

# SQL API

## *Instruction manual and examples*

### FURNISHINGS

---

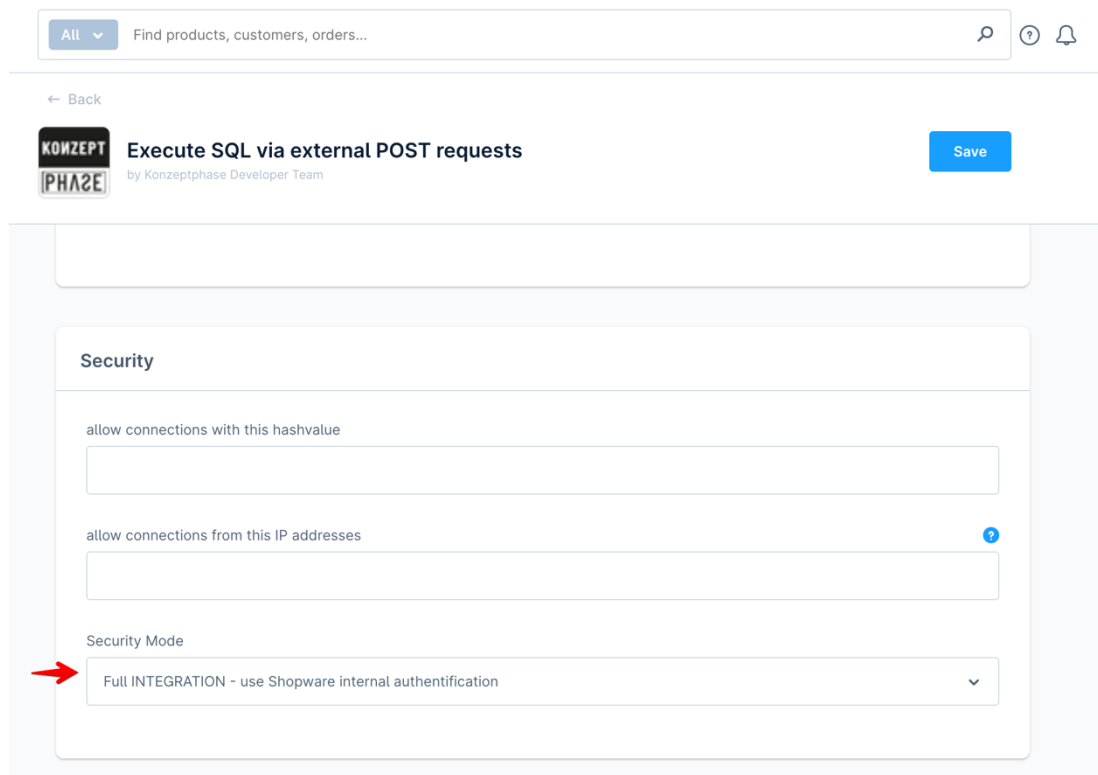
After the plugin is installed and activated, you can configure the security settings. Four different mechanisms are available to protect the API from unauthorized access.

1. Only IP addresses. This means that only requests that come from the defined IP address can be directed to the API. Multiple IP addresses can be entered using | to separate.
2. HASH only method. A hash is a secret key. Requests to the API are processed only with this secret key/word.
3. Combination of points 1 and 2.
4. Full integration via Shopware OAUTH token. A "system user" is created and an access token is generated, which is always valid for 10 minutes. You can also use this token to call other Shopware API methods.

# CONFIGURATION - FULL INTEGRATION

Let's start with the full integration - this is the most complex method, but you probably already use other Shopware API methods.

