

CIB Bank card payment

User manual – installation, configuration, daily use

CIB Bank servers

Live - merchant URL	<code>https://eki.cib.hu/market.saki</code>
Live - customer URL	<code>https://eki.cib.hu/customer.saki</code>
Test - merchant URL	<code>https://ekit.cib.hu/market.saki</code>
Test - customer URL	<code>https://ekit.cib.hu/customer.saki</code>

The module switches between them with **Live mode** on its configuration page.

Compatibility at a glance

PrestaShop	8.0.0 - 9.x (declared). Tested on 9.1.5 . Not for 1.7.x.
PHP	7.2 or newer (as required by your PrestaShop version). Tested on PHP 8.4 .
Server	Linux x86-64 only , PHP function <code>proc_open</code> enabled, programs allowed in the website folder – see Server requirements . Not for Windows or ARM servers.
PHP extensions	<code>curl</code> (with HTTPS), <code>mbstring</code> , <code>iconv</code> – all standard.
Web server	Apache recommended (the module protects its key folder with <code>.htaccess</code>). On nginx see section 10 .
Currencies	HUF (tested) and EUR (supported, not tested – needs a separate EUR terminal from CIB).
Bank	CIB Bank Zrt. internet card acceptance (EKI protocol), test server <code>ekit.cib.hu</code> , live server <code>eki.cib.hu</code> .

"Tested" means the full flow was run against CIB's test server: successful payment, declined card, retry, timeout, status query and reversal. On PrestaShop 8 the module uses only APIs that exist there, but it has not been run on a PrestaShop 8 shop yet – please test on a copy of your shop first.

1. What the module does

- Adds **“Bank card payment (CIB Bank)”** to the checkout. The customer ticks the data transfer consent box required by CIB, places the order and is redirected to CIB's secure payment page. The shop never sees card data.
- When the customer returns, the result is fetched from the bank server-to-server, the transaction is closed, and the page shows the data CIB requires: Transaction ID (TRID), result code (RC), result text (RT), paid amount (AMO), authorisation number (ANUM). The same data is e-mailed to the customer.
- A **cron job** checks payments the customer did not return from. As requested by CIB Bank, it queries a payment only after its 10-minute payment window, then at most every 10 minutes, and records the final result on the order.
- After a failed payment the customer gets hints about the cause and a **“Retry card payment”** button for the same order.
- In the back office each order shows its CIB transactions with **Bank status**, **History**, **Reverse** and **Refund** buttons.
- Shows “Card payment service provider: [CIB logo]” and “Accepted cards: [card logos]” on every page (CIB requirement).
- Every message to and from the bank is logged in the database. All transactions (TRID, ANUM, RC, RT, AMO) can be searched under **Orders → CIB transactions**.
- Hungarian and English texts included.

2. Before you start

1. **CIB contract.** CIB gives you a **shop ID (PID)**, e.g. ABC0001 (the 4th character is 0 for HUF, 1 for EUR), and two key files with the same name: a **test key** and later a **live key** (ABC.des). Use the .des file, not .reg.
2. **HTTPS** with a valid certificate on the shop.
3. **Friendly URLs** switched on (Shop Parameters → Traffic & SEO). CIB does not accept a return address containing ?.
4. The server must reach `https://ekit.cib.hu` and `https://eki.cib.hu` (port 443). Test with: `curl -I https://ekit.cib.hu/market.saki`
5. Access to **cron** on the server (or your hosting panel's scheduled tasks).
6. Working **e-mail sending** in PrestaShop (Advanced Parameters → E-mail) – the result e-mail is mandatory.
7. A **license code** for your shop domain (also needed for testing).

Server requirements and restrictions

The bank communication and the license check run in a small compiled program shipped with the module (`bin/cibeki-linux-x86_64`). This protects the module, but it means the module **does not run on every hosting**. Check these before buying or installing:

Requirement	Details
Operating system and processor	Linux on x86-64 (Intel/AMD 64-bit) only. Not supported: Windows hosting, ARM servers (e.g. AWS Graviton, Ampere, Raspberry Pi), macOS. Any Linux distribution works; the program needs no libraries.

PHP may run programs	The PHP function <code>proc_open</code> must be enabled. Many cheap shared hostings disable it in <code>disable_functions</code> – ask the provider before you buy. VPS, dedicated and most business hostings allow it.
The module folder may contain programs	The website folder must not be mounted with <code>noexec</code> , and the file must be executable. The module sets the permission itself; if it cannot: <code>chmod 755 modules/cibpayment/bin/cibeki-linux-x86_64</code> .
Security restrictions	SELinux/AppArmor or hosting sandboxes (e.g. CageFS) must allow the web server user to run the file. Usually no change is needed.
License code	Needed for testing and live payments. It is issued for your shop domain (subdomains included) and your CIB merchant code. For evaluation ask for a time-limited trial license.

Quick check on the server (as the web server user): `php -r 'var_dump(function_exists("proc_open"));` and `uname -m` (must print `x86_64`). After installing, the status box on the module's configuration page shows **License: Valid** when everything works, or the exact reason when not.

