User Guide

Backup Pro for Magento 2.x — Extmag_Backup

2026-05-24

1 About the module

- 1.1 What it covers
- 1.2 Restoration

2 Admin navigation

3 Module configuration

4 Storage Sets

- 4.1 Storage types
- 4.2 OAuth flow (Google Drive, OneDrive, Dropbox)
- 4.3 Amazon S3 credentials
- 4.4 Credentials at rest

5 Backup Sets

- 5.1 Creating a Backup Set
- 5.2 Backup pass — what happens
- 5.3 Frequency semantics
- 5.4 Manual run
- 5.5 Storage row tips

6 Backup Logs

- 6.1 Columns
- 6.2 Reading a *failed* Storages JSON row
- 6.3 Retention of log rows

7 Frontend behaviour

8 Restoring a store from a backup

9 Troubleshooting

- 9.1 Backup Log shows `Storages: Storage with id 0 does not exist`
- 9.2 OAuth-tied backup fails after weeks of success
- 9.3 Files archive exceeds 4 GB and one part fails to upload
- 9.4 `var/backups/extmag/` fills the disk
- 9.5 `Frequency = Hourly` but the job fires every 2 hours
- 9.6 Admin menu *Backup Pro* is missing
- 9.7 Notification: *Dropbox / Google Drive / OneDrive system message*

1 About the module

Backup Pro is a scheduled, multi-destination backup engine for Magento 2. It takes the database and the file tree on a per-hour cadence, compresses both, and uploads the archives to one or more cloud destinations — Google Drive, Microsoft OneDrive, Dropbox, Amazon S3 — alongside the local `var/backups/extmag/` directory.

Every run is recorded in an append-only log with overall status, per-storage upload status, the archive file names, the duration in seconds, and any error message. A daily cleanup job prunes log rows older than ten days; the archives themselves are pruned by the per-Set *Copies to Keep* retention knob, separately for local and each cloud destination.

1.1 What it covers

Aspect	What ships
Database backup	Compressed `db.sql.gz` produced by Magento's native backup factory.
Files backup	Compressed `.tgz` archives, auto-split into parts above 4 GB.
Exclude lists	First-level directories under Magento root; any DB table by name.
Schedule	1, 2, 3, 4, 5, 12, 18, 24, 36, 48 hours. Daily / every-2-days take time of day.
Storage	Local + multi-cloud per Set. Each storage row has its own retention.
Retention	Per-Set `keep_copies` for local and each cloud entry.
Audit log	`extmag_backup_log` — every run, overall + per-storage status, errors.
Admin permissions	Granular ACL — config, backups, storages, logs each have their own resource.
Credentials security	OAuth tokens and secrets encrypted with `EncryptorInterface` on save.

1.2 Restoration

Restoration of an archived backup is **not** automated by the module — it is a developer / DevOps task using standard Magento tooling against the `.sql.gz` and `.tgz` files produced by the backup pass. See the *Reference Manual* for the archive layout and the dump format.

