `.htaccess` files in all folders leading to the one hosting your site's `index.php` file. This *will* cause problems with the restored site which you will experience as 404, 403 and 500 error messages or blank pages. These have nothing to do with our software and / or the restoration. It's how your web server works. Use a subdomain instead.

If you are restoring on a subdomain, make sure that the subdomain's root directory is NOT a subdirectory of your main site. This is the same as the previous paragraph, really. Most hosting control panel software default to using a subdirectory of your site's root when creating a subdomain. For example, if your site is `www.example.com` and its root is `public_html` if you create the subdomain `dev.example.com` your hosting control panel will put its root in `public_html/dev`. Therefore you will have the problem we described above. In this case ask your host what is the best way to create a root folder for the subdomain next to `public_html`, not inside it.

Try restoring to as close a PHP version as possible. Not all third party extensions support all PHP versions. If your site was running on PHP 5.6 and you try to restore it on a server running PHP 5.3 or PHP 7.1 your site *may* break. This has, again, nothing to do with Akeeba Backup. Upholding the minimum / maximum PHP version requirements of the software running on your site is your responsibility. We have no way of knowing that information. All we can do is print out the PHP version of the site you backed up and the site you are restoring to during restoration. Everything else is up to you.

Don't try to restore to a different database technology. If your site runs on MySQL don't try to restore it on a server that only supports PostgreSQL or Microsoft SQL Server. Even though Joomla! 3 supports all of these database technologies they are incompatible with each other and you *cannot* transfer data between them. You can only restore a site on the same database technology it was backed up on. Clarification: MySQL, Percona and MariaDB are all using the same database technology, collectively called "MySQL". While you can a site between these different MySQL-type database servers we recommend against it. Subtle differences between them may cause restoration errors in some cases. In the few cases we can prevent that, we have added the necessary workarounds. There are some cases we can do nothing about. If you get a database restoration issue please check if you're trying to restore to a different MySQL-like database server than the one you backed up from.

Do not try to overwrite one Joomla! version family with a different one. Overwriting a major version with another (e.g. restoring a backup taken on Joomla! 3.7 on top of a site running Joomla! 2.5 or vice versa) or between different minor versions (e.g. restoring a backup taken on Joomla! 3.7 on top of a site running Joomla! 3.6 or vice versa) will NOT work. Joomla! moves files around between minor and major versions. Since the backup does not delete files not present in the backup archive this will end up with Joomla! being "confused" and malfunctioning. In these cases you should delete the existing files and folders (except, perhaps, user generated content) before restoring the backup. You can safely restore a sub-minor (path-level) version on top of another. For example, you can safely restore a Joomla! 3.7.5 site on top of a Joomla! 3.7.3 site or vice versa.

Pay attention when restoring to a different domain, subdomain or folder: you will need to enter the domain name, directory and database connection information where you are restoring to. If you don't pay attention you may overwrite a site you didn't intend to touch!

The restored site is a fully functional clone of your original site. There is no functional difference between a restored site and one you built from scratch. This means that you can always backup the restored site and then restore that new

backup on top of the original site. This makes Akeeba Backup ideal for live-to-development and development-to-live site transfers. If you are an advanced user such as a busy web agency do note that the process can be fully automated using Akeeba UNiTE: it can take a backup remotely, download it and restore it.

2. Guidelines for storing your backups remotely / "cloud backup"

Note

This only applies to Akeeba Backup Professional.

Uploading backups to a remote location requires setting up a Post-Processing Engine in the Configuration. By default, your backups are only stored locally, on the server where the site being backed up lives. If you want the backup to be uploaded to remote storage you have to go to the Configuration page and set up a Post-Processing Engine. As with all Configuration settings, this is set up *per backup profile*. Each profile can have a different post-processing setup - or no post-processing setup. Remember this if you're trying to figure out why your backups don't upload.

Uploading your backup archives can happen during or after taking the actual backup. All post-processing engines have an option to "Upload parts immediately". When this is enabled (checked), Akeeba Backup will upload each backup archive part file as it's finished being created. The first part of the backup is exempt from this rule: it is always uploaded after Akeeba Backup is done backing up your site. The first part contain special information about the number of part files and / or the number and size of files in the backup, information which is only known after the backup is complete. When the "Upload parts immediately" option is disabled (unchecked, the default state) Akeeba Backup will finish taking a backup of your site and only then will it upload the backup to remote storage. Therefore, when you are perceiving an issue with Akeeba Backup "not uploading your backups" first check if you have an issue preventing the backup to be taken at all!

A failed upload to remote storage does not cause the backup record to be reported as failed. As far as Akeeba Backup is concerned, backing up and uploading are two distinct operations. If the backup completes but the upload fails the backup record will appear as "OK" (green), NOT as Failed (red). If *both* the backup and upload are successful the backup record will appear as either OK (if the backup archive is kept on your site's server) or Remote (if the backup archive is deleted from your site's server, the default option). This distinction makes sense: if the backup is complete you can still restore your site from the generated backup archive.

Most failed uploads are caused by timeouts. PHP and your web server have time limits, i.e. the maximum time a PHP script 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. If that time is over either time limit your backup will fail. The time limit and the bandwidth are beyond our control. The only thing you can control is the size. Many post-processing engines support chunked uploading (breaking up the uploads in smaller bits and having the remote server piece together the file) and you can change their chunk size. A chunk size of 5 or 10 MB works best in most cases. For those post-processing engines which don't have an option for chunked uploads you will have to change the Part size for split archives in the Archiver Engine options. Again, a value of 5 or 10 MB works best in most cases. This setting will split our backup archive into multiple files (same base filename, the extensions are .j01, .j02, ..., .jpa; or .j01, .j02, ..., .jps; or .z01, .z02, ..., .zip;), the maximum size of each one being the value of this setting. To restore these backups just place ALL of these files in the same directory and choose the main .jpa, .jps or .zip file: the other parts are discovered and extracted automatically.

If you get no uploads / zero sized uploads but not error message, contact your host. We have seen many hosts putting a (broken) caching proxy in front of their web servers. Instead of letting Akeeba Backup communicate with the remote storage server they immediately return an HTTP 200 OK response *without contacting the remote storage server*. Unfortunately, for many remote storage services such as Dropbox, OneDrive and Google Drive *this would be the expected response when the upload succeeds*. How you can tell this happened? Check the Akeeba Backup log file. If the upload takes less than 2 seconds we GUARANTEE that your host is doing what we just described. Their

first level support may deny it; ask to escalate to a server technician. They will add a proxy server exception and your remote backups will work perfectly.

You can't upload to multiple locations. You can only set up a single post-processing engine. Multiple upload locations would increase the load on your server and the likelihood that something fails during backup. Moreover, this does not offer the kind of redundancy you might hope to achieve. Instead, use Dropbox, OneDrive or Google Drive to automatically download the backup archive to your computer. Use a regular desktop backup software to back up the local copies of your site's backups to a NAS.

If some part files of your backups failed to upload use the Manage remote backups in the Manage Backups page to retry uploading them. Sometimes a temporary network issue may prevent the upload from going through. Using the Manage remote backups to retry the upload usually works just fine!

If your uploads fail with long, cryptic errors about the signatures being wrong please check the time and the timezone of your server. Most remote storage engines require your server's time to be set within a reasonable accuracy to the true time. This can be automated on your server by setting and running the ntpd service. If your host hasn't done so the time will drift until it's so far off the true time that uploads will fail. If you get these cryptic error messages about signatures first *triple check* that your credentials are set up correctly in the post-processing engine options in the Configure page for the backup profile that fails. If you have triple checked them and found them to be working, contact your host and ask them to check the time and timezone on the server. It's silly, but this is the second most common cause of upload failures (after the part size discussed above) that we keep seeing.

3. Overview of the backup restoration procedure

Please watch our Video Tutorials [https://www.akeeba.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.

The backup restoration procedure generally consists of two discrete steps:

1. **Extraction of the backup archive**. The backups taken with the application are compressed to save space and, in the case of JPS archives, encrypted to protect their contents from prying eyes. The first thing you need is to extract the archive, getting access to the files contained in it.

   You can find information about the different ways to extract backup archives in Extracting your backup archives.

2. **Database restoration and site (re-)configuration a.k.a. "running the installer"**. All backups taken with Akeeba Backup contain a web-based restoration script called the installer. Its job is to restore the database backup, move the backed up off-site folders to their proper locations and, if your site script is supported, rewrite your site's basic configuration to match the new server.

   You can find information about the different restoration scripts in ANGIE: the Akeeba Backup restoration script.

Prerequisites

Before you begin the restoration procedure you will also need to have a database to hold your data and the connection information to it at hand. For this you need to create a database for your site's data, or note down the connection information to an existing database if you are installing on top of an existing site. You will need the following information:

- Database host name. This is usually `localhost`, but you may need to check with your host
- Database name. The name of the database you are restoring to. If you are on a host powered by cPanel or Plesk do note that the name of the database includes an account-specific prefix. If your account name is foo and the name of your database you asked to create is bar, the full database name is foo_bar.

- Database user name. The user name you use to connect to your database. The same thing about the naming prefix on cPanel and Plesk hosts is true for the username as well.
- Database user password.
- Your preferred table name prefix. This is not something your host will tell you, it's just a matter of your personal preference. You may use anything you want. It's best to pick a name consisting of three to four letters and a single trailing underscore, i.e. tst_ or test_. Do not use bak_ as it is a reserved prefix for keeping copies of replaced tables when you select the Backup old tables option in the installer later in the process.

  ### Warning

  DO NOT use any uppercase letters. There is a known issue with MySQL on case-insensitive filesystems such as on Windows and macOS which may make it difficult or outright impossible to restore a backup you take when a table contains uppercase letters. This is NOT a bug in Akeeba Backup, it's a known and well documented issue in MySQL itself.

If your host gives you such an option —or if you are using a local server— it's a good idea to set the default collation of the database to `utf8_general_ci`. If you are not given such an option, don't worry. The installer can work around this limitation with its Force UTF8 collation on tables option on MySQL databases. For other database types you have to specify the correct collation on the database when creating it.

Some PHP software, such as Joomla 3.7 and later, is using the `utf8mb4_general_ci` collation instead. This collation supports multi-byte Unicode characters such as Emoji, certain Tradition Chinese characters and so on. If you need to restore such a site please remember to turn on the Allow UTF8MB4 auto-detection option.

The server where you are restoring your backup must be using PHP 7.2.0 or later by default. Please note that many hosts claim to *support* a high version of PHP (e.g. PHP 8.0) but it's not the version they use by default. For instance, we've seen hosts which claim to support PHP 7.4, but the default version they are using is PHP 5.3. If the restoration fails with a notice that the PHP version is too old please do contact your host and ask them to make PHP version 7.2.0 or later as the default PHP version of your site. It's really easy for them: they have to change just one line in your site's configuration.

Alternatively, you can set the default PHP version before the restoration using your hosting control panel. Ask your host exactly how to do that. This typically creates a special line in your site's .htaccess file. Kickstart knows about it and keeps it active throughout the backup extraction. ANGIE, our restoration script, is also aware of this and lets you apply it to the restored site so that the restored site itself won't break because the default PHP version on your server is too old.

4. Extracting your backup archives

There are different ways to extract your backup archive, depending on its format and where you intend to restore it to. If you are restoring on the same hosting account as your Akeeba Backup installation the most convenient method is using the integrated restoration feature. If you are restoring to a different site or server using Akeeba Kickstart is the best option. Finally, if you are using a ZIP archive it is possible that some third party software will be able to extract it as well.

4.1. Using the integrated restoration feature (most common)

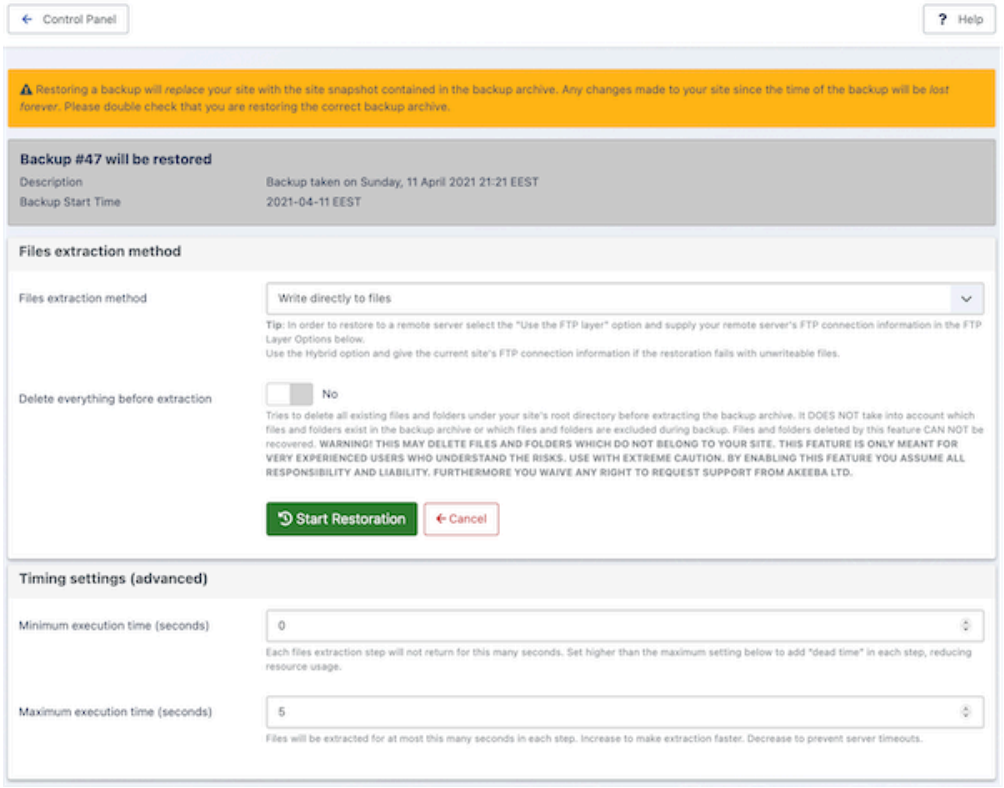
Note

This only applies to Akeeba Backup Professional.

The integrated restoration feature allows you to easily restore a previous backup directly on your server, where you took the backup from, as long as your backup archive still exists on your server of course.

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 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, the application 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 extracted, 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 you have already enabled the FTP mode in the application's System Configuration page. In this 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 probably cause the restoration script (ANGIE) to appear or your site will throw an error message. If you want to stop the restoration please remove the `installation` directory from your site's root manually, for example using FTP, before trying to access your site again. Please note that it is possible that your site is left in an unusable state by doing that. If this happens, please retry the restoration.

If you chose to use the FTP mode, there are some connection settings you have to take care of. They areL

Host name	The host name of your site's FTP server, without the ftp:// protocol prefix. 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. Copy (in FileZilla it appears on the right hand column, above the directory tree) and paste that path in the application'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 rest of the extraction process is 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 `restore.php` script inside the component's root directory. 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 the application. As a result, it can't be used as a potential attack vector.

2. Before the restoration begins, the application needs to create the `restoration.php` file with all the archive extraction setup parameters. It is intelligent enough to use any FTP / SFTP file writing mode which you have configured in the System Configuration page to overcome any permission problems, but you are ultimately responsible for ensuring that the permission settings are adequate for the application to create this file.

   If you are using the direct file writes in the System Configuration page the permissions of the application's directory should be 0777 for the integrated restoration to work. USING SUCH BROAD PERMISSIONS IS NOT RECOM-

MENDED AND MUST BE AVOIDED IF AT ALL POSSIBLE. On hosts which use suPHP, FastCGI or other methods which ensure proper file ownership 0755 permissions are recommended.