1. Jump to the plugin configuration and select the “Full Integration” security mode.

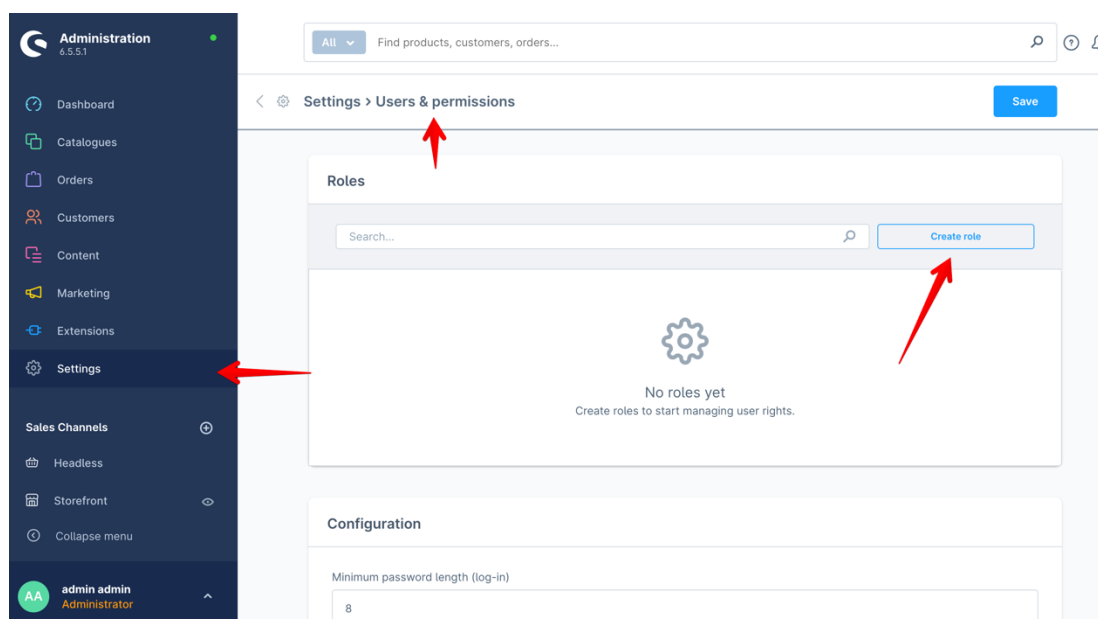


The screenshot shows the configuration page for the 'Execute SQL via external POST requests' plugin. At the top, there is a search bar with the text 'Find products, customers, orders...' and a 'Back' button. Below the search bar, the plugin name 'Execute SQL via external POST requests' is displayed, along with a 'Save' button. The main configuration area is titled 'Security' and contains three sections: 'allow connections with this hashvalue' with an empty text input field, 'allow connections from this IP addresses' with an empty text input field and a help icon, and 'Security Mode' with a dropdown menu. A red arrow points to the 'Full INTEGRATION - use Shopware internal authentication' option in the 'Security Mode' dropdown.

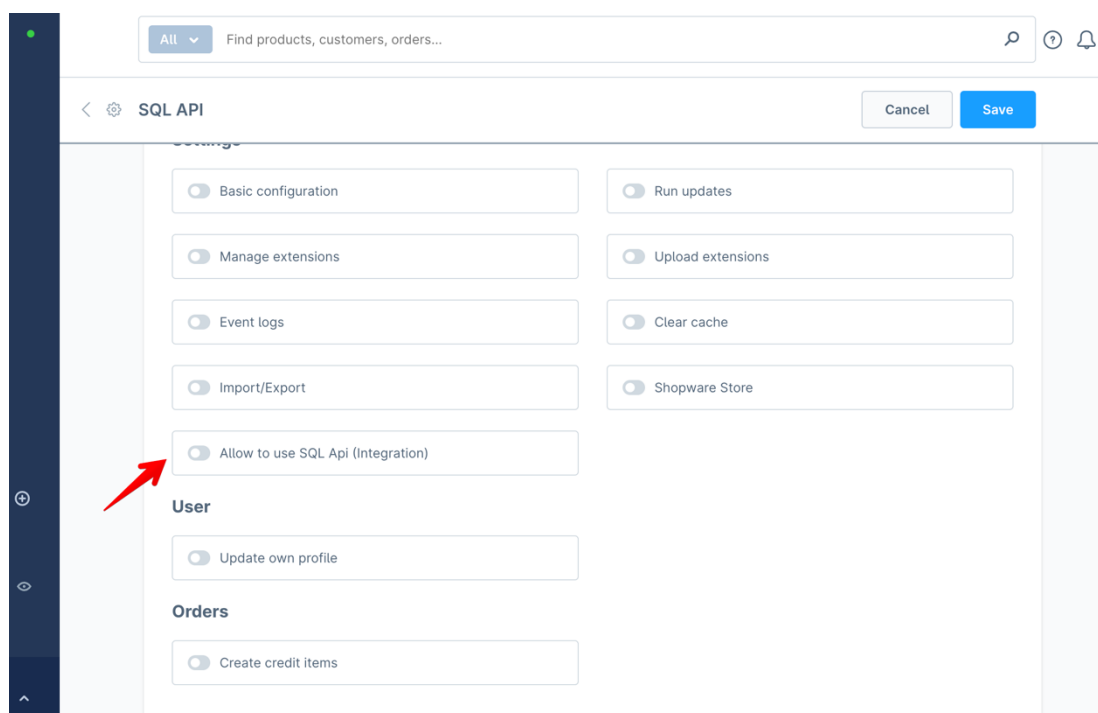
[If you prefer to use the simpler “Hash” method, select “Hash only required” and go to the “HASH METHOD CONFIGURATION” section.](#)

