

Publisher-Verwaltung über die V6 REST-API

Publisher-Verwaltung über die V6 REST-API

Authentifizierung

Erforderliche Header für alle Anfragen:

Header	Beschreibung
X-Network-ID	Netzwerk-ID, z.B. <code>-1</code> für global
X-Auth-Token	API-User-Token
X-Auth-ID	API-User-Login-ID
Content-Type	<code>application/json</code>

Endpunkt

Base URL: `https://SUBDOMAIN.de/ws/V6/admin/REST/Publisher`

Es werden die üblichen HTTP-Methoden verwendet: `POST` zum Anlegen, `GET` zum Abrufen.

CREATE Operation (POST)

Methode & Pfad: `POST /ws/V6/admin/REST/Publisher`

Erforderliche Parameter

Parameter	Typ	Beschreibung
<code>email</code>	String	Max. 255 Zeichen
<code>salutation</code>	String	z.B. <code>mr</code> / <code>mrs</code>
<code>prename</code>	String	Max. 255 Zeichen
<code>surname</code>	String	Max. 255 Zeichen

Optionale Parameter

`company`, `street`, `zip`, `city`, `telephone`, `country` (ISO 3166 ALPHA-3), `billing_sepa_owner`, `billing_sepa_iban`, `billing_sepa_bic`, `companytype` (z.B. `un`/`priv`), `language_interface` (Default: `DEU`), `billing_mode` (Default: `1`), `billing_media` (Default: `1`), `country_billing` (Default: `DEU`), `country_publisher` (Default: `DEU`), `billing_limit` (Default: `25`), `dialog_email` (Default: Wert von `email`)

“ Für `salutation`, `companytype` und `country` gibt es keine feste Werteliste, die serverseitig erzwungen wird — bitte trotzdem an die üblichen Werte halten, damit nachgelagerte Auswertungen (Rechnungsstellung, Sprachwahl etc.) korrekt funktionieren.

Response

```
{
  "data": {
    "id": 123,
    "name": "Max Mustermann",
    "email": "max.mustermann@example.com"
  },
  "log": "no logmessage provided",
  "sessionId": "",
  "session": null
}
```

Die eigentlichen Daten stehen im `data`-Feld.

READ Operation (GET)

Methode & Pfad: `GET /ws/V6/admin/REST/Publisher` bzw. `GET /ws/V6/admin/REST/Publisher/{id}`

Einzelnen Publisher abrufen

`GET /ws/V6/admin/REST/Publisher/{id}` — `id` ist Teil des URL-Pfads, keine Query-Parameter.

Liste abrufen (mit Filtern)

Filter werden als Query-Parameter im Format `params[feld]=wert` übergeben, z.B.:

```
GET /ws/V6/admin/REST/Publisher?params[status]=1
```

Filterbar sind sowohl die Basis-Felder des Publisher-Accounts (`id`, `status`, `type`) als auch alle Felder aus dem Registrierungsprozess (`email`, `prename`, `surname`, `company`, `street`, `zip`, `city`, `telephone`, `country`, Billing-Felder etc.). Mehrere Filter zusammen werden UND-verknüpft, z.B.:

```
GET /ws/V6/admin/REST/Publisher?params[status]=5&params[prename]=Max
```

Filter auf Registrierungsfelder sind exakte Treffer (kein Teilstring-/Wildcard-Match).

Status-Werte

Bekannte Werte für `status`:

Status	Bedeutung
1	Bestätigter, aktiver Publisher
5	Neu registriert, noch nicht freigegeben

(Für eine vollständige Übersicht aller möglichen Status-Werte bitte im Admin-Interface unter der Publisher-Verwaltung nachsehen.)

Response-Format der Liste

```
{
  "data": {
    "123": {
      "type": "pub",
      "id": 123,
      "status": 5,
      "settings": {
```

```

    "email": "max.mustermann@example.com",
    "prename": "Max",
    "surname": "Mustermann",
    "company": "",
    "street": "Musterstraße 1",
    "zip": "12345",
    "city": "Musterstadt",
    "country": "DEU",
    "...": "weitere im Registrierungsprozess gespeicherte Felder"
  }
},
"log": "no logmessage provided",
"sessionId": "",
"session": null
}

```

Das `settings`-Objekt enthält alle für den Publisher gespeicherten Registrierungsdaten — welche Felder konkret enthalten sind, hängt davon ab, was beim jeweiligen Publisher tatsächlich hinterlegt wurde (kein festes Schema).

“ **Bekannte Einschränkung:** Informationen zur Werbefläche (Werbefläche/Werbeflächen-URL), die im Registrierungsprozess ebenfalls abgefragt werden, sind über diese API aktuell **nicht** abrufbar. Das wird als Erweiterung nachgezogen.

Beispiele

- Einzelner Publisher: `GET /ws/V6/admin/REST/Publisher/123`
- Neu registrierte, noch nicht freigegebene Publisher: `GET /ws/V6/admin/REST/Publisher?params[status]=5`
- Publisher per E-Mail suchen: `GET /ws/V6/admin/REST/Publisher?params[email]=max.mustermann@example.com`

Fehlerbehandlung

Fehler werden in folgendem Format zurückgegeben:

```
{
  "error": true,
  "msg": "<Fehlerbeschreibung>",
  "type": "Ws\\Error",
  "data": []
}
```

Der HTTP-Statuscode ist bei Fehlern durchgängig `500`.

Fehler	Beispiel- <code>msg</code>
Fehlendes Pflichtfeld bei CREATE	<code>internal server error. Details: Missing required field: email</code>
Publisher nicht gefunden	<code>internal server error. Details: Model not found.</code>
Filter ohne Treffer	kein Fehler — liefert <code>"data": []</code>

Revision #2

Created 7 April 2026 13:47:26 by Daniel Hörenkamp

Updated 6 July 2026 15:46:17 by Leonie Engstfeld