2 Admin navigation

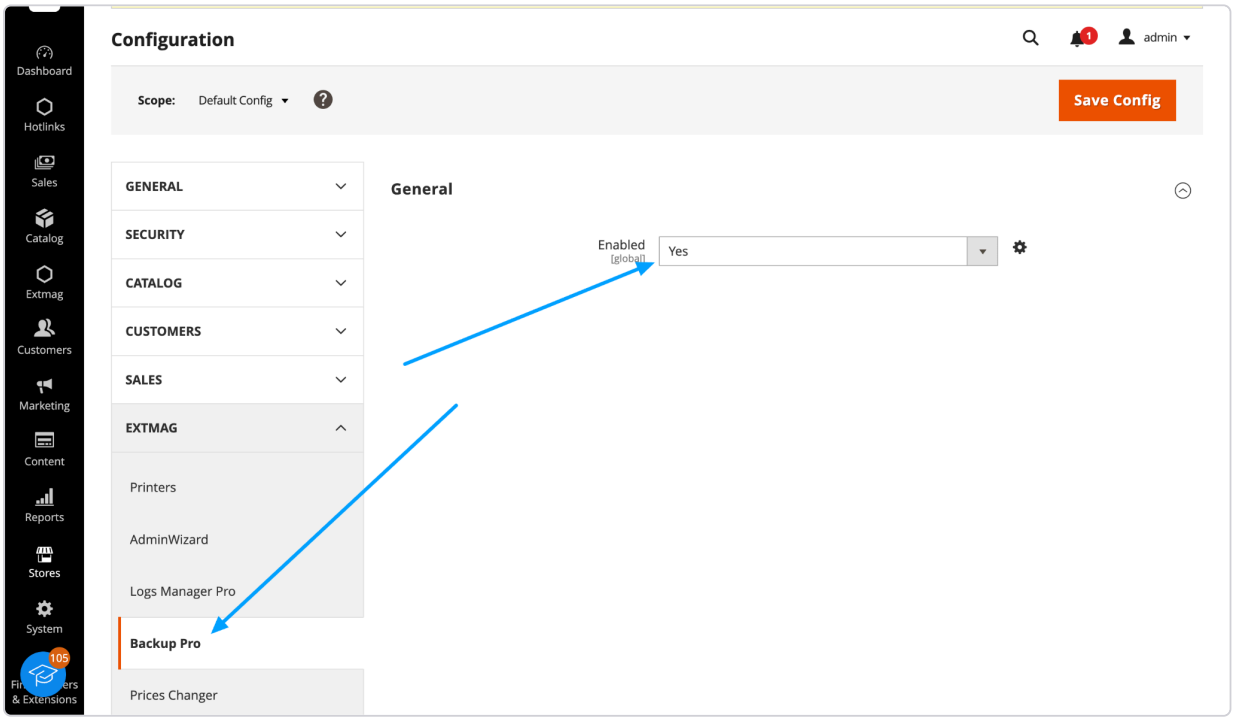
`Extmag → Backup Pro` menu:

Item	Purpose
Backup Sets	Sets grid + create / edit form.
Storage Sets	Cloud / S3 destination credentials.
Backup Logs	Append-only run audit grid.
Configuration	`Stores → Configuration → Extmag → Backup Pro`.

Each menu item has its own ACL resource (`Extmag_BackupCore::backups` , `Extmag_BackupCore::storages` , `Extmag_BackupCore::logs` , `Extmag_BackupCore::config`), so role assignment is granular. The save / delete actions under Backups and Storages are protected by separate nested resources (`backups_save` , `backups_delete` , `storages_save` , `storages_delete`).

3 Module configuration

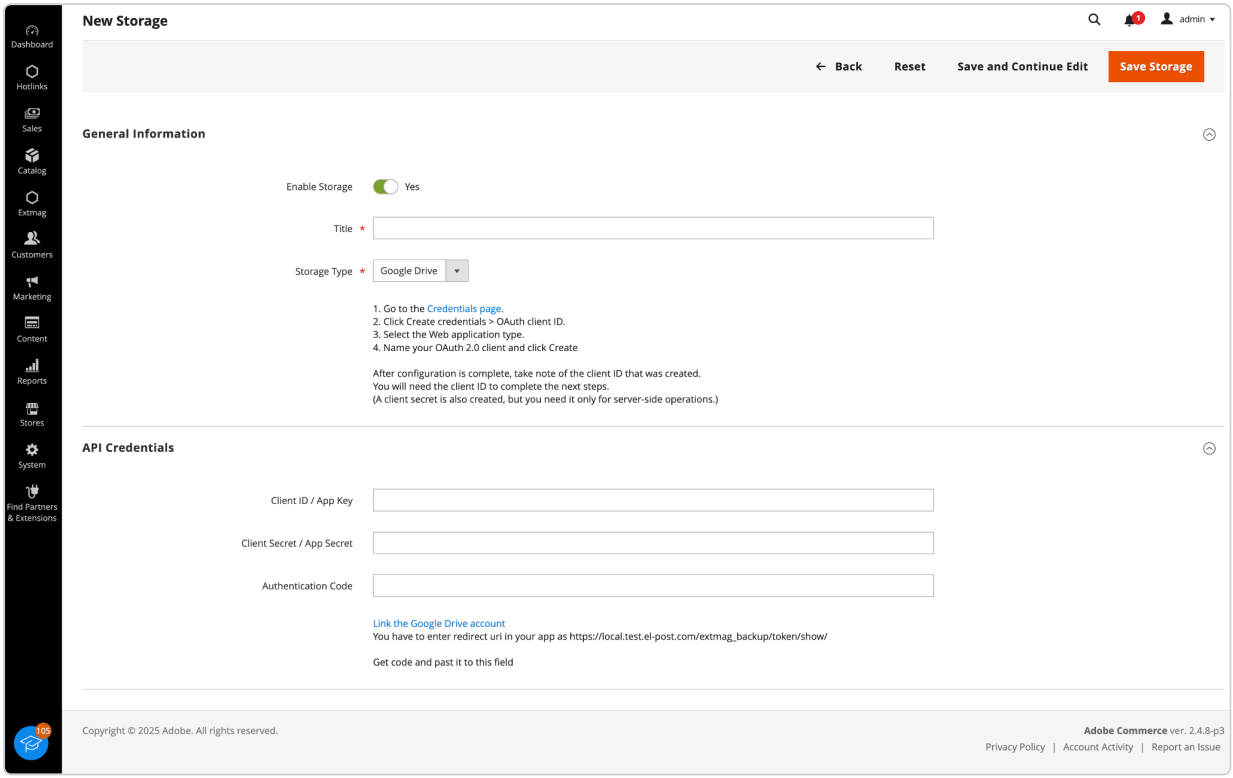
`Stores → Configuration → Extmag → Backup Pro` is the master switch — when *Enabled* is No, the cron driver short-circuits and no Backup Set fires, regardless of the per-Set flag.