2. After saving, you must create a system role. This controls access to the SQL API.

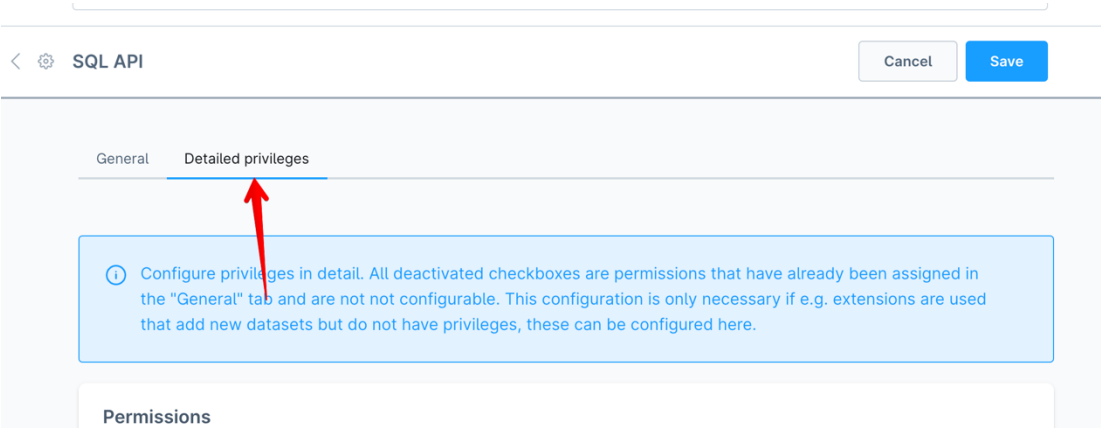
You can find this under “Settings” -> “System” -> “Users and permissions”. Scroll down a little and click on “Create role”.



3. Give the new role a name such as “Access to SQL API”. Also allow access to the SQL API at the very bottom of the page.



