



API-Security-Checkliste

Authentifizierung, Berechtigungen, Rate Limiting, Validierung und Monitoring für deine APIs.

Version 1.0 · Oktober 2026 · switech-solution.com

APIs sind das Rückgrat moderner Apps und ein beliebtes Angriffsziel. Diese Checkliste orientiert sich an den OWASP API Security Top 10 (Ausgabe 2023). Die Kürzel API1 bis API10 verweisen auf die jeweilige Kategorie.

SO NUTZT DU DIESE CHECKLISTE

Geh die Punkte der Reihe nach durch und hake ab, was erledigt ist. Beginne mit allen Punkten «Kritisch». Wiederhole die Prüfung vor jedem grösseren Release und mindestens einmal pro Quartal.

Prioritäten: **KRITISCH** **WICHTIG** **EMPFOHLEN**

1 AUTHENTIFIZIERUNG

- ☐ **Bewährte Standards nutzen** **KRITISCH**
OAuth 2.0 / OpenID Connect oder etablierte Auth-Anbieter. Keine eigene Kryptografie (API2).
- ☐ **Tokens vollständig prüfen** **KRITISCH**
Bei JWTs Signatur, Ablaufzeit (exp), Aussteller (iss) und Empfänger (aud) prüfen. Den Algorithmus «none» ablehnen.
- ☐ **Kurzlebige Tokens** **WICHTIG**
Access-Tokens mit kurzer Gültigkeit, Refresh-Tokens widerrufbar.
- ☐ **API-Schlüssel richtig einsetzen** **WICHTIG**
Im Header übertragen, nie in der URL. Rotierbar, pro Kunde getrennt, mit minimalen Rechten. Für Nutzerdaten nicht als einzige Authentifizierung verwenden.

2 AUTORISIERUNG

- ☐ **Objektebene prüfen (API1)** **KRITISCH**
Bei jedem Zugriff auf /orders/{id} prüfen, ob das Objekt dem anfragenden Nutzer gehört. Die häufigste API-Schwachstelle.
- ☐ **Eigenschaftsebene prüfen (API3)** **KRITISCH**
Nur benötigte Felder zurückgeben. Mass Assignment verhindern: Nutzer dürfen Felder wie «role» oder «isAdmin» nicht selbst setzen.
- ☐ **Funktionsebene prüfen (API5)** **KRITISCH**
Admin- und Verwaltungsendpunkte serverseitig auf die Rolle prüfen, nicht nur im Frontend verstecken.
- ☐ **Geschäftskritische Abläufe schützen (API6)** **WICHTIG**
Registrierung, Gutscheine, Bestellungen und Buchungen gegen automatisierten Missbrauch absichern.

3 EINGABEN & VALIDIERUNG

- ☐ **Schema-Validierung** KRITISCH
Jede Anfrage gegen ein Schema prüfen (z. B. JSON Schema, zod). Typen, Längen und Wertebereiche festlegen, unbekannte Felder ablehnen.
- ☐ **Größen und Formate begrenzen** WICHTIG
Maximale Payload-Grösse, erlaubte Content-Types, Limits für Arrays und Uploads.
- ☐ **Server-Side Request Forgery verhindern (API7)** KRITISCH
Werden URLs von Nutzern abgerufen: Zieladressen per Allowlist prüfen, interne Adressen und Metadaten-Dienste blockieren.

4 RESSOURCEN & RATE LIMITING (API4)

- ☐ **Rate Limiting** KRITISCH
Limits pro Nutzer, API-Schlüssel und IP. Bei Überschreitung Status 429 mit Retry-After-Header.
- ☐ **Pagination erzwingen** WICHTIG
Maximale Seitengrösse festlegen. Nie unbegrenzt viele Datensätze zurückgeben.
- ☐ **Timeouts und Kostenlimits** WICHTIG
Timeouts für Anfragen und Abfragen setzen. Bei kostenpflichtigen Diensten (z. B. KI-APIs, SMS) Budgetgrenzen einrichten.

5 KONFIGURATION (API8)

- ☐ **Nur HTTPS** KRITISCH
Unverschlüsselte Verbindungen ablehnen, HSTS aktivieren.
- ☐ **CORS restriktiv** WICHTIG
Nur bekannte Ursprünge erlauben, keinen Platzhalter «*» mit Credentials.
- ☐ **Fehlermeldungen ohne Interna** WICHTIG
Keine Stacktraces, SQL-Fehler oder Versionsnummern an Clients senden.
- ☐ **Unnötiges deaktivieren** EMPFOHLEN
Ungenutzte HTTP-Methoden, Debug-Modi und Standardkonten abschalten.

6 INVENTAR & VERSIONEN (API9)

- ☐ **API dokumentieren** WICHTIG
Alle Endpunkte z. B. mit OpenAPI dokumentieren, inklusive Authentifizierung und Datenarten.
- ☐ **Alte Versionen abschalten** WICHTIG
Veraltete API-Versionen und Test-Endpunkte geplant ausser Betrieb nehmen.
- ☐ **Umgebungen trennen** EMPFOHLEN
Test- und Staging-APIs nicht öffentlich erreichbar machen und nie mit Produktionsdaten betreiben.

7 EXTERNE APIS (API10)

- ☐ **Antworten nicht blind vertrauen** **WICHTIG**
Daten von Drittanbietern genauso validieren wie Nutzereingaben.
- ☐ **Timeouts und Weiterleitungen** **WICHTIG**
Timeouts setzen und Weiterleitungen nicht ungeprüft folgen.
- ☐ **Schlüssel sicher verwahren** **KRITISCH**
Zugangsdaten zu Drittanbietern nur serverseitig in einem Secret-Manager oder in Umgebungsvariablen.

8 LOGGING & MONITORING

- ☐ **Sicherheitsereignisse protokollieren** **WICHTIG**
Fehlgeschlagene Logins, verweigerter Zugriffe und Überschreitungen des Rate Limits.
- ☐ **Keine Geheimnisse in Logs** **KRITISCH**
Tokens, Passwörter und vollständige Personendaten maskieren oder weglassen.
- ☐ **Alarmer und Request-IDs** **EMPFOHLEN**
Alarmer bei Auffälligkeiten. Request-IDs erleichtern die Analyse von Vorfällen.

9 TESTEN

- ☐ **Berechtigungen automatisiert testen** **KRITISCH**
Tests, die sicherstellen, dass Nutzer A nicht auf Daten von Nutzer B zugreifen kann.
- ☐ **Abhängigkeiten scannen** **WICHTIG**
Automatische Prüfung auf bekannte Schwachstellen in Bibliotheken.
- ☐ **Penetrationstest** **EMPFOHLEN**
Vor grossen Releases oder bei sensiblen Daten einen externen Test durchführen lassen.

Diese Checkliste ersetzt keine individuelle Sicherheitsprüfung. Weitere Details: OWASP API Security Project (owasp.org).

Fragen oder Unterstützung nötig?

Ich helfe kleinen Unternehmen und Gründern bei sicheren KI-Tools und Automatisierungen. Das Erstgespräch ist kostenlos.

www.switech-solution.com/de/