System configuration — Extmag → Backup Pro

4 Storage Sets

A Storage Set holds the credentials and OAuth tokens for one cloud destination. Backup Sets reference Storage Sets by `storage_id` on their *Storages Configuration* fieldset, so the same Google Drive account can be shared across many Backup Sets without duplicating credentials.



Storage Set edit form — Google Drive example

4.1 Storage types

Type	Field semantics
Google Drive	OAuth — Client ID, Client Secret, Authentication Code, then refresh token.
Microsoft OneDrive	OAuth — same three fields, same flow.
Dropbox	OAuth — labels switch to *App Key* / *App Secret*.
Amazon S3	Static — AWS Access Key ID, Secret Access Key, region in *Authentication Code*.

The local filesystem destination does **not** require a Storage Set; it is hard-wired and selected by entering `local` in the Backup Set's *Storages Configuration* → *Storage* field.

4.2 OAuth flow (Google Drive, OneDrive, Dropbox)

1. Register the app in the provider's developer console.
2. Set the redirect URI to `https://<your-domain>/extmag_backup/token/show/`.

3. Copy the Client ID (or App Key) and Client Secret (or App Secret) into the Storage Set form.
4. Save and Continue Edit. Click *Link the account*.
5. Complete the OAuth consent in the provider's window.
6. Copy the authorization code into the **Authentication Code** field.
7. Save. The module exchanges the code for a refresh token immediately and stores it encrypted; from this point on the access token is rotated automatically.

When the refresh token is revoked (admin-side revoke, password change, long inactivity), the next backup attempt records the error in *Backup Logs* and posts an admin inbox notification. Re-link the account from the Storage Set edit form to recover.

4.3 Amazon S3 credentials

S3 uses static credentials, not OAuth.

Storage Set field	AWS meaning
Client ID / App Key	Access key id.
Client Secret	Secret access key.
Authentication Code	Region (e.g. `us-east-1`, `eu-central-1`). Defaults to `us-east-1` if empty.