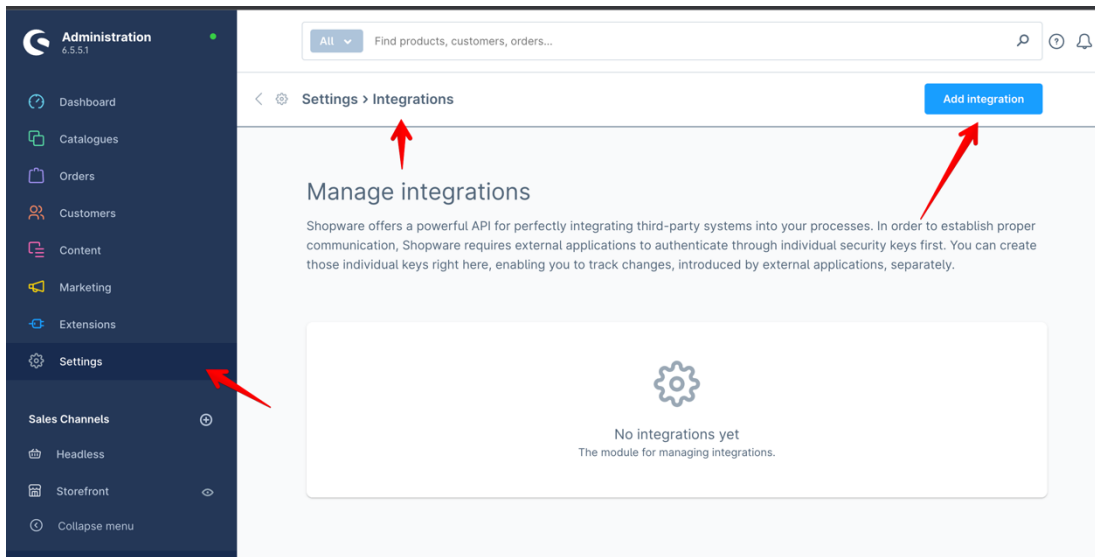
4. In order for the API to be able to write log entries, this must also be permitted in the role. To do this, click on the “Detailed Privileges” tab at the top of the role.



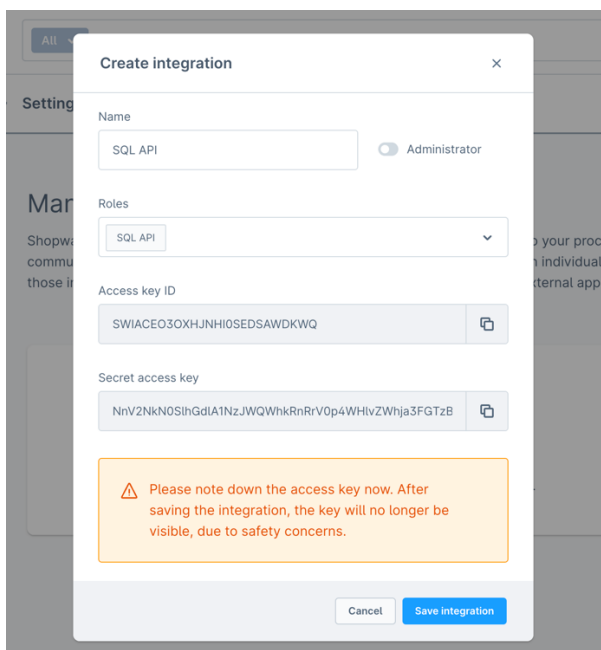
Here please search for `kzph_sqlapi_log` and tick all 4 boxes. Then save and the roll is ready.

import_export_profile_translation	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
integration	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
integration_role	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
kzph_sqlapi_log	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
landing_page	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
landing_page_sales_channel	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
landing_page_tag	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

5. Let's start with the "System User". This can be found under "Settings" -> "System" -> "Integrations". Users for the entire Shopware API can only be created here.



When you click on "Create integration", a window should open. Enter a name of your choice here and select the role. Important: Store the Access Key and Secret Key in a safe place. This will no longer be displayed to you. Save and you're done.



# CONFIGURATION OF THE HASH METHOD

---

In the last few sections we showed you how to prepare the full Shopware API in order to then use the SQL API.

Sometimes you want to keep the effort to a minimum or the other “side” doesn’t have the ability to retrieve OAUTH tokens.

To do this, define and save a hash value (secret word) in the plugin configuration.

That's it.

---

All ▾

Find products, customers, orders...

🔍

?

🔔

---

← Back

**KONZEPT**  
**PHASE**

**Execute SQL via external POST requests**  
by Konzeptphase Developer Team

Save

Security

allow connections with this hashvalue

→

WeLoveShopware!

allow connections from this IP addresses

?

