



PaymentSafe Protocol

API Reference & Integration Guide

Version 1.0 (Sandbox/Production)

Published: June 2026

Support: developers@paymentsafe.business

PaymentSafe — API Reference Documentation

Welcome to the **PaymentSafe Developer API**. This documentation outlines the REST endpoints, authentication standards, and webhook event formats required to integrate PaymentSafe's secure escrow vault directly into custom marketplaces, mobile applications, or storefronts (such as Jumia or Shopify).

1. Authentication & Base URL

All requests must be made over HTTPS. The sandbox API is base-routed to:

```
https://api.paymentsafe.business/v1
```

To authenticate your API requests, pass your secret key (`sk_sandbox_...`) in the `Authorization` header as a Bearer token:

```
Authorization: Bearer sk_sandbox_77c29be01ffb43aa9f7478da25ab8a4c
Content-Type: application/json
```

[!WARNING] Keep your secret keys secure. Never expose secret keys in client-side applications or public repositories.

2. REST Endpoints

2.1 Create an Escrow Draft

Initiates a new escrow protection vault. Usually called when a buyer clicks your checkout button.

- **Endpoint:** `POST /escrows`
- **Headers:**
 - `Authorization: Bearer <SECRET_KEY>`
 - `Content-Type: application/json`

- **Request Payload:**

```
{
  "merchantId": "cli_55f9a62b",
  "itemTitle": "Oraimo Smart Watch ES-1",
  "amount": 350.00,
  "currency": "GHS",
  "buyerEmail": "buyer@example.com",
  "sellerPhone": "+23324001111",
  "inspectionDays": 3,
  "tradeNature": "local"
}
```

- **Request Parameters (B2B Cargo Shipping Fallback):** If the `tradeNature` is set to `"cargo"`, you can include logistics details:

```
{
  "tradeNature": "cargo",
  "cargoShippingLine": "MSC Logistics",
  "containerNum": "MSCU1234567",
  "blNum": "BLMSC987654"
}
```

- **Response Payload (201 Created):**

```
{
  "id": "tx_88f910ab",
  "merchantId": "cli_55f9a62b",
  "itemTitle": "Oraimo Smart Watch ES-1",
  "amount": 350.00,
  "currency": "GHS",
  "buyerEmail": "buyer@example.com",
  "sellerPhone": "+233240001111",
  "status": "CREATED",
  "inspectionDays": 3,
  "tradeNature": "local",
  "escrowFee": 5.25,
  "totalDue": 355.25,
  "createdAt": "2026-06-20T09:30:15Z"
}
```

2.2 Retrieve Escrow Status

Query the vault to check if the buyer has completed payment, or if the funds have been dispatched.

- **Endpoint:** GET /escrows/:id
- **Response Payload (200 OK):**

```
{
  "id": "tx_88f910ab",
  "status": "FUNDED",
  "amount": 350.00,
  "currency": "GHS",
  "buyerEmail": "buyer@example.com",
  "sellerPhone": "+233240001111",
  "escrowFee": 5.25,
  "totalDue": 355.25,
  "payoutStatus": "AWAITING_DELIVERY",
  "createdAt": "2026-06-20T09:30:15Z",
  "fundedAt": "2026-06-20T09:32:04Z"
}
```

Escrow Statuses Lifecycle:

- **CREATED** : Vault initialized; awaiting payment from the buyer.
 - **FUNDED** : Funds successfully deposited and locked in the vault. Seller is safe to ship.
 - **SHIPPED** : Seller has submitted transit details / Bill of Lading.
 - **DELIVERED** : Package received by the buyer; inspection window is ticking.
 - **RELEASED** : Buyer approved or inspection window expired. Funds settled to the seller.
 - **DISPUTED** : Buyer opened a claim; vault is locked pending administrative review.
-

2.3 Release Funds (Admin Payout)

Programmatically releases locked vault funds to the seller's mobile money/bank account. Requires administrative/staff bearer tokens.

- **Endpoint:** `POST /escrows/:id/payout`
- **Response Payload (200 OK):**

```
{
  "id": "tx_88f910ab",
  "status": "RELEASED",
  "payoutAmount": 350.00,
  "payoutChannel": "mobile_money",
  "recipientNumber": "+233240001111",
  "transactionId": "paystack_settlement_9981024",
  "releasedAt": "2026-06-20T09:44:12Z"
}
```

3. Webhook Event Notifications

When transaction states change, PaymentSafe dispatches an outbound `POST` request to your registered **Webhook Target URL**.

