

Installation Guide

Backup Pro for Magento 2.x — Extmag_Backup

2026-05-24

- 1 Requirements
- 2 Composer install
- 3 Cron prerequisites
- 4 First-install configuration
- 5 Enabling backup destinations
 - 5.1 Google Drive
 - 5.2 Microsoft OneDrive
 - 5.3 Dropbox
 - 5.4 Amazon S3
- 6 Verification
- 7 Upgrade flow
- 8 Disabling the feature without uninstalling
- 9 Uninstall
- 10 Troubleshooting install-time issues
 - 10.1 Admin menu *Backup Pro* does not appear
 - 10.2 Cron job never fires
 - 10.3 OAuth redirect mismatch
 - 10.4 “Storage with id 0 does not exist”
 - 10.5 Backup archive larger than 4 GB
 - 10.6 “Permission denied” writing to `var/backups`
 - 10.7 OAuth refresh fails after long idle

1 Requirements

- Magento 2.3.x, 2.4.x (Open Source or Adobe Commerce).
- PHP \geq 7.4. PHP 8.1+ recommended for the cloud SDKs shipped as transitive dependencies.
- MySQL 5.7+ / MariaDB 10.4+ / MySQL 8.0+.
- Composer 2.x.
- Magento `default` cron group running every minute. The backup driver fires on the `1 * * * *` schedule; the log cleanup job runs at `0 3 * * *`.
- Outbound HTTPS to the configured cloud provider:
 - Dropbox — `api.dropboxapi.com`, `content.dropboxapi.com`.
 - Google Drive — `*.googleapis.com`, `oauth2.googleapis.com`.
 - Microsoft OneDrive — `login.microsoftonline.com`, `graph.microsoft.com`.
 - Amazon S3 — `s3.<region>.amazonaws.com` for the chosen region.
- Server-side disk space — at least 2x the size of the database dump plus the files archive. The runner writes both to `var/backups/extmag/...` before uploading to cloud storage.

The composer package pulls these libraries transitively:

Library	Purpose
<code>spatie/flysystem-dropbox</code>	Dropbox adapter (\geq 1.2 <4.0).
<code>google/apiclient ^2.8</code>	Google Drive REST client.
<code>aws/aws-sdk-php</code>	Amazon S3 client (pulled by AwsS3 helper).

2 Composer install

```
composer require extmag/backuppro
bin/magento module:enable Extmag_Backup Extmag_BackupCore
bin/magento setup:upgrade
bin/magento setup:di:compile
bin/magento cache:flush
```

The `extmag/backuppro` package is a thin wrapper that pulls `extmag/backup-core` (`Extmag_BackupCore`) as a dependency. Both modules must be enabled — `Extmag_Backup`

contains only registration glue; all logic lives in `Extmag_BackupCore`.

`setup:upgrade` creates the following tables:

Table	Purpose
<code>extmag_backup</code>	Backup Sets (profile + schedule + storages config).
<code>extmag_storage</code>	Storage Sets (cloud provider credentials, OAuth tokens).
<code>extmag_backup_log</code>	Append-only run audit (start / finish, status, files, errors).

After installation, flush the cache so the new admin menu entry shows up under `Extmag` → `Backup Pro`.

3 Cron prerequisites

The backup driver and the log cleanup job are registered as standard `<job>` entries in `crontab.xml` under the `default` cron group:

```
bin/magento cron:install
crontab -l | grep magento
bin/magento cron:status
```

Job	Schedule	Purpose
<code>extmag_backup_create</code>	<code>1 * * * *</code>	Per-hour pass. Fires due Backup Sets.
<code>extmag_backup_log_cleanup</code>	<code>0 3 * * *</code>	Daily prune. Removes logs > 10 days old.

The driver evaluates the per-Set `Frequency` (1–48 hours) and `Time of Day` on every tick. A Set with `Frequency = Hourly` fires on the first tick after its `last_time + 1h`; a Set with `Frequency = Daily` fires on the first tick of the configured hour and minute.

4 First-install configuration

Stores → Configuration → Extmag → Backup Pro.

Setting	Default	When to change
Enabled	Yes	Set to No to pause every Backup Set without deleting definitions.

The cron driver short-circuits on `extmag_backup/general/active = 0` — no Backup Set fires while the master switch is off.

5 Enabling backup destinations

By default, backups land in `var/backups/extmag/<folder>/<timestamp>/` on the local filesystem. To add cloud destinations, configure one or more entries under `Extmag → Backup Pro → Storage Sets` *before* creating Backup Sets that reference them.

5.1 Google Drive

1. Google Cloud Console → APIs & Services → **Credentials**.
2. **Create credentials** → **OAuth client ID**.
3. Application type: **Web application**.
4. Authorised redirect URI: `https://<your-domain>/extmag_backup/token/show/`.
5. Save and copy the **Client ID** and **Client Secret** into the Storage Set form.
6. Click *Link the Google Drive account*, complete the OAuth consent, copy the resulting code into **Authentication Code**, save.

5.2 Microsoft OneDrive

1. Microsoft Entra (Azure AD) → **App registrations** → **New registration**.
2. Redirect URI: `https://<your-domain>/extmag_backup/token/show/`.
3. API permissions → **Files.ReadWrite.All, offline_access**.
4. Certificates & secrets → New client secret.
5. Copy the **Application (client) ID** and the secret value into the Storage Set form; complete the OAuth handshake the same way as Google.