If you are using the FTP / SFTP layer, 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 asked to open the restoration script in a new tab or window. DO NOT CLOSE THE INTEGRATED RESTORATION PAGE'S TAB/WINDOW! 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.

4. After you have competed the restoration script's process you are supposed to return to the Integrated Restoration page and click on the Finalize button to:

   - remove the `installation` directory from your restored site's root, and
   - remove the `restoration.php` setup file from the application's directory.

   If you have restored the backup to a site different than the one you backed up from, the Finalize button may fail to work. In this case 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`.

4.2. Using Akeeba Kickstart

Tip

You can download Akeeba Kickstart from our site's download page [https://www.akeeba.com/download/official/akeeba-kickstart.html]. The latest release is always at the top of the list. Older versions are available for use with older versions of our backup software and older versions of PHP.

In order to use Kickstart, begin by downloading Kickstart itself from our site. You will download a ZIP archive which you have to extract first on your computer. Inside it you will find kickstart.php, several .ini files and a couple of .js file. You need to copy at least the PHP file into the root of the to be restored site. For example, if you are restoring on a subdirectory named `example` in the root of your XAMPP for Windows installation, you need to copy the file inside the `c:\xampp\htdocs\example` folder. Conversely, if you are restoring to a live host, upload the PHP file using FTP in your site's root.

You do not need to upload the .ini file. It's a translation file for people who want to translate the software. The English language is already included in Kickstart itself. Kickstart will query your browser's default language and look for the respective translation file. For example, if the default browser language is German (de-DE) Kickstart will try to load the de-DE.kickstart.ini file in order to present you with a localized interface. If you want to translate Kickstart feel free to do so, following this naming convention and upload your translation INI file to your server before running Kickstart.

The final step before being able to extract the archive is, of course, copying the archive itself in the same directory as `kickstsart.php`. If you have a multi-part archive remember to upload all of the archive parts, otherwise the extraction will, of course, fail. Moreover, if you are restoring to a live server please upload the backup archive and all of its parts using the Binary transfer mode. We suggest using CyberDuck to do that. As soon as you connect to your site and right before uploading any files click on the Transfer, Transfer Type, Binary menu item. This will ensure that the backup parts will not be corrupted during transfer.

Note

Special note for most shared live hosts

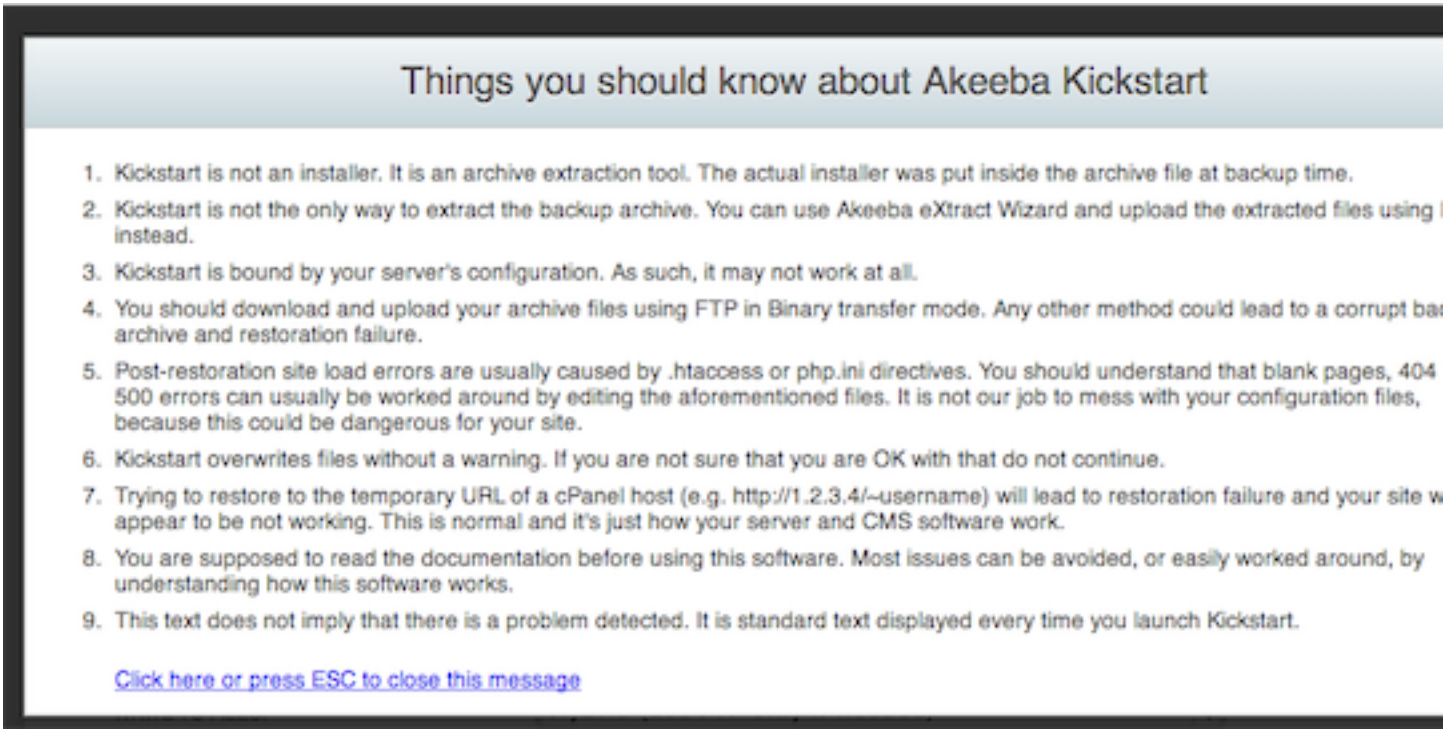
If you are restoring on a shared live server which doesn't use suPHP (the majority of live servers) you may need to use Kickstart’s FTP mode if the Direct mode (default) doesn't work. In this case, creating a temporary

directory is necessary. First make sure that your site's root directory has 0755 permissions. If unsure, ask your host. Then, create a directory named `kicktemp` inside the directory `kickstart.php` is in and give it 0777 permissions. Remember to remove this directory as soon as you are done restoring your site for security reasons!

Once you're ready with the preparation, launch Kickstart by visiting its URL which looks like `http://localhost/mysite/kickstart.php` on local hosts, or `https://www.example.com/kickstart.php` on live hosts.

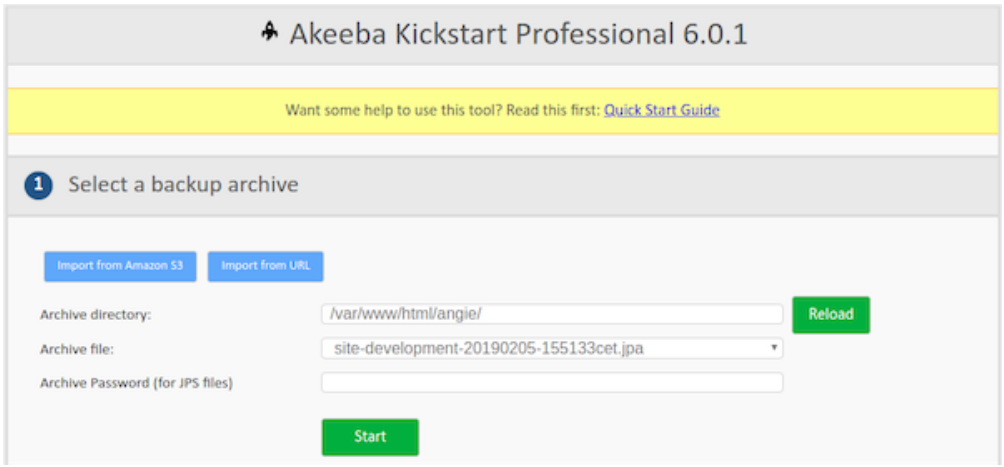
The first page is a reminder of key facts about Kickstart:

Kickstart's first page



After reading it, press ESC to close the information window and display the main interface:

Kickstart - Select a backup archive



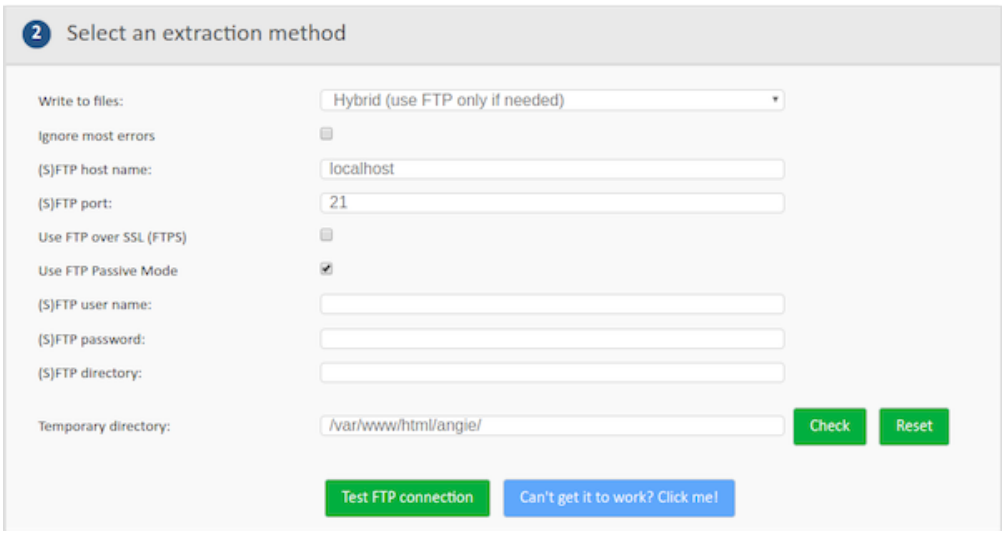
In the first step, select your backup archive file. Usually, there is only one file and it is pre-selected for you.

Important

If you have a multipart backup archive only the main part with the .jpa, .jps or .zip extension will be displayed. The part files (.j01, .j02, ... or .z01, .z02, ...) will be extracted automatically. In fact, all of these part files MUST be present for the extraction to proceed.

If you are restoring an encrypted archive (JPS) please enter the password to it in the Archive password (for JPS files) field. Please remember that the password is case sensitive: abc, ABC and Abc are three different passwords.

Kickstart - Select an extraction method



In the second step, you have to choose an extraction method. The `Directly` method is the fastest and should work on all local and most live hosts. If you get error messages about unwritable files in later steps, you'll have to use the `Use FTP` mode here. The best option is the `Hybrid` one. It will intelligently switch between Directly and Use FTP on a file-by-file basis, automatically detecting which mode is more appropriate for it.

If you are using the FTP or Hybrid modes, you will see the FTP-specific options:

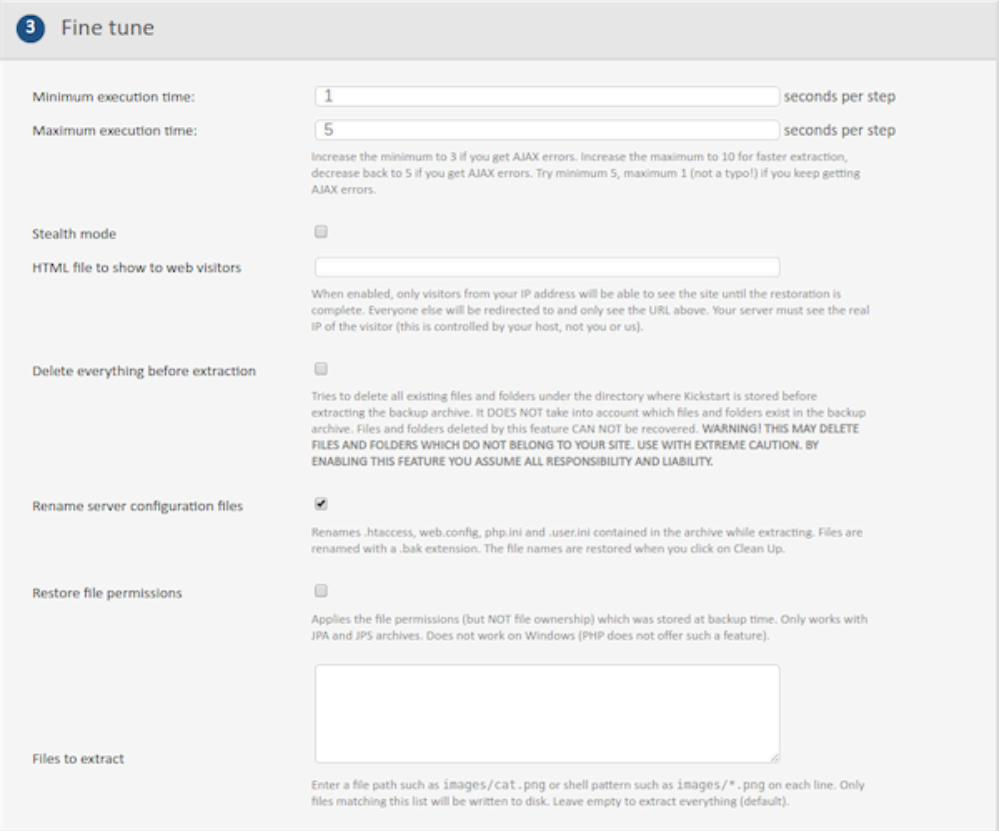
FTP host name	Use the domain name or IP address to access your site's FTP server. Do not use the ftp:// protocol prefix. For example, `ftp.example.com` is correct whereas `ftp://ftp.example.com` is wrong.
FTP Port	Leave the default value (21) unless your host tells you otherwise. Do note that Kickstart only supports plain FTP and FTP over SSL connections, but not SFTP. If your host tells you to use port 22 – which is used only by SFTP – it won't work.
Use FTP over SSL (FTPS)	Use only if your host tells you it is supported. FTPS is not the same as SFTP, do not confuse those two!
Use FTP Passive Mode	It's a good idea to turn it on, as most servers require it. If your host told you they require active mode, uncheck this option.
FTP user name and password	What they claim to be, the user name and password to connect to your site's FTP server
FTP directory	The absolute FTP path to your site's root. The easiest way to find it is using FileZilla to connect to your site and navigate to your site's root, which is usually a directory named `htdocs`, `http-`

`docs`, `http_docs`, `public_html` or `www`. Look at the right hand pane, above the folder tree (Remote site text box). This is what you want. Copy it and paste it in Kickstart's FTP directory box.

Temporary directory
: If you followed the instructions at the beginning of this chapter above, you have already created a `kicktemp` directory with 0777 permissions. If not, do it now. After that, just append `/kicktemp` to whatever is already written the Temporary Directory box. You can check that the directory exists and is really writeable by clicking on the Check button. The Reset button will reset the box to its initial value should you accidentally mess up its contents.

Click on Test FTP connection before proceeding to make sure that Kickstart can connect to your site through FTP before proceeding.

Kickstart - Fine tune



The fine tuning options are *for advanced users only*. Changing them might make it impossible to extract your backup archive. If you are not sure you understand what they do we advise you to not touch them.

Minimum execution time
: Akeeba Kickstart breaks down the archive extraction process in small chunks. This value determines the minimum amount of time each of these chunks should take. It's recommended to leave the default setting of 1 second. If you have a server with strict usage limits you are suggested to set this to 5 and the maximum execution time to 3 (yes, the maximum will be lower than the minimum, it's not a typo).

Maximum execution time
: Akeeba Kickstart breaks down the archive extraction process in small chunks. This value determines the maximum amount of time each of these chunks should take. Larger values will make the extraction marginally faster but more prone to server timeouts. Smaller values will make the extraction slower but less prone to server timeouts.