3.1 Webhook Headers

Webhooks include a cryptographic signature header `x-PaymentSafe-Signature` to verify that the request originated from our servers. The signature is computed as an HMAC-SHA256 hash of the request body using your app's **Secret Key**:

```
X-PaymentSafe-Signature: 6e982c7a5db7cf43b9df702ab92d04a6ee2d80a1c6a28d5789a6be32bc93c042
Content-Type: application/json
```

3.2 Event Payload Examples

Event: `escrow.created`

Fires when an escrow draft is initialized.

```
{
  "event": "escrow.created",
  "timestamp": "2026-06-20T09:30:15Z",
  "data": {
    "id": "tx_88f910ab",
    "amount": 350.00,
    "currency": "GHS",
    "itemTitle": "Oraimo Smart Watch ES-1",
    "buyerEmail": "buyer@example.com",
    "sellerPhone": "+233240001111"
  }
}
```

Event: `escrow.funded`

Fires immediately after the buyer completes the Paystack MoMo or Card checkout. **Safe to ship the items now.**

```
{
  "event": "escrow.funded",
  "timestamp": "2026-06-20T09:32:04Z",
  "data": {
    "id": "tx_88f910ab",
    "amount": 350.00,
    "currency": "GHS",
    "paymentChannel": "momo",
    "paymentProvider": "mtn"
  }
}
```

Event: `escrow.completed`

Fires when the inspection window passes or the buyer approves the items, resulting in a successful payout.

```
{
  "event": "escrow.completed",
  "timestamp": "2026-06-20T09:44:12Z",
  "data": {
    "id": "tx_88f910ab",
    "amount": 350.00,
    "currency": "GHS",
    "payoutChannel": "mobile_money",
    "payoutReference": "paystack_settlement_9981024"
  }
}
```

Event: `escrow.disputed`

Fires when a buyer files a dispute or a compliance hold is triggered.

```
{
  "event": "escrow.disputed",
  "timestamp": "2026-06-20T09:50:01Z",
  "data": {
    "id": "tx_88f910ab",
    "amount": 350.00,
    "currency": "GHS",
    "disputeReason": "The watch screen is cracked and doesn't turn on.",
    "raisedBy": "buyer@example.com"
  }
}
```

4. Alternative Escrow Triggering Mechanisms

In addition to standard REST API drafts, PaymentSafe supports several alternative triggering mechanisms optimized for peer-to-peer retail and international cargo logistics:

4.1 Chatbot Trigger (WhatsApp & Telegram)

Integrate escrow creation inside messaging apps where negotiations occur. * **Flow:** A user texts the PaymentSafe chatbot (e.g., `/escrow @seller 350 GHS "iPhone 11"`). * **Payload:** The bot processes the message body, calls `POST /escrows`, and replies with a secure Paystack MoMo check-out link.

4.2 Offline USSD Gateway (*224#)

Allows buyers and sellers to trigger and fund escrow contracts on feature phones without internet access. * **Flow:** Dial `*224#` in West Africa to input the seller's phone, price, and currency. * **Payload:** The USSD gateway sends a mobile money payment prompt (MoMo STK Push) directly to the buyer's phone.

4.3 QR Code "Protected Checkouts"

Allows in-person P2P checkouts at transit hubs or marketplace meetups. * **Flow:** The seller generates a product QR code. The buyer scans it to instantly trigger and fund the escrow agreement on their smartphone before receiving the items.

4.4 Carrier & Logistics Webhooks

Automates escrow locks based on shipping events from local couriers or international shipping lines. * **Flow:** Integrating courier apps (DHL, Bolt, local dispatchers) or shipping line tracking APIs. *

Payload: When the courier inputs container tracking or B/L parameters, the logistics webhook triggers the escrow vault funding lock.

4.5 Email Invoice Parser

A no-code solution for B2B cargo importers. * **Flow:** Importers forward shipping invoices or Bills of Lading to `bills@paymentsafe.business`. * **Payload:** An AI parser extracts transaction parameters (amounts, shipping details, seller phones) and drafts the escrow agreement automatically.