Security Mode

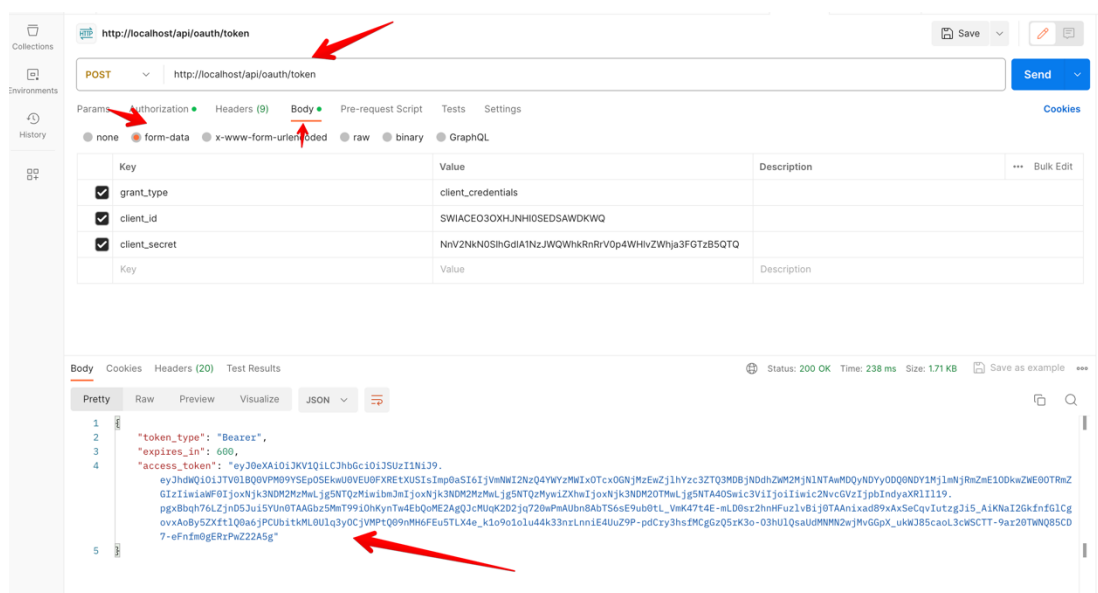
→

Only HASH required ▾

# TEST FUNCTION WITH POSTMAN

To test the API, we recommend the free “Postman” tool. This is available for all platforms and enables quick and easy testing.

Let's start by retrieving the system token. As already mentioned, this is important in order to access the Shopware internal API. **If you only use the internal hash or IP address method, you can skip to “ISSUE SQL COMMAND”.**



To do this, create a new request in Postman (+ button). This must be of type “POST”.

Url: **\*\*\*Your ShopwareUrl\*\*\*/api/oauth/token**

Now go to the “Body” tab, select “form-data” and enter the following parameters:

grant_type	client_credentials
client_id	***ACCESS KEY ID***
client_secret	***SECRET KEY***

You received the access and secret key during the “integration” and stored it in a safe place.

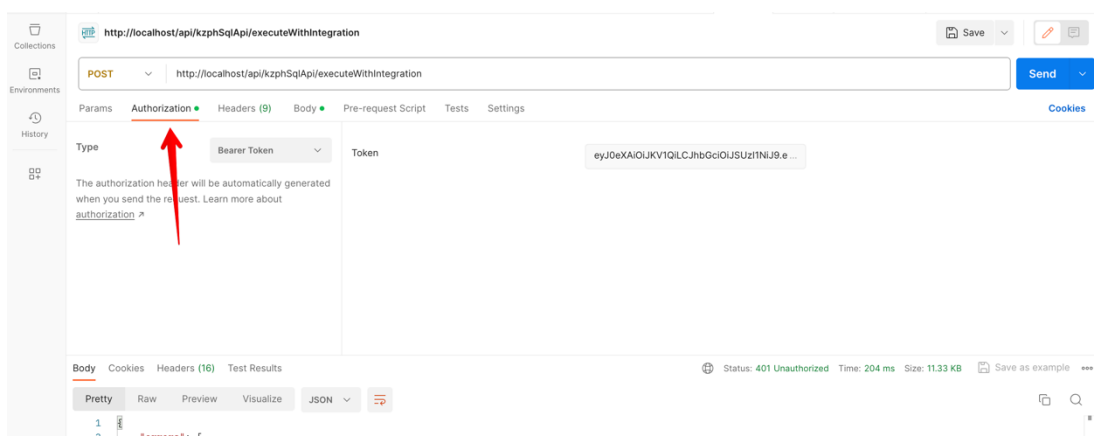
By clicking on “SEND” you should receive an access token in return.

