



Sureway

Payments
The Standard Gateway

INTEGRATION GUIDE

Integrate in a format you already know

For platform teams: what your team builds, what Sureway handles, and what we need to start.



Who this is for

This guide is for the team that owns gateway connectors in a billing, subscription or commerce platform. It is usually a connector type your team has built before.

Summary

Your team builds one connector in a format it already supports. Sureway handles processing, card storage and certification. Your platform keeps billing, schedules, retries and its gateway list.

DIVISION OF WORK

What you build, and what Sureway handles



You build

- 1 A connector in a gateway format your team already knows
- 2 A Sureway entry on your gateway list, with its credential fields
- 3 A run through the certification scenarios



Sureway handles

- ✓ Receiving and validating every request
- ✓ Processing on each merchant's account
- ✓ BIN-level optimization on each request
- ✓ Responses in your connector's format
- ✓ Card storage, and migration of existing stored cards
- ✓ Sandbox credentials and certification scenarios
- ✓ Merchant onboarding
- ✓ Go-live support for your team

What stays with your platform

Area	Detail
Billing	Schedules, amounts and when each charge runs. Sureway never starts a transaction.
Retries and dunning	Declines come back as declines. Your rules decide what happens next.
Your gateway list	Sureway is one entry on it. Merchants can choose it only once you add it.

CHOOSING

Choosing a format

Sureway accepts several gateway API formats. Look at the gateway connectors your platform already has and pick the format your team knows best. The format is confirmed during scoping.

Format	Shape	Best for
Format 1	Key-value, form-encoded	The most widely carried connector type among billing and subscription platforms
Format 2	JSON or XML envelope	Platforms with a versioned envelope connector and customer profiles
Format 3	XML schema	Platforms built on an XML schema connector
Format 4	Signed-request REST	Platforms with a signed-request REST connector
Format 5	Versioned JSON	Platforms with a versioned JSON payments connector
Format 6	JSON, stored payment methods	Platforms that charge stored payment methods server-side
Format 7	GraphQL, stored payment methods	Platforms that charge stored payment methods through GraphQL
Platform-native	The platform's own format	Platforms with their own API for adding external gateways

Each format is documented in full in the API specification, shared during technical review.

Integration outline

- 1 Add a connector in the supported gateway format, or reuse the one you have.
- 2 Add Sureway to your gateway list with its credential fields.
- 3 Get sandbox credentials from your Sureway contact.
- 4 Run the certification scenarios for the accepted operations.
- 5 Go live merchant by merchant, as each one chooses Sureway.

If your platform already has a connector in the chosen format, the work is mostly configuration: make its endpoint and credential fields configurable, or clone it under the Sureway label. If it does not, it is a new connector of a type your team has built for other gateways. Test credentials and a test environment are arranged with your Sureway contact during scoping.

EXAMPLE

A sample charge

A rebill sent in Format 1. Field names and response codes follow the gateway format, so your connector parses the answer as it does today.

Request

```
POST https://[api-endpoint]/api/transact
Content-Type: application/x-www-form-urlencoded

type=sale
security_key=YOUR_API_KEY
amount=49.00
currency=USD
orderid=SUB-10482-R7
ccnumber=4111111111111111
ccexp=1228
first_name=Jordan
last_name=Lee
address1=200 Main St
zip=94107
initiated_by=merchant
stored_credential_indicator=used
billing_method=recurring
```

Response (one field per line for reading)

```
response=1
responsetext=SUCCESS
authcode=123456
transactionid=10293847561
avsresponse=Y
cvvresponse=M
orderid=SUB-10482-R7
type=sale
response_code=100
```

Example values only.

Accepted operations

Operation	Status	Notes
sale	Accepted	Authorize and capture in one call. The main path for first charges and rebills.
refund	Accepted	Against the Sureway transaction ID, full or partial.
void	Accepted	Before the batch settles.
auth and capture	Planned	Separate authorization and capture.
Reporting and queries	Not accepted	Each response carries the full result and transaction ID.
Funding and settlement	Not applicable	Funds settle to the merchant through their acquiring bank.
Recurring schedules	Not offered	Your platform runs the billing schedule.

An operation outside this list returns a not-supported response in the format's own error style.

CARDS AND TOKENS

Stored cards and rebills

- ✓ Cards are stored in a tokenized vault. Your platform stores the token, not the card number.
- ✓ Merchants moving from another provider can migrate their stored cards to Sureway.
- ✓ Card export to another PCI DSS compliant provider is available on request.

Cards a merchant already has stored with their current gateway can be charged through Sureway without collecting them again, as long as that gateway account stays open. Merchants can also move those cards into the Sureway vault.

Stored credentials on rebills

Rebills reach the issuer flagged as merchant-initiated (MIT), linked to the network transaction ID (NTID) of the first customer-initiated (CIT) payment. Sureway accepts the stored-credential fields your connector sends today and returns the NTID for later charges, so each rebill stays linked to the cardholder's original authorization.

Declines

A decline comes back to your platform as a decline, with the issuer's response code in your connector's format. Your retry and dunning rules decide what happens next, as they do with your other gateways.

Duplicate requests

If your connector resends a sale with the same order reference and amount within the duplicate window, Sureway returns the original result and does not charge twice. Where the format has its own duplicate rule, Sureway follows it.

Timeouts and availability

If your connector times out, resend with the same order reference; inside the duplicate window Sureway returns the original result. If Sureway cannot be reached, your connector receives a standard gateway error in its format, and your platform's existing rules for an unavailable gateway apply.

ENABLEMENT

Turning it on for a merchant

Once your platform carries the connector, enabling a merchant is a short configuration task.

Step	Detail	Who
1. Merchant chooses Sureway	The merchant asks for Sureway the way they would ask for any new gateway.	Merchant
2. Merchant onboarding	Sureway onboards the merchant and confirms the merchant account their charges run on.	Merchant, Sureway
3. API credentials	Sureway issues credentials in the format your connector's credential fields expect.	Sureway
4. Connector configured	The merchant or your support team selects Sureway on your gateway list and enters the credentials.	Merchant or platform
5. Stored cards	Existing stored cards are kept as they are or migrated into the Sureway vault, as the merchant chooses.	Merchant, Sureway
6. Verification charge	A small authorization, or a test transaction where your platform supports one, confirms the setup.	Merchant, Sureway
7. Live	New charges and rebills run through Sureway.	All

GETTING STARTED

What we need from you to start

Question	Why it helps
Which gateway formats your platform already carries, and whether their endpoints are configurable.	Picks the format and tells us whether this is configuration or a new connector.
What your connector sends on first charge and on rebills: card data, or a stored-card token.	Tells us how card storage and migration fit your platform.
Which stored-credential fields your connector sends today.	Confirms how rebills are flagged to the issuer.
How your platform handles a response code it has not seen before.	Lets us test the code mapping against your actual handling.
What your support team needs to see on each transaction.	Confirms the response fields before certification.

Talk to the integrations team

Tell us which gateway connectors your platform ships. That tells us which format fits. Write to [contact@\[domain-tbd\]](mailto:contact@[domain-tbd]).