Import and Export

Sublogin Documentation

Guide for Magento administrators, technical project owners, and implementation teams.

LAST UPDATED
2026-09-26

SOURCE
Open this article online

EXTENSION
Magento 2 Extension Sublogin

Import and Export

`mageb2b/sublogin-importexport` adds Magento's `sublogin` import and export entity, an Admin grid export action and a command-line CSV importer. Use it for reviewed bulk changes, not as an unmonitored identity feed.

Install

```
composer require mageb2b/sublogin-importexport
php bin/magento module:enable MageB2B_SubloginImportExport
php bin/magento setup:upgrade
```

Start from an export

Open **Sublogin > All Sublogins**, select a small set of representative records and use the grid's export action. The export includes up to three sublogin-owned address groups, using columns such as `address_1_*` through `address_3_*`.

Passwords are not exported. Treat the file as sensitive because it still contains names, email addresses, company relationships and address data.

Use a copy of this export as the template for your first staging import. Do not build a production CSV from the package's fixture data.

Required import data

For create and update behavior, the importer requires:

- `entity_id`, the parent Magento customer ID;
- `email`;
- `firstname` and `lastname`;
- `active`;
- `store_id`;
- `password` when automatic password generation is disabled.

Include `website_id` whenever accounts span multiple websites. Role and group IDs are supported when Role is installed. `order_needs_approval` and `order_approval_amount_check` are supported when Order Approval is installed.

Address columns support three numbered address groups. Verify country, region and default billing or shipping fields with a small import before adding hundreds of records.

Password behavior

Stores > Configuration > MageB2B > Sublogin > Import/Export settings > Generate Missing Import Passwords is enabled by default. A row without `password`

receives a generated password.

Generated credentials are not a substitute for an account-setup process. Decide how the buyer will securely establish access before importing active records.

Run the CLI importer

The command is:

```
php bin/magento sublogin:import-sublogin
var/import/sublogins.csv \
  --behavior=add_update \
  --delete_file_after_import=0
```

Supported behaviors are `append`, `add_update`, `replace` and `delete`.

The separator options are:

```
--field_separator=,
--field_multiple_value_separator=,
--fields_enclosure=0
```

Important file-deletion default

`--delete_file_after_import` defaults to `1`. After a successful import, the command deletes the source CSV. Pass `--delete_file_after_import=0` while validating a file or when your workflow archives the source elsewhere.

Keep import files outside public web directories and remove them according to your data-retention policy.

Understand the behaviors

- **append / add_update** saves valid rows and updates matching records according to Magento's import behavior.
- **replace** can replace matching entity data, including address relationships. Test it with a backup.
- **delete** removes matching sublogin records. Email is required.

When the same email can match more than one scoped record, a delete row must also provide `store_id`, `website_id` or `entity_id`. This prevents a broad email-only delete across scopes.

Verify the result

1. Read the command output and stop on validation errors.

2. Open **Sublogin > All Sublogins** and compare the imported parent, store and website.
3. Check active state, role, group, addresses and method restrictions.
4. Use one imported buyer to sign in and complete checkout.
5. Confirm password setup or reset works without exposing the generated password.

For recurring external synchronization, consider the authenticated Sublogin Web API instead of repeatedly moving CSV files.