

PrestaMailchimp

PrestaShop ↔ Mailchimp synchronization

PrestaShop 1.6 to 9.x | PHP 5.6 to 8.3+

Author : Mediacom87

Contents

1. Overview and use cases

2. Compatibility and requirements

3. Installation and first run

4. Synchronization architecture

5. The “API Key” tab

6. The “Exclusion” tab

7. The “List” tab — audience, merge fields, segmentation

8. The “Products” tab — Mailchimp e-commerce catalog

9. The “Orders” tab

10. Initial synchronization (bulk import)

11. Real-time synchronization

12. Abandoned cart and recovery

13. Double opt-in and newsletter signup

14. Inbound Mailchimp webhooks

15. Multilang and multishop

16. Logging and diagnostics

17. Security

18. Performance and volume

19. License, updates and domain migration

20. FAQ

21. Troubleshooting

1. Overview and use cases

PrestaMailchimp connects your PrestaShop store to Mailchimp and pushes all e-commerce data: **customers, catalog products, past orders, carts and cart rules (promo codes)**. The goal is not just to send newsletter emails but to feed Mailchimp's marketing engine.

What it enables

- **Mailchimp automations** (welcome, re-engagement, post-purchase) fed with real customer and order data.
- **Abandoned cart**: PrestaShop carts are sent to trigger Mailchimp recovery emails.
- **Product retargeting and recommendations** via the catalog in Mailchimp e-commerce.
- **Revenue tracking**: synced orders feed Mailchimp revenue reports.
- **Fine segmentation**: merge fields, interests from customer groups, personal vouchers.

Two modes

1. **Initial sync**: a bulk import, launched from the back office, sending existing data.
2. **Real-time sync**: afterwards every change is mirrored to Mailchimp through PrestaShop hooks, no cron.

2. Compatibility and requirements

Item	Supported
PrestaShop	1.6 to 9.x (incl. thirty bees)
PHP	5.6 to 8.3+
Multishop	Yes — one Mailchimp audience per shop
Multilang	Yes — one audience per language, UI in 17 languages
Native PrestaShop tables	None altered. One dedicated table: <code>ps_med_pmc_sync_job</code>
Mailchimp account	Required — API key
HTTPS	Recommended (required for inbound Mailchimp webhooks)

3. Installation and first run

1. Back office > **Modules** > Upload a module > select `prestamailchimp.zip`.
2. Click **Install** then **Configure**.
3. An internal security key is generated on first open (protects the sync AJAX calls).
4. Upgrading from an older version applies database migrations automatically.

On PrestaShop 1.6, install via the back office (admin session required for tab permissions).

4. Synchronization architecture

The initial sync uses a **resumable engine**: work is split into chunks, each chunk builds a Mailchimp batch and submits it; progress (cursor, batches, errors) is persisted in `ps_med_pmc_sync_job`. A progress window opens on the configuration page; closing the tab, a timeout or a network drop does not break it — the import **resumes where it stopped**. Mailchimp processes batches asynchronously; the module then polls their status and shows the result (and the Mailchimp error-report URL if any). Four types share the same engine: customers, products, orders, vouchers.

5. The “API Key” tab

Paste your **Mailchimp API key**; the datacenter is detected automatically. This tab also shows the module **security key** used to sign sync calls and the webhook URL — keep it private.

6. The “Exclusion” tab

Defines **excluded customer groups**. A customer in an excluded group is not sent to Mailchimp. With the **“subscribers only”** option, only newsletter subscribers are synced and an excluded customer already in Mailchimp is removed for consistency.

7. The “List” tab — audience, merge fields, segmentation

For each language pick the target Mailchimp **list/audience** (or let the module create it).

Merge fields

Standard fields are populated: `FNAME`, `LNAME`, `OPTIN`, `BIRTHDAY`, `ADDRESS`, `PHONE`, `VOUCHER` (the customer's personal voucher). Required radio/dropdown fields are supported with a configurable default.

Interests = customer groups

PrestaShop **customer groups** are projected as Mailchimp **interests** (category “PrestaShop Customer groups”). Each contact is tagged by group, enabling direct marketing segmentation in Mailchimp.

8. The “Products” tab — Mailchimp e-commerce catalog

Associate the Mailchimp **store** first. Settings: image format, description type, tax handling (incl./excl.), vendor source (category, manufacturer or supplier — used for Mailchimp “Vendor” segmentation). All products and their variants are sent with id, title, URL, price, stock, image, type and vendor — the base for Mailchimp product blocks and recommendations.