Stealth mode

When an extraction and restoration takes place anyone with a web browser can navigate to Kickstart itself or the restoration script. This is problematic on the security front. Enabling the Stealth Mode will make your site available only to your IP address while the extraction and restoration is in progress. Anyone else visiting the site will see a static HTML page, given below. When you clean up after the restoration is over your visitors will see your site normally.

Important

This feature only has any effect on hosts using the Apache web server or a compatible server which understands .htaccess files. It will definitely not work on hosts using IIS or NginX.

HTML file to show to web visitors

This option allows you to define the name of the static HTML page to show to your visitors when the Stealth Mode above is enabled. The file and its resources (images, CSS, Javascript files) must reside inside your to-be-restored site's root. You must only define the name of the file to use, not its URL. This means that `offline.html` is a valid setting, whereas `http://www.example.com/offline.html` is INVALID and will result in a 404 error thrown to your visitors.

Delete everything before extraction

Tries to delete all existing files and folders under the directory where Kickstart is stored before extracting the backup archive. It DOES NOT take into account which files and folders exist in the backup archive. Files and folders deleted by this feature CAN NOT be recovered. WARNING! THIS MAY DELETE FILES AND FOLDERS WHICH DO NOT BELONG TO YOUR SITE. USE WITH EXTREME CAUTION. BY ENABLING THIS FEATURE YOU ASSUME ALL RESPONSIBILITY AND LIABILITY.

Enable Stealth Mode while restoring

When using the integrated restoration feature you are extracting the backup archive, which includes the site restoration script, into your live site. This means that any visitor to your site would be seeing the restoration script. This is confusing for legitimate visitors and a security risk (if you'd not used the “ANGIE Password” feature during backup) as the restoration script will display the database connection information to your site.

Enabling the Stealth Mode addresses this issue. Right before Akeeba Backup starts extracting the backup archive it will rename your site's `.htaccess` file to htaccess.bak and replace it with a new one. The new file will issue a temporary HTTP redirection (HTTP 307) for all visitors to your site to the file `installation/offline.html` which is included with the restoration script on all backups taken with Akeeba Backup and Akeeba Solo versions released after May 20th, 2021. This file tells your visitors that the site is temporarily under maintenance and they should come back later. Since this is a temporary redirection it does not cause any problems for your site's SEO; search engines will indeed come back later to re-index your site if you happen to start a site restoration right before or during their indexing your site.

You should be aware of some facts and limitations of the Stealth Mode:

- This feature only works on servers which support `.htaccess` files, i.e. web servers tunning the Apache or Lighttpd web server software with the option to use `.htaccess` files enabled by the hosting company. If unsure, please ask your host.
- This feature requires your site being able to write to the .htaccess file in the site's root. In most cases this is not a problem. If the .htaccess file is NOT replaced when you enable this feature you will need to temporarily give more open permissions (e.g. 0666 – read and write to everybody) to the .htaccess file. This is not a security concern because the file will be replaced as soon as you start the restoration and its permissions will be reset.

- The redirection applies to all visitors coming from a different IP address than the one your web server told PHP you are visiting from. If your IP address changes in the middle of the restoration — something which can happen automatically with some ISPs or if you have a temporary loss of network connectivity — you will also be locked out of the restoration yourself. There are manual override instructions after these bullet points.

- Some servers may be behind a load balancer, proxy, caching proxy, CDN or a security firewall (or a combination thereof!); we will call them collectively "proxies". If the web server is not configured to use the forwarded IP address from these proxies it may report the wrong IP to PHP. As a result, the Stealth Mode will not work properly. You will either be allowing everyone to access your site (as if Stealth Mode was not present at all) OR you may find yourself locked out. There are manual override instructions after these bullet points. If you suspect that this happened to you please contact your host and ask them to reconfigure your server so it honors the forwarded IP address, typically conveyed from proxies through the HTTP X-Forwarded-For header. Modern versions of Apache, NginX and IIS are fully capable of doing that with some very straightforward configuration.

- The Stealth Mode .htaccess will only be removed at the end of the restoration, when you click the “Finalize restoration” button in Akeeba Backup's / Akeeba Solo's interface OR the “Clean up” button in the restoration script (the latter only appears if your browser doesn't let the restoration script detect that it's running as part of the integrated restoration). If your restoration fails for any reason please take a look at the manual override instructions after these bullet points.

- Finally, please note that Stealth Mode .htaccess also addresses another issue. If you are using a security hardened .htaccess file e.g. one created by a security plugin on your site (such as our own Admin Tools); or one you created yourself following security hardening instructions for your CMS (e.g. the security hardened .htaccess file you can find in Joomla's documentation, which is based on an old version of our own security hardened .htaccess) you may find that it's impossible to run the integrated restoration. The archive is extracted but you cannot access the installation folder. Enabling Stealth Mode will remove the security hardened .htaccess temporarily, overcoming this problem.

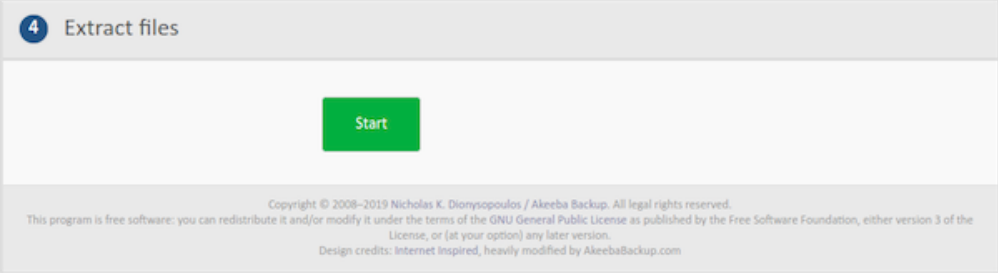
Manual override. If for any reason you are locked out of the restoration script or the restoration doesn't complete and you want to regain access to your site, please delete the .htaccess file from your site's root. Then rename the file htaccess.bak to .htaccess (note the dot in front of the file's name!). If this happens consistently to you, we advise you to ask your host how to apply password protection to your site's root and do that **right after** you start extracting your backup archive. Furthermore, you should take the following steps to mitigate the issues normally addressed by Stealth Mode *without having to use Stealth Mode on your site when performing an integrated restoration*:

- **Use the “ANGIE Password” feature in all your backup profiles**. This will require entering a password to access the restoration interface after extracting your backup archive. This will be less confusing for legitimate visitors. Moreover, it mitigates the potential security issue of divulging your database connection information as access to the database restoration page will require entering the ANGIE Password first.

- **Configure your security plugins**. If you are using a security plugin which creates a security strengthened .htaccess file, such as our Admin Tools Professional, please configure it so that access to the `administrator/components/com_akeebabackup/restore.php` file and the `installation` folder (and all of its files, including .php files) is allowed.

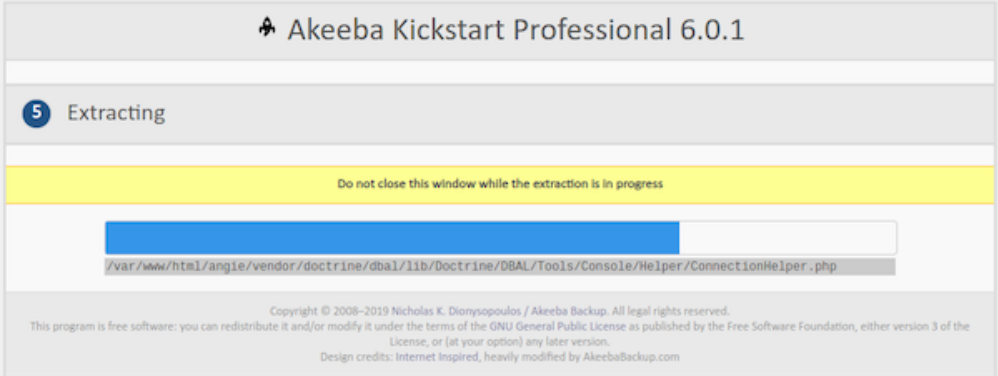
Rename server configuration files	Renames .htaccess, web.config, php.ini and .user.ini contained in the archive while extracting. Files are renamed with a .bak extension. The file names are restored when you click on Clean Up.
Restore permissions	Applies the file permissions (but NOT file ownership) which was stored at backup time. Only works with JPA and JPS archives. Does not work on Windows (PHP does not offer such a feature).
Files to extract	Enter a file path such as `images/cat.png` or shell pattern such as `images/*.png` on each line. Only files matching this list will be written to disk. Leave empty to extract everything (default).

Kickstart - Extract files



Click on the big green Start button. Kickstart will start extracting your site's files.

Kickstart- Extracting



The blue bar fills up while your site files are being extracted.

If you get an “Unwritable file” error message, go back and enable the Use FTP option before re-trying extraction. If all else fails, extract the archive locally and upload the extracted files to your site by FTP.

If you get an error message that the archive is corrupt, you have to check two things. First, make sure that you have uploaded all archive parts. In a multi-part archive situation, there is the main .jpa, .jps or .zip file and several “part files” with the same name as the main file but with extensions like .zip, .z01, .z02, etc (ZIP) or .jpa/.jps, .j01, .j02, etc (JPA/JPS). You have to upload all of those files for the extraction to work.

The other think you must check is how you downloaded and uploaded the backup archives. As mentioned in "The recommended method' section, you should use FTP in Binary transfer mode. This holds true for uploads as well. Try uploading the backup archive again, using FTP in Binary Transfer mode and retry. This usually does the trick.

When the extraction is over, Kickstart offers you a link to open the restoration script which was included in your archive file and is just extracted to your site:

Kickstart- Restoration and Clean Up

Click on this button and start reading the next chapter, detailing the use of the restoration script. Do not close Kickstart's window/tab yet! You will need it to clean up after the restoration is over.

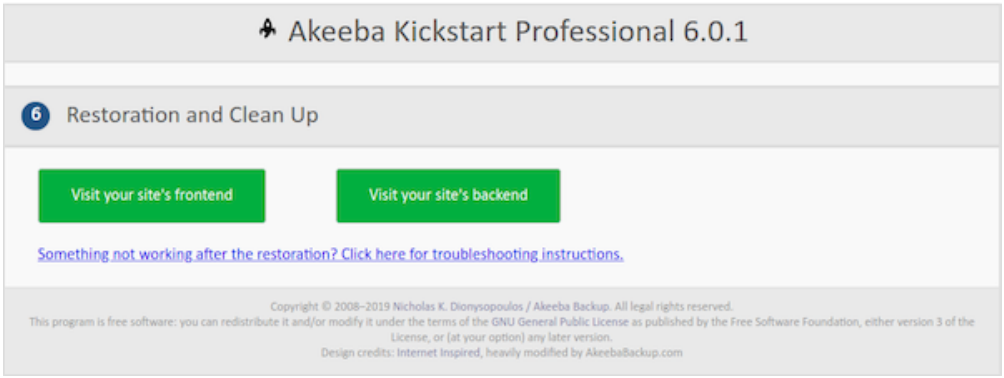
Note

Kickstart extracts the `.htaccess` and `php.ini` files in your site's root (if they exist in the backup archive) as `htaccess.bak` and `php.ini.bak` respectively to avoid any incompatibilities during site restoration. Do note that these files are restored to their original names during the last step of the restoration procedure. Should you use Kickstart just for extracting your site's files, please keep this information in mind as you will have to rename those files manually.

Kickstart- Restoration and Clean Up

Once you are done with the restoration script, click on the Clean Up button. This will remove the installation directory which holds the restoration script, Kickstart's files, the backup archive (and all its part files) and rename the file mentioned above back to their normal names.

Kickstart- Restoration and Clean Up



Use the View your site's front-end button to visit your site. Ignore the other button; it's only used for backups taken with Akeeba Backup for Joomla!.

4.3. Using third party software

Warning

We strongly advise you against using third party software to extract backup archives. Backup archives taken with Akeeba Backup are best extracted using Akeeba Kickstart.

If you are using the ZIP format it is possible to use third party software to extract the backup archives locally. Please note that the only third party software which follows the official ZIP specification and can extract backup archives in ZIP format is PKZIP for Windows and WinZIP. Other software such as the Linux and Mac OS X ZIP utility, Windows Compressed Folder, WinRAR, 7-Zip etc use the InfoZIP library which does not understand the official multipart ZIP archive format specification which is used by Akeeba Backup.

Moreover, large files do have an invalid CRC32 checksum. This is done on purpose. Calculating the CRC32 checksum of large files would lead to timeouts and backup failure. This means that you may receive warning about corrupt files when extracting backup archives in ZIP format. Usually this is not a problem, the files are extracted successfully.

After extracting the files you will need to move them to your web root.

5. ANGIE: Akeeba Backup's restoration script

All full site and database backup archives created by the application contain a restoration script (also called "installer") inside the `installation` directory. This script is used to restore your database and, depending on the type of installer script selected and the PHP-based application used by your site, possibly reconfigure your site.

The restoration scripts used by the application are all based on ANGIE (Akeeba Next Generation Installer Engine), a common platform for backup restoration scripts. The only difference in each installer is the first page (requirements check) and its final page (site configuration). Everything else is common among them.

Since the installer is stored in the backup archive at backup time, it is important to choose the correct installer for your site. For example, if you have a Joomla site you have to choose "ANGIE for Joomla". Selecting the wrong installer is not the end of the world; you will still be able to restore your site's database content but you will have to modify your site's configuration file(s) manually.

All installers can be accessed by visiting the `installation/index.php` URL on your site after extracting the backup archive. For example, if your site is located in `http://www.example.com` you can access the ANGIE

installer after extracting the backup archive by visiting `http://www.example.com/installation/index.php`.

5.1. Common instructions for all ANGIE installers

This section of our documentation will guide you step-by-step through the restoration process with ANGIE, the restoration script included in the `installation` directory of every backup archive taken with the application. When the instructions depend on a specialised ANGIE installer (e.g. ANGIE for Joomla!, etc) we will point you to the right page for more information. Please read this section in the order presented for best results.

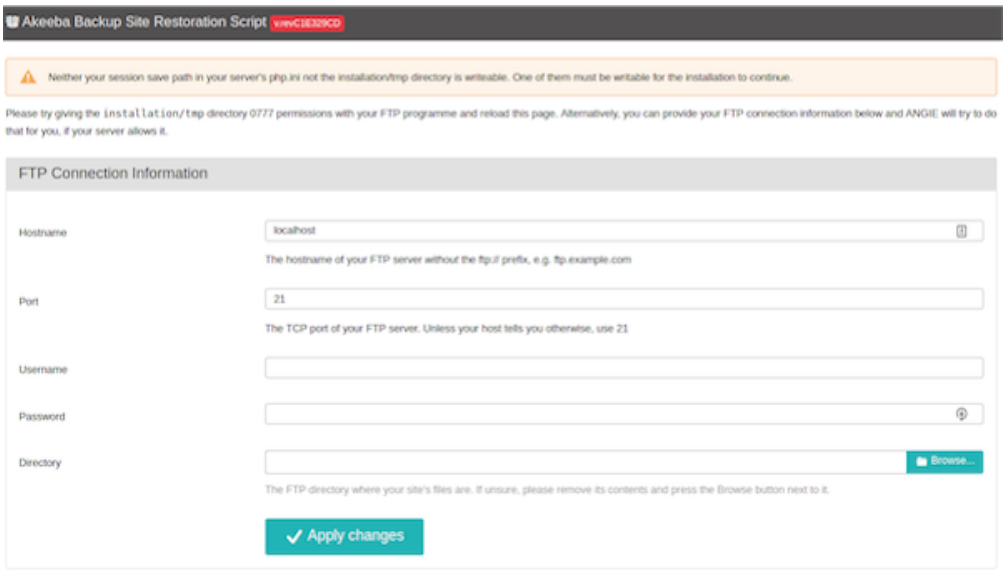
5.1.1. The session fix page

Important

This is an optional page and you may not ever see it during your site's restoration. Read below for more information on when it is shown.

