API Add-On

Sales Staff Documentation

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

LAST UPDATED
2026-10-13

SOURCE
Open this article online

EXTENSION
Magento 2 Extension Sales Staff

API Add-On

The Staff API add-on (`MageB2B_StaffApi`) exposes the Staff service contracts through Magento Web API. Use it to maintain representatives and customer assignments from an ERP or CRM, or to build a Staff-authenticated client.

Installation

```
composer config bearer.repo.softwaresilo.io <token>
composer config repositories.softwaresilo composer
https://repo.softwaresilo.io/
composer require mageb2b/staff-api:*
php bin/magento module:enable MageB2B_StaffApi
php bin/magento setup:upgrade
php bin/magento cache:flush
```

Where to find the endpoints

- Magento Swagger UI: `https://<your-domain>/swagger`
- REST base URL: `https://<your-domain>/rest`

Available REST Endpoints

The add-on registers the following endpoints:

- `GET /V1/staff/:staffId`: get a Staff user by ID
- `GET /V1/staff/search`: search or list Staff users with Magento `searchCriteria`
- `POST /V1/staff`: create a Staff user
- `PUT /V1/staff/:staffId`: update a Staff user
- `DELETE /V1/staff/:staffId`: delete a Staff user
- `POST /V1/staffcustomers`: assign customers to Staff
- `POST /V1/unassign-staffcustomers`: unassign customers from Staff
- `POST /V1/staff/token`: create a token for Staff context
- `GET /V1/staff/customers`: list customers for the current Staff context
- `GET /V1/staff/:staffId/customers`: list customers for a specific Staff ID
- `POST /V1/staff/customers/:customerId/token`: create a customer impersonation token

Authentication and permissions

Staff CRUD/assignment routes require admin/integration permissions for Staff management:

- Read permissions for staff data
- Write permissions for staff and assignments
- Delete permissions for staff cleanup

In practice you usually call these endpoints with an **Admin Token** or an **Integration Token** that has the matching permissions.

The token and customer-context routes are declared `anonymous` in `webapi.xml` so a Staff bearer token can reach the service. They are not unrestricted: the service reads the bearer token, checks the Staff identity, limits a representative to their own account and assigned customers, and enforces the Staff website boundary. Admin and integration contexts may use the explicitly supported management paths.

Treat customer tokens as credentials. Send them only over HTTPS, keep them out of logs and URLs, and never expose a long-lived admin token in a browser application.

Examples (REST)

Get a staff user by ID

```
curl -sS -H "Authorization: Bearer <ADMIN_TOKEN>" \
  "https://<your-domain>/rest/V1/staff/123"
```

Search staff users

```
curl -sS -H "Authorization: Bearer <ADMIN_TOKEN>" \
"https://<your-domain>/rest/V1/staff/search?searchCriteria[pageSize]=20"
```

Create a staff user

```
curl -sS -H "Authorization: Bearer <ADMIN_TOKEN>" \
  -H "Content-Type: application/json" \
  -X POST "https://<your-domain>/rest/V1/staff" \
  -d '{
    "staff": {
      "firstname": "Elena",
      "lastname": "Fischer",
      "email": "elena.fischer@northstar-industrial.example",
      "phone": "+49 30 5550 1840",
      "status": 1,
      "website_id": 1
    }
  }'
```

Staff token flow

`POST /V1/staff/token` accepts the Staff email and password, applies Magento's credential validation and request throttling, rejects inactive accounts, and returns a Staff access token. Use that token as `Authorization: Bearer <STAFF_TOKEN>` for the customer-context endpoints.

`POST /V1/staff/customers/:customerId/token` creates a Magento customer token only when the Staff account may access that customer. A representative with **Access all customers** is still restricted to the Staff account's website.

Extension attributes on core entities

The base extension registers extension attributes that are returned by Magento's own API responses:

- `GET /V1/orders/...` order items carry `sales_staff_id`, `sales_staff_group_id` and `sales_staff_commission` in `extension_attributes`.
- `GET /V1/customers/...` records carry `primary_staff_id` in `extension_attributes`.

`primary_staff_id` is a protected customer attribute: it can be read and set with admin-level credentials but is rejected in customer-side `custom_attributes` writes — see Protected customer attributes.

Notes

- Exact payload fields depend on your installed Staff add-ons (e.g. order editing, quotes).
- For bulk operations, consider the Import/Export Add-On.