3. Installation

1. In the back office open Modules → Module Manager and click **Upload a module**.
2. Select `cibpayment-1.3.3.zip`. PrestaShop installs it and creates:
 - the order status **“Awaiting CIB card payment”**,
 - two database tables (transactions and message log),
 - the menu item **Orders → CIB transactions**.
3. Click **Configure**.

Command-line alternative: copy the `cibpayment` folder to `modules/`, then run `php bin/console prestashop:module install cibpayment` as the web server user.

4. Configuration

Setting	What to enter
License code	The license code you received with the module. It is issued for your shop domain (subdomains such as <code>www.</code> included) and your CIB merchant code (the letters of your shop ID and key file). Test and live payments both need a valid license. The status box shows whether the license is valid and, if not, why. If a license expires, new payments are blocked, but payments already started are still finished and reversal/refund keep working.
Live mode	No while testing (CIB test server + test key). Yes only after CIB has approved your shop and sent the live key.
Shop ID (PID) for HUF	Your PID from CIB, e.g. <code>ABC0001</code> .
Shop ID (PID) for EUR	Only if CIB gave you a EUR terminal (<code>ABC1xxx</code>). Leave empty otherwise – the payment option then only appears for HUF carts.

Test key / Live key (.des)	Upload the matching file. Test and live keys have the same name; the module stores them separately. The status box above the form shows each key's MD5 checksum – compare it with the one CIB gives you.
Merchant legal name	Your company's legal name as in the CIB contract (e.g. “Example Kft.”). It appears in the mandatory consent text and in the country line at checkout.
Data processing information page	CMS page with CIB's “Information on Data Processing” (see section 6).
Card payment information page	CMS page with CIB's customer information about card payment. The CIB logo links here.
Send customer name and address	Leave No . Country, postcode and e-mail are always sent for 3-D Secure. CIB's test server rejects accented letters, spaces and punctuation in name/city/street; enable only if CIB confirms the format.
Show CIB and card logos in the footer	Yes adds the logo strip at the bottom of every page. If your theme builds the footer with a page builder (e.g. Creative Elements), you can place the module as a widget there instead and switch this off.

Also check in Payment → Preferences that the module is enabled for your currencies, countries and customer groups.

5. Cron job (required)

The status box on the configuration page shows the exact line for your server, for example:

```
* * * * * www-data php /var/www/prestashop/modules/cibpayment/cron.php >/dev/null 2>&1
```

- Put it in `/etc/cron.d/cibpayment`. The file must be owned by root with permissions `644` – otherwise cron ignores it (“INSECURE MODE” in the system log).
- Hosting panels: create a task that runs every minute: `php /path/to/modules/cibpayment/cron.php`
- The script only runs from the command line; web access is blocked.
- Check: the configuration page shows **Last cron run**; it should never be more than a minute or two old.

6. CIB requirements checklist

CIB tests your shop before activation. The module covers the technical points; the rest is shop content:

Requirement	Covered by
-------------	------------

Bank and card scheme logos on every page and at the payment method: CIB's official combined image (CIB_es_kartyalogok_30px_hrz) between the “Bank card payment” and “Information on Data Processing” links), linking to the information page	Module (footer strip + payment option)
Card logos with “Accepted cards:”	Module
Consent checkbox with the required text	Module (checkout)
Result page with TRID, RC, RT, AMO, ANUM and the required labels	Module
Result sent by e-mail	Module – needs working e-mail sending
Status query (MSGT33) for payments the customer did not return from – only after the 10-minute window, then at most every 10 minutes (CIB's request)	Module – needs the cron job
Retry after a failed payment	Module
Customer information page about card payment	You: create a CMS page from CIB's eCom_CIB.fiz.taj (and the Q&A) documents, select it in the settings
“Information on Data Processing” page	You: create a CMS page from CIB's Adatkezelesi_tajekoztato document, fill in your company data, select it in the settings
Merchant contact data (tax number, registered office, phone, e-mail), T&C, warranty terms; country and country code visible during payment	You (Contact page, CMS). The payment option already shows “Country and country code of the registered office of [merchant]: Hungary (HU)”.
Prices in HUF on every product, shipping cost and total (also EUR if you settle in EUR)	You (catalogue settings)
Short description for every product	You

7. Testing

With **Live mode = No**, CIB's test page is marked “for testing only” and no real money moves. CIB's test cards:

Card number	Expiry / CVC	Result
4111 1111 1111 1111	any future date / any CVC	Successful payment
4999 9999 9999 9999	any future date / no CVC	Declined payment

Recommended tests:

1. **Success** with 4111... → “Successful payment”, order becomes Payment accepted.

2. **Decline** with 4999... → “Unsuccessful payment” with hints, order becomes Payment error; **Retry card payment** with 4111... → the same order becomes paid.
3. **Reverse**: on a successful test order press Bank status (should be 10), then Reverse → status 40, order Cancelled.
4. **Closed browser**: pay with 4111... and close the tab on CIB's page before returning to the shop → CIB reverses the unclosed payment after 10 minutes (the reserved amount is released); about 11 minutes after the start the cron reads the result and the order shows the failed/timed-out payment, with the “Retry card payment” option for the customer.
5. **Abandoned payment**: open CIB's page and do nothing → after 10–15 minutes the order becomes Payment error.