In order for ANGIE to work it needs to be able some temporary data between page loads. This data is stored in files inside the `installation/tmp` directory of your site during the restoration. Depending on your server and the way you extracted the backup archive this directory may end up being unwritable for PHP and, of course, ANGIE itself. The easiest way to work around it is giving this directory 0777 permissions using your favourite FTP programme. We have, however, found out that many of our users feel lost when they are told they have to change permissions. As a result we have made ANGIE very smart and capable of fixing this problem by itself, as log as you give it your FTP connection information first. Therefore, in case that this problem is detected, you will see the following page:

ANGIE - The session page



You have to provide the following settings:

Host name	Use the domain name to access your site's FTP server
Port	Leave the default value (21) unless your host tells you otherwise. Do note that Joomla! only supports plain FTP. If your host tells you to use port 22 – which is used only by SFTP – it won't work.
Username and password	What they claim to be, the user name and password to connect to your site's FTP server

Directory — The absolute FTP path to your site's root. The easiest way to find it is using FileZilla to connect to your site and navigate to your site's root, which is usually a directory named `htdocs`, `http-docs`, `http_docs`, `public_html` or `www`. Look at the right hand pane, above the folder tree (Remote site text box). Copy it and paste it in the Directory box.

Tip

You can instead fill in all of the other information, leave this field blank and click on the Browse button next to it. If your FTP information is correct a popup directory browser appears. You can now browse to the site root directory. It's the one where you can see your site's `installation`, `includes` and `libraries` directories. Once you're in there click on the Use this button.

Then click on the Apply changes button. If all of the FTP connection information is correct, PHP can connect to your site by FTP and your FTP server allows changing permissions you will see the main page of ANGIE. Otherwise you will see the session fix page again. In this relatively rare (less than 5% of servers out there, according to our experience) case you will have to change the permissions of the `installation/tmp` directory manually, using your FTP software or your hosting control panel.

5.1.2. The password page

Important

This is an optional page and you may not ever see it during your site's restoration. Read below for more information on when it is shown.

Your backup contains all of the database connection information and, sometimes, the FTP connection information as well. This means that anyone who has access to the backup restoration script over the web can learn this information. This is a security risk. While Kickstart allows you to use Stealth Mode (which locks the site for everyone except your own IP address) this only works on servers using the Apache web server. With alternative web server software such as IIS, NginX, Litespeed and Lighttpd becoming more popular this doesn't cut it any more.

This is why we added support for password protection to ANGIE. You can enabled it in the application, in the Configuration page of the component. You also get the chance to override that password when taking a backup from the backend of your site, by providing it your desired password to the "ANGIE password" field in the Backup Now page. The password is stored in the backup archive encrypted, making it more difficult to be accidentally exposed to a hacker.

If you have specified a password for ANGIE you will see the ANGIE password page when you launch it. Just provide your password and ANGIE will show its main page. If the password is incorrect you will be shown the password page again.

Important

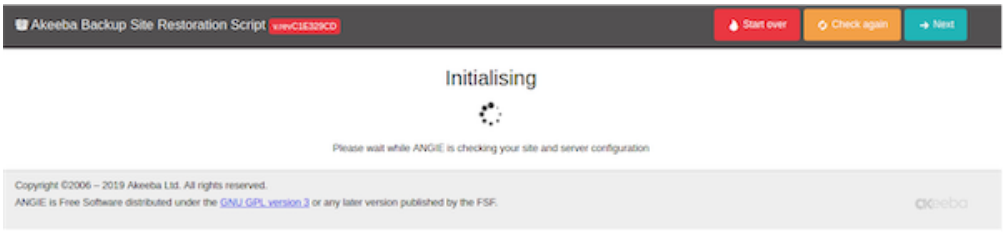
Just like with passwords on most web sites and devices you may have used, ANGIE passwords are case sensitive. This means that ABC, abc and Abc are three different passwords.

In some cases you may have forgotten your ANGIE password or not remember setting one. The latter could happen if your browser auto-fills a password in the backup page. You can still continue the restoration by deleting the file `installation/password.php` from your site which is currently being restored. Then reload ANGIE and the password page will go away without having to enter a password.

5.1.3. The main page

First you see ANGIE's initialisation page:

ANGIE - The initialisation page



At this point ANGIE is performing system checks. If your browser gets stuck at this page please try disabling any browser plugins, firewalls and antivirus applications which might be interfering with the Javascript on the page. The most likely culprits we've seen are NoScript (a Firefox plugin which disables Javascript) and AVG Antivirus' "Link Checker" feature which ultimately breaks Javascript on most pages. If that doesn't help please try using a different computer, with a different browser and connected via a different ISP to the Internet. Sometimes computer policies, proxies and other network settings can interfere with web applications such as ANGIE or the site script you are restoring.

The next page you will see is the main system checks page. In this page ANGIE checks your server configuration and makes sure it matches the requirements of both ANGIE and the site script you are currently restoring. The exact layout of the page depends on which ANGIE installer you are using. Here you can find links to these specialised installers, in alphabetical order:

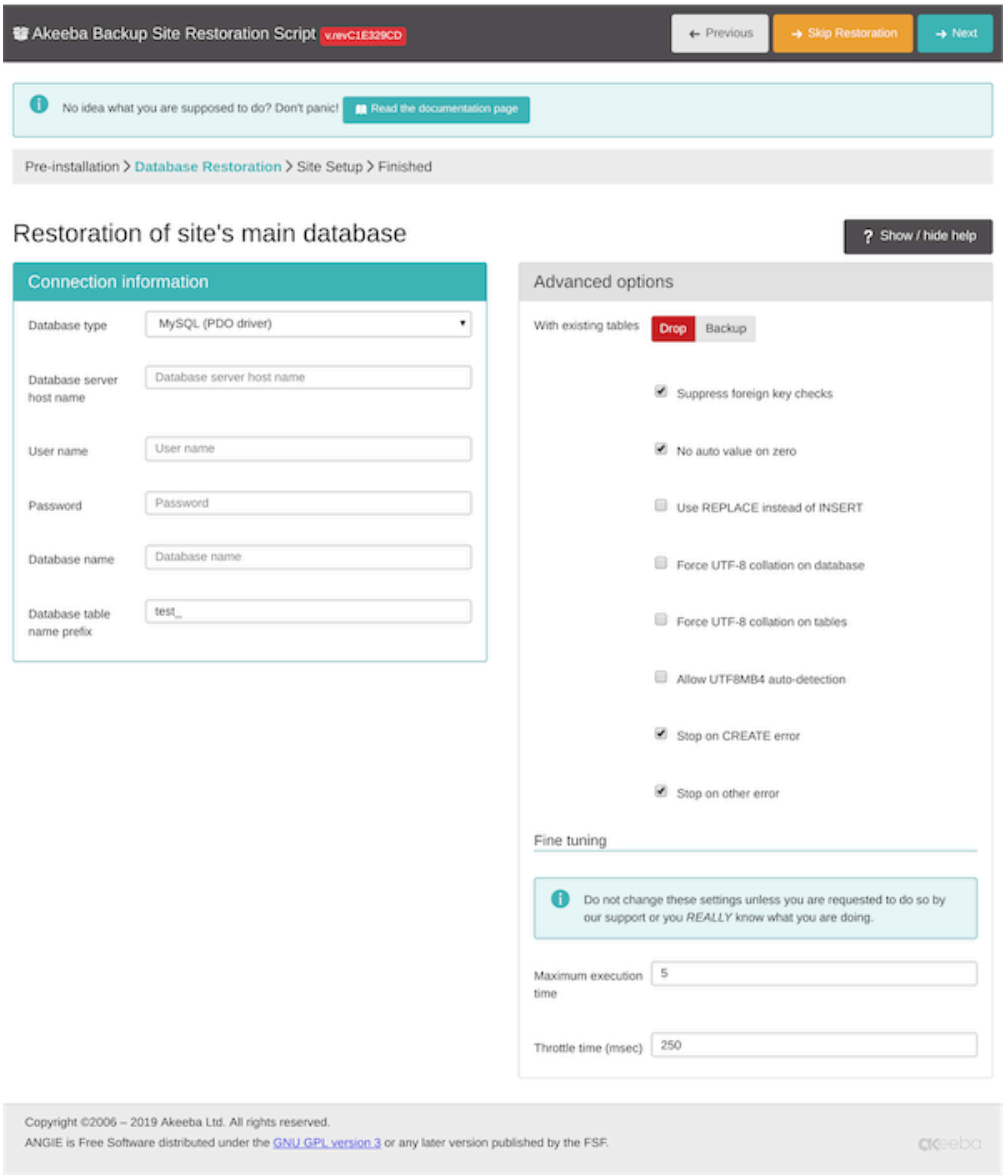
- ANGIE for Joomla! [angie-joomla-first]
- ANGIE for Miscellaneous PHP Applications [angie-misc-first]

5.1.4. The database restoration page

Important

If you are restoring to a new site -even if it is a subdomain or directory of the main site- you MUST create a new database before the restoration. ANGIE can not do that for you. It's not that we didn't think of it (the code is there, if you take a look), it's that on most server environments you don't have adequate database server permissions to create a new database. For detailed instructions on creating new databases, take a look at the relevant page of our Troubleshooting Wizard [https://www.akeeba.com/documentation/troubleshooter/abidatabase.html].

ANGIE - The database restoration page



At the top of the page you see which database you are about to restore. The first database you will be restoring is the "site's main database" which means that it's the database your web script uses to store its data (the one you set up in the configuration wizard or the application's Configuration page). If you had defined multiple databases you will be restoring one database after the other, but the "site's main database" is always the first one to restore. If you do not wish to restore a particular database you can click the orange Skip Restoration button in the top right hand corner of the page.

The first thing you see is the Connection information column in the left hand side of the page. It is pre-populated with the connection settings of the site you backed up. If you are restoring to a different site than the one you backed up from it will be blank, indicating that you must specify the correct connection information for the new server.

Start with the database type. It's usually `MySQLi` (with the "i" at the end), unless you have a server with an outdated PHP version or suboptimal server settings. The other option, `MySQL` (without the "i" at the end) is the old MySQL driver which is slower and was ultimately removed in PHP 5.5.0 released on June 2013. The final option is the `PDO MySQL` which is another modern driver for MySQL. It's just as fast as the MySQLi driver. If you see no option then your new server lacks the PHP modules which are necessary to connect to MySQL databases. Do note that it's not

necessary to have a MySQL-compatible database server (MySQL, MariaDB, Percona, Amazon RDS etc). Your version of PHP also needs to include the mysqli or pdo_mysql module which allows it to talk to MySQL servers. If the list is empty you need to contact your host.

The other available options you need to set are:

- Database server host name. The database server host name. Usually it's localhost, but you must ask your host for this setting, or consult your hosting account control panel, as this setting is usually displayed there.

  ## Important

  Despite what you think, `localhost` and `127.0.0.1` are two completely different things for PHP's MySQL drivers. It is possible -especially on a Mac OS X local or live server- that your database server does not connect when using `localhost`. Just use `127.0.0.1` and it will! This will save you a lot of hair pulling if you're dealing with MAMP or ever try to test restore your site to a Mac.

- Username. The username of the database server user. Please consult your host.
- Password. The password of the database server user. Please consult your host.
- Database name. The actual name of the database you want to restore to. If you choose a database which already has tables in them, existing tables with the same name as the ones being restored will be overwritten (replaced) by default. You may want to ask your host for the correct value of this setting.

The Database table name prefix is up to your liking. For security purposes it's best to not use the default prefixes used by most common site scripts (e.g. jos_ or wp_). Ideally, you should use three to four letters followed by an underscore, e.g. tst_ or test_.

Warning

Only use two to five lowercase letters a-z (without accents or diacritics) and numbers 0-9 followed by exactly one underscore. Anything else may cause restoration problems due to the way databases work on most operating systems. Especially avoid uppercase letters! On many servers it will cause problems due to the way MySQL stores its data on the server's disk.

After entering this information, take a look at the the Advanced options column on the right hand side of the page.

The With existing tables option lets you decide what to do with tables which have the same name as those currently being restored. The default Drop option will delete same-named tables during restoration without asking. The Backup option will keep a copy of those tables, changing their name prefix to bak_, i.e. an existing abc_users table will be renamed to bak_users. Existing bak_ tables will be deleted before the rename. Both Drop and Backup options apply to tables that are being restored by ANGIE.

There are two more options which apply to tables in your database even if they are NOT included in the backup archive you are restoring or if you chose not to restore them via the Select which tables to restore option. Drop same prefix will delete all tables with the same prefix as the Database table name prefix you defined in the Connection Information. Drop All is the most dangerous option. It will remove all tables from the database, regardless of their name. If you are sharing a database with another site its database tables will be erased. Both of these options are dangerous and can cause permanent data loss. Be very careful when you use them!

Select which tables to restore allows you to perform a partial restoration. Select which of the tables backed up should be restored. Make a multiple selection with CTRL-click (Windows, Linux) or CMD-click (macOS). Any table you have not selected here will not be restored; it will be skipped over. This an advanced feature. Use it if you are absolutely sure about what you're doing. Skipping the restoration of the wrong table may cause your site to malfunction or not work at all.

Suppress Foreign Key checks allows you to restore cross-linked tables without MySQL errors. Leave it on. Use REPLACE instead of INSERT may be required if you keep getting MySQL errors about rows already existing in your tables. Force UTF-8 collation on database and Force UTF-8 collation on tables should be enabled on all sites which use non-ASCII characters in their contents, e.g. accented Latin characters, German umlauts, Cyrillic, Greek, Chinese or any other characters which are not normally used in the English language. If unsure, make sure it's checked.

No auto value on zero works around a problem with some software which uses an auto incrementing counter column in a database table but also inserts the numerical value 0 in it. This can cause problems during restoration which why this option exists. If unsure, keep it enabled.

Use REPLACE instead of INSERT tells ANGIE to use REPLACE database commands to apply the restored table contents instead of INSERT. This is rarely necessary, if you have edited your SQL files by hand and removed the CREATE TABLE lines. If you don't know what it is you most likely don't need it.

Some PHP software, such as Joomla 3.7 and later, is using the `utf8mb4_general_ci` collation instead. This collation supports multi-byte Unicode characters such as Emoji, certain Traditional Chinese characters and so on. If you need to restore such a site please remember to turn on the Allow UTF8MB4 auto-detection option.

In case you have taken a backup on a server supporting UTF8MB4 but trying to restore on a server that doesn't you will observe that this results in an error. This is by design: UTF8MB4 supports more characters (notably: emoji) than UTF8. If the database server allowed you to restore UTF8MB4 data to a UTF8 table / table field you'd lose data. If, however, you are *absolutely sure* you want to downgrade your UTF8MB4 data to UTF8 and complete the restoration you can do that in newer versions of our restoration script by selecting (checking) the Force UTF-8 collation on tables option but NOT selecting (leave unchecked) the Allow UTF8MB4 auto-detection option. Please note that if you have extended, 4-byte characters such as Emoji, certain traditional Chinese characters etc they will be replaced with the Unicode Replacement Character (#) as they cannot be mapped to plain UTF-8.

As a final note on the subject of UTF8MB4, if you back up a site on a server supporting only UTF8 and restore it to a server supporting UTF8MB4 you will NOT have a problem. Therefore you do not need to do anything special.

