

# Zammad hinter Zoraxy: CSRF-Token-Probleme lösen

## Problem

Beim Zugriff auf ein Zammad-System über den Reverse Proxy **Zoraxy** kam es zu folgender Fehlermeldung:

```
{"error":"CSRF token verification failed!","error_human":"CSRF token verification failed!"}
```

Zammad blockierte POST-/PUT-Anfragen, da kein gültiges CSRF-Token erkannt wurde.

## Ursache

Rails-basierte Systeme wie Zammad schützen sich gegen CSRF (Cross-Site Request Forgery). Damit das funktioniert, muss Zammad wissen, ob die Verbindung über **HTTPS** erfolgt. Wenn hinter einem Reverse Proxy wie Zoraxy **der Header `X-Forwarded-Proto` fehlt oder falsch gesetzt ist**, denkt Zammad, die Verbindung sei unsicher (HTTP):

- **Secure-Cookies werden nicht gesetzt** → Session geht verloren
- **CSRF-Token wird nicht akzeptiert** → Aktionen scheitern

## Lösung

In der **Zoraxy-Oberfläche** wurde ein benutzerdefinierter Header gesetzt, um Zammad das HTTPS-Protokoll korrekt mitzuteilen.

## Schritte in Zoraxy

1. Reverse Proxy für `support.example.de` öffnen
2. Reiter **Custom Headers** auswählen
3. Im Abschnitt **Zoraxy → Origin** folgenden Header hinzufügen:
  - **Header Key:** `X-Forwarded-Proto`
  - **Header Value:** `https`

4. Mit „**Add Header**“ speichern
5. Einstellungen übernehmen und ggf. Zoraxy neu starten

## Optional: Weitere hilfreiche Header

Header Key	Value	Hinweis
X-Forwarded-Proto	https	Pflicht für CSRF und Cookies
X-Forwarded-Ssl	on	Optional, für bessere Kompat.
X-Real-IP	(auto)	Optional, für Logging

## Ergebnis

Nach dem Setzen des Headers erkennt Zammad wieder korrekt:

- Die Verbindung ist über **HTTPS**
- **Secure-Cookies** werden korrekt gesetzt
- **CSRF-Token-Prüfungen** funktionieren wieder

Anmeldung über den Browser ist nun problemlos möglich.

---

### Tipp:

Wenn weitere Dienste hinter Zoraxy laufen (z. B. GitLab, Nextcloud), lohnt es sich, diesen Header standardmäßig in den jeweiligen Reverse-Proxy-Konfigurationen zu setzen.

---

Revision #2

Created 15 July 2025 07:37:51 by Joachim Hummel

Updated 15 July 2025 07:38:48 by Joachim Hummel