Installation

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

Installation

Install and test the base package before adding roles, budgets or approval workflows. Sublogin adds account, quote, wishlist and order data, so take a database backup and run the first installation in staging.

Before you begin

You need:

- PHP 8.1 or newer within your Magento release's supported range;
- Magento 2.4;
- the Magento Integration module;
- Composer 2;
- repository access for `mageb2b/sublogin`;
- a valid SoftwareSilo license;
- maintenance access to run Magento setup and deployment commands.

Check the package version Composer can resolve before changing the store:

```
composer show mageb2b/sublogin --all
```

Install the base package

From the Magento project root:

```
composer require mageb2b/sublogin
php bin/magento module:enable MageB2B_ExtensionManager
MageB2B_Sublogin
php bin/magento setup:upgrade
```

For a production-mode store, rebuild generated code and static assets using your normal deployment process:

```
php bin/magento setup:di:compile
php bin/magento setup:static-content:deploy
```

Refresh only the caches your deployment has invalidated. A blanket cache flush is not required for every installation.

Confirm the installation

Run:

```
php bin/magento module:status MageB2B_Sublogin
composer show mageb2b/sublogin
```

Then sign in to Magento Admin and open **Stores > Configuration > MageB2B > Sublogin**. Confirm that the module information is visible and **Enabled** is set for the intended website or store view.

The base package adds the main Sublogin tables and associates the current sublogin ID with quotes, orders, the sales order grid, wishlists and integration tokens. `setup:upgrade` must complete before anyone signs in as a sublogin.

Install optional packages only when needed

Each add-on is a separate Composer package and license entitlement. Install the base package first.

Workflow	Composer package	Magento module
Roles, permissions and groups	`mageb2b/sublogin-role`	`MageB2B_SubloginRole`
Spending limits and Pay on Budget	`mageb2b/sublogin-budget`	`MageB2B_SubloginBudget`
Allowance for the next order	`mageb2b/sublogin-budget-one-time`	`MageB2B_SubloginBudgetOneTime`
Order approval	`mageb2b/sublogin-orderapproval`	`MageB2B_SubloginOrderApproval`
Sublogin-specific catalogs	`mageb2b/sublogin-custom-catalog`	`MageB2B_SubloginCustomCatalog`
REST and SOAP endpoints	`mageb2b/sublogin-api`	`MageB2B_SubloginApi`
CSV import and export	`mageb2b/sublogin-importexport`	`MageB2B_SubloginImportExport`
Customer and sublogin conversion	`mageb2b/sublogin-transform`	`MageB2B_SubloginTransform`
Disposable development records	`mageb2b/sublogin-sample-data`	`MageB2B_SubloginSampleData`

Reports are included in `mageb2b/sublogin`. Do not install the retired `mageb2b/sublogin-report` package.

Install an add-on with the same sequence used for the base package. For example:

```
composer require mageb2b/sublogin-role
php bin/magento module:enable MageB2B_ExtensionManager
```

```
MageB2B_SubloginRole
php bin/magento setup:upgrade
```

After installation, open the add-on's guide and configure it before exposing its screens to customers.

Hyvä packages

Hyvä compatibility is split by feature. Install the bridge that matches each enabled add-on:

Storefront feature	Package
Base Sublogin screens	`mageb2b/sublogin-hyva`
Roles and groups	`mageb2b/sublogin-role-hyva`
Budgets	`mageb2b/sublogin-budget-hyva`
Sublogin Custom Catalog	`mageb2b/sublogin-custom-catalog-hyva`
Order Approval	`mageb2b/sublogin-orderapproval-hyva`

These packages require the Hyvä default theme and the corresponding parent Sublogin packages. Let Composer resolve the complete dependency set rather than copying one bridge package between stores.

See Hyvä Compatibility Packages for the exact package relationships.

First verification

Use a clearly fictional staging customer, for example `northstar-buyer@example.com`.

1. Create one active sublogin from the main customer account.
2. Sign out and sign in with the new credentials.
3. Add a product to the cart and complete checkout with an allowed address, payment method and shipping method.
4. Confirm the order appears under the correct sublogin and follows the configured order-visibility rule.
5. If impersonation is enabled, return to the main account, sign in as the sublogin and confirm the context banner is visible.
6. Test account setup or password reset without placing a password in an email or log.

Do not import a full company structure until this path works.

Updating

Review the package changelog and take a backup before updating:

```
composer update mageb2b/sublogin --with-dependencies
php bin/magento setup:upgrade
```

When add-ons are installed, include them in the update plan and verify that Composer selects compatible versions. Recompile and deploy static content when required by your Magento mode and release process.

Retest login, address selection, checkout, order visibility and every installed add-on after the update.

Removing the module

Disabling code is different from deleting business data. Start with:

```
php bin/magento module:disable MageB2B_Sublogin
```

Do not run `module:uninstall` or manually delete Sublogin tables on a live store without a reviewed data-retention plan. Sublogin IDs are connected to customer, quote, wishlist and sales records. Preserve the database backup and decide how historic orders should be presented before removing data.

Next steps

- Configure Sublogin
- Create the First Sublogin
- Common Installation and Login Issues