Handling database errors

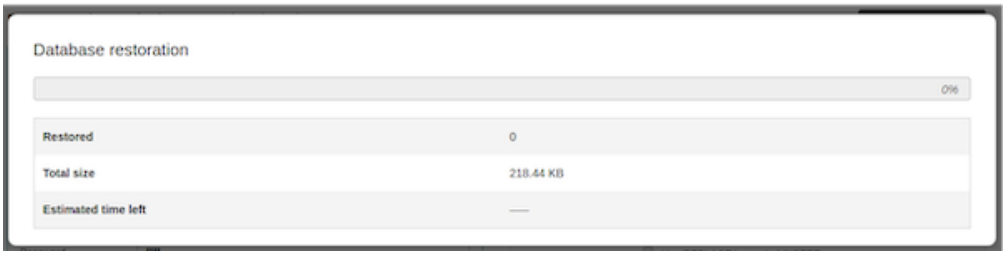
On rarely cases during the restoration you could get a database error. This is not caused by Akeeba software, but by the data you're trying to restore (in other words: you would get the error even manually importing the SQL file using PhpMyAdmin). However, if that data is not crucial, your site would work perfectly fine without it. Think about logging data, statistical information or some article metadata that is too large for your new database: if one page is missing an image or some formatting that's something you can manually fix later.

This is when the options **Stop on CREATE error** and **Stop on other error** come into play. By default ANGIE will stop when something wrong happened, if you uncheck those fields, ANGIE will ignore any database error, showing you a report of the failed queries. In this way you can review them and know which part of the old database wasn't restored in the new one.

You will find the Fine Tuning area towards the bottom. The setting Maximum execution time should be left at its default values unless you get AJAX or timeout errors while ABI is restoring your database. In this case, try setting it to 3, 2 or even 1. This will slow down the restoration a bit, but it will make it more resilient to timeout issues. The Throttle time (msec) should be left at 250 unless you are told by our support staff to change it.

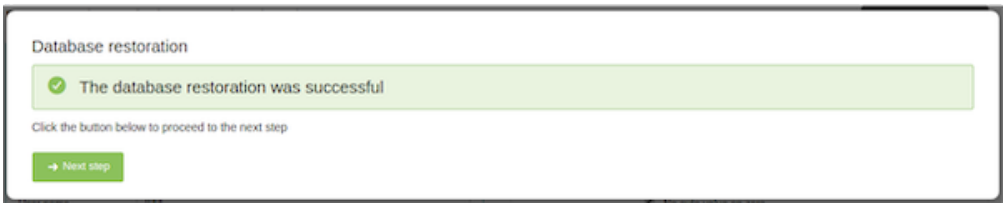
When you're ready with all those settings, click on the Next button on the top right corner of the installer page to start restoring your database. The restoration dialog appears:

ANGIE - Stepping through the database restoration



While your database is restoring, you will see the progress bar filling up and the information line below informing you of the processed and total size of the database dump file. The estimated time left is a very rough approximate. Should an error occur, you can close the dialog, modify the settings and retry by clicking the Next button again. Once the restoration is over, the Next step button appears:

ANGIE - Database restoration is complete



Just click on it. If you have more databases to restore (only backups made with the Professional version, using the Multiple Database Definitions feature of Akeeba Backup Professional) you will see the database setup page again, but the header will read the name of the extra database instead of Site's main database. If you have extra databases to restore you will see the same database restoration page for the next database.

If there are no more databases left to restore but you had defined off-site directories to be backed up, upon clicking Next step you will be taken to the Off-site directories restoration page.

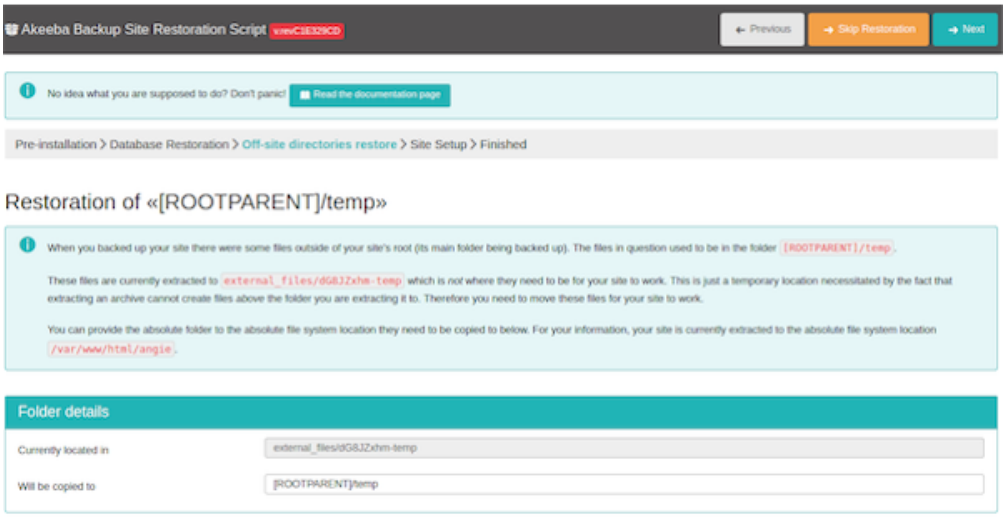
If, however, you do not have any off-site directories upon clicking Next step you will be taken to the Site Setup page. The exact page depends on the installer you are using:

- ANGIE for Joomla! [angie-joomla-setup]
- ANGIE for Miscellaneous PHP Applications [angie-misc-setup]

If you don't see that page, maybe you want to take a look at this troubleshooting page [https://www.akeeba.com/documentation/troubleshooter/abiafterdb.html].

5.1.5. Off-site directories restoration page

ANGIE - Off-site directories restoration page



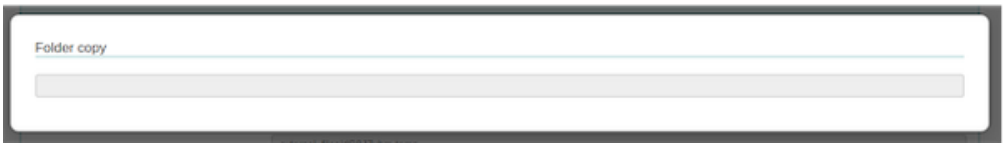
When Akeeba Backup backs up an off-site directory its contents are placed inside the backup archive in a directory called "virtual folder". By default this is `external_files`. Naturally, this is extracted inside the site's root which is not where these files need to be. Therefore, they have to be moved. This is what this page does: it moves the off-site folder from where they were extracted to where they should be.

The Virtual folder area shows you the directory inside the extracted site's root where the files and folders to be copied reside. In the Target folder you can enter the absolute filesystem path of where the files and folders contained in the displayed Virtual folder should be moved to. Click on the Next button to perform the move.

You can always click on Skip Restoration to skip moving the files and folders from the displayed virtual folder, e.g. when you would rather do that manually.

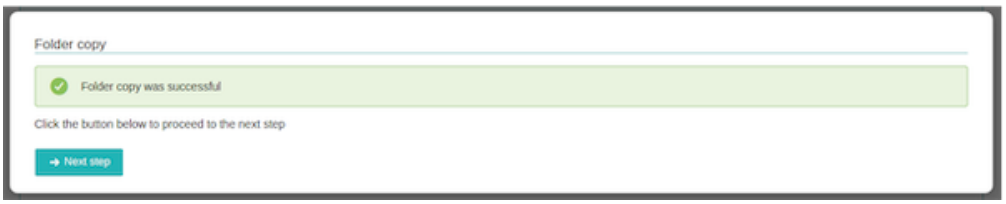
While the move is in progress you will see a progress bar filling up:

ANGIE - Off-site files being copied



When the process is finished, ANGIE will let you know:

ANGIE - Off-site files finished copying



Clicking on the Next step button will let you move the next off-site folder.

When you have move all off-site folders, upon clicking Next step you will be taken to the Site Setup page. The exact page depends on the installer you are using:

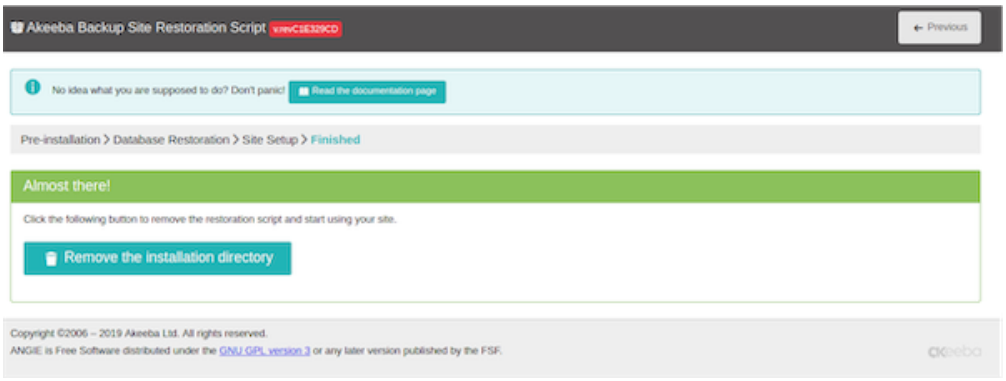
- ANGIE for Joomla! [angie-joomla-setup]
- ANGIE for Miscellaneous PHP Applications [angie-misc-setup]

If you don't see that page, maybe you want to take a look at this troubleshooting page [https://www.akeeba.com/documentation/troubleshooter/abiafterdb.html].

5.1.6. The “Finished” page

Normally, you should see a success message like this:

ANGIE - Off-site files finished copying



If ANGIE could not write to your site's configuration file, it will present you with a dialog box informing you of this fact. You can close the message by clicking on the "X" button on its top right corner. You can then copy the contents of the text area and paste it into your configuration file - replacing any and all existing content - manually, following the instructions printed on this page. This is only required if your configuration file was not writable in the first place. Under most circumstances this won't happen. Do note that if you get this message and you do not follow the steps printed on the page your restored site will not work.

You now have three options to proceed, depending on how you extracted the backup archive.

If you used Kickstart to extract the backup archive, close the window/tab of the restoration script and return to Kickstart's window. You will see a big “Clean Up” button. Just click on it and it will automatically remove the installation directory, the backup archive, `kickstart.php` and its translation files, as well as rename `htaccess.bak` to `.htaccess` and `php.ini.bak` to `php.ini` respectively. No further action is necessary. Your restored site is ready for use.

Important

If you had created a `kicktemp` directory for Kickstart's FTP mode to work properly, you must remove it manually with your FTP client application. Kickstart will not do that automatically.

If you were not using Kickstart, you can try clicking on the Remove the installation directory button to have ANGIE try to remove the installation directory automatically. If it succeeds, it will present you with a success dialog. Just click on the Visit your site's front-end button to visit your site's front page. However, if you used Kickstart to extract your files and clicked on this link accidentally instead of using Kickstart's Clean Up button you need to manually rename `htaccess.bak` to `.htaccess` and `php.ini.bak` to `php.ini`, as well as remove the archive file, kickstart.php file and all its INI files and the `kicktemp` directory (if one is present) manually.

If you uploaded the extracted files manually, you must remove the installation directory from your site using your FTP client application before visiting your restored site. If you don't, the restoration page may appear again. In this case do not run the restoration again. Just remove the installation directory through FTP and retry visiting your site.

5.2. ANGIE for Joomla!

These are the specific instructions for the ANGIE for Joomla! site restoration script which is supposed to be used when backing up sites based on the Joomla! content management system. Only the pages specific to this script are described here. For the common instructions among all ANGIE installers please consult the relevant section [angie-common].

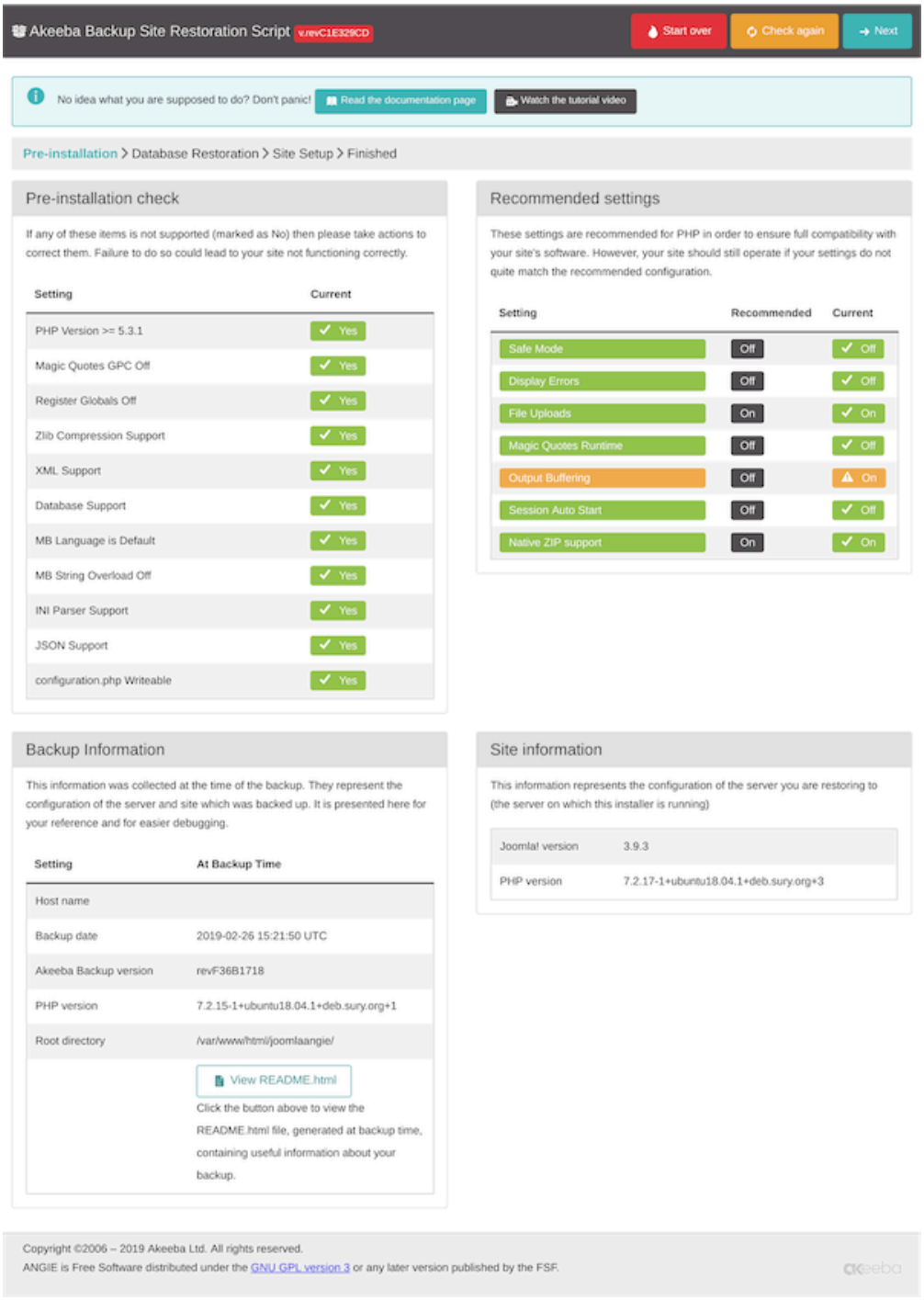
5.2.1. First page

If you are restoring to a different site or a server with a different PHP version than the one you backed up from you will see warnings in yellow background at the top of the page. Please note that the existence of these warnings doesn't mean that the restoration won't work. They are there to warn you to the fact that your restored site may or may not work in the new server because the server configuration may be different and some extension may not be compatible with it. It's the kind of information that is impossible to know before finishing the restoration.