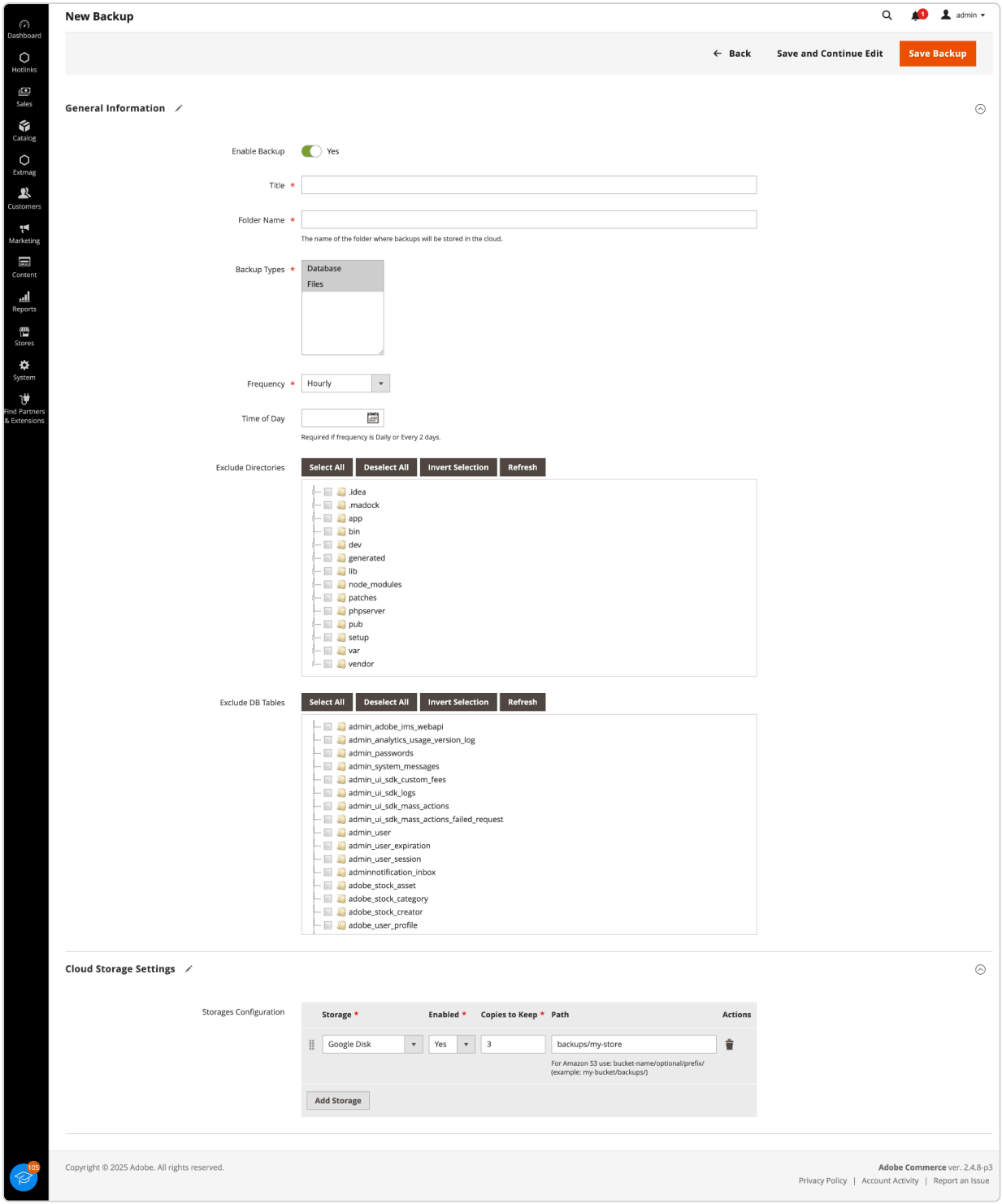
The bucket and path prefix are configured per Backup Set on the *Storages Configuration* → *Path* field — format `bucket-name/optional/prefix/`. The bucket must exist; the module does **not** create it.

4.4 Credentials at rest

`client_secret`, `oauth_refresh_token`, and `oauth_access_token` are encrypted via `Magento\Framework\Encryption\EncryptorInterface` before write and decrypted on load. Restoring a database dump from one environment into another requires the same Magento encryption key, otherwise the credentials cannot be decrypted and the operator must re-link the account.

5 Backup Sets

A Backup Set is a complete backup profile — what to back up, how often, where to send it, and how many copies to keep.



Backup Set edit form

5.1 Creating a Backup Set

`Extmag → Backup Pro → Backup Sets → Create New Backup`.

1. **General Information**
    - **Enable Backup** — master switch for this Set.
    - **Title** — unique. Enforced by `EXTMAG_BACKUP_TITLE_UNIQUE`.

- **Folder Name** — unique. The folder name used under `var/backups/extmag/<folder>/` and as the upload root in cloud storage. Enforced by `EXTMAG_BACKUP_FOLDER_NAME_UNIQUE`.
- **Backup Types** — multiselect: *Database*, *Files*. At least one.
- **Frequency** — 1, 2, 3, 4, 5, 12, 18, 24, 36 or 48 hours.
- **Time of Day** — required only for Frequency = Daily (24) or Every 2 days (48). The driver fires on the first hourly tick whose `H:M >= Time of Day`.
- **Exclude Directories** — first-level directories under the Magento root. Recommended excludes: `var`, `generated`, `node_modules`, `.idea`, `.git`.
- **Exclude DB Tables** — any tables that contain noise. Recommended: `*_log`, search indexes, session and cache tables.

