Update / easy.affiliate REST-API (Orders)

Mit diesem Aufruf ändern Sie bestehende Transaktionen (Orders): Status setzen, Umsatz korrigieren, Stornogrund hinterlegen. Einzeln oder viele auf einmal. Neue Transaktionen legt dieser Aufruf nie an, dafür gibt es Create Orders.

Den kompletten Ablauf eines Sales-Abgleichs, auch per Datei-Upload und SFTP, beschreibt Automatisierter Sales-Abgleich.

Endpunkte

Login	URL
Admin	`https://<Ihre-Domain>/ws/V6/admin/REST/Orders`
Advertiser	`https://<Ihre-Domain>/ws/V6/advertiser/REST/Orders`

Methode: `PUT`. Statt `REST` funktioniert auch `JSON` im Pfad, beide verhalten sich gleich.

Welche Transaktionen ein Login ändern darf:

- **Advertiser-Login:** nur Transaktionen der Kampagnen seines Advertisers.
- **Admin-Login mit Mandant:** nur Transaktionen der Kampagnen dieses Mandanten.
- **Admin-Login ohne Mandant:** alle Transaktionen der Installation.

Transaktionen außerhalb dieses Bereichs behandelt die API wie nicht vorhanden.

Header

Header	Wert
`Content-Type`	`application/json`
`X-Auth-ID`	Login-ID des Benutzers

Header	Wert
`X-Auth-Token`	Access-Token des Benutzers, zu finden unter **API → Daten-API**
`X-Network-ID`	ID des Mandanten. Bei nur einem Mandanten oder mandantenübergreifend `-1`.

Wo die Login-ID steht: Wo ist die User ID hinterlegt?

Beispiel

```
curl -X PUT "https://<Ihre-Domain>/ws/V6/admin/REST/Orders" \
  -H "Content-Type: application/json" \
  -H "X-Auth-ID: <LOGIN-ID>" \
  -H "X-Auth-Token: <ACCESS-TOKEN>" \
  -H "X-Network-ID: -1" \
  -d '[
    { "id": 12345, "status": 1 },
    { "id": 12346, "status": 2, "cancel_reason": "Retoure" },
    { "id": 12347, "status": 1, "turnover": 89.90 },
    { "campaign_id": 1, "ordertoken": "123a456b", "status": 2, "cancel_reason": "Storno" }
  ]'
```

Der Body ist eine Liste von Objekten, je Objekt eine Transaktion. Ein einzelnes Objekt ohne Liste geht auch.

Transaktion finden

Weg	Felder	Hinweis
Transaktions-ID (empfohlen)	`id`	eindeutig, mehr ist nicht nötig. Geben Sie zusätzlich `campaign_id` mit, muss sie zur Transaktion passen.
Bestellnummer	`campaign_id` + `ordertoken`	`campaign_id` ist hier Pflicht. Gibt es zur Bestellnummer mehrere Transaktionen in der Kampagne (z. B. Lead und Sale), ist die Zuordnung nicht eindeutig. Dann immer `id` verwenden.

Weg	Felder	Hinweis
Klick-ID	`emid` + `ordertoken`	Eine 24-stellige `emid` ersetzt die `campaign_id`, die Kampagne wird aus dem Klick ermittelt.

Felder

Feld	Bedeutung	Typ
`status`	`0` offen, `1` bestätigt, `2` storniert. Auch als Wort: `offen`, `bestätigt`, `storniert`, `open`, `confirmed`, `cancelled`. Andere Werte werden abgewiesen.	Integer
`turnover`	Netto-Umsatz in Kampagnenwährung, Punkt als Dezimaltrenner	Float
`original_turnover`	Umsatz in Fremdwährung. **Überschreibt `turnover`**, wenn gesetzt.	Float
`commission`	fester Provisionsbetrag. Nur mitgeben, wenn er bewusst von der Provisionsgruppe abweichen soll.	Float
`cancel_reason`	Stornogrund, Freitext	String
`status_change_date`	Datum der Statusänderung (ISO 8601). Ohne Angabe setzt das System es bei einem Statuswechsel selbst. Bestimmt auch das Abrechnungsdatum.	Datetime
`sub_status`	Substatus, frei belegbar	String
`description`	Beschreibung, höchstens 255 Zeichen	String
`order_currency`	Bestellwährung als ISO-Code, z. B. `EUR`	String
`subid_value`	SubID der Transaktion. Heißt hier `subid_value`, nicht `subid` wie beim Anlegen.	String
`get_parameters`	Objekt `{ "key": "value" }`, wird je Transaktion ergänzt oder überschrieben	Object

Felder, die Sie nicht mitschicken, bleiben unverändert.

Provision

Ohne `commission` gilt: Bei prozentualer Provision rechnet das System die Provision aus dem neuen `turnover` neu. Feste Provisionen bleiben, wie sie sind. Der Wert `"commission": "autocalculate"` bewirkt dasselbe und ist nicht nötig. Ein Zahlenwert in `commission` wird genau so übernommen.

Nicht änderbar

Ausgezahlte Transaktionen (Status 3) sind abgerechnet und lassen sich nicht mehr ändern. Status 3 kann auch nicht per API gesetzt werden, er entsteht bei der Auszahlung.

Antwort

```
{
  "data": [
    { "id": 12345, "status": 1, "turnover": 49.9, "commission": 4.99, "cancel_reason": null,
"status_change_date": "2026-10-28T14:50:44+00:00", ... },
    { "id": 12346, "status": 2, "turnover": 19.99, "commission": 2, "cancel_reason":
"Retoure", ... },
    { "id": 12347, "status": 1, "turnover": 89.9, "commission": 8.99, ... }
  ]
}
```

- `data` enthält jede geänderte Transaktion mit ihrem neuen Stand, so wie sie jetzt im System steht.
- **Gleichen Sie die IDs in `data` mit den gesendeten ab.** Fehlt eine, wurde sie nicht geändert: nicht gefunden, bereits ausgezahlt, außerhalb Ihres Bereichs oder ungültiger Status. In einer Liste gibt es dafür keine eigene Fehlermeldung.
- Schicken Sie ein einzelnes Objekt ohne Liste, steht die Transaktion direkt in `data`. Kann sie nicht geändert werden, kommt `"error": true`, der Grund steht in `msg`, zum Beispiel `status 7 is not allowed, use 0 (open), 1 (confirmed) or 2 (cancelled)`.

Im Beispiel oben fehlt die vierte Transaktion in der Antwort, wenn es die Bestellnummer `123a456b` in Kampagne 1 nicht gibt.

Fehler

Antwort	Bedeutung
`"message": "Authentification failed."`	Login-ID oder Access-Token falsch, oder der Login ist gesperrt
`"error": true` bei einem einzelnen Objekt	Transaktion nicht gefunden, nicht änderbar oder ungültiger Status. Der Grund steht in `msg`.
Transaktion fehlt in `data`	siehe Antwort

Jede Änderung steht in der Historie der Transaktion mit Zeitpunkt und Benutzer. Ein eigener Benutzer für die Anbindung, etwa „Salesabgleich API“, macht dort sofort sichtbar, welche Änderung aus der Schnittstelle kam.

Revision #9
Created 2026-03-12 09:23:25 UTC by Daniel Hörenkamp
Updated 2026-10-28 15:38:42 UTC by Markus Wigbels