If any of the settings under the Pre-installation check header is red, most probably the restoration will fail, or Joomla! will not run properly. Several users have reported that even when the MB language is default is set to `No` your site does get restored and does work properly. Take this reported success with a grain of salt, as the Joomla! project recommends otherwise and continue the restoration at your own risk. Please note that if any of these settings is shown in red you have to ask your host for support. Please do not ask us for support; we are not your host and we cannot change your server's configuration.

Recommended settings contains a series of optional settings and their recommended values. If any of those values is in orange, your site will be restored and will most likely work without a problem. It is common to have 2-4 orange items on most commercial hosts and we can attest that Joomla! works just fine on them. If you want more information about that, please take a look at our relevant troubleshooting page [https://www.akeeba.com/documentation/troubleshooter/abiredsettings.html].

ANGIE for Joomla! - The main page



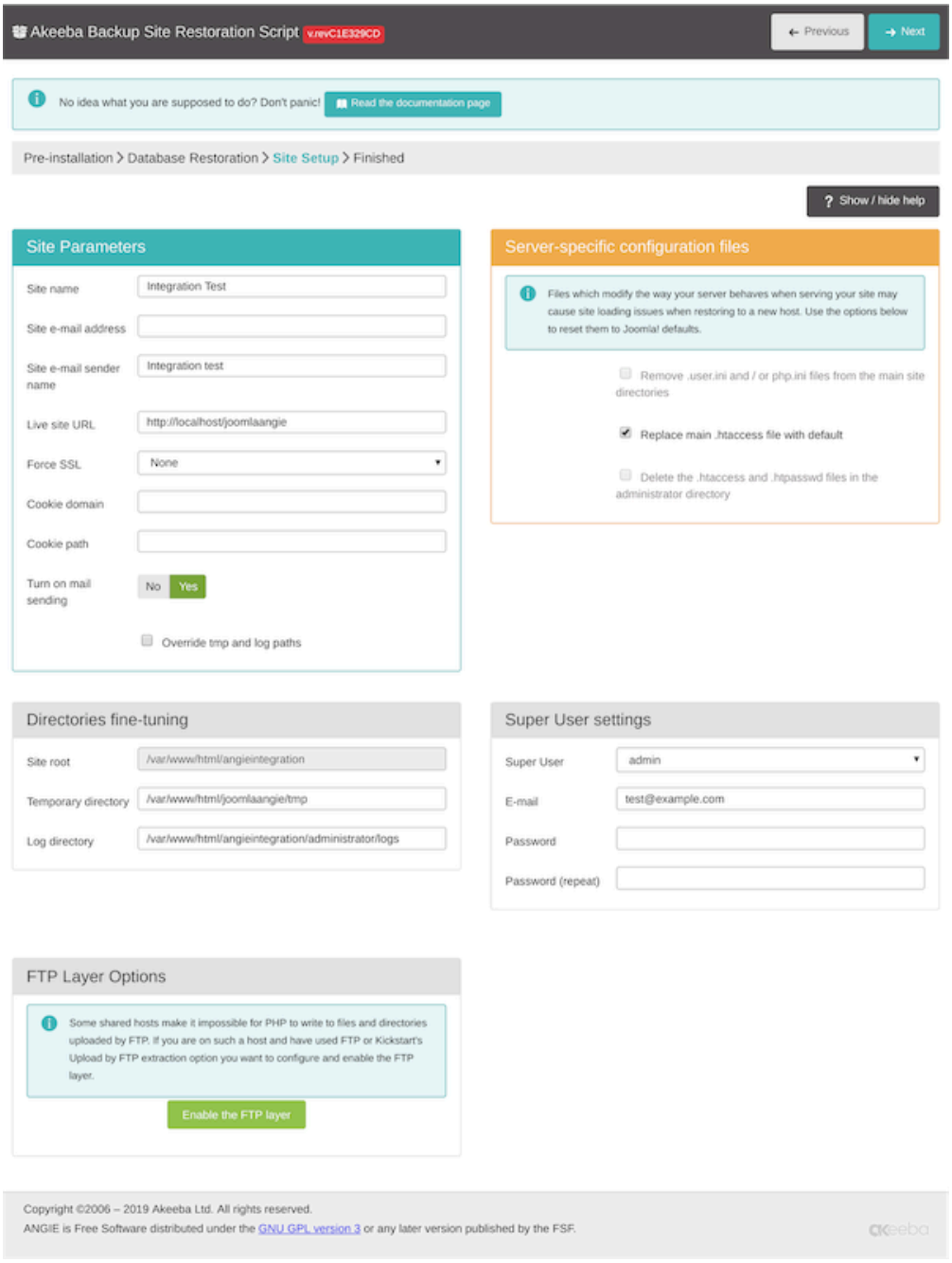
Below these areas you can find two information sections. The Backup Information column shows you information about the site you backed up from. Please note that this is the information Akeeba Backup recorded while taking the backup and they are presented here for your information only. Next to it you will find the Site Information column. This shows information about the current site you are restoring to. This is shown for your information.

When you're ready, please click on the blue Next button in the upper right hand corner of the page to proceed to the database setup page.

5.2.2. Site setup page

The Site Setup page can be used to optionally modify details of your restored site. It's not mandatory to go through its options if you are restoring on top of the same site you backed up from. All of the information in this page is pre-populated with the values read from the configuration.php file which was present in your backup archive's root.

ANGIE for Joomla! - The site setup page



The Site Parameters area contains the most basic options for your site.

The Site Name is the name of the restored Joomla! site which appears throughout the Joomla! application. The Site e-mail address is the e-mail address from which all e-mail sent out from your site will appear to originate from. Similarly, the Site e-mail sender name is the sender's name appearing in those e-mails' From field.

The Live site URL is optional and normally not required on the vast majority of hosts. If your site doesn't seem to work properly – e.g. missing pictures, all links resulting in 404 errors, etc – you may want to fill in your site's URL, for example `http://www.example.com` (include the `http://` part, but not a trailing slash or `index.php`!).

The Force SSL option will work exactly as the one in Joomla! Global Configuration. You can set it to **None**, **Administrator Only** or **Entire Site**. This is useful when you are restoring an SSL site on a server where you don't have any SSL certificate (for example localhost).

You will also see two more settings here regarding cookies. Under normal circumstances, both of them should be left empty. You only need to edit them if they are not blank and you are transferring your site to a different directory or domain name. The Cookie domain is the domain name of your site, without the protocol and, usually, without the www part. For example, if you are restoring to http://www.example.com, the Cookie domain is example.com (I will stress that again: there is NO http:// in there!!). The Cookie path is the subdirectory of your site, relative to the domain's root. If you are restoring to the root of a domain, e.g. http://www.example.com, then it is / (a single forward slash). If you are restoring to a subdirectory it's a slash followed by the directory's name. For example, if you're restoring to http://www.example.com/joomla then the Cookie path is /joomla (WITH a leading slash, but WITHOUT a trailing slash).

Warning

If the Cookie domain and/or Cookie path settings are non-empty and do not correspond to the location (domain name and directory) you are restoring your site to, **YOU WILL NOT BE ABLE TO LOG IN** in the front- or the back-end of your site. On most servers you can just leave them blank (strongly recommended). Be advised that if you request support for this issue you will be ignored because there is nothing we can support you with; you are simply entering the wrong values in these fields. You have to either retry the restoration or edit your configuration.php file and modify the cookie_domain and cookie_path parameters.

Turn on mail sending. Sometimes you don't want to enable mailing on your restored site, this option will modify your Joomla! Global Configuration to disable emails being sent using Joomla! framework.

Reset session options sets the Session Handler to Database and removes any customised Session Save Path; these are the Joomla! defaults. This allows you to use the restored site on a different server where the session handling settings of the original site and server don't work, e.g. if the original site used Redis or memcached, or if the original site used a custom Session Save Path.

Reset caching options sets the Cache to OFF - Caching disabled and the Cache Handler to File; these are the Joomla! defaults. This allows you to use restored site on a different server where the caching handling settings of the original site and server don't work, e.g. if the original site used Redis or memcached.

The Override tmp and log paths is a handy feature if you are restoring to a subdomain or subdirectory of the site you backed up from. It will force the paths to the `tmp` and `log` directories to point inside the restored site's `tmp` and `log` directory respectively. If you don't check this box, it is possible that the restored site will reference the old site's `tmp` and `log` paths, potentially causing issues in the long run. As a rule of thumb: always check this option unless you know what you are doing!

The Server-specific configuration files section helps you address issues relating to files which can change how your server performs when loading the site. Not all options are available or visible. ANGIE detects which ones can be used and disables or hides the other ones.

The .htaccess Handling drop-down allows you to decide what to do with your site's main .htaccess file, stored in your site's root. Please note that if you have a file named htaccess.bak the options here apply to that file as well. The reason is that upon clicking on Clean Up either in ANGIE or in Kickstart this file will be renamed to .htaccess. This entire feature or some of its options may not be available to you; ANGIE detects what is possible for your site and server.

None
: This option leaves the .htaccess file as-is.

Use default
: This option replaces the .htaccess file with the contents of the latest version of Joomla's default .htaccess file, the one that ships as htaccess.txt with Joomla itself. This option retrieves the latest version of htaccess.txt from Joomla's GitHub repository. It is conceivable that it might fail due to network issues.

Remove PHP handler lines
: Many hosts allow you to select which version of PHP you want to use with your site through their hosting control panel. In most cases this is done by adding some special lines which begin with AddHandler or SetHandler in your .htaccess file. However, these lines are specific to the host and sometimes to the server for which they were created. If you use the wrong AddHandler/SetHandler line you will get a 500 Internal Server Error or you will see the raw PHP source of your files when you try to access your site. Selecting this option removes the AddHandler/SetHandler lines, preventing this problem.

  Do note that if you select this option and after clicking Next you might get an error or you may find that the Clean Up button in Kickstart no longer works. If this is the case please log into your hosting control panel and select a newer version of PHP.

Replace PHP handler lines
: Some hosts use a very old PHP version by default. We've seen hosts defaulting to PHP 5 versions which went end of life in December 2019 or even as early as 2014! These old versions of PHP are unsafe for use on live sites. Most importantly, neither our backup restoration software nor modern CMS (like Joomla and WordPress) will work with these old versions.

  Typically, you'd go to your hosting control panel and select a newer PHP version before running Kickstart, the Site Transfer Wizard or the restoration script (ANGIE). What happens is that your host modifies your .htaccess file with an AddHandler or SetHandler line which tells the server to use a newer PHP version. However, your backup may already contain a .htaccess file with different or no AddHandler/SetHandler lines. If you were to proceed with the restoration of the .htaccess file as-is you'd end up in a situation where the restoration cannot proceed.

  This option tells ANGIE to look in the .htaccess file for AddHandler/SetHandler lines and apply them to the htaccess.bak file; this is the file that is renamed back to .htaccess when you click on Clean Up in ANGIE or Kickstart. By doing that it ensures that your restored site will continue using the newer PHP version you selected.

  ANGIE tries to automatically detect when this is needed and preselect this option for you. If you see it preselected we strongly recommend keeping it this way.

The Remove .user.ini and / or php.ini files from the main site directories option tells ANGIE to delete any .user.ini and php.ini files from the site's root and the administrator folder. These files are used to modify PHP configuration parameters. When transferring your site between servers they are likely to cause problems, preventing you from accessing your site. If this is the case repeat the restoration and select this option.

The Delete the .htaccess and .htpasswd files in the administrator directory option removes the password protection from the site's administrator directory. You need to do that if you are transferring your site between domains. The reason is that the .htaccess file in the administrator folder contains an absolute filesystem path to the .htpasswd file which is different on each server. Moreover, .htpasswd files may require different formats on each server. Removing the administrator password protection is the best approach to ensure you can log into your site's administrator after restoration. You can use your hosting control panel or third party software, such as Admin Tools, to reapply administrator login password protection on your restored site.

The FTP Layer Options area contains the necessary settings for enabling Joomla!’s FTP layer.

The Enable the FTP layer will activate Joomla!'s FTP layer, which forces the Joomla! core and several conforming extensions to write to your site's files using FTP instead of direct file access through PHP. This is designed to work

around permissions issues with the majority of shared hosts. If you had to use Kickstart's Use FTP option or if you uploaded the extracted files manually through FTP you must enable it and go through these settings, unless your host told you that they are using suPHP.

The rest of the FTP settings are exactly the same as those you had to fill in Kickstart:

Host name	Use the domain name to access your site's FTP server
Port	Leave the default value (21) unless your host tells you otherwise. Do note that Joomla! only supports plain FTP. If your host tells you to use port 22 – which is used only by SFTP – it won't work.
Username and password	What they claim to be, the user name and password to connect to your site's FTP server
Directory	The absolute FTP path to your site's root. The easiest way to find it is using FileZilla to connect to your site and navigate to your site's root, which is usually a directory named `htdocs`, `http-docs`, `http_docs`, `public_html` or `www`. Look at the right hand pane, above the folder tree (Remote site text box). Copy it and paste it in the Directory box.

Tip

You can instead fill in all of the other information, leave this field blank and click on the Browse button next to it. If your FTP information is correct a popup directory browser appears. You can now browse to the site root directory. It's the one where you can see your site's `installation`, `includes` and `libraries` directories. Once you're in there click on the Use this button.

The next area is called the Super User settings.

In this pane you can change the details of one of the Super Administrators on your site. First, select the username of the Super User you want to modify from the User name drop-down list. Then, simply type and retype the new password in the two fields below. The final field, E-mail address, is the e-mail address linked to that Super User. Make sure the address you type in here is not used already used by another user of the site or you will be unable to reset your Super User password if you forget it.

Important

This feature can change the password of **exactly one** Super User account (the one selected in the drop-down box). Its reason of existence is to allow you to reset the Super User password should you forget it or quite simply don't know it (e.g. restoring a client's site to a dev server).

Then, we have the Directories fine-tuning pane with advanced settings meant for power users and site builders.

The two options you can modify are the Temporary Directory and Log Directory paths. For your convenience, the absolute path to the site's root is displayed above. You should only need to use these fine-tuning parameters if you want to place the `tmp` and `log` directories outside your site's root. Both of them must be absolute paths. For your convenience the absolute path to the site's root is printed above so that you can get them right every time.

Finally, click on the Next button to let ANGIE write your site's new `configuration.php` file and display the final page.

5.3. ANGIE for Miscellaneous PHP Applications

These are the specific instructions for the ANGIE for Miscellaneous PHP Applications site restoration script which is supposed to be used when backing up sites which are not based on a content management system or script supported

by a specific ANGIE installer. Only the pages specific to this script are described here. For the common instructions among all ANGIE installers please consult the relevant section [angie-common].

5.3.1. First page

If you are restoring to a different site or a server with a different PHP version than the one you backed up from you will see warnings in yellow background at the top of the page. Please note that the existence of these warnings doesn't mean that the restoration won't work. They are there to warn you to the fact that your restored site may or may not work in the new server because the server configuration may be different and some extension may not be compatible with it. It's the kind of information that is impossible to know before finishing the restoration.