9. The “Orders” tab

Pick the **order states** to sync (logable states checked by default), the **period** (all / this year / this month / since last sync), the **product control** (verify each line on Mailchimp before sending — stricter, slower), and the order-state → Mailchimp **financial status** mapping. Each order sends the customer, currency, totals, taxes, shipping, addresses and product lines — usable for post-purchase automations and revenue reports.

10. Initial synchronization (bulk import)

Launch a type's sync from its button on the configuration page. A **modal window** opens: live progress (items generated, then validated by Mailchimp); **automatic resume** from the exact cursor after a tab close or drop; **large volumes** processed in adaptive chunks without loss; on **Mailchimp errors** the window stays open and shows the count and the **report file link** (also kept in Logs). Recommended order: customers → products → orders → vouchers.

11. Real-time synchronization

After the initial sync, the module keeps Mailchimp up to date automatically through PrestaShop hooks (no cron): customer add/update/delete, customer group changes (interests), product add/update/delete, order validate/update (with financial status), cart save (abandoned cart), cart rule add/update/delete (promo codes).

12. Abandoned cart and recovery

When cart sync is enabled, PrestaShop carts are sent to Mailchimp to feed its **abandoned cart** automations. An optional setting can automatically log the customer back in when they return via the Mailchimp recovery link.

13. Double opt-in and newsletter signup

Double opt-in: if enabled, Mailchimp sends a confirmation email before subscribing the contact; the module aligns this with PrestaShop's native email verification to avoid double confirmations. The **Signup Forms** tab integrates newsletter signup; on PrestaShop 1.7+ a signup field can be added to the account creation form. A **blocknewsletter** override keeps subscription consistent.

14. Inbound Mailchimp webhooks

The module exposes a receiver controller that Mailchimp calls on audience events: **subscribe** (contact subscribes on Mailchimp → PrestaShop customer marked subscribed), **unsubscribe** (mirrored), **cleaned** (invalid address handled). The webhook is registered automatically on Mailchimp at configuration. HTTPS required.

15. Multilang and multishop

Configuration is stored per language and shop: audience, last-sync dates, merge fields, interests and order states are independent per language/shop combination. No native PrestaShop table is altered.

16. Logging and diagnostics

Logs tab, three modes: log everything (testing), errors only (production), disabled. The log lists every action and error; on Mailchimp failures it stores the report URL for traceability.

From each sync tab (customers, products, orders, vouchers), an « **Analyze errors** » button queries the latest Mailchimp error report and identifies, for each error cause, the corrective action and affected items — without downloading or interpreting any raw file. A button purges all logs.

17. Security

- Sync calls signed with the module's internal security key.
- Inbound webhooks token-protected (constant-time compare, HTTPS, anti-replay).
- SQL escaped, PrestaShop validation, no native table altered.
- AJAX responses are strictly JSON: a server error is surfaced cleanly (class, file, line) instead of a silent loop.

18. Performance and volume

The engine handles very large catalogs and histories without exhausting memory thanks to **adaptive chunks** sized on the server's real execution time. Chunk size auto-adjusts; on a network error the module retries with backoff then shows a clear state instead of failing silently.

19. License, updates and domain migration

Mediacom87 license registration is **free** for every customer. Sync works without a license; registration unlocks priority support, in-back-office update notifications, major-version alerts and security advisories. License tab: enter email and order reference (Addons or Prestatoolbox), or your key. On a domain change use the **migration request** — the license follows your shop after validation.

20. FAQ

Does sync stop if I close the tab?

No, it resumes at the exact cursor.

Is a cron required?

No.

Are my PrestaShop tables modified?

No, one dedicated table is added.

Multiple languages/shops?

Yes, one audience per language and shop.

Excluded-group customer?

Not synced; removed from the audience in “subscribers only” mode.

Are product variants sent?

Yes, each variant is a Mailchimp variant.

21. Troubleshooting

- **Server error in the window:** the exact message (class, file, line) is shown — send it to support.
- **Mailchimp operations in error:** use the « **Analyze errors** » button on the relevant tab — direct diagnosis, no raw file needed; the report URL is also kept in Logs.
- **Nothing syncs in real time:** ensure the initial sync of that type was done and API/audience are set for the language.
- **Webhook inactive:** ensure the shop is HTTPS and publicly reachable.
- Enable **Logs > all actions** while diagnosing.

22. Glossary

Term	Definition
Audience / List	A set of Mailchimp contacts (one per language/shop)
Merge field	Mailchimp personalization variable (FNAME, ...)
Interest	Segmentation tag — here PrestaShop customer groups
E-commerce store	Mailchimp store receiving products/orders/carts
Batch	A set of operations processed asynchronously by Mailchimp
Webhook	Inbound HTTP notification from Mailchimp to the shop