5.3 Dropbox

1. Dropbox App Console → **Create app**.
2. Scoped access, app folder (or full Dropbox, whichever the operator prefers).
3. Redirect URI: `https://<your-domain>/extmag_backup/token/show/`.
4. Copy the **App Key** and **App Secret** into the Storage Set form (labels switch to *App Key / App Secret* automatically when the Dropbox type is selected). Complete the OAuth handshake.

5.4 Amazon S3

S3 uses static credentials, not OAuth.

Field	Maps to
Client ID / App Key	AWS access key id.
Client Secret	AWS secret access key.
Authentication Code	AWS region (e.g. <code>us-east-1</code> , <code>eu-central-1</code>).

The bucket name is configured per Backup Set on the *Cloud Storage Settings* fieldset under the **Path** column — format `bucket-name/optional/prefix/`.

6 Verification

```
bin/magento module:status Extmag_Backup Extmag_BackupCore
```

Expect *Module is enabled* for both modules. Then visit `Extmag → Backup Pro → Backup Sets` — the grid should render even when empty.

`Stores → Configuration → Extmag → Backup Pro` should show the *Enabled* toggle. If the section is missing, run `bin/magento cache:flush` and reload the admin.

After saving the first Backup Set with at least one storage configured, you can either wait for the next cron tick or trigger a manual run from the Backup Set edit page and watch `Backup Logs` for the result row.

7 Upgrade flow

```
composer update extmag/backuppro extmag/backup-core
bin/magento setup:upgrade
bin/magento setup:di:compile
bin/magento cache:flush
```

Schema migrations are declarative — no data loss on minor / patch versions. OAuth credentials in `extmag_storage` stay encrypted across upgrades; the encryption uses Magento's `EncryptorInterface`, so the encryption key must match the source environment when restoring a database dump that contains Storage Sets.

8 Disabling the feature without uninstalling

- `extmag_backup/general/active = 0` — turns the cron driver into a no-op. Saved Backup Sets and Storage Sets stay in the database.
- Per Set: uncheck *Enable Backup* on the Set edit form.
- Per Storage: uncheck *Enable Storage* on the Storage edit form.
- Disable the module: `bin/magento module:disable Extmag_Backup` (and `Extmag_BackupCore` if the operator wants the menu and cron jobs gone).

9 Uninstall

```
bin/magento module:disable Extmag_Backup Extmag_BackupCore
bin/magento setup:upgrade
composer remove extmag/backuppro
```

The schema is **not** dropped automatically. To remove the audit and configuration tables, back up first, then run:

```
DROP TABLE IF EXISTS extmag_backup_log;
DROP TABLE IF EXISTS extmag_backup;
DROP TABLE IF EXISTS extmag_storage;
```

Local backup archives under `var/backups/extmag/` are not removed by the uninstall flow. Delete that directory manually after confirming the archives have been transferred to a separate location.

10 Troubleshooting install-time issues

10.1 Admin menu *Backup Pro* does not appear

- Confirm both modules are enabled: `bin/magento module:status | grep Extmag .`
- `bin/magento cache:flush` and reload the admin.
- Confirm the admin user has the `Extmag_BackupCore::config` ACL resource assigned (System → Permissions → User Roles).

10.2 Cron job never fires

- `bin/magento cron:status` — confirms cron is wired in.
- `crontab -l | grep magento` — confirms the OS-level entry is in place.
- Stores → Configuration → Extmag → Backup Pro → Enabled must be Yes.
- Each Backup Set has its own *Enable Backup* flag; cron skips disabled Sets.
- The driver runs on the `default` cron group only — operators who isolated the group must adjust their `crontab` accordingly.

10.3 OAuth redirect mismatch

The redirect URI registered in the cloud provider’s console must match `https://<your-domain>/extmag_backup/token/show/` **exactly**, including the trailing slash. The `extmag_backup` route lives in `Extmag_BackupCore/etc/frontend/routes.xml` and is served by `Extmag\BackupCore\Controller\Token\Show`.

10.4 “Storage with id 0 does not exist”

The Backup Set references a Storage Set that has been deleted, or the `storage_id` was set to `0` (no storage chosen) on the *Storages Configuration* row. Edit the Set, remove the orphan row from the *Storages Configuration* table, save.

10.5 Backup archive larger than 4 GB

The file backup pass auto-splits archives over 4 GB into multiple `.tgz` parts. No operator action is required; the log row lists every part under *Files*.

10.6 “Permission denied” writing to `var/backups`

The Magento application user must own (or have write access to) `var/backups/`. After install:

```
mkdir -p var/backups/extmag
chown -R <web-user>:<web-group> var/backups
chmod -R 775 var/backups
```

10.7 OAuth refresh fails after long idle

The Storage Set stores a refresh token; the access token is rotated on expiry. When the provider revokes the refresh token (admin-side revoke, long inactivity, password change), the next backup attempt logs an OAuth error and posts an admin inbox system notification (see `Notification/System/Message/*`). Re-link the account from the Storage Set edit form to recover.