Important

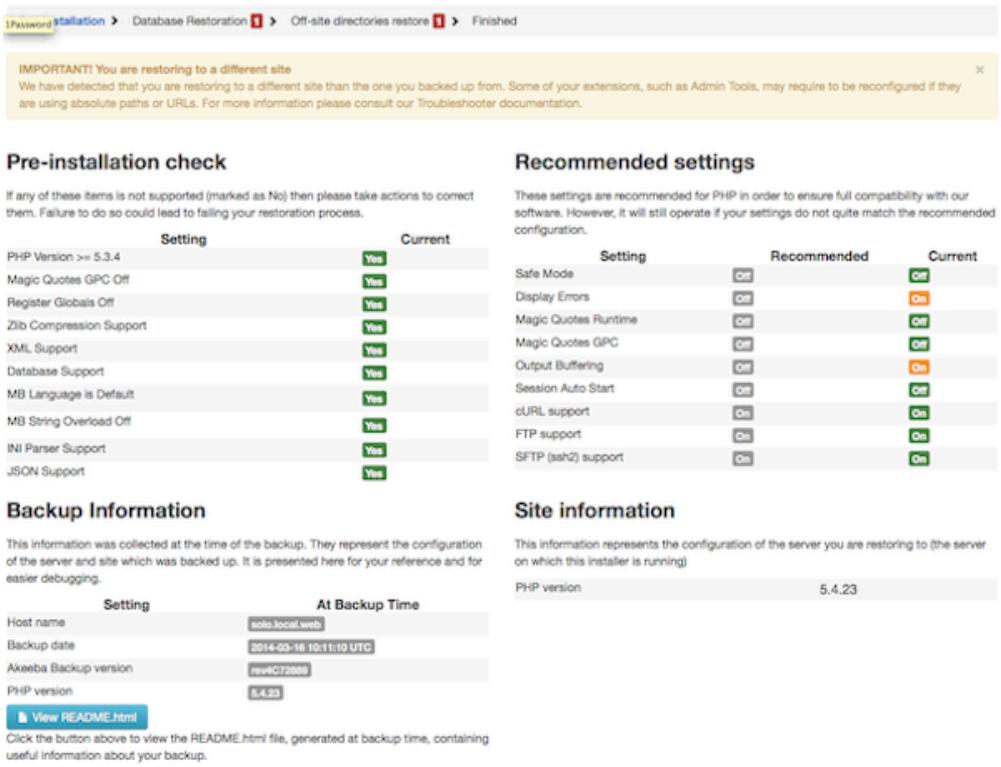
Everything you see in the pre-installation check and recommended settings are NOT the minimum requirements or recommendations for the site script you are restoring. These are the minimum requirements and recommended settings for ANGIE itself. It is possible that some of the features detected as missing or sub-optimal are actually not used at all by your site's script.

Please keep that in mind that before posting a support request in the spirit of "I have backed up a site running WhateverScript 2.0 and when I try to restore it ANGIE complains about missing gzip compression which WhateverScript isn't even using". It's not a bug, it's not an issue and can't be "fixed" unless we add an installer for your specific script (which is unlikely if we have not announced plans to do so). If you are using a specially supported script such as Joomla! please use the specific ANGIE installer for that script.

If any of the settings under the Pre-installation check header is red, most probably the restoration will fail. Several users have reported that even when the MB language is default is set to `No` your site does get restored. Take this reported success with a grain of salt, as we do understand the problems caused by this sever setting. If you choose to continue the restoration it is at your own risk. Please note that if any of these settings is shown in red you have to ask your host for support. Please do not ask us for support; we are not your host and we cannot change your server's configuration.

Recommended settings contains a series of optional settings and their recommended values. If any of those values is in orange, your site will be restored and will most likely work without a problem. Some restoration features (e.g. using FTP) might not work.

ANGIE for Miscellaneous PHP Applications - The main page



Below these areas you can find two information sections. The Backup Information column shows you information about the site you backed up from. Please note that this is the information Akeeba Backup recorded while taking the backup and they are presented here for your information only. Next to it you will find the Site Information column. This shows information about the current site you are restoring to. This is shown for your information.

When you're ready, please click on the blue Next button in the upper right hand corner of the page to proceed to the database setup page.

5.3.2. Site setup page

There is no such page in this installer. Since it's a generic installer which doesn't know which kind of site script you are using it is unable to reconfigure your site. Instead, you will be taken directly to the display the final page.

6. Restoration (ANGIE) troubleshooting

6.1. ANGIE reports that the session write path and the installation directory is unreadable

Your first approach should be asking your host to fix the PHP session save path to something which is writable by your hosting account. This is a very common misconfiguration issue on Plesk-based hosts. If your host can't fix it, they might be able to give you instructions for creating a `.htaccess` to override the PHP session save path. No matter which solution they provide, you don't have to run Kickstart again or upload the extracted files again; you can access ANGIE again by visiting `http://www.example.com/installation/index.php` where `www.example.com` is the domain name you are restoring to.

Another solution is to change the permissions of the `installation/tmp` directory after Kickstart has extracted your archive (just right before you click the Run the Installer button) or after you have uploaded the extracted files if you're not using Kickstart. In both cases, try using your FTP client to change the permissions of the `installation/tmp` directory (but NOT its contents!) to 0777. Then, click on the Run the Installer button in Kickstart or access ANGIE again.

If both of these solutions don't work out for you, you will unfortunately have to go through the scenic route, i.e. perform a manual restoration. The complete instructions are described in the "Unorthodox: the emergency restoration procedure [https://www.akeeba.com/documentation/akeeba-backup-joomla/unorthodox-emergency-restoration.html]" section of our User's Guide. Our suggestion: it's best to switch to a different host. There is a strong possibility that Joomla! might not work properly unless you have a working PHP session storage.

6.2. PHP errors , warnings, notices or a blank page upon accessing ANGIE / restoration

If you get a PHP fatal error, Internal Server 500 or a blank page you have to make sure that your server is indeed running on a supported version of PHP for the version of Akeeba Backup you are installing. Please consult our version compatibility matrix [https://www.akeeba.com/compatibility.html].

In order to determine which PHP version you're using, please create a file named `info.php` with the following contents:

```
<?php phpinfo(); ?>
```

Upload it to your server and access it as `http://www.example.com/info.php`, where `www.example.com` is your web site's domain name. You will see your PHP version on top. If the PHP version is unsupported please ask your host to immediately upgrade your PHP. Do not ask us for such instructions. This is absolutely server-specific; only your host can help you with this.

If, however, you are getting PHP Warning, Notice, Strict Standards or Deprecated messages, this is nothing to worry about, meaning that ANGIE will work properly but its interface might be so loitered with the messages that it's impossible to use. In that case please ask your host about instructions on either disabling PHP error output to the browser or setting the error reporting level to E_ERROR. If you are on a local server (e.g. XAMPP, MAMP, WAMPServer, etc) you have to edit your `php.ini` file and change the following two lines:

```
error_reporting = E_ERROR
display_errors = 0
```

If you don't know where your `php.ini` file is stored, please consult the documentation or support forum of your server stack (e.g. XAMPP) you are using. We know about a few of them, but not all of them. It's best to ask such questions on the official site of each server stack where you can get proper support about those non-Akeeba products.

Important

You MUST restart Apache (the web server service) after applying those changes. If you are unsure about how to do that, you'd better shut down and restart the server stack (e.g. XAMPP) you are using.

6.3. Some required or optional settings are red in ANGIE's first page

If any of the required settings are marked in red the restoration may be able to proceed, but your site will most likely not work. Do note that these are the required settings of Joomla!, not Akeeba Backup. You will get the exact same

errors if you try to install a brand new Joomla! site on that server. In order to solve them, please consult Joomla!'s documentation or the official Joomla! forums [http://forum.joomla.org].

6.4. I can't restore my database, or receive AJAX Error, timeout or other errors while restoring my database with ANGIE

Let's try some common troubleshooting steps. In 100% of cases we've seen (yes, we really do mean all of them) it's one of the following.

1. If you are not restoring on the same host as the one you backed up from, if you are transferring a site from a live server to a local server or if you are transferring a site from a local server to a live server you need to ensure the following:

   - You MUST create a new database BEFORE trying to restore your database. ANGIE can not create a new database unless you are restoring on a local host and using the root credentials. In other words, almost never. As a rule of thumb: always create a database before restoring your site. There are instructions for creating a new database below.
   - If you are on a cPanel host you have to add the user to the database. Consult your host's documentation about that.
   - You MUST supply the database connection information of **your new host** BEFORE clicking on the Next button. ANGIE will present you with the database connection information of the site you backed up from. These are no longer valid on the new host and you MUST supply the new host's database connection information.

     Further to that you DO NOT have to create databases and users on the new to match your old site's information. This is usually the extremely hard way to go. Simply, type in the new host's database connection information in ABI's interface BEFORE clicking on the Next button.

2. First try toggling between the mysql (no trailing i) and mysqli (with a trailing i) database drivers. Some hosts support both, most hosts support only one or the other.

3. Then check your database host name. It's not always localhost. Some hosts use an IP address (e.g. 1.2.3.4) others use a full domain name (i.e. mysql.myhost.com). If in doubt, please ask your host.

   One very picky case is using MAMP on Mac OS X. Using localhost will not allow you to connect to your database. Using 127.0.0.1 will work! The reason is that PHP treats these two values differently. Using localhost instructs PHP to use "named pipes" to access the MySQL server. Unfortunately, MAMP can't provide this feature. Using 127.0.0.1 instructs PHP to use TCP/IP networking which what MAMP supports. This trick may also work on a live host, try it!

   If your site is hosted on GoDaddy, other users have reported that the actual database server name is hidden in a small footnote on the bottom of the page where it displays the list of databases. If in doubt, contact GoDaddy's support. If you can't get an answer, tell to the support person (an underpaid, misinformed, script reading youngster overseas) that you want to "escalate this issue". After a short while you will be connected to a "real" support technician who can help you.

4. Check your database name. Hosts based on cPanel and Plesk, as well as GoDaddy, prefix the database name with your account name. For example, if your user account name on the host's server is "foo" and you try to create a database named "bar" the real database name is "foo_bar", *not* just "bar"! Do note that this prefix *has nothing to do with the table name prefix (e.g. jos_) used by Joomla!*. If in doubt, ask your host.

5. Check your username. The same principles with the database name apply, i.e. those hosts will prefix the database username with your account's name, e.g. myuser_mydatabase. If in doubt, ask your host.

6. Check your password. Many servers will not work with password which contain special characters (e.g. at sign, hash, dollar sign, percent sign, caret, ampersand, start, parentheses, equals sign, etc). The best approach is to make sure that your password consists only of alphanumeric characters, i.e. a-z, A-Z and 0-9, don't use any accented characters or characters with diacritic marks.

   This is not a limitation or bug in Akeeba Backup Installer. The problem comes from the host's Apache setup, namely the mod_security2 rule set. Such servers are set up to filter potentially dangerous and suspect query parameters. A password with special characters like a hash sign, dollar sign, percent sign, caret and star will most likely trigger that protection, throwing a 406 (Not Accepted) error. Since most servers do not have a server error document for the extremely rare 406 error, you also get a 404 error (Not Found) when Apache tries to retrieve the error document for the real error, the 406 error. Of course, all of that makes it completely impossible for a regular human to figure out what is going on and think that it is a bug in Akeeba Backup Restoration Script. It's not. We'd like to sincerely thank the engineers at Rochen who helped us troubleshoot this issue and confirm that it's not a bug in our code.

7. If you have created your database and user on a cPanel host (e.g. Rochen) this is not sufficient to connect to it. You also have to add the user to the database. Otherwise your server doesn't know that the username you are using should be able to access your database. This is an often overlooked step. Yours truly confesses that he is doing the same mistake every time he's restoring a site on a cPanel host.

8. Finally, if you had created your backup archive with Akeeba Backup 3.0.x or if you are unsure about the version of Akeeba Backup which created the archive, please try using a database password which consists only of unaccented alphanumeric characters (a-z, A-Z and 0-9, no accents, umlauts, tildes or other decorations).

Please *do not* ask us for support regarding this matter if you have not gone through this list very carefully at least twice. All the possibilities are listed above.

If you are a subscriber and have gone through the above list at least twice and it still doesn't work, please file a ticket in our ticket system indicating that you have gone through the list of this page and mentioning all of the following information:

- Akeeba Backup version number ("latest" is not a version number; something like 1.2.3 is)
- Exact PHP version number
- Exact MySQL version number
- Be advised that you may have to tell us the password you tried to use.

Creating a database on GoDaddy

If you're on GoDaddy, just like on most commercial hosts, the Akeeba Backup Restoration Script can not automatically create the database for your restored site. In order to create a database you must do the following:

- Go to your Hosting Control Center
- Click on Databases, then MySQL and finally on Create Database.
- Fill in your description, name, etc.
- Click on OK
- Go back to MySQL.
- The new database will be there or if you keep adding them there will be more than one.
- Hover of the "Pencil" on the Action Tab and then click the pencil. All the MySQL Database Information will now appear here. You fill in the Akeeba Installation field with this info so make a note of it.

- When you come to install the Akeeba Backup installation fill in the details from it. Don't forget to replace localhost with the new godaddy host name: similar to this: yourakeebanamehere.db.12345.hostedresource.com.

Credits go to our user jcjweb for kindly providing this step-by-step guide on our ticket system.

Creating a database on a cPanel host

If your server is using cPanel, here's what you have to do:

1. First, login to your CPanel account. If you don’t have your username and password, you’ll need to contact your web hosting provider.

2. Once you’ve logged in, select MySQL databases from the list.

3. Type in a name for your new database and click the Add DB button.

4. Then, type in a Username and a Password into the relative fields and click the Add User button.

5. Once your Database and User have been created, you need to give that user permission to use that database. We do this by selecting the username we created from the drop down and the database we created from the drop down, making sure the All checkbox is selected and clicking the Add User to DB option.

6. The last thing you will need is the MySQL server hostname. If this hasn’t been supplied to you by your web host, you can get it from the MySQL databases page.

Important

Note that the Username and Database will be prepended with your Cpanel login name. For example, if you created a database called DB and your cpanel login was mysite then your new database will be mysite_db.

Credits: These instructions were adapted from a blog post [http://www.interspire.com/content/2006/04/04/how-to-create-a-mysql-database-in-cpanel/] at Interspire

Creating a database on a Plesk host

If your server is using Plesk, here's what you have to do

1. First navigate to the hosting control panel for the domain that you want to create the new MySQL database for.

2. Click on the Databases icon.

3. Click on Add New Database.

4. You will be asked to enter the following details:

   - **Database name**. This is the name that you want to give the actual database, Plesk suggests that you begin the database name with your Plesk login followed by an underscore, it is after the underscore that you should enter the desired database alias.

   - **Database type** (may not be available on your host). This is where you select what database program you want the new database to be created in, if there is no more than one option then this field will not be shown – in the cases of Windows based Plesk website hosting, you are able to choose between MySQL and MSSQL databases, although the options are dependent on what database engines your web host has installed on your hosting server and what options they have enabled within their Plesk license. You MUST create a MySQL database for Joomla! to be restored to.

5. After clicking the OK button after entering the relevant details, you will be redirected to the management screen for the database that you have just created. In order to make your database usable from Joomla!, you will need to select the Add New User button.

6. You will then be prompted to enter the following details:

   - **Database username**. This is the username that you will use to gain access to your newly created database
   - **Password**. Enter your desired password here.

Credits: These instructions were adapted from a blog post [http://blog.eukhost.com/webhosting/creating-a-database-in-plesk/] at EUKHOST

Create a database under XAMPP, WAMPServer, MAMP or any other local server using phpMyAdmin

Google is your friend. Here's a good tutorial I came across by searching on Google: http://complete-concrete-concise.com/web-tools/creating-a-mysql-database-using-xampp

Even though the instructions are for XAMPP, it's the same thing for all local servers, as they're all using phpMyAdmin to administer the MySQL database.