2. **Cloud Storage Settings**
   - One row per destination.
   - **Storage** — `local` or the title of any enabled Storage Set.
   - **Enabled** — per-row toggle.
   - **Copies to Keep** — retention; older folders under that destination are deleted after a successful run.
   - **Path** — for cloud destinations, the folder name (Drive / OneDrive / Dropbox) or `bucket-name/optional/prefix/` (S3). Local destinations ignore this field — local archives always land under `var/backups/extmag/<folder>/<timestamp>/`.
3. Save. The Set is now active; the next cron tick that satisfies the frequency window fires it.

5.2 Backup pass — what happens

1. A log row is created with `overall_status = pending` and `started_at` filled in.
2. If `Backup Types` includes *Files*, the archiver streams the Magento root into one or more `files.tgz` archives under `var/backups/extmag/<folder>/<timestamp>/files/`, honouring *Exclude Directories*. Archives over 4 GB split into multiple parts (`files.tgz.001`, …) automatically.
3. If `Backup Types` includes *Database*, the dumper writes `db.sql.gz` to `.../<timestamp>/db/`, honouring *Exclude DB Tables*.
4. The runner enumerates every file under `<timestamp>/` and records the list in the log row.
5. The local retention sweep deletes any older `<timestamp>/` folder beyond `Copies to Keep` for the `local` row.
6. For each enabled cloud row: upload every archive to the configured Path, then sweep older remote folders beyond that row's `Copies to Keep`.

7. The log row is finalized — `overall_status` is `success` only when every storage row reported success; otherwise `failed`. `finished_at` and `duration_sec` are filled in. `files_json` lists the archive files, `storages_json` records per-row status / message.

5.3 Frequency semantics

Frequency setting	When the next pass fires
1 / 2 / 3 / 4 / 5 / 12 / 18 / 36 hours	First cron tick at or after `last_time + N hours`. Time of Day ignored.
24 (Daily)	First tick of the configured hour whose minute is at or after the configured minute, and at least 24 h have passed.
48 (Every 2 days)	Same as Daily, but the 48 h gap rule applies.

The driver only fires when both gates pass — the elapsed-time gate (`last_time + N hours`) **and** the time-of-day gate where applicable. After a successful or failed run, `last_time` is advanced to `now + frequency hours`, so the next firing window opens at that point.

5.4 Manual run

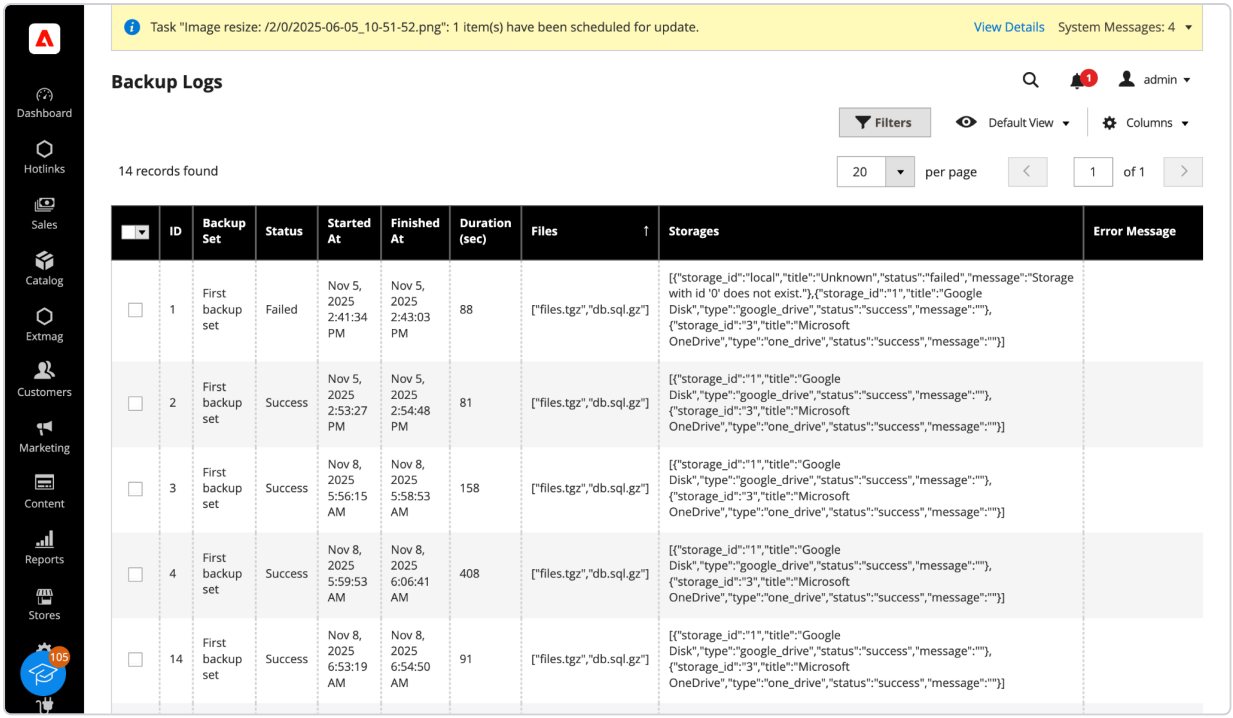
The Backup Set edit page exposes a *Create Backup* button that calls the same service as the cron driver — same log row, same retention sweep. Use it before significant store changes (theme deploys, catalog imports, version upgrades).