Copy the access token and create a new request again by clicking on (+). This should also be of type “POST”.

The URL is now:

\*\*\*Your ShopwareUrl\*\*\*/api/kzphSqlApi/executeWithIntegration

Now you have to select “Bearer Token” from the dropdown under “Authorization”. Then copy the access token into the input field on the right.



Now everything should be prepared to finally send the first SQL command.



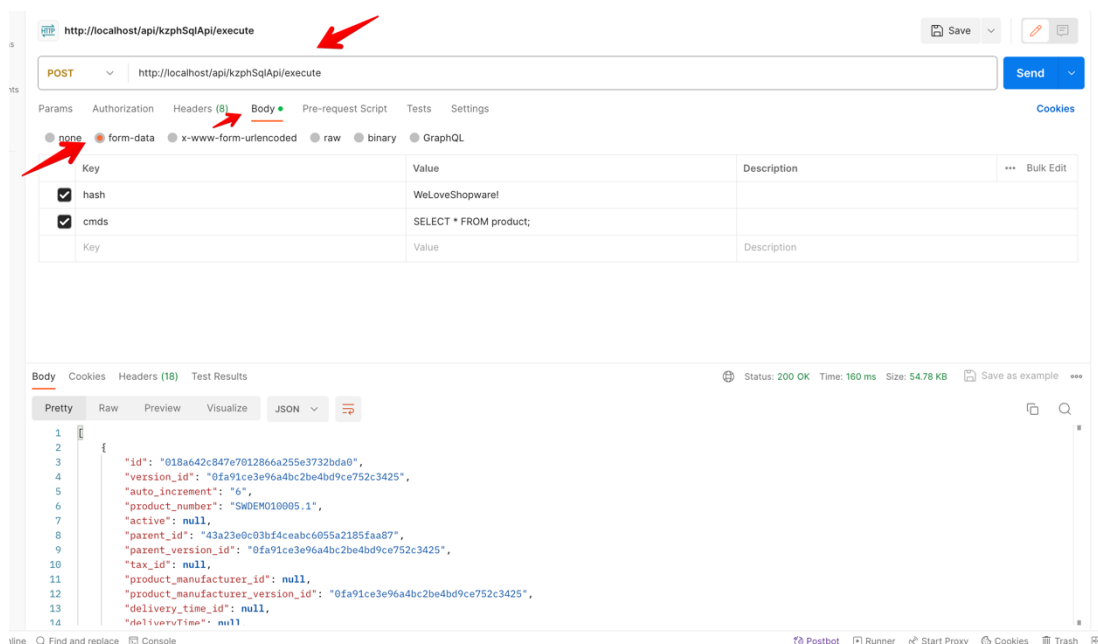
# ISSUE SQL COMMAND

Important: do you work with the hash method? The URL for this is:

\*\*\*Your ShopwareUrl\*\*\* /api/kzphSqlApi/execute

Jump to the “body” tab -> select “form -data” again and define the following parameters:

hash	Only necessary for the “hash” security method.
cmds	The SQL commands come in here. Multiple commands can be started with a ; be separated.



As a return value you will receive a JSON array with the results.

# SQL EXAMPLES FOR TESTING

---

1. Retrieve inventory and reserved inventory:

```
SELECT id, stock, available_stock FROM `product`;
```

2. Retrieve all products with name and item number  
\*multiple languages\*:

```
SELECT A.product_number, B.name FROM `product` A  
RIGHT JOIN `product_translation` B ON A.id =  
B.product_id
```

3. Retrieve customers with email address:

```
SELECT id, customer_number, email FROM `customer`
```