6.5. I restored my database but can't proceed to the next page of ANGIE

If you had backed up multiple databases, after restoring the first database (the main Joomla! site database) you will be asked for the connection information of the second database. If you don't look closely at the page you might believe that you are stuck on the same page. That said, you can see the name of the database ANGIE is trying to restore on the top of the page, above the connection information. You can not skip a database. You have to restore all of them.

6.6. My configuration.php wasn't written to disk after ANGIE ran

Follow the on-screen instructions. They tell you what you have to do. Essentially, you copy the code from the last page of ANGIE, save it as `configuration.php` and upload it to your site. There is no simpler way to do that.

6.7. Any other ANGIE error

Please note that ANGIE (Akeeba Next Generation Installer Engine) runs after Kickstart, albeit some people confuse the two of them.

Unfortunately, other issues are usually very server-specific and can not be dealt with through the troubleshooter. Direct access to the server is required to solve most of them. If you are a subscriber you can file a support request in our ticket system so that we can provide you with one-on-one help, resolving the issue directly on your server. In this case please indicate that you have gone through the troubleshooter and mention all of the following information:

- Akeeba Backup version number ("latest" is not a version number; something like 3.2.1 is)
- Exact PHP version number
- Exact MySQL version number

Please try to describe your issue as better as possible. If possible, please also post screenshots of your problem, especially if your copy of ANGIE is displayed in a language other than English.

7. Troubleshooting restored sites

Sometimes, when restoring a site on a different host or when transferring a site between a local and a live server, you will end up with a misbehaving site. The following pages will allow you to troubleshoot such issues.

7.1. Common issues on restored sites and how to solve them

You have to make sure that there are no settings transferred in files from your old server which are not compatible with your new host. The most notable culprits are `.htaccess` and local `php.ini` directives.

If your site was using Admin Tools' .htaccess Maker

If you have used Admin Tools' .htaccess Maker please remember that after restoring the site to a different location you will need to reconfigure and create a new .htaccess. Login to the back-end of your site, go to Components, Admin Tools, .htaccess Maker, then change the domain and directory names at the bottom of the page and finally click on Save & Create .htaccess. This is mandatory, every time you move your site to a different host, domain name, subdomain or directory.

I can't log in to my site after I restored it to a new location

This is a very common mistake with Joomla!1.6 and later versions. What you probably not remember is that you modified the cookie setup parameters in your site's Global Configuration page. The thing is, if you modify the cookie domain name and/or path, it's very likely that you will no longer be able to log in to your site if the domain name, subdomain or directory changes - exactly what happens when you restore a site to anywhere except its original location! Luckily, the workaround is very simple. Please edit the `configuration.php` file in the root of your site and find the lines starting with `public $cookie_domain` and `public $cookie_path`. Modify them so that they read:

```
public $cookie_domain = '';
public $cookie_path = '';
```

Save the file, clear your browser's cookies and cache, quit and restart your browser and try logging in to your site. You should be able to login without any problem now.

Another thing that you should be aware of is that the same problem could be caused by your .htaccess file. It's always a good idea to at least temporarily rename .htaccess to something else (e.g. htaccess.bak) when you're trying to troubleshoot a login issue. A .htaccess file may define redirections which get in the way during login.

Tip

You can change these parameters when restoring your site. We advise to clear (delete the contents of) these settings on most servers. In fact, if you need to set them to something other than blank (blank means "Joomla!, figure it out yourself") you would have already known what to set them to and why you need to do that.

I cannot log in or my site crashes with a blank page / Internal Server Error 500 page as soon as I try accessing it

There is one more thing which has to do with your caching and session storage options in your site's `configuration.php` file. If you are using redis, memcached and so on it is possible that these are not supported by your new host. Just to be on the safe side, edit your `configuration.php` file and change the following lines to read:

```
public $cache_handler = 'file';
public $caching = '0';
public $session_handler = 'database';
```

The first two lines set the cache handler to use files and disables caching which addresses blank page / 500 error when accessing your site. The latter line sets the session handler to the default (database) which addresses log in issues.

PHP memory issues

If you are restoring to a local server, please make sure that your PHP memory limit is adequately high. On some local hosts the default setting is 8Mb, which is too low for Joomla!. You can determine this be editing your local server's `php.ini` file. Look for a line like this (the value 8M might be different):

```
memory_limit = 8M
```

Change it so that it reads:

```
memory_limit = 128M
```

If you are on a live host, please ask your host and make sure that your PHP memory limit is at least 32M. If it's not, ask your host for the proper way to increase it.

.htaccess directives

Look in your site's .htaccess file for directives such as `php_value`, `php_flag` and `AddHandler`. Try commenting them out (putting a hash in front of the line) to see if it helps.

Usually they have a format like this:

```
AddHandler application/x-httpd-php5 .php
```

You most probably have to remove those lines beginning with `AddHandler`, especially if your problem is that you get a bunch of code, or the web browser offers to download index.php, instead of your site's front page. You most certainly have to remove such lines if you are restoring on a local server.

You should also try commenting out lines (by placing a single # character in front of them) which look suspicious to you, because any of those directives may cause trouble. If in doubt, get a fresh Joomla! package, extract the `htaccess.txt` file from it, rename it to `.htaccess` and upload it to your host.

Redirections in .htaccess

Check if you have redirections in your `.htaccess` file, for example directing all traffic to the www prefixed site or to a specific domain, e.g. all traffic to `www.example.com`, even if it referenced `example.com` or `example.net` in the URL. Such problems are easy to spot because you have put this code in the `.htaccess` file and you should know about what it does. Just remove it or comment it out.

RewriteBase in .htaccess

Another thing you should look into is the `RewriteBase` line. Normally, you need something `RewriteBase /` if your site is on the root of the domain, or `RewriteBase /mydirectory` if it's inside a directory named `mydirectory`.

You should note that some servers do not accept `.htaccess` files. Putting such a file on your site's root will make the server throw an HTTP Error 500: Internal Server Error as soon as you try to access your server. If this happens, you need to have a little chat with your host.

Enabling Apache's mod_rewrite

If you are restoring on a local host, you have to make sure that your server is loading the mod_rewrite module, otherwise you will most assuredly get a blank page or a 500 Internal Server Error.

If you are using WAMPserver on Windows you must note that mod_rewrite is not loaded by default. In order to enable it, you have to click on WAMPserver's tray icon, Apache, Modules and make sure that Rewrite is checked. If not, click on it and wait for the server to restart. This is required only the first time you restore to a WAMPserver installation and only if you have SEF URLs turned on and you are using Joomla!'s `.htaccess` file.

Other local servers, like XAMPP, also come with the mod_rewrite Apache module disabled. These servers require you to edit the `httpd.conf` or run other system commands. Please consult your server package's documentation for more information on enabling mod_rewrite.

Some live hosts also do not have Apache's mod_rewrite enabled. If trying to use Joomla!'s stock htaccess.txt renamed to .htaccess causes an immediate blank page or Internal Server Error 500 page on your site, please consult your host.

Special note for GoDaddy users

GoDaddy users will find out that the `.htaccess` changes need 10-30 minutes to take effect. This is a limitation of your host. Normally, these changes should take effect immediately, as happens with pretty much every other host including local installations. Unfortunately this behavior is specific to GoDaddy and neither you nor us have any way to change it.

The $live_site variable in configuration.php

Sometimes you might be getting URL errors. For example, the first page might display but clicking on any link returns a 404 error, while some other times the first page displays very weird, like the CSS and images are not loading. Both of those issues have nothing to do with the restoration itself, but your server setup and a clash with how Joomla! works. The easiest way to work around it is using the `$live_site` variable in your configuration.php file, if you haven't already set it up in ANGIE's Site Setup page. Edit the `configuration.php` file in the root of your site and modify it so that the line starting with var `$live_site` looks like this:

```
public $live_site = "http://www.mysite.com";
```

or (if you have installed in a subdirectory):

```
public $live_site = "http://www.mysite.com/mypath";
```

This will let Joomla! figure out the correct URLs to your site's CSS files, images and links and these errors will go away.

If you restored to a server which required the `$live_site` hack, next time do yourself a favour: use ANGIE's feature for changing the `$live_site` variable. It is available in the second to last step of the restoration procedure, just under the text boxes where you define your site's name and email details.

Redirection and SEF components and plugins

Sometimes restored websites redirect to the original site even when there is no such parameter in .htaccess and $live_site is correcty set in the configuration.php. In this case, please check if you have any SEF or redirection plugins installed, including any plugins which might be redirecting non-SSL to SSL URLs or vice-versa. Many such plugins and components store absolute URLs (URLs which include the domain name) causing wrong redirections. If this is not the case, read further down this page.

Problems with php.ini

If none of this helps, look for a file named `php.ini` inside your site's root. If it exists, try renaming it to `php.ini.bak` and retry loading your site. Also do the same thing in your site's administrator directory.

Third party components with absolute paths / URLs

As a side note, we might also add that some third party components, such as DOCman and VirtueMart, store absolute paths in their configuration. If you restored to a different location / server than the one you originally had the site you backed up, trying to access your new web site's public front-end might result in blank pages or HTTP Error 500. You will have to edit the configuration of those components and ensure that you have changed the paths to reflect the correct paths on your new server / location.

Some other software store the database table prefix of your site in their configuration. For instance, SQL2Excel stores the database table prefix of your site inside the SQL queries attached to each worksheet. If you changed the database table prefix when restoring the site you also have to change these SQL queries. If unsure, ask the developer of that specific software. We can't know how all several thousands of Joomla! extensions listed on the Joomla! Extensions Directory work. We can only provide support for our own software.

7.2. Common issues on restored sites due to PHP incompatibilities between the source and target server

PHP version incompatibilities

You have to consider the PHP version of the original and new server. If your original server was running on a newer version of PHP than the new server, you might end up with a blank page. This is especially true if your new host is running a different PHP version family (e.g. 5.6 vs 5.5 or 7.4 instead of 5.6) or has disabled some critical PHP functions. This is not a restoration problem, rather than a hosting configuration error. Some modules, components or plugins you have installed might be using functions which are not available on your new host. The only way to understand if this is the case is to have your host take a look at the error log and reconfigure your hosting environment to fix this issues.

If you are restoring to a local server, you might have one version of PHP which is too new or too old for Joomla! to work with. Please consult Joomla.org for PHP version compatibility information.

PHP memory issues

If you are restoring to a local server, please make sure that your PHP memory limit is adequately high. On some local hosts the default setting is 8Mb, which is too low for Joomla!. You can determine this be editing your local server's `php.ini` file. Look for this line:

```
memory_limit = 8M
```

Change it so that it reads:

```
memory_limit = 128M
```

If you are on a live host, please ask your host and make sure that your PHP memory limit is at least 64M. If it's not, ask your host for the proper way to increase it.

Error reporting matters

Finally, some people see PHP errors (Deprecated, Notice, Warning) when accessing their website which most of the times this is not a problem with the restoration but an issue with the server configuration. In most cases you can simply set Error Reporting to None in Joomla!'s Global Configuration page. If this doesn't work, please ask your host for information on disabling PHP's error output to the browser. Anyway, it's a good idea to do so in the first place! You don't want any minor glitch to reveal sensitive server configuration information to potential hackers.

If you are your own host, e.g. using a local installation of WAMPserver, XAMPP, MAMP, etc., the easiest way to do that is by editing your php.ini file and setting error_reporting = E_ERROR. Remember to restart Apache for the change to have any effect at all!

7.3. When updating the restored site, the original site changes as well (Entangled web sites)

Sometimes people get back to us claiming that whenever they change something in their restored site, the original site changes as well and vice versa. The problem is a small, but important, oversight when restoring the site.

During the restoration you are presented with your database connection information. By default, ANGIE displays the parameters pertaining to your original (old, backed up) web site if it believes that your site's URL and absolute location on disk are the same as those on the backed up site. Otherwise it shows you nothing there, expecting you to enter the connection information to a different database.

Unfortunately, some users get confused and believe that they need to enter the database information from the site they backed up. **This is wrong**. It will cause both the original and the restored site to use the same database data, meaning that changing one will change the other with unpredictable results.

You will need to restart the restoration of your site. Before doing that please create a new database. If that is not possible, for example your host limits the number of databases you can create, please use a different database table naming prefix in ANGIE. This will allow the restored site to use a different set of tables than the backed up site and the two sites will no longer be entangled.

7.4. Clicking on a link on the restored site takes me to the original site (link migration issues)

Please check if you have redirections in your `.htaccess` file, for example directing all traffic to the www prefixed site or to a specific domain, e.g. all traffic to `www.example.com`, even if it referenced `example.com` or `example.net` in the URL. Such problems are easy to spot because you have put this code in the `.htaccess` file and you should know about what it does. Just remove it or comment it out.

Maybe you have to set or change the `$live_site` parameter in your site's `configuration.php`. Normally, Joomla!™ is able to determine the absolute URL to your site's root without your help. However, it looks like that some servers lie, most notably Microsoft™ IIS™, so we have to help our favorite CMS with that. You had the chance to set this URL during the restoration, but it's never too late. Just edit the `configuration.php` file on your site's root with a plain text editor, e.g. Notepad++ or Kate, and find the line which starts with `public $live_site`. Change the rightmost part (the one between single quotes) to your site's URL, without a trailing slash. For example `http://www.example.com` - see? No trailing slash!

Finally, on some hosts it's required to change or enable the `RewriteBase` parameter in your .htaccess file. Most often than not it's enough to edit the file and find the line beginning with `RewriteBase`. If there is a hash in front, remove it. Change this line so that it reads:

```
RewriteBase /
```

or, if you restored in a subdirectory named myjoomla:

```
RewriteBase /myjoomla
```

7.5. Issues arising from your computer configuration, browser, ISP, antivirus and firewall incompatibilities

First try using a different browser. We recommend using the latest version of Google Chrome, Mozilla Firefox, Apple Safari or Microsoft Edge (after the Summer 2020 update). Make sure you disable all of its extensions first.

If you have AVG antivirus, please disable the Link Checker feature and *reboot* your computer. That feature of AVG is very intrusive and kills timing-sensitive Javascript procedures, like the backup procedure.

If this doesn't work, try disabling any antivirus/firewall/Internet security application. Windows Defender is safe to leave enabled.

If nothing of the above works, please try using a different PC, ideally connected to the Internet through a different ISP. We had many cases where the PC setup was broken and using a different computer did the trick.

If the above information doesn't help, please go back to the previous page and continue reading.

8. Unorthodox: the emergency restoration procedure

Warning

THIS IS NOT THE REGULAR RESTORATION PROCEDURE.

The following instructions are meant to be used in absolute emergencies, when the regular restoration procedure does not work.

If you are not sure what the regular restoration process is STOP NOW and go to our site to watch the Video Tutorials.

Note

These instructions are meant to be first read before disaster strikes. They are not meant as a checklist you follow when you're stressed out by your site being down. Please keep that in mind while reading them. Thank you!

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.

You'll have to use Akeeba Kickstart, available without charge from our website.

If you have a ZIP package you might be able to extract it using third party software. Typically, PKZIP for Windows, WinZIP and 7-Zip work best.

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 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) or pdomysql. 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!

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. Akeeba Backup fully supports Joomla's access control features, allowing you to set specific permissions for specific user groups. You can change this behavior from the component's Options button in the Control Panel page - just like with any other Joomla! component.

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.

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_akeebabackup/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, 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. Using a well known location would allow an attacker exploiting a security issue in a third party component to gain access to the backup archives. The only way around that is using a different directory, ideally one above your site's root.

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.akeeba.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 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 Ltd `http://www.akeeba.com`
Described version 1.9		
Revision History		
	January 2017	NKD, Akeeba Ltd `https://www.akeeba.com`
Described version 2.0		

Appendix C. 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-net-

work 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 Foun-

> dation; 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.