5.5 Storage row tips

- Put the local row first. It catches the archive even when every cloud destination fails the upload.
- Set per-cloud `Copies to Keep` independently — Drive can hold 30 daily copies cheaply, S3 can hold 365 with lifecycle policy on Glacier.
- Never set `Copies to Keep = 0`. The retention sweep then deletes the just-uploaded archive on the next pass.
- For S3, the prefix in *Path* is folder-style: keep the trailing slash.

6 Backup Logs

`Extmag → Backup Pro → Backup Logs` is the per-run audit grid.



ID	Backup Set	Status	Started At	Finished At	Duration (sec)	Files	Storages	Error Message
1	First backup set	Failed	Nov 5, 2025 2:41:34 PM	Nov 5, 2025 2:43:03 PM	88	["files.tgz","db.sql.gz"]	[{"storage_id":"local","title":"Unknown","status":"failed","message":"Storage with id '0' does not exist."},{"storage_id":"1","title":"Google Disk","type":"google_drive","status":"success","message":""},{"storage_id":"3","title":"Microsoft OneDrive","type":"one_drive","status":"success","message":""}]	
2	First backup set	Success	Nov 5, 2025 2:53:27 PM	Nov 5, 2025 2:54:48 PM	81	["files.tgz","db.sql.gz"]	[{"storage_id":"1","title":"Google Disk","type":"google_drive","status":"success","message":""},{"storage_id":"3","title":"Microsoft OneDrive","type":"one_drive","status":"success","message":""}]	
3	First backup set	Success	Nov 8, 2025 5:56:15 AM	Nov 8, 2025 5:58:53 AM	158	["files.tgz","db.sql.gz"]	[{"storage_id":"1","title":"Google Disk","type":"google_drive","status":"success","message":""},{"storage_id":"3","title":"Microsoft OneDrive","type":"one_drive","status":"success","message":""}]	
4	First backup set	Success	Nov 8, 2025 5:59:53 AM	Nov 8, 2025 6:06:41 AM	408	["files.tgz","db.sql.gz"]	[{"storage_id":"1","title":"Google Disk","type":"google_drive","status":"success","message":""},{"storage_id":"3","title":"Microsoft OneDrive","type":"one_drive","status":"success","message":""}]	
14	First backup set	Success	Nov 8, 2025 6:53:19 AM	Nov 8, 2025 6:54:50 AM	91	["files.tgz","db.sql.gz"]	[{"storage_id":"1","title":"Google Disk","type":"google_drive","status":"success","message":""},{"storage_id":"3","title":"Microsoft OneDrive","type":"one_drive","status":"success","message":""}]	

Backup Logs grid

6.1 Columns

Column	Notes
ID	`entity_id` — primary key.
Backup Set	Snapshot of the Set title at run time. Stays readable after the Set is deleted (FK is `ON DELETE SET NULL`).
Status	`success`, `failed`, `partial`, `pending`.
Started At	When the runner picked up the Set.
Finished At	When the runner finalised the log row.
Duration (sec)	Wall-clock seconds.
Files	JSON list of archive filenames produced by the pass.
Storages	JSON list of `{storage_id, title, type, status, message}` rows — one per storage row in the Set.
Error Message	Top-level error if the entire pass failed before the storage phase.