8. Daily use

Order statuses

Status	Meaning
Awaiting CIB card payment	Order created, customer is on (or on the way to) CIB's page.
Payment accepted	The bank authorised and the shop closed the payment. Ship the order.
Payment error	Declined, timed out or could not be started. The customer can retry from the result page. Nothing was charged.

The module only moves an order out of Awaiting CIB card payment or Payment error; statuses you set by hand are never overwritten.

Finding a transaction

Orders → CIB transactions lists every card transaction with date, TRID, order, customer, amount (AMO), response code (RC) and text (RT), authorisation number (ANUM) and status. Every column can be filtered, e.g. to find a payment by TRID or ANUM; the order number opens the order. The list can be exported to CSV.

Bank statuses and back-office actions

Open the order; the **CIB card payment** panel lists every attempt.

Bank status	Meaning	What you can do
10	Authorised – amount held on the card, not yet debited	Reverse : releases the hold, no money moves; the order becomes Cancelled
30	Debited by the bank's daily settlement (automatic)	Refund : full or partial, minimum 100 HUF / 1 EUR; a full refund sets the order to Refunded
40	Reversed	–
50	Refunded	–

60	Closed (failed, or reversed/ refunded after settlement)	-
99	No settled transaction (e.g. payment not finished)	-

- **Bank status** asks CIB for the current status and refundable amount.
- **History** shows the steps (10 = customer reached payment page, 21 = authorised, 30 = closed by shop, 55/56 = timeout reversal ...).
- Debiting happens automatically at the bank's daily settlement – the shop cannot delay it. Cancel before settlement with Reverse; after it only Refund is possible.
- A refunded transaction cannot be refunded again, but the amount can be set several times before the refund is sent.

9. Going live

1. Finish the checklist in section 6 and test (section 7).
2. E-mail **ecommerce@cib.hu** from the address in your contract: shop URL, shop ID (PID), a test customer login if needed, and a short description of the purchase process. CIB runs a test purchase (usually 2–3 working days per round).
3. CIB sends the live key (encrypted zip; the password is handed over at a branch).
4. Upload it as **Live key**, compare the MD5 checksum, check that the **license** shows as valid, set **Live mode = Yes**, save. Live mode cannot be switched on without a valid license. If your shop moves to another domain, ask for a new license code first.
5. Make a small real payment and reverse it from the back office.
6. Delete test customers and test orders.

10. Security notes

- Keys are stored in `modules/cibpayment/keys/test/` and `.../keys/live/`. On Apache, `.htaccess` blocks web access. **On nginx** add a rule, e.g. `location ^~ /modules/cibpayment/keys/ { deny all; }`, and check that `https://your-shop/modules/cibpayment/keys/live/ABC.des` returns 403. Block `/modules/cibpayment/bin/` the same way.
- Keep backups of the key files outside the web root. Never e-mail them unencrypted.
- The module never sees card numbers; the bank only returns a masked number.
- The transaction ID (TRID) and log must be kept for at least one year (CIB requirement). Uninstalling the module keeps both tables.

11. Troubleshooting

Symptom	Cause / solution
Payment option not shown	License not valid (see the status box), no PID for the cart's currency, key file missing for the current mode, or the module is not enabled for the currency/country/group (Payment → Preferences).

Status box: “proc_open is disabled”, “program cannot be run” or “server is not supported”	The server does not meet the server requirements . Ask the hosting provider to enable <code>proc_open</code> and allow running programs from the website folder, or move to a Linux x86-64 hosting that does.
Status box: license not valid for this domain / another merchant / expired	Ask for a new license code: after moving the shop to another domain, after receiving a different CIB shop ID, or when the license has expired.
Log line “NOT SENT - module program/ license error”	The message was not sent to the bank because of the license or the server check; the status box shows the reason.
“The card payment could not be started”	Check the log (<code>cibpayment_log</code> table). RC=S01 / S06 : wrong key (test key on live server or vice versa, or wrong file). RC=S05 : data rejected – switch off “Send customer name and address”. RC=D04 : too many requests, try again later. Connection error: firewall / DNS towards <code>ekit.cib.hu</code> / <code>eki.cib.hu</code> .
Orders stay “Awaiting CIB card payment”	Cron is not running. Check Last cron run, the cron file permissions (644, owner root) and the system log.
Return address warning on the settings page	Switch on friendly URLs.
Customer says money was held but order shows error	Press History and Bank status on the order. A timed-out payment is released by CIB automatically (status 55/56 in the history).
No result e-mail	Fix PrestaShop's e-mail settings and send a test e-mail from Advanced Parameters → E-mail.

For CIB support write to **ecommerce@cib.hu** with the subject “Tranzakció vizsgálati kérelem <shop ID>”, the TRID, the time, your server's IP address and the related log lines.

cibpayment 1.3.3 · provided as-is, without warranty. CIB Bank, Visa, Mastercard, Maestro and V PAY logos are trademarks of their owners and are included as supplied by CIB Bank for merchant use.