6.2 Reading a *failed* Storages JSON row

The *Storages* column carries a JSON array of per-row statuses. A typical mixed-result row looks like:

```
[
  {"storage_id":"0","title":"Unknown","status":"failed","message":"Storage with
        id 0 does not exist."},
  {"storage_id":"1","title":"Google
        Disk","type":"google_drive","status":"success","message":""},
  {"storage_id":"3","title":"Microsoft
        OneDrive","type":"one_drive","status":"success","message":""}
]
```


`overall_status = failed` because at least one storage row did not succeed. The fix in the example above: remove the row with `storage_id = "0"` from the Set (it points at a deleted Storage Set).

6.3 Retention of log rows

The `extmag_backup_log_cleanup` cron job runs daily at 03:00 and deletes rows older than ten days, 500 at a time, to avoid table-level locks. This retention is independent of the *Copies to Keep* knob on Backup Sets — that knob governs the archive folders, not the log rows.

7 Frontend behaviour

The module has no storefront UI. The only frontend route (`extmag_backup/token/show/`) is the OAuth redirect target for cloud provider consent flows and renders the authorization code on screen for the operator to copy into the Storage Set edit form.

8 Restoring a store from a backup

Restoration is a developer / DevOps task. The module produces standard artifacts that any operator can restore with native tooling:

- `db.sql.gz` — gzip-compressed MySQL dump. Restore with `gunzip -c db.sql.gz | mysql -u <user> -p <database>`.
- `files.tgz` (or `files.tgz.001`, …) — gzip-compressed tarball of the Magento file tree, minus the directories in *Exclude Directories*. Restore with `tar -xzf files.tgz -C <magento-root>` (concatenate split parts first: `cat files.tgz.* | tar -xz -C <magento-root>`).

After a restore, run:

```
bin/magento cache:flush
bin/magento setup:upgrade
bin/magento setup:di:compile
bin/magento indexer:reindex
```

Restoration into a different environment may also require updating `app/etc/env.php` for the new database credentials and base URL, and re-encrypting the Storage Set OAuth secrets using the destination environment's encryption key (or re-linking the cloud accounts).

9 Troubleshooting

9.1 Backup Log shows `Storages: Storage with id 0 does not exist`

The Backup Set has a *Storages Configuration* row whose Storage was deleted. Edit the Set, remove the orphan row from the table, save. The next run logs only the remaining rows.

9.2 OAuth-tied backup fails after weeks of success

The provider's refresh token was revoked or expired. An admin inbox notification posts on the next failed pass. Re-link the account from the Storage Set edit form. Existing archives in cloud storage are preserved.

9.3 Files archive exceeds 4 GB and one part fails to upload

Each part is uploaded as a separate file. The failed part is logged in *Storages* JSON. A retry on the next pass uploads a fresh full archive. For very large file trees, exclude `pub/media/catalog/product/cache/`, `pub/static/`, `var/cache/`, `var/log/`, `var/page_cache/`, `var/session/`, `var/view_preprocessed/` from *Exclude Directories* — those are reproducible from source assets.

9.4 `var/backups/extmag/` fills the disk

Lower *Copies to Keep* on the `local` row, or remove the `local` row entirely and rely on cloud destinations. The local row primarily exists as a safety net for runs where every cloud destination fails.

9.5 `Frequency = Hourly` but the job fires every 2 hours

The system cron runs every minute, but the Magento cron group has its own schedule. Check `bin/magento cron:status` and inspect the `default` group in `crontab.xml` (Magento's own per-group throttling can defer ticks). Also confirm the Set's `last_time` is moving forward each pass — a stuck `last_time` indicates the runner is failing before it can persist the update.

9.6 Admin menu *Backup Pro* is missing

- `bin/magento module:status | grep Extmag` — both modules enabled.
- `bin/magento cache:flush` and reload the admin.
- The admin user must hold either `Extmag_BackupCore::backups`, `Extmag_BackupCore::storages`, `Extmag_BackupCore::logs`, or `Extmag_BackupCore::config` — the parent menu item is hidden when no child resource is granted.

9.7 Notification: *Dropbox / Google Drive / OneDrive system message*

The relevant Storage Set has `oauth_error_notify_mess` populated. Open the Storage Set, re-link the account, save. The notification clears on the next successful authenticated call.